المرجع الكامل

شرح أكثر من 100 عنصر وأكثر من 115 خاصية
وشرح كل القيم المدعومة في آخر إصدار

وائل منير

wailmonir@gmail.com

هذا المُصنَّف مرخص بموجب رخصة المشاع الإبداعي نَسب المُصنَّف - غير تجاري - منع الاشتقاق 4.0 دولي

نظراً لرفض معظم المجموعات groups نشر كتب سلسلة المرجع الكامل، ورفض تعليقي في البعض الآخر الذي وافق على النشر، فسوف أتوقف قريباً عن نشر الكتب في أي مجموعة وسيكون نشر الكتب حصرياً على المجموعة الخاص بي فقط، انضم للمجموعة حتى تتمكن من تحميل الكتب فور نزولها

https://www.facebook.com/groups/wailmonir

wailmonir@protonmail.com

هذا الكتاب صدقة جارية على روح أمي وأبي رحمهما الله، وأرجو منك عزيزي القارئ أن تتذكرهما بدعوة بظهر الغيب أن يتغمدهما الله سبحانه وتعالي بواسع رحمته وعظيم مغفرته

وجزاكم الله خيراً

wailmonir@proomail.com

إهـداء

إلى أمي و أبي

"غفر الله لهما ورحمهما رحمة واسعة"

wailmonir0@protonmail.com

wailmonir0@protonmail.com

مقدمة

الحمد لله رب العالمين والصلاة والسلام على رسول الله، نبينا محمد، المبعوث رحمة للعالمين، وعلى آله وصحبه أجمعين، ومن اهتدى بهديه، واستن بسنته إلى يوم الدين.

وبعد، فإنه بعد عناء شديد، وجهد جهيد، استمر لأشهر عديدة، قد تقارب العام، فإنني أقدم لكم هذه السلسلة المرجعية، في تصميم المواقع الإلكترونية، وهي تضم كتابي هذا، بالإضافة إلى كتابين آخرين في مراحلهما الأخيرة، ليكونا جاهزين للنشر في أقرب فرصة، وهما، المرجع الكامل لكل عناصر وخصائص وقيم SVG ، والمرجع الكامل لكل خصائص وقيم CSS والذي سيصدر في جزئين بمشيئة الله.

وأظنك عزيزي القارئ، قد لاحظت أنني دائماً أضع كلمة "كل" في شرح عناوين كتبي، ولعلك تسأل عن الإلحاح والتأكيد على هذه الكلمة، والإجابة ببساطة، هي أني لا أؤلف كتاباً مختصراً، ولا كتاباً متخصصاً في جزء أو أجزاء معينة من مادة أو لغة معينة، وإنما أقدم مرجعاً شاملاً بمعنى الكلمة، يضم كل العناصر وكل الخصائص وكل القيم، المدعومة في الوقت الحالي في معظم متصفحات الويب، لأنني شخصياً قد عانيت كثيراً من البحث عن كيفية تنفيذ فكرة معينة، رأيتها على أحد المواقع أو إحدى الصفحات، لأفاجأ بعد فترة من البحث، أن التنفيذ كان باستخدام عنصر لم أدرسه من قبل، لا في دروس الفيديو ولا في الكتب التي درستها، ثم أفاجأ أن هناك عنصراً آخر وخاصية أخرى وقيم أخرى وأخرى لم أدرسها، ووجدت أنني أضيع وقتاً طويلاً في البحث، في مواقع الفيديو والكتب والمقالات، حتى أعثر على كيفية عمل هذا العنصر أو تلك الخاصية أو هذه القيمة، ولو باللغة الإنجليزية.

لذلك فقد رأيت أن أوفر على غيري من الدارسين، الوقت والمعاناة المبذولين خلال بحثهم، وأن أؤلف مراجعاً باللغة العربية، تضم كل العناصر والخصائص والقيم المدعومة في HTML5 و CSS3، والتي قد تصادفك بعضها يوماً ما، في أحد المواقع أو

wailmonir@protonmail.com

الدروس، فبدلاً من أن تجوب المواقع والكتب والمقالات، بحثاً عن نصف إجابة ولو بلغة لا تتقنها، فأنا أقدم لك إجابة شاملة، بالشرح والصور والكود المجرب، عن أي عنصر أو أي خاصية أو أي قيمة مدعومين في متصفحات الويب الشهيرة، لتستغل وقتك في التعلم والتجربة، بدلاً من أن تضيع وقتك في بحث قد لا يصل بك إلى شيء، سوى الضيق والملل، وربما يصيبك اليأس، فتضطر أن تهجر ما درسته وما أضعت وقتاً طويلاً في تعلمه وإتقانه، لأنك وجدت - رغم جدك واجتهادك في الدراسة - أنه يفوتك الكثير الذي لم تحصله.

أما مرجعي هذا، فهو مرجع شامل لكل العناصر المدعومة في HTML5 ، وكل الخصائص التي يتم تعيينها لكل عنصر، وكل القيم التي يتم إسنادها لكل خاصية، وهي كل العناصر والخصائص والقيم المدعومة حالياً في جميع المتصفحات أو معظمها، وقد اخترت لتجربة الأكواد متصفحين فقط، حتى لا يتشتت القارئ من كثرة المتصفحات، وما هو مدعوم هنا وليس مدعوماً هناك، واختلاف السلوك في هذا عن ذاك، وهما المتصفحان Chrome و Firefox .

وكان مرجعي الأساسي في تأليف هذا الكتاب، هو الموقع الشهير MDN ، بجانب العديد من المواقع الأخرى، وبتوفيق الله، لن تجد - عزيزي القارئ - أي عنصر أو خاصية أو قيمة، مدعومين في HTML5 إلا وقد شرحتها بالقول المفصل، وبالكود المجرب، وبالصور التي توضح مخرجات هذا الكود، ليكون أمام القارئ كل الوسائل التي تجعله مدركاً لطبيعة عملها وسلوكها وتأثيرها في بناء الصفحة، وقد بذلت غاية الجهد، آملاً أن يكون كتابي هذا، مساهماً ولو بقدر بسيط في تسهيل التعلم واكتساب المهارات، التي تؤهلك لإتقان بناء المواقع الإلكترونية، في أقرب وقت ممكن.

وائل منير

waimonir@protonmail.com

التواصل والنقاش

هذا العمل نتاج عمل فردي شاق ومضنٍ، وللأسف الشديد لم أجد من يقوم بمراجعة هذا العمل الكبير ، لما يحتويه من كثير المعلومات، التي لم يسبق أن رأيت شرحاً لها باللغة العربية، على أي موقع أو فيديو أو كتاب أو مقال.

وقد تحريت كل وسائل البحث والتدقيق والمراجعة، حتى أصل إلى معلومة صحيحة ومدققة لا يشوبها أي شبهة خطأ ولو بسيط، ولكن ودائماً فوق كل ذي علم عليم، فربما أكون قد أخطأت فهم نقطة معينة، أو سهوت عن شرح موضوع معين، لذلك فإني أعتذر منكم مقدماً، عن أي خطأ أو سهو قد يبدر مني، وأرجو منك عزيزي القارئ، أن تبادر بالتواصل معي لتوضيح وجهة نظرك، ولنتناقش فيها، وأصوب إن كان الخطأ مني، أو أقنعك بوجهة نظري إن كانت صحيحة، فالتواصل والنقاش قد يفتح لنا أبواباً من الفهم والإدراك، ربما لم تكن لتفتح لولا تواصلنا ونقاشنا، ويسعدني أن أتلقي ملاحظاتكم القيمة ونصائحكم الغالية، على صفحتي على الفيسبوك أو البريد الإلكتروني:

https://facebook.com/wailmonir0
e-mail: wailmonir0@outlook.com

المؤلف

wailmonir0@hotmail.com

wailmonir0@protonmail.com

فهرس الموضوعات

الموضوع	رقم الصفحة
مقدمة عن HTML	1
كيف تنشئ صفحة ويب	2
الدخول إلى HTML	4
أنواع عناصر HTML حسب تدفقها في الصفحة	9
الخصائص Attributes	11
الأنماط التي يستخدمها المتصفح في العرض	16
هيكل كود صفحة HTML	17
تقسيم عناصر HTML	25
عناصر الكتابة وتنسيق النصوص	26
التشابه بين العناصر في التنسيق الظاهري	58
عناصر تجزئة الصفحة	59
الروابط التشعبية Hyperlinks	77
عناصر القوائم List Elements	81
عناصر الجدول Table Elements	92
عناصر النموذج Form Elements	104
عناصر الدمج Embedding Elements	171
العناصر الوصفية Metadata Elements	202
الخصائص العامة Global Attributes	223
خصائص الحدث Event Attributes	241
ملحق 1 خصائص مشتركة	275
ملحق 2 تثبيت خادم محلي	284
ملحق 3 إنشاء ملف ترجمة	292

wailmonir0@protonmail.com

فهرس العناصر مرتبة أبجدياً وموضعها في المرجع

Element	Page No.
<a>	77
<abbr>	26
<address>	37
<area>	179
<article>	68
<aside>	71
<audio>	190
<b>	32
<base>	204
<bdi>	48
<bdo>	50
<blockquoute>	45
<body>	59
 	64
<button>	160
<canvas>	187
<caption>	94
<cite>	35
<code>	41
<col>	101
<colgroup>	100
<data>	31
<datalist>	169
<dd>	90
<del>	40
<details>	76
<dfn>	36
<dialog>	75
	63
<dl>	90
<dt>	90
	34
<embed>	195
<fieldset>	167
<figcaption>	185
<figure>	185

Element	Page No.
<footer>	74
<form>	104
<head>	202
<header>	73
<h1>…<h6>	61
<hr>	66
<html>	19
<i>	34
<iframe>	198
<img>	171
<input>	107
<ins>	39
<kbd>	42
<label>	116
<legend>	167
	86
<link>	206
<main>	70
<map>	179
<mark>	46
<menu>	89
<meta>	211
<meter>	56
<nav>	72
<noscript>	221
<object>	196
	81
<optgroup>	166
<option>	164
<output>	170
<p>	27
<param>	197
<picture>	182
<pre>	43
<progress>	55
<q>	44

Element	Page No.
<rp>	54
<rt>	54
<ruby>	54
<s>	41
<samp>	42
<script>	218
<section>	69
<select>	162
<slot>	53
<small>	47
<source>	182
	30
	33
<style>	209
<sub>	46
<summary>	76
<sup>	47
<svg>	188
<table>	92
<tbody>	98
<td>	99
<template>	53
<textarea>	158
<tfoot>	98
<th>	94
<thead>	94
<time>	52
<title>	203
<tr>	99
<track>	193
<u>	38
	87
<var>	35
<video>	192
<wbr>	51

فهرس الخصائص مرتبة أبجدياً وموضعها في المرجع

Attribute	Page No.
accept	130
accept-charset	104
accesskey	225
action	105
allow	198
allowfullscreen	199
allowpaymentrequest	199
alt	172, 180
as	207
async	219
autocapitalize	110
autocomplete	105
autofocus	110, 159
autoplay	191
capture	130
charset	211
checked	119, 143
cite	39, 40, 44
class	226
cols	158
colspan	96
content	212
contenteditable	227
controls	190
coords	179
crossorigin	173, 191
data	196
data-*	228
datetime	39, 40, 52
decoding	173
default	193
defer	219
dir	48,50,229
dirname	111
disabled	112, 206
download	78, 180
draggable	230
enctype	105
for	116, 170

Attribute	Page No.
form	111
formaction	112
formenctype	113
formmethod	113
formnovalidate	113
formtarget	113
headers	97
height	135, 173
hidden	227
high	56
href	77, 180
hreflang	78, 180
http-equiv	214
id	231
imagesrcset	207
imagesizes	208
integrity	219
inputmode	237
ismap	174
itemprop	238
kind	193
label	194
lang	232
list	113, 169
loading	174, 199
loop	191
low	56
max	56,113
maxlength	114
minlength	114
media	182, 207
method	105
min	56, 113
multiple	127, 163
muted	190
name	106, 212
nomodule	220
nonce	220
novalidate	106

Attribute	Page No.
open	75
optimum	56
orient	145
pattern	128
ping	78, 181
placeholder	114, 127
playsinline	192
poster	192
preload	191
readonly	114, 129
refferrerpolicy	78, 199
rel	106, 206
required	114, 140
reversed	84
rows	158
rowspan	95
sandbox	199
scope	97
selected	164
shape	179
size	114, 162
sizes	177
span	100
spellcheck	115
src	134, 171
srcdoc	200
srclang	193
srcset	174, 182
start	82
step	115,138
style	233
tabindex	235
target	77, 106
title	36, 234
type	77,84,107
usemap	174
value	56,83,115
width	135, 173
wrap	159

* بعض الخصائص قد تكررت مع أكثر من عنصر، فذكرت ثلاثة مواضع على الأكثر، وباقي المواضع بنفس السلوك.

wailmonir0@protonmail.com

مقـــدمـــة

Inroduction

wailmonir@protonmail.com

wailmonir0@protonmail.com

مقدمة عن HTML

بما أن هذا العمل يهدف لشرح أكبر قدر ممكن من عناصر HTML ، وخصائصها ، وقيم هذه الخصائص، فقد رأيت الدخول مباشرة إلى ممارسة HTML ، والبدء في التطبيق العملي، لذلك فإنني لن أضيع وقتاً في مناقشة ما أعتبره معلومات عامة، مثل تاريخ شبكة الويب وأهميتها في حياتنا، أو ما أظنه غير مؤثر في عملية التعلم والدراسة مثل تاريخ HTML ونشأتها وإصداراتها الكثيرة، يكفيني فقط أن تعلم أننا سوف نتناول كل العناصر والخصائص والقيم المدعومة في الإصدار الأخير HTML5 ، حيث إن كل المواقع والمتصفحات - تقريباً - أصبحت تعمل بهذا الإصدار، لذلك، لن أناقش أي عنصر أو خاصية أو قيمة مدعومين في الإصدارات السابقة مثل HTML4 ، إلا إذا كانوا مدعومين في HTML5 ، فلا حاجة لإضاعة الوقت، في شرح المهمل من العناصر، والخصائص والقيم، والتي يوصَى بعدم استخدامها.

ولن أناقش الموضوعات التي لا يرجى منها نفعاً، مثل مناقشات المبتدئين والمتفلسفين، هل HTML لغة برمجة أم ليست لغة برمجة، لأن الإجابة أياً كانت، لن تغير من الواقع شيئاً، فسواء كانت HTML لغة برمجة أو لغة توصيفية، فدراستها ضرورية ولازمة وإجبارية لأي دارس أو مهتم بكيفية إنشاء وتطوير المواقع الإلكترونية، وأي مجال آخر متعلق بها، لذلك فلا حاجة لإضاعة الوقت في هذا النوع من الجدال، لأنه عبث لا طائل من ورائه.

والاسم HTML هو اختصار Hyper Text Markup Language ، وهي لغة بناء المواقع الإلكترونية، حيث إن عناصرها تمثل الأساس الذي تقوم عليه الصفحة، من عناوين رئيسية، وفقرات نصية، وقوائم، ونماذج إدخال بيانات، وصور، وروابط تشعبية، وملفات صوت، وملفات فيديو، وغيرها، ثم يأتي دور اللغات الأخرى، مثل CSS التي تقوم بتنسيق هذه العناصر، لعرضها بشكل يجذب المستخدمين، ثم لغة Javascript لتقوم بعمل تفاعل بين المستخدم وهذه العناصر، لأداء وظيفة معينة.

وخلاصة القول، أن كل اللغات التي ستستخدمها في بناء موقعك، لا فائدة منها، ولن ترى لها أي إنجاز حقيقي أو أداءً سلساً يجذب الانتباه، بدون إنشاء بناء قوي ومتماسك من HTML ، يستطيع أن يقوم عليه موقعك الإلكتروني.

wailmonir0@protonmail.com

كيف تنشئ صفحة ويب؟؟

لكي تنشئ صفحة ويب، تحتاج إلى محرر نصوص text editor ، لتستخدمه في كتابة كود html الخاص بك، ولكي تحول هذا الكود إلى صفحة ويب، فإنك تحتاج إلى مستعرض أو متصفح ويب web browser ، ليستعرض الصفحة التي قمت بإنشائها.

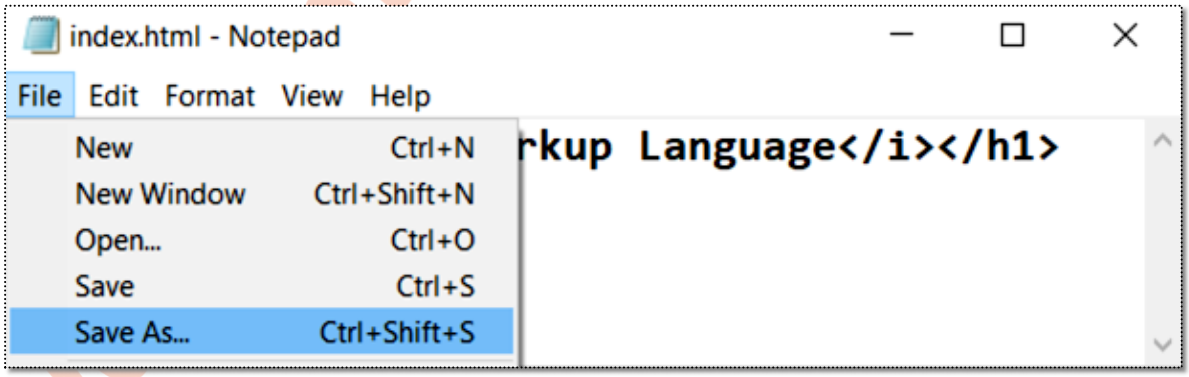
خطوات إنشاء أول صفحة ويب

1- افتح أي محرر نصوص يمكنك الاستعانة به، وليكن أبسطها، وهو برنامج Notepad الذي يأتي ضمن حزمة برامج نظام windows ، واكتب فيه السطر التالي:

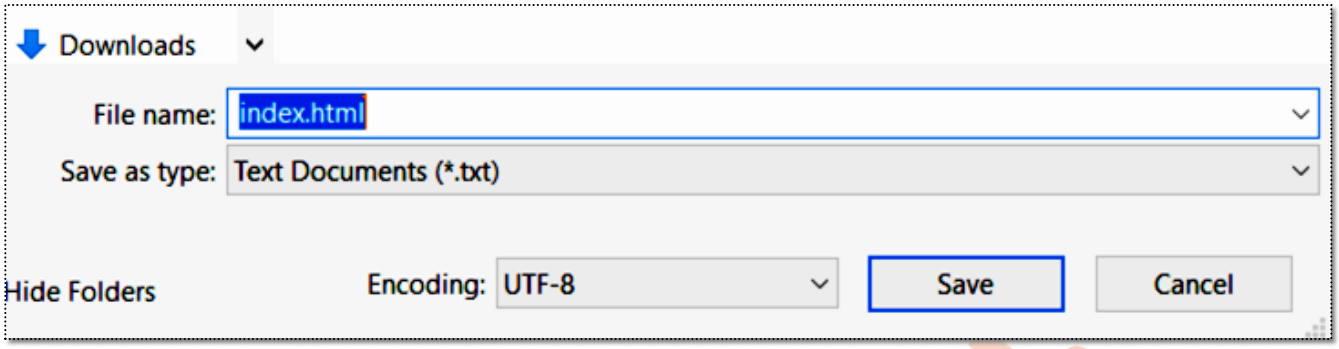
```
<div><i> Hyper Text Markup Language </i></div>
```

2- الآن أصبح هذا السطر من الكود هو source code ، أو كود المصدر الخاص بالصفحة التي نحن بصدد إنشائها.

3- أيا كان نوع محرر النصوص الخاص بك، قم بعمل حفظ save as من قائمة File :

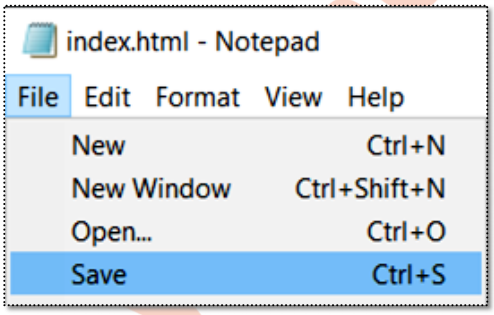


4- اختر المكان الذي ستحفظ فيه الملف، ثم اختر اسماً له وليكن index ، وأضف له امتداد html بعد الاسم، ليصبح اسم الملف index.html ، هو اسم صفحة الويب التي أنشأتها، ثم اضغط save أو حفظ لهذا الملف كما بالشكل التالي:



5- لاحظ أن الملف الذي أنشأته موجود في المكان الذي حفظته به، وهو هنا desktop وسيظهر لك على هيئة أيقونة للمتصفح الافتراضي لديك، كما بالشكل التالي:

6- اضغط على هذا الملف ضغطاً مزدوجاً، ليفتح لك متصفحك الافتراضي صفحة الويب البسيطة التي أنشأتها، ولتظهر لك بالشكل التالي:



7- تذكر جيداً بعد أن تحفظ الكود لأول مرة، فإن أي تعديل سوف تقوم به بعد ذلك على هذا الكود، لن يظهر له أي تأثير ، ما لم تقم بعمل حفظ أو save لهذا التعديل.

8- إذا أجريت أي تعديل على كود المصدر، ثم قمت بحفظه، فيمكنك معاينة تأثير هذا التعديل في المتصفح بالصغط على أيقونة refresh أو مفتاح F5 من لوحة المفاتيح.

الدخول إلى HTML

تعتبر لغة HTML بالنسبة لصفحة الويب بمثابة الهيكل العظمي للإنسان، فهي لغة بناء هيكل structure لصفحات الويب، ويتم ذلك عن طريق إنشاء مجموعة من العناصر elements ، وكل عنصر يتحكم في بنائه مجموعة من الخصائص attributes ، وتتحكم مجموعة من القيم values في سلوك وتأثير كل خاصية على العنصر.

العناصر الدلالية Semantic Elements

بعض عناصر html وإن كانت متشابهة السلوك ظاهرياً، واعتقدت للوهلة الأولى أنه يمكنك استبدال أحدها بآخر، دون أن يكون لذلك أي تأثير على صفحتك أو موقعك، إلا أن هذه العناصر ليست مجرد تنسيق بصري فقط، بل إن استخدامها له دلالة معينة عن طبيعة وظيفتها في بناء الصفحة، حيث يختص كل منها بأداء وظيفة معينة، لبناء جزء معين من الصفحة، كما أن نطق أسماء هذه العناصر قد يحمل دلالة معينة لمن يسمعها، أو دلالة معينة عن طبيعة محتوى هذه العناصر، وهذه المعاني والدلالات تسمى semantics .

لذلك، عندما تبدأ في تصميم صفحتك باستخدام html ، يفضل دائماً أن تلتزم التخصص الموضوع طبقاً للمعايير القياسية، لاستخدام كل عنصر من عناصر html ، للأسباب التالية:

1- عندما تقوم باستخدام عنصر في غير الوظيفة التي أُنشأ من أجلها، مثل أن تقوم باستخدام العنصر الذي يقوم بإنشاء أي جزء من الصفحة بشكل عام، بدلاً من العنصر <main> المخصص لإنشاء الجزء الرئيسي في الصفحة، ثم تحتاج بعد فترة أن تقوم بعمل تعديل على الجزء الرئيسي من الصفحة، هنا سوف يصعب عليك تذكر مكان هذا العنصر وأي عنصر هو من عناصر المنتشرة بالصفحة، وسوف تضطر لاختبار الكثير من العناصر للوقوف على طبيعة سلوكها وتأثيرها، وبالتالي تكتشف بعد فترة وظيفتها التي تقوم بها في الصفحة، حتى تصل للسلوك الذي كنت تطلبه عند إنشائك هذا العنصر، وربما وقتها سوف تستعين بتعليق كتبته لتتأكد من طبيعته وسلوكه، وقد يكون التعليق غير مؤثر في عملية البحث، لأنه تعليق واحد بين تعليقات كثيرة، لذلك فإنك ستوفر على نفسك الكثير من عناء البحث وضياع الوقت، لو اخترت العنصر المناسب، خاصة عندما يكون حجم مشروعك كبيراً، وعدد أسطر الكود فيه كبيراً جداً.

2- إذا كنت تعمل في فريق، أو كنت جزءاً من فريق، وانتهت علاقتك بهذا المشروع سيكون من السهل على الفريق أن يكمل الجزء من العمل الذي كنت تقوم به، إذا اتفقتم جميعاً على نظام عمل واحد ومرجع واحد، وهو اختيار العنصر المناسب، لشغل المكان المناسب، لأداء الوظيفة المناسبة، طبقاً للمعايير القياسية.

3- اعلم أن محركات البحث تهتم عند ترتيب نتائج البحث بطبيعة العنصر الذي تبحث عنه، فيمكنك مثلاً أن تكتب عنواناً رئيسياً للصفحة من خلال عنصر أو عنصر <p> ، ثم تقوم بعمل بعض التنسيقات لهذا العنصر ، لتجعل النص يبدو كعنوان رئيسي للصفحة، وسوف تنجح في ذلك بالتأكيد، وقد يبدو العنوان جذاباً ظاهرياً، إلا أن محركات البحث لن تلتفت لهذا العنصر كعنوان رئيسي للصفحة، بنفس السرعة التي يمكن أن تلتفت بها، إذا اخترت العنصر المعد لكتابة العنوان الرئيسي، وهو <h1> ، لأنك ستجعل محركات البحث تبحث عن عنوان رئيسي للصفحة بين عناصر أخرى كثيرة ليست معدة لهذا الغرض، بعد أن تفشل في إيجاد العنوان الرئيسي في العنصر المعد لإنشائه، وهو كما ذكرنا <h1> .

4- سماع اسم بعض العناصر يفيد أصحاب الاحتياجات الخاصة، في التعرف على جزء الصفحة الذي يعرض أمامهم، كما أن نطق بعض العناصر بطريقة معينة يفيد في التأكيد على معنى معين، أو أهمية نص معين، كما سنرى لاحقاً.

📌 عليك أن تعلم جيداً، أن صفحتك أو موقعك ليسا مجرد شكل جميل يجذب المستخدمين إليه، بل سهولة البحث والعثور على الموقع ومحتوياته، وسهولة التعامل والتفاعل مع الموقع، وسهولة إنجاز المهمة المطلوبة من الموقع، من الأمور التي يجب أن تهتم بها، بقدر أكبر من اهتمامك بمظهر الموقع الخارجي، لأن المستخدم قد يتحمل بعض القصور في الشكل، إذا كان الموقع يؤدي وظيفته بطريقة جيدة، ولكنه لن يتحمل أي قصور في أداء الوظيفة التي يطلبها، حتى وإن كان الموقع هو أجمل مواقع الويب على الإطلاق.

أجزاء عنصر HTML

تتكون عناصر html من عدة أجزاء ثابتة، وقد تختلف بعض العناصر في شكل هذه الأجزاء، وعددها، وهذه الأجزاء كالتالي:

1- وسم بداية start tag ويبدأ بعلامة "<" ثم اسم العنصر ثم علامة ">".
2- وسم نهاية end tag ويبدأ بعلامة "<" ثم شرطة مائلة "/" ثم اسم العنصر ثم علامة ">" ، لتكون هذه هي نهاية العنصر ونهاية تأثيره على محتواه الداخلي.
3- المحتوى content وهو كل ما يتواجد بين وسم البداية ووسم النهاية للعنصر.

والشكل التالي يوضح أجزاء عنصر html :

العناصر الفارغة Empty Elements

هي بعض العناصر التي لا يكون لها وسم نهاية end tag يغلق العنصر، وتكتفي بوسم البداية start tag فقط، وبالتالي لا يكون لهذه العناصر أي محتوى داخلي content ، سواء نص أو أي عناصر أخرى من عناصر html ، حيث لا يوجد ما يمكن أن يحتويها داخله، وقد يطلق عليها "العناصر الفارغة " void elements أو "الوسوم الفارغة" أو empty tags ، ومن أمثلها، عنصر إدراج صورة <img> ، وعنصر الخط الأفقي <hr> وعنصر حقل الإدخال <input> وغيرها.

```
<img  src="images/boat.jpg"  width="20%"><hr>
<input  type="text"  id="usr"  name="user_name"  value=""/>
```

يمكن وضع شرطة مائلة " / " قبل قوس الإغلاق ">" ، كما في العنصر <input> في المثال السابق، ويمكن الاستغناء عنها، كما في العنصرين <img> و <hr> .

ملاحظات على أجزاء عنصر HTML

1- قد يطلق على وسم البداية اسم "وسم الافتتاح" opening tag ، وقد يطلق على وسم النهاية اسم "وسم الإغلاق" closing tag .

2- أسماء العناصر غير حساسة لحالة الأحرف case-insensitive ، أي أن اسم العنصر هو نفسه <DIV> هو نفسه <Div> هو نفسه <dIV> ، لا فرق بينها جميعاً ، ولكن التطبيق والممارسة الأمثل، هي استخدام الأحرف الصغيرة دائماً عند كتابة أسماء العناصر.

3- لا تترك مسافة فارغة في وسم الافتتاح، بين قوس البداية "<" واسم العنصر، وكذلك لا تترك مسافة فارغة في وسم الإغلاق، بين قوس البداية "<" والشرطة المائلة، ولا مشكلة في أن تترك مسافة فارغة أو أكثر بعد اسم العنصر في وسم الافتتاح، أو بعد الشرطة المائلة في وسم الإغلاق، وعليه فهذه الوسوم غير صحيحة < div > و < /div > ، بينما هذه الوسوم صحيحة < div > و < div /> و < div > /> والتطبيق والممارسة الأمثل، هي عدم ترك أي مسافات فارغة بين أجزاء وسم الافتتاح، أو بين أجزاء وسم الإغلاق، إلا إذا احتوى وسم الافتتاح على خصائص للعنصر ، كما سنرى لاحقاً.

4- في حالة العناصر التي تحتاج إلى وسم إغلاق، يتم اعتبار كل ما بعد وسم الافتتاح هو محتوى لهذا العنصر، ويتم تطبيق تأثير وتنسيق العنصر عليه، حتى وإن كان المحتوى عنصراً آخر، إلى أن يتم كتابة وسم نهاية العنصر الأول، وعندها ينتهي تأثير هذا العنصر.

افتح محرر النصوص الخاص بك وقم بكتابة الكود التالي، ثم قم بحفظه كما تعلمنا سابقاً:

```
<i>This italic text contains some <b> bold</b> text</i>
```

اضغط ضغطاً مزدوجاً على أيقونة الملف ليفتح لك المتصفح صفحتك بالشكل التالي:

*This italic text contains some **bold** text*

لاحظ أن النص كله مائل italic ، لأنه داخل محتوى عنصر الخط المائل ، أي واقع بين وسم البداية ، ووسم النهاية ، ولاحظ أن الكلمة bold هي في نفس الوقت نص سميك، لأنها داخل محتوى عنصر الخط السميك، أي واقعة بين وسم البداية ووسم النهاية .

5- في حالة الوسوم المتداخلة nested tags ، فإن الوسم الذي يُفتح أولاً يجب أن يتم إغلاقة أخيراً، ثم الذي فُتح بعده يجب أن يغلق قبله، أي أن العنصر الذي فتح داخل العنصر يجب أن يغلق قبل وسم الإغلاق والعنصر الذي يفتح داخل العنصر يجب أن يغلق قبل وسم الإغلاق .

علاقة الأب والابن Parent-Child relationship

في حالة تداخل العناصر - وهذا ما يحدث دائماً عند استخدام HTML - فإن العنصر الخارجي يسمى الأب parent والعنصر المحتوى داخله يسمى بالعنصر الابن child :

```
<i>Text contains <b><u> underline</u> bold</b> text</i>
```

```
<i>Text contains <b><u> underline</u> bold</b> text</i>
 1                2   3                1        2       3
```

لاحظ ترتيب الأرقام، وأن العنصر التي تم فتحه أولاً أغلق أخيراً ، ليحتوي العناصر التي فتحت داخله، ويكون العنصر الخارجي هو "الأب" parent ، والعنصر الداخلي هو "الابن" child ، أي أن العنصر هو أب للعنصرين و <u> ، لأنهما فتِحا وأغلقا داخله، والعنصر هو أب للعنصر <u> الذي فُتح وأغلق داخل العنصر .

Text contains ***underline*** ***bold*** *text*

لاحظ أن النص كله تم تنسيقه بخط مائل، لأنه واقع بين وسمي البداية والنهاية للعنصر ، والجملة underline bold تم تنسيقها بخط سميك bold ، لأنها داخل وسمي البداية والنهاية للعنصر ، بينما الكلمة underline تم تنسيقها بخط تحتها underline ، لأنها داخل وسمي البداية والنهاية للعنصر <u> .

محررات النصوص الحديثة، تتكفل بكتابة وسم البداية ووسم النهاية، بمجرد أن تفتح وسم البداية وتكتب أول حروف من العنصر، دون أن تستكمل أنت باقي الحروف أو الأجزاء، حيث تقوم هذه المحررات باستكمالها تلقائياً، ولكن عليك أن تتعلم قواعد كتابة الوسوم جيداً، وتجربها بنفسك مرات ومرات، فربما في يوم ما، تجد نفسك مضطراً لاستخدام محرر نصوص بسيط، لا يقوم باستكمال الكود تلقائياً.

أنواع عناصر HTML حسب تدفقها في الصفحة

يوجد أنواع عديدة لعناصر HTML حسب تدفق هذه العناصر داخل صفحة الويب، ولكن هذا الكتاب ليس مجال دراسة هذه الأنواع، وإنما سنتحدث عنها بالتفصيل في المرجع الكامل في css ، ولكن من الضروري أن ندرس أهم نوعين من هذه العناصر، لأهمية دورهما وشدة تأثيرهما في بناء هيكل الصفحة الأساسي، وهما:

Inline Element -

العنصر الخطي، هو عنصر يظهر هو والعنصر الذي قبله، والعنصر الذي بعده، على نفس السطر ، إذا كانوا جميعاً من نفس النوع inline ، والعنصر الخطي ليس له عرض أو ارتفاع أو هامش علوي أو هامش سفلي، ومن أمثلته العناصر و<data> و <input> .

افتح محرر الأكواد الخاص بك واكتب الكود التالي:

```
One <span> Two </span> <span> Three</span> Four
```

قم بحفظ التغييرات وافتح الملف لتظهر لك صفحة الويب كالتالي:

لاحظ أن العناصر ظهرت جميعاً على سطر واحد، وهذا ما يمكن ملاحظته الآن من خصائص هذا النوع من العناصر، وعند دراسة css سوف نتبين طبيعة باقي خصائصه.

مثال آخر:

```
<img  src="blue.jpg"  width="10%">
<img  src="girl.jpg"  width="20%">
<img  src="graff.jpg"  width="15%">
```

اعلم أن الكود السابق للتوضيح فقط، فإن كنت مبتدئاً، فلا تنشغل كثيراً بمفهوم العناصر والخصائص والقيم داخله ، فسوف نشرح كل ذلك بالتفصيل في حينه.

وكما ترى فالصور الثلاث ظهرت بجانب بعضها البعض على نفس السطر.

📌 العنصر <img> ليس عنصراً خطياً خالصاً، وإنما هو inline-block ، وهو نوع وسط، حيث تظهر عناصره على سطر واحد، ولكن لها عرض وارتفاع وهامش من كل الجهات.

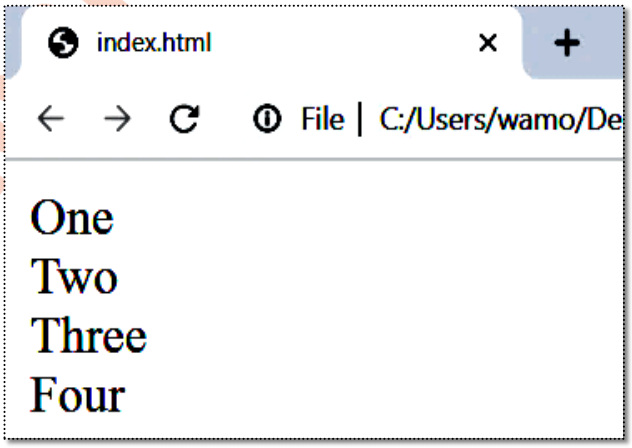
Block Element -

عنصر الكتلة، وسمي بذلك لأنه عنصر له عرض وارتفاع وهامش من كل الجهات، وهو عنصر يظهر على سطر جديد بعد العنصر السابق له، ويتم إضافة العنصر التالي له على سطر جديد، غير السطر الذي عليه هذا العنصر، لأن هذا العنصر يشغل كامل عرض العنصر الأب الذي يحتويه، مهما كان عرض هذا العنصر ، فلا يشاركه أي عنصر آخر السطر الأفقي الموجود عليه، ومن أمثلته العناصر و <p> و<main> و<section> و<article> .

اكتب الكود السابق ولكن استبدل العنصر بالعنصر كالتالي:

```
One <div> Two </div> <div> Three</div> Four
```

قم بحفظ التغييرات وافتح الملف لتظهر لك صفحة الويب كالتالي:



لاحظ أن نفس الكلمات السابقة، خارج وداخل العنصر ، والتي ظهرت على سطر واحد، قد ظهرت كل كلمة منها على سطر جديد، بوضعها خارج وداخل العنصر .

الخصائص (السمات) Attributes

هي مزايا إضافية، يتم تعيينها للعنصر، داخل وسم الافتتاح opening tag ، ويسند لها العديد من القيم للتحكم في العنصر، وتحديد تأثيره واستخداماته.

```
<div  id = "dv_id"  class = "dv_cls" > Web Design </div>
```

الخاصية — اسم الخاصية — قيمة الخاصية

لاحظ أنه قد تم تعيين خاصيتين للعنصر ، الأولى هي الخاصية id ، وأسند لها القيمة "dv_id" ، والثانية هي الخاصية class ، وأسند لها القيمة "dv_cls" .

ملاحظات على الخصائص

- قد لا يتم تعيين أي خاصية للعنصر وقد يتم تعيين خاصية واحدة أو أكثر له.
- تُكتب الخصائص دائماً داخل وسم البداية للعنصر.
- يجب أن يفصل دائماً بين الخاصية، واسم العنصر الذي توجد داخله بمسافة فارغة واحدة على الأقل، وعليه فهذا الوسم خطأ <divid = "dv_id"> .
- كل خاصية لها اسم name ولها قيمة value ، ويفصل دائماً بينهما بعلامة "=".
- يمكن أن يكون بين اسم الخاصية و "=" ، كسر للسطر أو مسافة فارغة أو أكثر أو بدون، ونفس الشيء بين "=" وقيمة الخاصية، وعليه فالكود التالي صحيح:

```
<div  id =  "dv_id" class="attr" name=
"dv_nm"> Web Design </div>
```

- والقيمة يمكن أن تكتب داخل أقواس أو بدون أقواس، والأقواس قد تكون مزدوجة أو مفردة، وعليه، فالكود التالي صحيح:

```
<div  id=dv_id  class='attr' name="dv_nm"> Web Design </div>
```

- يمكنك أن تترك مسافة فارغة واحدة أو أكثر ، بين قيمة الخاصية واسم الخاصية التي تليها، وعلى الرغم من أن كتابة قيمة الخاصية داخل أقواس، تسمح لك بألا تترك أي مسافة فارغة بين قيمة الخاصية واسم الخاصية التي تليها، إلا أن التطبيق والممارسة الأمثل، هي أن

تفصل بين قيمة الخاصية واسم الخاصية التي تليها بمسافة فارغة واحدة على الأقل، ومع ذلك، فالكود التالي صحيح رغم عدم وجود أي مسافة فارغة بين قيمة الخاصية id وهي "dv_id" واسم الخاصية class ، لوجود قيمة الخاصية id بين قوسين.

```
<div  id="dv_id"class="attr"> Web Design </div>
```

ملاحظات على استخدام الأقواس حول قيمة الخاصية

يمكن إسناد قيم الخصائص بدون أقواس حولها مع مراعاة الآتي:

1- أن يكون بين قيمة الخاصية واسم الخاصية التي تليها - إن وجدت - مسافة فارغة.

2- أن يسند للخاصية قيمة واحدة فقط، لأن بعض الخصائص تقبل أكثر من قيمة للخاصية، مثل الخاصية class التي يمكن أن يسند لها أكثر من قيمة، وكل قيمة منهما تمثل العنصر، ولكن يجب وضعها بين أقواس كالتالي:

```
<div  class="clss   attr   clssAttr"> Class Attrbiute </div>
```

قيمة الخاصية class داخل الأقواس، هي في الحقيقة ليست قيمة واحدة، وإنما هي ثلاث قيم لنفس الخاصية، داخل أقواس وبين كل منها مسافة فارغة.

3- إذا كان العنصر يقبل قيمة واحدة، فيجب أن تكون كلمة واحدة أو عدة كلمات متصلة ببعضها، مثل أن يكون بينها "-" أو "_" ، دون مسافة فارغة بين الكلمة والأخرى.

```
<div  di=div_ attr_id> Class Attrbiute </div>
```

4- أما إذا كانت القيمة تتكون من أكثر من كلمة، وبينها مسافة أو أكثر، فيجب أن توضع القيمة داخل قوسين مزدوجين أو منفردين، وإلا فإن الخاصية سوف تعتبر أن أول كلمة هي القيمة، وتهمل باقي الكلمات، كما في المثال التالي بالنسبة للخاصية style :

```
<div  style="color : orange ; background : darkgreen;">
Quotes around the attribute value </div>
```

Quotes around the attribute value

لاحظ أن وضع قيمة الخاصية style داخل أقواس، نتج عنه لون أمامي للنص ، ولون خلفية للعنصر، بينما عدم وضع الأقواس سينتج عنه الشكل التالي:

```
<div  style=color : orange ; background : darkgreen;>
Quotes around the attribute value </div>
```

Quotes around the attribute value

لاحظ فشل الخاصية في التأثير على العنصر لوجود مسافات فارغة بين قيمها وعدم وضعها داخل أقواس وبالتالي، اعتبرت الخاصية أن أول كلمة هي قيمة لها، وهي بالتأكيد قيمة غير صحيحة، فظهر العنصر بقيم خصائصه الافتراضية، وهي لون أمامي (لون النص)أسود، ولون خلفية شفاف.

قم بإلغاء المسافات الفارغة واجعل القيمة كلمة واحدة متصلة بدون أقواس:

```
<div  style=color:orange;background:darkgreen;>
Quotes around the attribute value </div>
```

Quotes around the attribute value

لاحظ رجوع تأثير الخاصية على العنصر، بعد إلغاء المسافات الفارغة بين الكلمات داخل القيمة، بالرغم من عدم وجود القيمة داخل أقواس، لأن القيمة أصبحت كلمة واحدة، يفصل بين أجزائها رموز محددة لها دلالة معينة.

ملاحظات عامة على استخدام الأقواس

- التطبيق والممارسة الأمثل، هي أن تضع دائماً قيم الخصائص داخل أقواس، سواء كانت كلمة واحدة أو عدة كلمات، متصلة أو منفصلة، لتجنب أي أخطاء أنت في غنى عنها.

- لا يوجد فارق بين استخدام الأقواس المفردة single-quotes ، أو الأقواس المزدوجة double-quotes ، طالما كان الاستخدام مرة واحدة لأي منهما حول القيمة.

- إلا أنه في حالة استخدام الأقواس المتداخلة nested quotes ، يجب أن يكون نوع الأقواس الخارجي حول القيمة، مختلفاً عن نوعها داخل القيمة.

فعلى سبيل المثال، إذا كانت الأقواس حول القيمة من النوع المزدوج double-quotes ، فيجب أن تكون داخل القيمة من النوع المفرد single-quotes ، وعكس ذلك صحيح أيضاً، لأن استخدام نفس النوع داخل القيمة وحولها، سينتج عنه خطأ.

انسخ الكود التالي إلى محرر النصوص، وقم بالحفظ ثم افتح الملف:

```
<img  src="mount.jpg"  alt="The magic of 'graffiti' is here">
```

لاحظ أن النص الذي يمثل قيمة الخاصية alt ، والذي يظهر عند حدوث خطأ في مسار الصورة، قد أحاطت به أقواس مزدوجة، لأن داخله أقواس مفردة، نريدها أن تظهر في المتصفح حول الكلمة 'graffiti' ، وهو ما حدث بالفعل كما بالشكل المقابل.

The magic of 'graffiti' is here

فإذا حاولت أن تجعل نوعي الأقواس، حول وداخل القيمة من نفس النوع، فسوف يحدث خطأ، لأن الخاصية ستعتبر أن قيمتها قد انتهت عند فتح القوس الداخلي:

```
<img  src="grf.jpg"  alt="The magic of "graffiti" is here">
```

لاحظ أن قيمة الخاصية لم تكتمل بسبب تشابه الأقواس داخل القيمة وحولها.

The magic of

- أسماء الخصائص غير حساسة لحالة الحروف case-insenstive ، فيمكن أن تكون بأحرف كبيرة أو أحرف صغيرة أو خليط بينهما، فالخاصية class هي نفسها الخاصية CLASS هي نفسها الخاصية clAss، ولكن التطبيق والممارسة الأمثل، أن تكتب أسماء الخصائص دائماً بحروف صغيرة.

- يجب ألا تتكرر نفس الخاصية داخل نفس العنصر أكثر من مرة واحدة، وإن حدث ذلك، فإن العنصر سوف يتأثر بقيمة أول خاصية فقط، وسيتجاهل القيم التالية، وهو ما يمكن أن ينتج عنه بعض الارتباك، لعدم معرفة سبب تجاهل تأثير الخاصية الأخيرة على العنصر، إذا لم تنتبه، أن الخاصية مكررة، وذلك كما في الكود التالي:

```
<div  style="color : blue" class="repeat"  id="div_id"
style="background-color : orange"> Repeated Attribute</div>
```

Repeated Attribute

سوف تشعر بكثير من الارتباك، عندما تجد أن لون الخلفية شفاف وليس orange كما كنت تتوقع، ولكن عندما تكتشف أن الخاصية مكررة وأن التكرار الثاني تم تجاهله، فعليك

عندئذ، أن تضع قيمة الخاصية الموجودة في التكرار الثاني داخل قيمة الخاصية الأولى، وتحذف التكرار الثاني، لتجد أن الكود قد ظهر بالشكل المتوقع :

```
<div class="repeat"  id="div_id" style="color : blue; background-color :
orange ">Repeated Attribute</div>
```

Repeated Attribute

ما سبق ينطبق على الخاصية التي تقبل قيم متعددة، فتوضع جميع القيم كقيمة لأول تكرار للخاصية، أما الخاصية التي تقبل قيمة وحيدة، فيجب أن تحذف أي تكرار لها، وتبقى على خاصية واحدة فقط.

الخصائص المنطقية Boolean Attributes

ذكرنا من قبل، أن الخاصية تتكون من اسم وقيمة يفصل بينهما علامة "=" ، إلا أنه يوجد نوع آخر من الخصائص، وهو boolean attributes أو الخصائص المنطقية، حيث يكون اسم الخاصية هو نفسه قيمتها، بدون علامة "=" أو أي قيمة بعدها، وboolean تعني أنه في حالة وجود اسم الخاصية، تكون قيمتها true ، وذلك يعني وجود تأثير لهذه الخاصية على العنصر ، وفي حالة عدم وجوده فهذا يعني أن قيمتها تكون false ، أي لا تأثير لها على العنصر.

```
<input  type="text"   autofocus>
<input  type="password"   disabled>
```

autofocus disabled

لاحظ أن تعيين الخاصية autofocus للعنصر الأول، نقل إليه تركيز لوحة المفاتيح ووجود مؤشر الكتابة داخله، في حين أن تعيين الخاصية disabled للعنصر الثاني، جعله غير فعال وغير متاح للتعامل من قبل المستخدم، بالرغم من عدم إسناد أي قيم للخاصيتين، ولكن مجرد تعيينهما للعنصر، وذكر اسمهما فقط داخله، فهذا يعني أن قيمتهما هي true .

الأنماط التي يستخدمها المتصفح في العرض

في بدايات الانترنت أو الويب، كانت الصفحات تكتب بنسختين، نسخة للعرض على متصفح Microsoft internet explorer ، ونسخة أخرى للعرض على متصفح Netscape navigator ، وبعد أن وضعت المعايير القياسية للويب، كان من الصعب إعادة بناء المواقع القديمة وفقاً لهذه المعايير، لذلك تم وضع أنماط لعرض صفحات الويب، حتى يمكن عرض الصفحات والمواقع القديمة بشكل مناسب، وهي:

1- نمط المعايير الكاملة Full Standards Mode

وهو نمط عرض يتوافق مع المواصفات القياسية لكل عناصر html وخصائص css .

2- نمط المعايير شبه الكاملة Almost Standards Mode

وهو نمط عرض يتوافق مع معظم المواصفات القياسية لعناصر html وخصائص css ، ما عدا القليل من المواصفات التي تعرض عن طريق النمط الثالث.

3- نمط الملاءمات Quirks Mode

وهو نمط عرض يتوافق مع المتصفح Microsof internet explorer 5 والمتصفح Netscape navigator 4 .

كيف يحدد المتصفح النمط الواجب استخدامه؟؟

يتم تحديد النمط المستخدم في عرض صفحة الويب، عن طريق الوسم الموجود في بداية صفحة html ، فاستخدام الوسم <!DOCTYPE html> يعني أن المتصفح سيعمل في نمط المعايير الكاملة full standards mode ، والذي يتوافق مع معايير HTML5 ، وعدم استخدام هذا الوسم في بداية الصفحة، سيجعل المتصفح يستخدم نمط الملاءمات Quirks mode .

wailmonir0@protonmail.com

هيكل كود صفحة HTML

يتكون كود صفحة html من عدة أجزاء أو أقسام، وهذه الأجزاء مقسمة حسب وظيفة كل جزء في صفحة الويب، والكود التالي هو أبسط أشكال كود صفحة html ، وسوف نتناوله بالشرح، لتوضيح طبيعة وتأثير هذه الأجزاء في بناء صفحة الويب:

```
<!DOCTYPE html>
<html>
    <head>
            <meta charset="utf-8">
            <title>Learn HTML</title>
    </head>
    <body>
            <p>First HTML Page</p>
    </body>
</html>
```

<!DOCTYPE html>

وسم البداية في صفحة html ، وهو الذي يحدد أن النمط أو الوضع الذي سيستخدمه المتصفح في عرض الصفحة، هو وضع المعايير الكاملة full standards mode ، وهو النمط الذي يتوافق مع المعايير القياسية لـ HTML5 .

وهذا الوسم مثل كل وسوم html ، غير حساس لحالة الأحرف case-insenstive أي أن الوسم <!doctype html> صحيح، وسيعرض وضع المعايير الكاملة .

<html>....</html>

هو العنصر الأب parent لكل عناصر الصفحة، بمعنى أن كل عناصر الصفحة هم أبناء children لهذا العنصر، أي أن كل عناصر الصفحة يجب أن تقع داخل وسم البداية ووسم النهاية لهذا العنصر، فتكون محتوى أو content لهذا العنصر.

ويوجد خاصيتان، يمكن أن يتم تعيينهما لهذا العنصر :

lang

وهي خاصية تحدد لغة النص الثابت داخل الصفحة، أو النص القابل للإدخال والتعديل، وتحديد نوع اللغة يعتبر هاماً لعمل بعض الخصائص الأخرى، وقيمة الخاصية تتكون من حرفين أو ثلاثة، وهذه الحروف تعبر عن اللغة المستخدمة وسوف نتناولها بالشرح في فصل الخصائص العامة.

xmlns

وهي خاصية تحدد ملف xml المرتبط بصفحة html ، وهي خاصية اختيارية في HTML5 ، أي أنه لا مشكلة إذا لم يتم تعيينها، لأن قيمتها هي قيمة افتراضية، وهي كالتالي:

```
<html  lang="en"  xmlns="http: //www.w3.org/1999/xhtml">
```

<head>....</head>

هو عنصر يحتوي عناصر metadata ، وهي عناصر إعدادات الصفحة التي يقرؤها المتصفح تلقائياً ، وهي عناصر ليست مقروءة أو مرئية للمستخدم في جسم الصفحة، وهي العناصر <title> و <script> و <base> و <link> و <style> و <meta> .

سوف تلاحظ لاحقاً، أن العنصر <title> مقرؤ للمستخدم، ولكن ليس في جسم الصفحة، وإنما في شريط عنوان الصفحة، أو تبويب الصفحة.

<body>....</body>

هو العنصر الأب، الذي يحتوي كل العناصر المادية المرئية في جسم الصفحة، أي كل ما يراه المستخدم على الشاشة، كما أنه يحتوي العناصر المادية المخفاة hidden elements ، التي لا يراها المستخدم، والتي يتم إخفاؤها لأسباب تتعلق بالأمان، أو لأغراض برمجية.

wailmonir0@protonmail.com

المسافات الفارغة وانكسار الأسطر

Whitespaces and Line Breaks

المسافات الفارغة، ومسافة tab ، وانكسار الأسطر، والالتفاف إلى الأسطر الجديدة، وأي إعدادات أو تنسيقات، تستخدم في محرر النصوص، لنكتب بها كود المصدر، لا تأثير لها على تنسيق الصفحة في المتصفح، إلا في حالات قليلة جداً، فأياً كان عدد المسافات الفارغة أو الأسطر الجديدة التي ستقوم بإدخالها بين أي كلمة والكلمة التي تليها، فسوف تظهر الكلمتان على السطر الأول، وبينهما مسافة فارغة واحدة فقط.

اكتب الكود التالي في محرر النصوص الخاص بك:

```
<!DOCTYPE html>
<html>
 <body>
  My                    name
          is      Wail
       Monir
 </body>
</html>
```

قم بحفظ التغييرات، ثم افتح الصفحة، ليظهر لك الكود بالشكل التالي:

لاحظ اختفاء كل المسافات الفارغة والأسطر الجديدة، التي كانت موجودة في كود المصدر source code ، بين أي كلمة والتي تليها، وتم الاستعاضة عنها جميعاً بمسافة فارغة واحدة فقط، بين الكلمة والأخرى.

الحروف الخاصة HTML Entities

بعض الرموز، لا يمكن لها أن تظهر في المتصفح بشكلها المتعارف عليه، إذا كتبت بنفس هيئتها في محرر النصوص، لأن بعضها رموز محفوظة لأكواد html ، وبعضها رموز يصعب كتابتها بشكلها الرمزي الذي توجد عليه، لذلك تستخدم html طريقتين عند كتابة هذه الرموز، وهما تركيبة خاصة من الحروف والرموز، وتركيبة أخرى من الأرقام والرموز، كما بالجدول التالي:

Symbol	Letters	Numbers
space		
<	<	<
>	>	>
&	&	&
"	"	"
'	'	'
¢	¢	¢
£	£	£
¥	¥	¥
€	€	€
©	©	©
®	®	®
-		

قم بكتابة الكود التالي في محرر النصوص الخاص بك:

```
<!DOCTYPE html>
<html>
 <body>
 &quot; Personal Info &#34;<br>
    Name:  &#39;Wail Monir&#39;<br>
    Age:   &lt; 47&gt;<br>
    Salary: 500&euro;<br>
 &copy;Copy rights reserved &amp; &#174;registred <br>
```

```
</body>
</html>
```

قم بحفظ التغييرات وافتح الصفحة، وانظر كيف تم كتابة الحروف الخاصة عن طريق ترجمة تركيبتي الحروف والرموز، والأرقام والرموز، كما بالشكل التالي:

التعليقات Comments

ترجع أهمية التعليقات إلى سهولة ترتيب العمل على الكود، ولتوضيح وظيفة كود معين، أو التنبيه على خطوة معينة يجب إتمامها، أو مشكلة تواجهك وتبحث عن حل لها، أو ملاحظة موجهة لمن يتشارك العمل معك على نفس الكود، والتعليقات مقرؤة ضمن كود المصدر الخاص بالصفحة، ولكن يتم تجاهلها عند العرض في المتصفح.

والتعليقات في html تكتب بين رمزين، رمز الافتتاح "<!--" ، ورمز الإغلاق "-->" ، وبينهما الملاحظة المراد كتابتها على سطر واحد أو عدة أسطر، والتي سيتجاهلها المتصفح.

```
<!-- This part about personal information-->
<div>Name: Wail Monir</div>
<div>Age: 47</div>
<div>Job: Web developer </div>
```

لاحظ أن سطر التعليق الموجود في أول سطر في كود المصدر، غير موجود عند العرض في المتصفح.

wailmonir0@protonmail.com

HTML عناصر

HTML Elements

wailmonir0@protonmail.com

wailmonir0@protonmail.com

تقسيم عناصر HTML

كل عنصر من عناصر HTML له دوره في بناء صفحة الويب، وقد تتشابه أدوار بعض العناصر، وبناءً على تشابه هذه العناصر في دورها ووظيفتها، قمت بتقسيمها بما رأيت أنه يتناسب مع وظيفتها وسلوكها، إلا أن هذا التقسيم قد يختلف قليلاً في موقع عن آخر ، وكلها صحيحة، لأنك قد تجد أكثر من وظيفة للعنصر، المهم أن تعلم أن التقسيم بغرض الترتيب والتوضيح، واختلاف القسم الذي يوجد تحته العنصر، لا يؤثر مطلقاً في طبيعة عمله أو سلوكه، وكثير من المحاضرين، لا يتطرقون في الأصل إلى تقسيم العناصر في دروسهم، لأن التقسيم غير مؤثر في طبيعة عمل وسلوك العنصر، فلا تنزعج، إن اختلف تقسيم عنصر ما في مرجع عن مرجع آخر، أو رأيت أن عنصراً ما يقع تحت قسم آخر، غير القسم الذي درسته في هذا الكتاب أو أي كتاب أو مرجع آخر، المهم أن تستوعب، طبيعة عمل، وتأثير العناصر المختلفة في بناء الصفحة.

وقد اتبعت منهجاً معيناً في شرح عناصر HTML ، حيث سأبدأ بشرح طبيعة عمل ووظيفة العنصر، ثم سأعرفك بكل الخصائص التي يمكن أن يتم تعيينها له مع الشرح، ثم نتناول كل القيم التي يتم إسنادها لكل خاصية، بالشرح والكود وصورة المخرج النهائي للكود، وكل ذلك سوف يقرب إليك طبيعة عمل العنصر، وتأثير خصائصه، وتأثير اختلاف قيم هذه الخصائص.

واعلم أن بعض الخصائص أو القيم أو الجزئيات، ستكون مشتركة بين العناصر، لذلك سأشرحها باختصار في المواضع المتكررة، وسأحيلك إلى الشرح المفصل في ملحق 1 من هذا الكتاب، أو أي موضع آخر شرحت فيه هذه الجزئية بشيء من التفصيل، حتى يسهل عليك فهم واستيعاب طريقة عمل هذه الخاصية أو الجزئية، دون تكرار ممل.

وأخيراً سأوضح لك التنسيق الافتراضي للعنصر وهو عبارة عن كود css ، وهو خارج نطاق هذا الكتاب، فلا تهتم كثيراً إذا لم تفهم محتواه أو طبيعة عمله، لأنك سوف تعود إليه بالتأكيد عندما تدرس css ، فأردتك أن تتعرف عليه حتى تألفه وتعتاده.

wailmonir0@protonmail.com

عناصر الكتابة وتنسيق النصوص

هي العناصر التي تقوم بتنسيق نص معين تنسيقاً خاصاً.

<abbr>....</abbr>

عنصر يقوم بعرض نص على هيئة تلميح منبثق أو tooltip ، ليوضح معنى نص مختصر بين محتوى العنصر <abbr> ، ويظهر هذا التلميح عند مرور الفأرة فوق المصطلح أو الاسم المختصر، ويتم تعيين خاصية واحدة لهذا العنصر :

title

خاصية تحدد نص التلميح، الذي سيظهر عند مرور الفأرة فوق المصطلح المختصر.

```
<abbr title="North Atalntic Treaty Organization"> NATO</abbr>
```

NATO

North Atalntic Treaty Organization

لاحظ ظهور خط منقط تحت النص المختصر، كإشارة إلى أنه سيظهر منه التلميح عند مرور الفأرة فوقه، ويمكن أن تكتب التلميح على سطرين أو أكثر، بأن تقوم بكسر السطر في قيمة الخاصية title ، وسوف ينتقل كسر السطر إلى المتصفح.

```
<abbr title="North Atalntic Treaty
        Organization"> NATO</abbr>
```

NATO

North Atalntic Treaty Organization

التنسيق الافتراضي:

```
abbr {
        display: inline;
       }
```

....

عنصر مخصص في العادة، لإدخال فقرات نصية إلى الصفحة، وإن كان بإمكانه أن يحتوي أي عنصر من عناصر html ، مثل الصور والنماذج والقوائم والجداول وغيرها، ويمكن للصفحة أن تحتوي أكثر من عنصر .

```
<div>
    Lorem ipsum
    <p style="width:400px;">
        Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do
        eiusmod tempor incididunt ut labore et dolore magna aliqua.
    </p>
    Lorem ipsum
</div>
```

Lorem ipsum

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Lorem ipsum

والعنصر هو عنصر كتلة block element ، أي ينتقل إلى سطر جديد بعد العنصر الذي قبله، وينقل العنصر بعده إلى سطر جديد، ويكون بين العنصر والعنصر الذي قبله هامش علوي بقيمة 1em (16px) ، كما يكون بينه وبين العنصر الذي بعده هامش سفلي بنفس القيمة.

التنسيق الافتراضي:

```
p {
    display:  block;
    margin: 1em   0em;
   }
```

خاصية margin الموضحة بالكود السابق تتكون من قيمتين، القيمة الأولى تمثل الهامشين العلوي والسفلي للعنصر ، والقيمة الثانية تمثل الهامشين الأيمن والأيسر.

إلغاء وسم العنصر الأب تلقائياً Tag Omission

بعض عناصر html من النوع block element قد يتم إلغاء وسم البداية له opening tag وكأن العنصر لم يتم إنشاؤه، وبعضها قد لا يحتاج إلى وسم إغلاق closing tag ، والحالتان تكونان بفعل إنشاء عنصر ابن child من النوع block element داخل العنصر الأب، ومن هذه العناصر:

العنصر <p>

إذا أنشأ داخله عنصر ابن وكان هذا الابن عنصر <p> أو أي عنصر من هذه العناصر:

<address>,<aside>,<article>,<blockqoute>, ,<dl>,<ul>,<fieldset>, <footer>,<form>,<h1>..<h6>,<header>,<hr>,<main>,<menu>,<nav>,<ol>, <p>,<pre>,<section>,<table>

فتعتبر بداية إنشاء العنصر الابن الداخلي، هي إغلاق للعنصر الأب الخارجي، حتى وإن أُغلق العنصر الأب بعد ذلك - حسب التسلسل الذي سبق دراسته - بعد إغلاق العنصر الابن.

```
<p style="width:400px; background-color:tomato"> Lorem ipsum
   <div style="width:400px; background-color: yellow">
    Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed
    do eiusmod tempor incididunt ut labore et dolore magna aliqua.
    </div>
    Lorem ipsum
</p>
```

Lorem ipsum

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Lorem ipsum

لاحظ أن الجملة الأخيرة، الواقعة بين وسم نهاية العنصر الابن ووسم نهاية العنصر الأب <p> ، لم يتم تنسيقها بنفس تنسيق محتوى العنصر <p> - خلاف المتوقع - وظهرت بدون هامش وبدون لون خلفية، بالرغم من أنها تقع داخل العنصر <p> ، وذلك لأن العنصر <p> أغلق بفعل إنشاء العنصر الابن ، أي أن وسم بداية العنصر الابن قد أغلق العنصر

الأب، وبالتالي أصبحت هذه الجملة تقع خارجه، وإن أغلق العنصر الأب بعدها.

قم باستبدال العنصر <p> بالعنصر <main> ، أو أي عنصر كتلة آخر، ولاحظ الفرق بين العنصرين :

```
<main style="width:400px; background-color:tomato">Lorem ipsum
    <div style="width:400px; background-color: yellow">
     Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed
     do eiusmod tempor incididunt ut labore et dolore magna aliqua.
     </div>
     Lorem ipsum
</main>
```

Lorem ipsum
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.
Lorem ipsum

لاحظ أن الجملة الأخيرة تم تنسيقها – كما هو متوقع – بنفس تنسيق العنصر <main> الأب، لأنها تقع داخل وسمي البداية والنهاية له، فكان من الطبيعي أن تتأثر بسلوكه، حيث إن معظم عناصر block elements لا يمتلك ميزة tag omission مثل العنصر <p> .

وسوف نتناول هذه الميزة بالشرح في العديد من العناصر التي سندرسها لاحقاً.

...

عنصر مخصص لاحتواء النصوص، وهو عنصر خطي inline element ، ليس له عرض أو ارتفاع، وتكون أبعاده هي أبعاد المحتوى داخله، وهو عنصر لا ينتقل للسطر التالي بعد العنصر الذي قبله، إذا كان هذا العنصر عنصراً خطياً مثله، ولا ينقل العنصر بعده للسطر التالي، إذا كان كذلك عنصراً خطياً مثله، أي أن العنصر قبله والعنصر بعده يكونان معه على نفس السطر ، طالما كانا عناصر خطية مثله.

```
<span>Before image</span>
<img src="images/colors.jpg" width="200" height="100">
<span>After image</span>
```

لاحظ أن العنصر <img> وعنصر قبله، وعنصر بعده، جميعهم ظهروا على سطر واحد، لأنهم جميعاً عناصر خطية.

التنسيق الافتراضي:

```
span {
        display: inline;
       }
```

لاحظ أن الصورة تظهر على نفس السطر، إلا أن لها عرضاً وارتفاعاً، وذلك لأن عرض وارتفاع الصور خصائص أصلية فيها، فنحن لا نقوم بإنشاء الصور من العدم ومنحها عرضاً وارتفاعاً مثل العنصر على سبيل المثال، وإنما الصورة موجودة بكامل أبعادها، نحن فقط نتحكم في العرض والارتفاع الأصليين بالتكبير والتصغير.

<data>...</data>

عنصر خطي، يشبه إلى حد كبير العنصر ، حيث يحتوي نصاً بين وسمي البداية والنهاية، ويعرض هذا النص في المتصفح بشكل عادي، بالإضافة إلى خاصية أخرى لهذا العنصر، لا تعرض في المتصفح ولكن يتم قراءتها آلياً، وهي الخاصية value .

value

خاصية يتم قراءتها آلياً لتعطي العنصر قيمة جديدة، بجانب النص المقروء في المتصفح، فيكون العنصر عبارة عن، نص مكتوب وقيمة مسموعة.

```
<h4>Drinks list</h4>
<data value="20">Coffee</data><br>
<data value="10">Tea</data><br>
<data value="15">Juice</data><br>
<data value="25">Milk</data>
```

Drinks list

Coffee
Tea
Juice
Milk

لاحظ أن العنصر <data> ليس به ما يميزه ظاهرياً، ولكن ما يميزه سمعياً هو الخاصية value ، التي تُقرأ آلياً لتربط النص بين وسمي البداية والنهاية للعنصر، بقيمة معينة يتم قراءتها، مثل سعر منتج أو الكمية المخزنة، وغير ذلك من القيم.

لاحظ أن محتوى العناصر ظهر تحت بعضه، وليس بجانب بعضه، بالرغم من أن العنصر <data> هو عنصر خطي، مثل العنصر ، وذلك لأني استخدمت العنصر
 بعد وسم الإغلاق للعنصر <data> ، فقام بكسر السطر والانتقال لسطر جديد، لتظهر العناصر على شكل قائمة.

....

عنصر يقوم بتنسيق نص معين بخط سميك **bold** لجذب الانتباه إليه، ولكن هذا العنصر لا يعطي النص المنسق أي أهمية عن باقي النص، وإنما فقط للفت النظر إليه، ومع وجود **css** ، لا ينصح باستخدام هذا العنصر في تنسيق النصوص التي ترغب في أن تظهرها بخط سميك، لوجود الخاصية **font-weight** في **css** ، والتي تقوم بنفس العمل، وتتيح للمصمم اختيارات أكثر تنوعاً وأكثر دقة.

```
The b tag shows the text in <b>bold</b>.
```

The b tag shows the text in **bold**.

التنسيق الافتراضي:

```
b {
    display: inline;
    font-weight:  bold;
   }
```

....

عنصر يقوم بتنسيق نص معين بخط سميك، وهذا النص يكون له أهمية أكثر من باقي أجزاء النص، وهذا هو وجه الاختلاف بينه وبين العنصر <b> .

```
This text contains some <strong>important</strong> parts.
```

This text contains some **important** parts.

التنسيق الافتراضي:

```
strong {
        display: inline;
        font-weight:  bold;
       }
```

لاحظ أن العنصر له نفس سلوك وتأثير العنصر <b> ، ولعلك تسأل نفسك، لماذا هذا التعدد في العناصر ، بالرغم من تشابه وتطابق التنسيق والشكل؟
وسوف أجيبك على هذا السؤال في آخر هذا الجزء، لتكون الإجابة شاملة هذه العناصر ، أو أي عناصر أخرى متشابهة.

....

عنصر يقوم بتنسيق نص معين بخط مائل، للخروج من وتيرة النص الأصلي، فهذا النص المائل يمكن أن يكون تصنيفاً معيناً (مثل رجل، امرأة، طفل)، أو مصطلحات فنية، أو ترجمة من لغة أخرى، أو شخص يحدث نفسه بفكرة ما، أو قراءة صوتية وغيرها.

```
<i>HTML</i> is used to build web pages structure.
```

HTML is used to build web pages structure.

التنسيق الافتراضي:

```
i {
    display: inline;
    font-style:  italic;
    }
```

....

عنصر يقوم بتنسيق النص بخط مائل، للتأكيد والتشديد على نقاط مهمة في هذا النص المائل، تميزه عن باقي النص العادي.

```
We have to <em> stay at home </em> to avoid covid-19.
```

We have to *stay at home* to avoid covid-19.

التنسيق الافتراضي:

```
em {
    display: inline;
    font-style:  italic;
    }
```

<cite>....</cite>

عنصر يقوم بتنسيق النص بخط مائل، وهو مخصص لتنسيق العناوين داخل نص، مثل عناوين الكتب والأفلام والمسرحيات والألعاب واللوحات الفنية والمقالات، والأعمال الإبداعية عامة، لتمييزها عن باقي النص (أسماء الأشخاص ليست عناوين أعمال إبداعية ليتم تمييزها).

```
The <cite>Mona Lisa</cite> is painted by Leonardo da Vinci.
```

The *Mona Lisa* is painted by Leonardo da Vinci.

التنسيق الافتراضي:

```
cite {
    display: inline;
    font-style:  italic;
    }
```

<var>....</var>

عنصر يقوم بتنسيق النص بخط مائل، وهو مخصص لتنسيق أسماء المتغيرات في الرياضيات ولغات البرمجة.

```
The area of the rectangle = <var>width</var> * <var>length</var>.
```

The area of the rectangle = *width* * *length*.

التنسيق الافتراضي:

```
var {
    display: inline;
    font-style:  italic;
    }
```

<dfn>....</dfn>

عنصر يقوم بتنسيق النص بخط مائل، وهو مخصص لتنسيق مصطلح معين، سوف يتم شرحه في سياق النص الحالي.

```
The <dfn>CSS</dfn> is used to format HTML elements.
```

The *CSS* is used to format HTML elements.

ويتم تعيين خاصية واحدة للعنصر <dfn> :

title

خاصية تقوم بإظهار تلميح أو tooltip ، عند مرور الفأرة فوق المصطلح.

```
The <dfn title="Cascading Style Sheets">CSS</dfn> is used .....
```

The *CSS* is used .....

Cascading Style Sheets

لاحظ أن العنصر <dfn> يشبه العنصر <abbr> في سلوكه، ولكنه يختلف عنه في أن النص بين وسمي البداية والنهاية للعنصر يكون مائلاً، كما لا يوجد خط منقط تحت المصطلح أو الاختصار الذي سيتم شرحه.

CSS

Cascading Style Sheets

كذلك يمكن كتابة التلميح على أكثر من سطر ، عن طريق كسر السطر في قيمة الخاصية title ، وسيظهر ذلك في المتصفح.

```
<dfn title="Cascading Style
      Sheets">CSS</dfn>
```

التنسيق الافتراضي:

```
dfn {
      display: inline;
      font-style:  italic;
      }
```

<address>....</address>

عنصر يقوم بتنسيق النص بخط مائل، وهو مخصص لتنسيق معلومات الاتصال لشخص أو مكان معين، وقد يتضمن الاسم والعنوان ورقم الهاتف والبريد الإلكتروني ومواقع التواصل الاجتماعي، وغيرها من وسائل التواصل.

```
Contact us:
<address>Name: Tourism and travel Co.</address>
<address>Address: Egypt, Alexandria. </address>
<address>Phone: 0203441552989.</address>
<address>Email: www.tourtravel.com</address>
```

Contact us:
Name: Tourism and travel Co.
Address: Egypt, Alexandria.
Phone: 0203441552989.
Email: www.tourtravel.com

وبما أن العنصر <address> هو عنصر كتلة block element ، فإنه يشغل السطر بالكامل عند إنشائه، وينقل العنصر بعده إلى سطر جديد.

وبناء على وظيفته ، فإن أنسب عنصر يمكن أن يحتوي العنصر <address> داخله هو العنصر <footer> ، المخصص لاحتواء معلومات الاتصال بجهة معينة، والذي سندرسه في الجزء التالي من هذا الكتاب.

التنسيق الافتراضي:

```
address {
        display:  block;
        font-style:  italic;
        }
```

<u>....</u>

عنصر يقوم بوضع خط مصمت تحت نص معين، وهذ النص قد يكون خطأً في التهجئة، أو أسماء بلغة أخرى، لتمييز هذا الجزء من النص عن باقي النص.

```
Make the wrong words <u>underline ones</u> to find it easily.
```

Make the wrong words <u>underline ones</u> to find it easily.

التنسيق الافتراضي:

```
u {
    display: inline;
    text-decoration:  underline;
    }
```

يفضل عدم استخدام هذا العنصر أو هذا التنسيق مع النصوص، حتى لا يحدث خلط بين النص المنسق بخط تحته، وبين الروابط التشعبية، التي دائماً يكون تنسيقها underline.

wailmonir0@protonmail.com

<ins>....</ins>

عنصر يقوم بوضع خط مصمت تحت النص الذي تم تعديله إو إدخاله.

```
<ins> <p> This paragrah is updated today</p></ins>
```

This paragrah is updated today

ويتم تعيين خاصيتين للعنصر <ins> :

cite

وهي خاصية تحدد الموقع URL أو الصفحة، التي تحتوي توضيحاً للتعديل أو النص الذي تم إدخاله، مثل محضر اجتماع أو خطوات متابعة مشكلة ما.

datetime

وهي خاصية تحدد تاريخ إجراء التعديل على النص، أو تاريخ محضر الاجتماع، وتقبل قيمتها على هيئة نص، يحتوي تاريخاً متوافقاً مع معايير العنصر <time> ص 52 .

```
<ins cite="www.company.com"  datetime="2021-03-16">
<p> This paragrah is updated today</p>
</ins>
```

التنسيق الافتراضي:

```
ins {
    display: inline;
    text-decoration:  underline;
    }
```

~~....~~

عنصر يقوم بتنسيق النص، بخط يمر في وسطه، وهذا النص يدل على معلومة تم حذفها، إلا أن لها علاقة بباقي أجزاء النص الأصلي.

```
The old color was <del>blue</del> and the new one is red.
```

The old color was ~~blue~~ and the new one is red.

ويتم تعيين خاصيتين للعنصر <del> :

cite

وهي خاصية تحدد الموقع URL أو الصفحة، التي تحتوي توضيحاً لأسباب حذف هذا النص.

datetime

وهي خاصية تحدد تاريخ حذف النص، وتقبل قيمتها على هيئة نص، يحتوي تاريخاً متوافقاً مع معايير العنصر <time> ص 52 .

```
The old color was
        <del cite="www.clr.com" datetime="2020-05-31">blue</del>
 and the new one is red.
```

التنسيق الافتراضي:

```
del{
    display: inline;
    text-decoration:  line-through;
    }
```

<s>....</s>

عنصر يقوم بتنسيق نص معين، بخط يمر في وسطه، لأن هذا النص أصبح غير دقيق، أو غير مستخدم، أو غير ذي صلة بالنص الأصلي في الوقت الحالي.

```
I though the job salary is <s>500$</s> but, it was 300$ only.
```

I though this job salary is ~~500$~~ but I was wrong, it was 300$ only.

التنسيق الافتراضي:

```
s {
    display: inline;
    text-decoration:  line-through;
    }
```

لا تخلط بين العنصر <del> والعنصر <s> ، فالعنصر <del> مخصص للنص المحذوف أو المعدل، بينما العنصر <s> مخصص للنص الغير دقيق أو الغير مستخدم.

<code>....</code>

عنصر مخصص لتنسيق كود برمجي، سطر أو جملة في نص، يستخدم ككود في لغة برمجة أو تعليمات من الحاسب، ويتم تنسيقه بنوع خط من العائلة monospace ، الذي يشبه خط الآلة الكاتبة.

```
To resize text use <code> font-size </code> property.
```

To resize text use `font-size` property.

التنسيق الافتراضي:

```
code {
        display: inline;
        font-family:  monospace;
        }
```

<kbd>....</kbd>

عنصر مخصص لتنسيق نص معين، بخط من نوع خط الآلة الكاتبة، ويكون هذا النص قد تم إدخاله عن طريق لوحة المفاتيح أو الميكروفون، أو أي وسيلة إدخال input أخرى، ويكون تنسيق الخط من نوع العائلة monospace .

```
Enter the author name: <kbd>Wail Monir</kbd>
```

Enter the author name: Wail Monir

التنسيق الافتراضي:

```
kbd {
        display: inline;
        font-family:  monospace;
       }
```

<samp>....</samp>

عنصر مخصص لتنسيق مخرجات output كود برمجي، بخط من نوع خط الآلة الكاتبة، ويتم تنسيق النص بخط من نوع monospace .

```
When I tried to delete some files the computer asked me
<samp>are you sure</samp>.
```

When I tried to delete some files the computer asked me are you sure.

التنسيق الافتراضي:

```
samp {
        display: inline;
        font-family:  monospace;
        }
```

<pre>....</pre>

عنصر يقوم بتنسيق النص في المتصفح، من حيث المسافات وانكسار الأسطر، بنفس تنسيق كود المصدر source code ، أي بنفس المسافات الفارغة والأسطر الجديدة التي تستخدم في محرر الأكواد، ويكون نوع الخط من العائلة monospace .

```
<pre>
    My      name is
        Wail        Monir
</pre>
```

```
    My        name is
          Wail            Monir
```

التنسيق الافتراضي:

```
pre {
        display: block;
        font-family:  monospace;
        margin-top:  1em;
        margin-bottom:  1em;
        white-space:  pre;
      }
```

<q>....</q>

عنصر مخصص لاحتواء الاقتباسات النصية الخطية الصغيرة داخل نص أكبر، ويكون النص المقتبس بين قوسي اقتباس، وهذا الاقتباس غير مخصص للفقرات النصية الكبيرة.

```
<div style="width:400px">Lorem ipsum dolor bhas jkp sit amet, <q>consectetur adipiscing</q> elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad <q>minim veniam</q> quis nostrud exercitatn esse cillum dolore eu fugiat nulla pariatur.</div>
```

Lorem ipsum dolor bhas jkp sit amet, "consectetur adipiscing" elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad "minim veniam" quis nostrud exercitatn esse cillum dolore eu fugiat nulla pariatur.

ويتم تعيين خاصية واحدة للعنصر <q> :

cite

خاصية تحدد مصدر النص، أو الرسالة المطلوب إيصالها من هذا الاقتباس، وقد تشير إلى مصدر معين، يفسر هذا الاقتباس.

التنسيق الافتراضي:

```
q {
   display: inline;
   }
q:before {
            content: open-quote;
            }
q:after {
          content: close-quote;
          }
```

<blockquote>....</blockquote>

عنصر مخصص للاقتباسات النصية الكبيرة ذات الفقرات، وهذا النص لا يتم إحاطته بأقواس الاقتباس، وهو منسق ليكون فقرة نصية منفصلة عن باقي النص، ويكون لها هامش افتراضي من الجهات الأربع.

ويتم تعيين خاصية واحدة للعنصر <blockquote> :

cite

خاصية تحدد URL الصفحة أو الموقع المأخوذ منه الاقتباس.

```
Lorem ipsum dolor
<blockquote cite="www.lorumipsum.com">
Lorem ipsum dolor sit amet, consectetur adipiscing elit, <br>
esse cillum dolore eu fugiat nulla pariatur.
</blockquote>
Lorem ipsum dolor
```

Lorem ipsum dolor

Lorem ipsum dolor sit amet, consectetur adipiscing elit,
esse cillum dolore eu fugiat nulla pariatur.

Lorem ipsum dolor

التنسيق الافتراضي:

```
blockqoute {
        display: block;
        margin-top:  1em;
        margin-bottom:  1em;
        margin-right:  40px;
        margin-left:  40px;
        }
```

<mark>....</mark>

عنصر مخصص لتظليل خلفية نص معين، لتمييزه عن باقي النص وسهولة العثور عليه، مثل تظليل كلمة، يتم البحث عنها داخل نص معين.

```
Write the student names and mark <mark> the best ones</mark>
```

Write the student names and mark the best ones

التنسيق الافتراضي:

```
mark {
        background-color: yellow;
        color:  black;
       }
```

_{....}

عنصر مخصص لتنسيق نص، بحجم أصغر من النص العادي، وأسفل مستوى النص العادي، بمسافة تقدر بنصف ارتفاع النص تقريباً **subscript**، ويستخدم هذا العنصر في الغالب، في المعادلات والرموز الكيميائية وفي ترقيم الهامش.

```
Chemical formula for water is H<sub>2</sub>O
```

Chemical formula for water is H_2O

التنسيق الافتراضي:

```
sub {
      vertical-align: sub;
      font-size:  smaller;
      }
```

<small>....</small>

عنصر يعرض النص بخط أصغر من الخط الافتراضي، ويستخدم كتعليق جانبي.

```
<span>Bigger Text<small> Not Bigger Enough</small></span>
```

Bigger Text Not Bigger Enough

التنسيق الافتراضي:

```
small{
        font-size:  smaller;
        }
```

^{....}

عنصر مخصص لتنسيق نص، بحجم أصغر من النص العادي، وأعلى قاعدة النص العادي، بمسافة تقدر بنصف ارتفاع النص تقريباً supscript، ويستخدم هذا العنصر في الغالب، في كتابة الأس الرياضي، أو كتابة ترتيب الأرقام Ordinal numbers .

```
5<sup> 3</sup> = 125
```

$5^3 = 125$

مثال آخر:

```
Ordinal no. : 1<sup>st</sup>, 2<sup>nd</sup>, 3<sup>th</sup>
```

Ordinal no. : 1st, 2nd, 3th

التنسيق الافتراضي:

```
sup{
      vertical-align:  super;
      font-size:  smaller;
      }
```

<bdi>....</bdi>

عنصر يقوم بعزل جزء من النص، له اتجاه مختلف عن اتجاه النص الأصلي، فيظهر النص بصورة صحيحة بالرغم من أنه ثنائي الاتجاه **bidirectional** .

```
Info : Name | Age | Job <br>
Info1:  47  | وائل | Developer.<br>
Info2: <bdi >  وائل </bdi> | 47 | Developer.
```

Info : Name | Age | Job
Info1: 47 | وائل | Developer.
Info2: وائل | 47 | Developer.

لاحظ أن اتجاه النص "وائل" من اليمين إلى اليسار (لأنه باللغة العربية)، بينما اتجاه النص الأصلي من اليسار إلى اليمين، فتجد أن النص في السطر الثاني ظهر غير مرتب، بينما ظهر النص في السطر الثالث مرتباً، نظراً لاستخدام العنصر <bdi> في عزل النص مختلف الاتجاه عن النص الأصلي، مع العلم أن النص داخل محتوى العنصر <bdi> لا يؤثر ، ولا يتأثر باتجاه النص خارج هذا المحتوى.

ويمكن تعيين خاصية واحدة للعنصر <bdi> :

dir

خاصية تحدد اتجاه النص داخل العنصر، ولها قيمتان:

- ***rtl*** : قيمة تعني أن اتجاه النص داخل العنصر يكون من اليمين إلى اليسار.
- ***ltr*** : قيمة تعني أن اتجاه النص داخل العنصر يكون من اليسار إلى اليمين.

ولكن في كثير من الأحيان ، يقوم المتصفح دائماً باختيار الاتجاه الصحيح للنص داخل العنصر، وأحياناً قد يحتاج لتعيين هذه الخاصية وإسناد القيمة المطلوبة لها.

قم بإضافة الخاصية للكود السابق، ولاحظ عدم حدوث أي تغيير في اتجاه النص الأصلي، أو اتجاه النص مختلف الاتجاه، حيث حدد المتصفح اتجاه النص مختلف الاتجاه، دون الحاجة إلى هذه الخاصية.

```
Info2: <bdi dir="rtl">  وائل </bdi> | 47 | Developer.
```

Info2: وائل| 47 | Developer.

التنسيق الافتراضي:

```
bdi {
      unicode-bdi: isolate;
     }
```

wailmonir0@protonmail.com

<bdo>....</bdo>

عنصر يقوم بعكس اتجاه جزء من النص، عن اتجاه النص الأصلي.

```
Learn Web Design Starting with <bdo dir="rtl">HTML</bdo>
```

Learn Web Design Starting with LMTH.

ويمكن تعيين خاصية واحدة للعنصر <bdo> :

dir

خاصية تحدد الاتجاه المطلوب أن يتبعه النص داخل العنصر، والخاصية هنا تقبل قيمتين:

- ***rtl*** : قيمة تعني أن اتجاه النص داخل العنصر يكون من اليمين إلى اليسار.
- ***ltr*** : قيمة تعني أن اتجاه النص داخل العنصر يكون من اليسار إلى اليمين.

التنسيق الافتراضي:

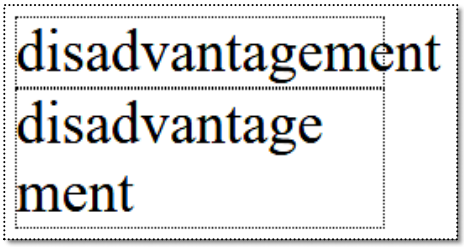
```
bdo {
        unicode-bdi: bidi-override;
       }
```

wailmonir@protonmail.com

<wbr>

عنصر empty element مخصص لكسر الكلمة، والتفاف النص إلى سطر جديد، إذا زاد عرض الكلمة عن عرض السطر، أو عن عرض العنصر الذي يحتويها، وليس شرطاً أن يكون موضع كسر الكلمة، هو آخر حروف لا يستوعبها السطر، بل يمكن أن يكون في أي موضع من الكلمة، إذا تحقق الشرط السابق.

```
<div style="width: 100px; border: 0.5px dotted;">
 disadvantagement</div>
<div style="width: 100px; border: 0.5px dotted;">
 disadvantage<wbr>ment</div>
```



وانكسار الكلمة والتفاف النص يكون عند أي موضع يوجد فيه العنصر <wbr> ، بشرط أن يزيد عرض الكلمة عن عرض السطر أو عرض العنصر الذي يحتويها، وإلا فلن يكون للعنصر <wbr> أي تأثير، ولن يحدث انكسار للكلمة، كما بالشكل المقابل والكود التالي:

```
<div style="width: 100px; border: 0.5px dotted;">
advant<wbr>age</div>
```

لاحظ أن الكلمة لم تنكسر، برغم وجود العنصر <wbr> ، لأن السطر يتسع للكلمة، ولم يزد طولها عن عرض السطر الذي يحتويها.

إنكسار الكلمة عن طريق العنصر <wbr> ، لا يضع شرطة وصل تلقائية hyphen ، في موضع الكسر والتفاف النص، ولوضع شرطة وصل في موضع الكسر، يجب وضع رمز شرطة الوصل التلقائية ، وهي تلقائية، لأنها لا تظهر إلا إذا حدث انكسار للكلمة.

```
<div style="width: 50px;"> advant<wbr>&shy;age</div>
```

<time>....</time>

عنصر مخصص لاحتواء التاريخ والوقت، ويمكن تعيين خاصية وحيدة له :

datetime

خاصية تحدد التاريخ والوقت، حتى يمكن قراءتهم عن طريق برامج قراءة النصوص، فتستخدم للتذكير بتاريخ معين، أو تستخدمها محركات البحث في ترتيب نتائج البحث.

```
<!-- local time-->
<time datetime="2020-11-16T01:53:16.929">16 Nov. 2020</time>
<br>
<!-- Globl time-->
<time datetime="2020-11-16T01:53:16.929Z">16 Nov. 2020</time>
```

الصيغة المستخدمة داخل الخاصية datetime

Type	Value
years	2020
month	2020-11
date	2020-11-25
yearless	10-31 الشهر واليوم بدون العام
week	2020-W32 الأسبوع 32 من السنة 2020
time	17:34 or 17:34:25 or 17:34:25.763
local date & time	2020-11-25 17:34:25.763
	2020-11-25T17:34:25.763
global date & time	2020-11-25 17:34:25.763Z Z يعني توقيت جرينتش
	2020-11-25T17:34:25.763Z
	2020-11-25 17:34:25.763-0400
	2020-11-25 17:34:25.763-04:00
period of time	PT8H32M50S الفترة الزمنية 8 ساعات و32 دقيقة و50 ثانية

<template>....</template>

عنصر لا يتم عرضه أو عرض محتوياته مباشرة عند تحميل صفحة html ، ولكن يتم تنشيط عملية العرض لاحقاً باستخدام الجافاسكربت Javascript .

```
<h1>Hidden Elements</h1>
<div> Any elements inside template will be hidden</div>
<template>Text
</template>
```

لاحظ أن النص داخل محتوى العنصر template غير ظاهر عند بداية تحميل الصفحة.

Hidden Elements

Any elements inside template will be hidden

إظهار العنصر template أو محتواه عن طريق javascript يستلزم معرفة جيدة بهذه اللغة، وهذا يقع خارج نطاق هذا الكتاب، لذلك يكفيك حالياً أن تعرف ماهي طبيعة عمل العنصر template وتأثيره على العناصر داخله.

<slot>....</slot>

عنصر يشترك مع العنصر template لإنشاء عناصر جديدة في الصفحة، وذلك عن طريق بناء ما يسمى web component ، وهو عبارة عن كود موضوع في قالب معين، يمكن استخدامه أي عدد من المرات، في مواضع أو صفحات متعددة، ويستخدم عن طريق الجافا سكربت، أو أحد إطاراتها frameworks .

ومجال التطبيق على هذا العنصر مثل العنصر template ، خارج نطاق هذا الكتاب حيث يحتاج - كما ذكرنا سابقاً - إلى دراية بلغة javascript وكذلك الدراية بأحد إطاراتها.

wailmonir0@protonmail.com

<ruby>....</ruby>

عنصر يستخدم لإدخال الكتابة تحت نص معين أو فوقه أو بعده، وفي الغالب يستخدم هذا العنصر في إدخال اللغات الأسيوية (الصينية واليابانية والكورية)، ولكنني استخدمت اللغة العربية والإنجليزية، ليسهل استيعاب عمل هذا العنصر.

وهذا العنصر يحتاج إلى عنصرين داخليين كي يظهر تأثيره:

<rp>....</rp>

ruby parenthesis وهو عنصر مخصص لاحتواء الأقواس، التي تفصل بين الكلمات، أو بين جزء من النص وجزء آخر، وهو يحتوي إما قوس البداية، أو قوس النهاية.

<rt>....</rt>

ruby text وهو عنصر مخصص لاحتواء النص الذي يتم إدخاله.

```
<ruby>
      نرحب<rp>(</rp><rt> بكم </rt><rp>)</rp>
      Welcome <rp>(</rp><rt> home </rt><rp>)</rp>
</ruby>
```

<progress>....</progress>

عنصر يقوم بعرض معلومات معينة، على هيئة شريط يوضح التقدم في إنجاز عملية ما، عن طريق تعيين خاصيتين لهذا العنصر :

max

خاصية تحدد أعلى قيمة للتقدم يمكن أن تظهر على هذا الشريط.

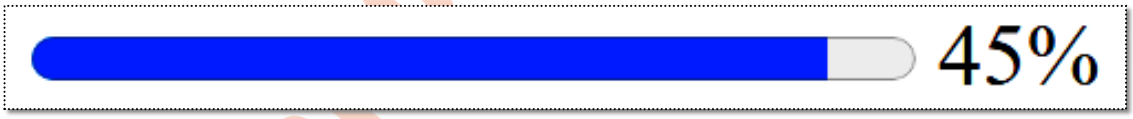
value

خاصية تحدد القيمة التي تظهر مدى التقدم في إنجاز العملية.

```
<progress max="100" value="45"></progress> 45%
```

يجب أن تكون قيمة الخاصية max قيمة موجبة، والقيمة الصغرى تكون 0 افتراضياً، وتكون قيمة الخاصية value أصغر من قيمة الخاصية max، أو تساويها ليظهر شريط التقدم بالشكل الصحيح:

```
<progress max="50" value="45">/progress> 45%
```



لاحظ أن تغيير قيمة الخاصية max قد أثر على نسبة التقدم، حيث إن التقدم في إنجاز العملية، يكون دائماً هو النسبة بين قيمة الخاصية value إلى قيمة الخاصية max .

<meter>....</meter>

هو عنصر مشابه للعنصر <progress> حيث يظهر هذا العنصر على هيئة شريط تقدم في إنجاز عمليه ما، والفرق بينهما، أن القيمة الصغرى min للعنصر <meter> متغيرة، ولا يشترط أن تكون ثابتة، أو تساوي 0 مثل العنصر <progress> ، كما أن العنصر <meter> يحدد ظاهرياً، إذا ما كان التقدم في إنجاز العملية مقبولاً أم لا، عن طريق تغيير لون الشريط المتري، ويمكن تعيين عدة خصائص للعنصر <meter> :

min

خاصية تحدد القيمة الصغرى التي لن تقل عنها قيمة الشريط المتري.

max

خاصية تحدد القيمة الكبرى التي لن تزيد عنها قيمة الشريط المتري.

value

خاصية تحدد القيمة التي تظهر على الشريط المتري.

optimum

خاصية تحدد القيمة المثلى للشريط المتري.

low

خاصية تحدد القيمة التي سيتغير عندها لون الشريط إذا انخفض عنها.

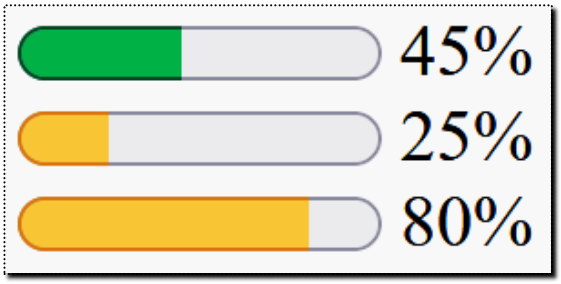
high

خاصية تحدد القيمة التي سيتغير عندها لون الشريط إذا ارتفع عنها.

```
<meter min="0" max="100" value="45" optimum="50"></meter> 45%
<br>
<meter min="0" max="100" value="25" low="50"></meter> 25%<br>
<meter min="0" max="100" value="80" high="75"></meter> 80%
```

لاحظ أن لون الشريط المتري كان باللون الأخضر في حالته العادية، وعندما تم تحديد قيمة الخاصية low وانخفضت قيمة الخاصية value عنها تلون الشريط باللون الأصفر، وكذلك

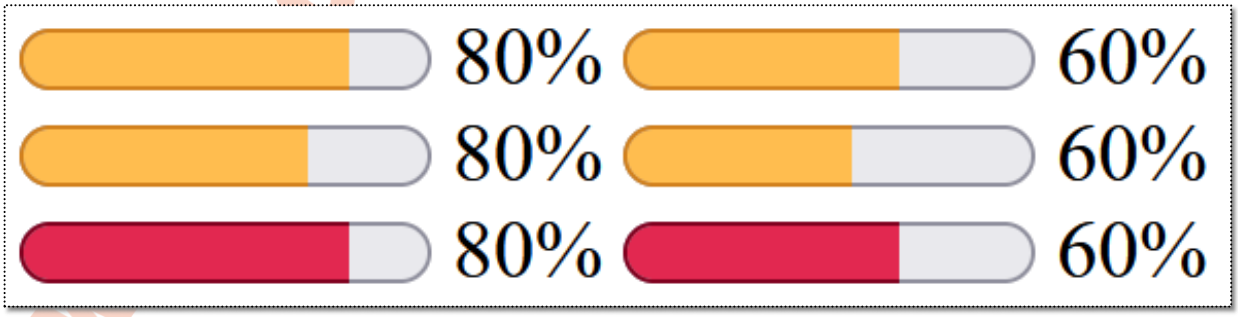
wailmonir0@protonmail.com



الحال عندما زادت قيمة الخاصية value عن قيمة الخاصية high .

لاحظ أن لون الشريط المتري يتحول للون الأحمر ، إذا زادت قيمة الخاصية low عن متوسط مجموع قيمتي الخاصيتين min و max ، وفي نفس الوقت كانت قيمة الخاصية value أكبر من قيمة الخاصية high :

```
<!-- low value <= (min value+max value)/2 & value > high value -->
<meter min="0" max="100" value="80" low="50"
high="75"></meter> 80%
<meter min="0" max="90" value="60" low="45"
high="55"></meter> 60% <br>
<!-- low value > (min value+max value)/2 & value < high value -->
<meter min="0" max="100" value="70" low="51"
high="75"></meter> 80%
<meter min="0" max="90" value="50" low="46"
high="55"></meter> 60% <br>
<!-- low value > (min value+max value)/2 & value > high value -->
<meter min="0" max="100" value="80" low="51"
high="75"></meter> 80%
<meter min="0" max="90" value="60" low="46"
high="55"></meter> 60%
```



لاحظ أنه، في حالة عدم تحقق أحد الشرطين السابقين، يعود اللون لحالته السابقة في المثال الأول، بنفس الشروط المذكورة في المثال.

التشابه بين العناصر في التنسيق الظاهري

بما أنك قد لاحظت وجود بعض العناصر التي تتشابه في تنسيقها ظاهرياً، فاعلم أن التشابه بين العناصر هو تشابه في الشكل فقط، ولكن وراء هذا الشكل توجد بعض التفاصيل التي تؤثر في عمل صفحة الويب، فهذه العناصر قد تختلف في وظيفتها، وقد تختلف في معناها، لأن لها مفاهيم ودلالات معينة أو semantics .

فالعنصر وإن تشابه في التنسيق مع العنصر <b> ، إلا أن النص داخل العنصر يتمتع بدرجة أكبر من الأهمية، لذلك عند استخدام برامج قراءة النصوص التي يستعين بها فاقدو البصر، تكون النصوص المنسقة بهذا العنصر، أشد تأكيداً عند النطق بها، أكثر من النصوص الغير منسقة، أو المنسقة بالعنصر <b> .

وكذلك الحال بالنسبة للعنصر <i> والعنصر ، فالنص داخل العنصر يكون أكثر تأكيداً عند النطق به، من النص داخل العنصر <i> .

بالإضافة إلى أنك عندما تدرس عناصر تجزئة الصفحة، ستعلم أن العناصر المتشابهة لها دور معين في بناء صفحة الويب، يدرك من يسمع اسمها، أي جزء من الصفحة يعرض أمامه.

وهناك آراء تقول بأن محركات البحث تميز بين هذه العناصر، وبالتالي تكون نتائج البحث أسرع إذا توافقت الكلمات المفتاحية التي نبحث عنها مع وظيفة ودلالة العنصر.

وعلى الرغم من أن هذا الرأي يواجه معارضة تقول: إنه لا تأثير لنوع العناصر المتشابهة ظاهرياً في السلوك على نتائج البحث، إلا أن رأيي الشخصي، أن الرأي الأول يبدو منطقياً وأقرب للصواب، لأنه إذا كان الهدف من تنوع العناصر هو هدف دلالي فقط، أي أن كل عنصر له دلالة معينة، مثل تشديد قراءة العنصر عن العنصر <i> ، فكيف سيختلف تأكيد أهمية عنصر عن تشديد العنصر ، وكيف سيتم التمييز صوتياً بين العناصر الأخرى التي تجعل الخط مائلاً، مثل العنصر <i> والعنصر <cite> والعنصر <var> والعنصر <address> ، وكل هذه العناصر يظهر النص داخلها بشكل مائل، فكيف سيتم التمييز بين دلالة كل هذه العناصر صوتياً بشكل يلاحظه المستمع، وهل ينبغي للمستخدم العادي أن يحفظ وظيفة كل عنصر ودلالته، حتى يميز بين العناصر المتشابهة، ويتخيل شكل العنصر في الصفحة عندما يسمع اسمه من خلال القارئ الآلي؟؟

wailmonir0@protonmail.com

عناصر تجزئة الصفحة

هي عناصر تقسم الصفحة إلى أجزاء، مثل العناوين الرئيسية، ورأس وذيل الصفحة، وأجزاء أساسية، وأخرى فرعية، والعناصر كلها تقريباً متشابهة، ويمكن أن يحل أحدها مكان الآخر دون حدوث أي خلل في تصميم الصفحة شكلياً، إلا أن عليك أن تعلم أن لكل عنصر وظيفة معينة، يشغل على أساسها جزءاً معيناً من الصفحة، وكلما استخدمت العنصر المناسب للوظيفة المناسبة، كانت الصفحة أكثر سهولة في العثور عليها، وفي تصفحها وقراءتها آلياً، وفي التعامل معها من قبل ذوي الاحتياجات الخاصة.

<body>....</body>

هو العنصر الرئيسي المختص باحتواء كل عناصر جسم الصفحة، وهي كل العناصر المادية المرئية في المتصفح، والعناصر المادية المخفاة hidden ، ولا يجب أن يكون بالصفحة أكثر من عنصر <body> واحد.

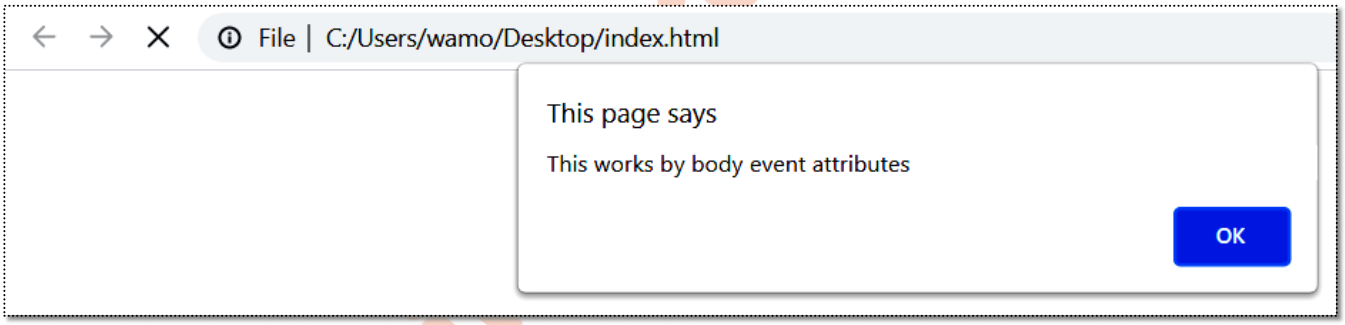
وتغيير بعض خصائص العنصر <body> يوثر على العناصر بالصفحة فهو - على سبيل المثال - يمكنه التحكم في لون خلفية الصفحة، ولون النصوص الافتراضي، ونوع وحجم وتنسيق الخط الافتراضي داخل الصفحة، وغيرها من الخصائص.

```
<!DOCTYPE html>
<html>
<head>
<title> The Body Element</title>
</head>
<body style=background:lightgray;color:navy;font-style:italic;>
<h2>We are in the body element</h2>
<p>Lovely Scenes</p>
<img src="images/road.jpg" width="200" height="100">
<img src="images/colors.jpg" width="200" height="100">
<img src="images/mount.jpg" width="200" height="100">
</body>
</html>
```

والعنصر <body> قد نعين له event attributes ، وهي خصائص عبارة عن شرط لتنفيذ حدث معين، أي عند تحقق هذا الشرط يتم تنفيذ سكريبت (نص برمجي)، ويوجد العديد من هذه الخصائص، وسوف نشرحها في فصل خاص بها، ومن أمثلتها :

onafterprint | onbeforeprint | onbeforeunload | onmessage | onload | onhashchange | onunload | onpopstate| onresize

```
<body onload="alert('This works by body event attributes')">
```



التنسيق الافتراضي:

```
body {
        display: block;
        margin: 8px;
       }
body:focus {
       outline: none;
       }
```

<h1>....</h1>

عناصر headings أو العناوين الرئيسية للصفحة وأجزاء الصفحة.

والعناصر headings عبارة عن 6 عناصر، تبدأ بالعنصر <h1> وهو أكبرها حجماً، وتنتهي بالعنصر <h6> وهو أصغرها، وبينهما العناصر من <h2> إلى <h5> .

العنصر <h1> مخصص كعنوان رئيسي للصفحة، ويفضل ألا يستخدم أكثر من مرة واحدة في نفس الصفحة، كتطبيق وممارسة أمثل.

يفضل أن تستخدم العناصر headings بالترتيب وعدم تخطي أحدها.

يفضل ألا يتم أي تعديل على تنسيق العناصر headings باستخدام css ، وإذا كان تنسيقها لا يفي بالتصميم المطلوب، فيمكن استخدم نص عادي وتنسيقه بدلاً منها.

```
<h1>Welcome to html !!</h1>
<h2>Welcome to html !!</h2>
<h3>Welcome to html !!</h3>
<h4>Welcome to html !!</h4>
<h5>Welcome to html !!</h5>
<h6>Welcome to html !!</h6>
```

Welcome to html !!

Welcome to html !!

Welcome to html !!

Welcome to html !!

Welcome to html !!

Welcome to html !!

لاحظ التدرج في حجم الخط من الأكبر <h1> إلى الأصغر <h6> ، ولاحظ أن كل عنصر من عناصر headings قد تم إغلاقه قبل فتح أي عنصر آخر.

Tag Omission

ذكرنا من قبل وجود الميزة في بعض عناصر block elements ومن ضمن هذه العناصر، عناصر headings إذا أنشأ داخلها عنصر heading آخر.

```
<h2 style="background: lightblue">
      Welcome to html !!<br>
      Welcome to html !!
     <h3 style="background: orange"> Welcome to html !!</h3>
      Welcome to html !!
</h2>
```

لاحظ أن النص قبل بداية العنصر <h3> قد تم تنسيقه بنفس تنسيق العنصر <h2> لأنه يقع داخله، في حين أن النص الواقع بعد إغلاق العنصر <h3> وقبل وسم إغلاق العنصر <h2> ، قد عاد للتنسيق الافتراضي للخصائص، بالرغم من وجوده قبل وسم إغلاق العنصر <h2> ، مما يعني أن وسم بداية العنصر الابن الداخلي <h3> ، قد أغلق العنصر الأب <h2> وتم تجاهل وسم إغلاقه.

استبدل العنصر <h2> بأي عنصر كتلة آخر ، ولاحظ الفرق.

التنسيق الافتراضي:

```
h1,h2,h3,h4,h5,h6 { display: block; font-weight: bold; }
h1 { margin: 0.67em   0; font-size:  2em; }
h2 { margin: 0.83em   0; font-size:  1.5em; }
h3 { margin: 1em      0; font-size:  1.17em; }
h4 { margin: 1.33em   0; font-size:  1em; }
h5 { margin: 1.67em   0; font-size:  0.83em; }
h6 { margin: 2.33em   0; font-size:  0.67em; }
```

....

عنصر مخصص لاحتواء أي نوع من العناصر، سواء عناوين رئيسية، أو فقرات، أو صور ، أو روابط تشعبية، أو جداول، أو قوائم وغيرها، وليس له أي تأثير على العناصر داخله، حتى يتم تنسيقه بـ css ، فيكون له عرض وارتفاع ولون خلفية ولو ن أمامي وغيرها من الخصائص.

```
<div>
    <h3>Web Design</h3>
    <div>HTML</div>
    <div>CSS</div>
    <div>Javascript</div>
    <div><img src="images/apple.png" width="10%"></div>
    <div><a href="https://google.com">Google</a></div>
</div>
```

لاحظ أن وجود العناصر داخل العنصر لم يؤثر على مظهرها لأنه غير منسق تنسيقاً خاصاً، وكل خصائصه لها القيم الافتراضية.

لاحظ أن الصفحة يمكن أن تحتوي على أي عدد من العنصر ، وأن العنصر يمكن أن يحتوي أي نوع من العناصر، وأنه كعنصر block element ومثل معظم العناصر في هذا الجزء من الكتاب، ينتقل للسطر التالي بعد العنصر الذي قبله، وينقل العنصر الذي بعده إلى سطر جديد، أي أنه لا يكون على سطر واحد مع أي عنصر، وهذا هو التنسيق الافتراضي، والذي ستتعلم كيف تغيره عند دراستك css .

التنسيق الافتراضي:

```
div{
    display: block;
}
```


عنصر empty element مخصص لكسر السطر، ونقل العنصر أو النص الذي بعده إلى سطر جديد، لاحظ مخرجات الكود التالي بدون استخدام العنصر
 .

```
<div>
    Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do
    eiusmod
</div>
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod

لاحظ مخرجات نفس الكود السابق ولكن باستخدام العنصر
 .

```
<div>
    Lorem ipsum dolor sit amet, consectetur adipiscing elit,<br>sed
    do eiusmod
</div>
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit,
sed do eiusmod

لاحظ حدوث انكسار للسطر ، والتفاف للنص في موضع العنصر
 .

مثال آخر:

لاحظ مخرجات الكود التالي بدون استخدام العنصر
 بين الصورتين.

```
<img src="images/girl.jpg"  width="200"  height="100">
<img src="images/graff.jpg"  width="200"  height="100">
```

لاحظ مخرجات نفس الكود السابق ولكن باستخدام العنصر
 بعد الصورة الأول :

```
<img src="images/girl.jpg" width="200" height="100"><br>
<img src="images/graff.jpg" width="200" height="100">
```

لاحظ انكسار السطر في موضع العنصر
 ، والتفاف الصورة الثانية لسطر جديد أسفل الصورة الأولى.

العنصر
 لا يقوم بوضع هامش بين السطرين عند كسر السطر والتفاف العناصر لسطر جديد، كما بالشكل المقابل.

wailmonir0@protonmail.com

<hr>

عنصر empty element ، مخصص لرسم خط أفقي بين العناصر، وطوله الافتراضي هو عرض الصفحة، أو عرض العنصر الذي يحتويه.

```
<div style=width:200px>
        before horizontal line <hr> after horizontal line
</div>
```

before horizontal line

after horizontal line

لاحظ أن الخط الأفقي قام بكسر السطر ، وحدث التفاف للنص لسطر جديد، وتم رسم خط أفقي بعرض العنصر بين السطرين، وكانت قيمة الهامش بين الخط الأفقي والسطر العلوي، مساوية لقيمتها بين الخط الأفقي والسطر السفلي وهي 8px .

مثال آخر:

نفس مثال الصورتين المتجاورتين السابق ، ولكن هذه المرة باستخدام العنصر <hr> بعد الصورة الأولى :

```
<img src="images/girl.jpg"  width="200" height="100"><hr>
<img src="images/graff.jpg"  width="200" height="100">
```

لاحظ انكسار السطر والتفاف الصورة الثانية أسفل الصورة الأولى، مع رسم خط أفقي بينهما، مع وجود هامش أعلى وأسفل الخط بين الصورتين.

استخدام العنصر <hr> في رسم خط أفقي أصبح أسلوباً قديماً وتقليدياً، ورسم الخط الأفقي يجب أن يتم باستخدام css .

التنسيق الافتراضي:

```
hr {
    display: block;
    margin: 0.5em   auto;
    border: 1px   inset;
    }
```

في الكود السابق، تمثل الخاصية border الحدود حول العنصر، والخاصية في هذا الكود تقبل قيمتين، الأولى تمثل عرض حدود العنصر border-width وهو 1px والقيمة الثانية، تمثل خاصية شكل الحدود border-style ، وهى inset .

<article>...</article>

عنصر مخصص لكتابة نص، على هيئة مقالة أو بوست أو تعليق على خبر، ويمكن أن يحتوي هذا العنصر على فقرات نصية أو صور أو روابط تشعبية وغيرها من العناصر.

```
<span style=color:green>Before article element</span>
<article style=width:420px><h2>Roses</h2>
  <p>Pick your favorite roses</p>
  <section> Lorem ipsum dolor sit amet, consectetur adipiscing elit,
     sed do eiusmod tempor incididunt ut labore et dolore magna
     enim ad minim veniam, quis nostrud exercitation ullamco
  </section>
  <section><img src="roses.jpg" width="200">
              <img src="rosa.jpg"  width="200">
  </section>
</article>
<span style=color:green>After article element</span>
```

يمكن للصفحة أن تحتوي أكثر من عنصر <article> وكل عنصر يجب أن يتم تقديمه باستخدام أحد العناوين الرئيسية headings (<h1>...<h6>).

يمكن للعنصر <article> أن يقسم داخلياً إلى عدد من الأقسام، كل قسم عبارة عن عنصر <section>.

التنسيق الافتراضي:

```
article {
        display: block;
        }
```

<section>...</section>

عنصر عام ليس له وظيفة محددة، ويستخدم لشغل أي جزء من الصفحة، ولا يوجد جزء محدد يمكن أن يشغله، ويمكن أن يحوي عدة عناصر أخرى، ويتم تقديمه أو تعريفه باستخدام أحد عناصر headings .

```
<span style=color:blue>Before section element</span>
<section  style=width:500px>
  <h3>Section </h3>
  <p> Lorem ipsum </p>
  <span>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed
   do eiusmod tempor incididunt ut labore et dolore magna aliqua.
   Ut enim ad minim veniam, quis nostrud exercitation ullamco
  </span>
</section>
<span style=color:blue>After section element</span>
```

Before section element

Section

Lorem ipsum

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco

After section element

التنسيق الافتراضي:

```
section {
        display: block;
        }
```

<main>...</main>

عنصر مخصص لاحتواء الجزء الرئيسي من الصفحة، والذي يكون في مركز الصفحة غالباً، ويتناول الموضوع الأساسي الذي تناقشه الصفحة، لذلك يجب ألا يوجد بالصفحة أكثر من عنصر <main> واحد.

```
<span style=color:blue>Before main aside</span>
<main  style=width:420px>
  <h2 style=text-align:center;>main Element</h2>
  <img src="images/green.jpg" width="200" height="150">
  <img src="images/blue.jpg"  width="200" height="150px">  <br>
   <span>Lorem ipsum dolor sit amet, consectetur adipiscing elit,
    sed do eiusmod tempor incididunt ut labore et dolore magna
    aliqua.Ut enim ad minim veniam, quis nostrud exercitation ullamco
  </span><br>
</main>
<span style=color:blue>After main element</span>
```

و يمكن للعنصر <main> أن يحتوي أكثر من نوع من العناصر، ولكن لا يجب أن يحتوي أي عناصر مكررة موجودة بالفعل في الصفحة بشكل أو بآخر، والعنصر <main> ينتقل إلى سطر جديد بعد العنصر قبله، وينقل العنصر بعده إلى سطر جديد.

التنسيق الافتراضي:

```
main {
        display: block;
        }
```

<aside>...</aside>

عنصر يكون على هيئة شريط جانبي للصفحة، لعرض معلومات فرعية، تعتبر جزءاً من المعلومات الأساسية المعروضة في العنصر <main> ، وقد يكون مربع استدعاء call-out box ، وهو مربع صغير، يعرض مقدمة عن موضوع معين، وإذا أردت المزيد من المعلومات، تضغط على زر more الموجود داخله، لتستدعي صفحة تحتوي معلومات أكثر.

```
<span style=color:darkred>Before aside element</span>
<aside  style=width:200px>
  <h3>Aside Element</h3>
  <div> <a href=" "> Lorem ipsum</a> </div><br>
  <span>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed
   do eiusmod tempor incididunt ut labore et dolore magna aliqua.
   Ut enim ad minim veniam, quis nostrud exercitation ullamco
  </span><br>
   <img src="images/blue.jpg" width="200" height="150">
</aside>
<span style=color:darkred>After aside element</span>
```

لاحظ أن العنصر aside يعامل تقريباً معاملة الشريط الجانبي، كما أنه يمكن أن يحتوي العديد من العناصر، ولا يتشارك مع أي عنصر آخر السطر الذي يوجد عليه.

التنسيق الافتراضي:

```
aside {
        display: block;
        }
```

<nav>...</nav>

عنصر مخصص لإنشاء قائمة تنَقُل، تحتوي الراوبط التشعبية للمواقع والصفحات، لتنقل المستخدم بين الصفحات والمواقع، ويمكن أن تكون الروابط للتنقل بين أجزاء الصفحة نفسها، كما سنعرف لاحقاً عند دراستنا للعنصرين <a> و<area> .

كما يمكن للعنصر <nav> أن يستخدم كجدول لمحتويات موقع معين، أو فهرس لمجموعة مقالات أو صفحات.

```
<nav style="background: lightblue; width: 245px; padding: 3px">
    <a href="https://google.com">Google | </a>
    <a href="https://facebook.com">Facebook | </a>
    <a href="https://twitter.com">Twitter |</a>
    <a href="https://amazon.com">Amazon </a>
</nav>
```

Google | Facebook | Twitter | Amazon

والعنصر <nav> ينطبق عليه نفس خصائص عناصر الكتلة block elements حيث ينتقل إلى سطر جديد بعد العنصر قبله، وينقل العنصر بعده إلى سطر جديد.

يمكن للصفحة أن تحتوي أكثر من عنصر <nav> ، وليس من الضروري أن تنحصر الروابط التشعبية داخل العنصر <nav> ، بل يمكن أن تتواجد داخل العنصر <footer> ، كما سنرى لاحقاً.

التنسيق الافتراضي:

```
nav {
    display: block;
}
```

<header>...</header>

عنصر مخصص ليحتوي رأساً للصفحة، أو مقدمة تعريفية للصفحة، وقد يحتوي شعاراً، أو اسم الصفحة، أو قائمة التنقل بين صفحات الموقع، أو أي معلومات عن المسئول، أو عن الهيئة صاحبة الصفحة وغيرها.

```
<span style=color:darkred>Before header element</span>
<header style=width:500px;background:lightgray;>
 <img src="images/stain3.png" width="170" height="100"
  style=vertical-align:middle;>
 <span style="font-size: 2.25em">Stain For Printing</span>
 <nav style=text-align:right;margin-right:10px;>
    <a href=" ">Home  | </a><a href=" ">Products | </a>
    <a href=" ">Gallery | </a><a href=" ">Total Cost |</a>
    <a href=" ">Our Stuff  | </a><a href=" ">Contact us </a>
  </nav><br>
</header><span style=color:darkred>After header element</span>
```

خصائص هذا العنصر هي نفس خصائص عناصر الكتلة block elements .

التنسيق الافتراضي:

```
header {
        display: block;
    }
```

<footer>...</footer>

عنصر مخصص لاحتواء آخر محتويات الصفحة أو ذيل الصفحة، ويحتوي على معلومات حقوق الطبع والملكية، ووسائل الاتصال بمسئولي الهيئة صاحبة الصفحة، وخريطة موقع الهيئة، وأي معلومات ذات صلة بالهيئة.

```
<footer style="background: orange; text-align: center;">
<address>Feedback: <a href="">one@email.com</a></address>
<address>Contact us: <a href="">www. site.com</a></address>
<address>&copy; Copy rights reserved</address>
</footer>
```

Feedback: one@email.com
Contact us: www. site.com
© Copy rights reserved

وبما أن العنصر <footer> يحتوي وسائل الاتصال والعناوين الخاصة بالصفحة، فإن أنسب عنصر لتنيسق وإضافة هذه البيانات هو العنصر <address> .

والعنصر <footer> يتمتع بنفس خصائص عناصر الكتلة block elements السابق شرحها.

التنسيق الافتراضي:

```
footer {
        display: block;
        }
```

<dialog>...</dialog>

عنصر مخصص لإنشاء مربع حواري، أو نافذة صغيرة، تحتوى نصاً معيناً، أو رسالة، أو تنبيه للمستخدم.

ويمكن أن يسند للعنصر <dialog> خاصية واحدة:

open

إحدى خصائص boolean attributes ، وجودها يعني أن قيمتها هي true ، وأن العنصر dialog يعرض مفتوحاً حتى يتمكن المستخدم من قراءة محتواه.

```
<div>We study dialog element here.</div>
<dialog  open>This dialog will be opened</dialog>
<div>We study dialog element here.</div>
```

We study dialog element here.
We study dialog element here.

This dialog will be opened

لاحظ أن العنصر المتواجد بعد العنصر <dialog> ، قد تدفق في الصفحة مباشرة بعد العنصر المتواجد قبل العنصر <dialog> ، ولم يتم الفصل بينهما، وكأن العنصر <dialog> غير موجود في الصفحة.

في حالة عدم تعيين الخاصية open سيختفي العنصر dialog ، وسيبدو كأنه غير موجود في الصفحة من الأساس.

We study dialog element here.
We study dialog element here.

<details>...</details>

عنصر مخصص لإنشاء قائمة منسدلة، تحتوي عدة بنود، تفتح وتعرض هذه البنود عند الضغط على عنوان هذه القائمة، وهو عبارة عن العنصر <summary> ، والذي يحتوي نصاً معيناً بالإضافة إلى مثلث صغير، يدور عند فتح وغلق القائمة.

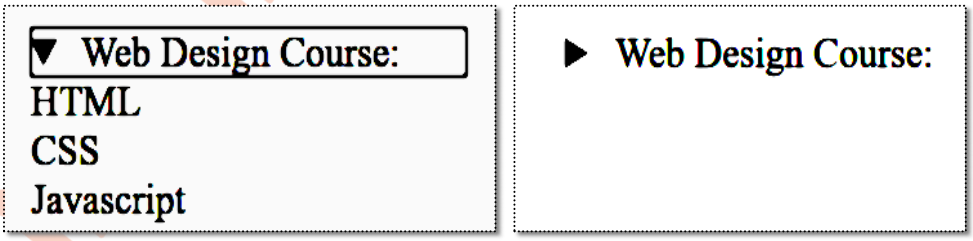
التنسيق الافتراضي:

```
details {
        display: block;
        }
```

<summary>...</summary>

عنصر داخلي للعنصر <details> مخصص لإنشاء عنوان للقائمة ، وعند الضغط عليه تفتح وتغلق هذه القائمة بالتبادل، ويكون العنصر <summary> أول عنصر داخل العنصر <details> ، وإن تغير ترتيبه داخل الكود، فلن يؤثر ذلك على ظهوره كعنوان للقائمة.

```
<details style="width: 180px">
   <summary>Web Design Course:</summary>
            <div> HTML </div>
            <div> CSS</div>
            <div> Javascript</div>
</details>
```



التنسيق الافتراضي:

```
summary  {
         display: list-item;  /*  block means no triangle  */
         }
```

الروابط التشعبية Hyperlinks

هي عناصر، عند الضغط عليها، تقوم بنقل المستخدم إلى صفحات أو مواقع أخرى، خارج الصفحة، أو تنقله إلى أجزاء داخل نفس الصفحة، إذا كان عرض أو ارتفاع الصفحة كبيراً، والصفحة لا تظهر بكامل عرضها أو ارتفاعها في شاشة العرض.

ويوجد عنصران مسئولان عن إنشاء الروابط التشعبية، أحدهما هو العنصر <area> ، وسوف ندرسه ضمن عناصر الدمج، والعنصر الآخر هو العنصر <a> .

<a>....

أو anchor element حيث يقوم بنقل المستخدم إلى صفحات أخرى، أو مواضع أخرى، داخل نفس الصفحة، ويوجد عدة خصائص تحدد طبيعة عمل هذا العنصر:

href

خاصية تحدد الصفحة أو الجزء، الذي سينتقل إليه المستخدم عند الضغط على الرابط، والرابط يكون دائماً كل ما يقع بين وسمي البداية والنهاية للعنصر <a> .

target

خاصية تحدد الموضع الذي ستفتح فيها الصفحة التي سيتم الانتقال إليها، عند الضغط على الرابط، ويسند إليها عدة قيم وشرحها ملحق 1 .

```
<a href="https://facebook.com" target="_blank"> Facebook</a>
```

Facebook

rel

خاصية تحدد العلاقة بين الصفحة الحالية، والصفحة التي سيتم الانتقال إليها، ويسند إليها العديد من القيم، وشرحها في ملحق 1 .

type

خاصية تحدد نوع الملف الذي سينتقل إليه المستخدم، عند الضغط على الرابط، ويمكن تحديد نوع الملف بأكثر من طريقة، وشرحها ص 130 .

wailmonir0@protonmail.com

download

خاصية تسمح بتحميل الملف إلى الحاسب، عند الضغط على الرابط، على أن يتم فتح الصفحة عن طريق خادم محلي localhost ، كما يجب أن يكون الملف على نفس العنوان الرئيسي origin ، ولتوضيح مفهوم origin انظر ص 282 .

```
<div> Click to download this
<a href ="some_file.pdf" download >Download file</a></div>
```

ويمكنك إضافة اسم جديد للملف، بإضافته كقيمة download="filename" ويمكن عدم ذكر الامتداد، وسيقوم المتصفح بالتعرف على الامتداد عن طريق الخاصية href .

hreflang

خاصية تحدد لغة الملف الذي سيتم الانتقال إليه، عند الضغط على الرابط.

ping

خاصية تحدد قائمة بالمواقع، التي سيتم اختبارها لتتبع حركة الملف، الذي سيتم الانتقال إليه عند الضغط على الرابط.

referrerpolicy

خاصية تحدد إمكانية نقل معلومات، من الصفحة الحالية إلى الصفحة التي سيتم الانتقال إليها، والخاصية تقبل عدة أنواع من القيم، وشرحها ملحق 1 .

ربط عنوان بريد إلكتروني Linking to an email address

يمكن ربط عنصر <a> مع برنامج إرسال بريد إلكتروني، عن طريق إضافة الكلمة mailto: ثم عنوان البريد الإلكتروني المطلوب مراسلته، كقيمة للخاصية href ، وعند الضغط على الرابط، يتم فتح برنامج إرسال البريد الإلكتروني، ويقوم المستخدم بكتابة الرسالة المطلوبة، وإرسالها للشخص المطلوب.

```
<a href="mailto:someone@mail.com " target="_blank "> email</a>
```

wailmonir0@protonmail.com

ربط جزء من الصفحة بجزء آخر في نفس الصفحة

Linking to an element on the same page

يمكن للرابط التشعبي التنقل داخل الصفحة الواحدة، إذا كانت الصفحة ذات عرض أو ارتفاع كبيرين، يتخطي عرض أو ارتفاع شاشة العرض، وخاصة إذا كان بها العديد من الموضوعات التي لا يمكن أن تتجمع في شاشة واحدة، فيمكن تجزئة الصفحة إلى أجزاء fragments ، والانتقال إلى جزء معين fragment مباشرة، بالضغط على الرابط الذي سيتم إعداده خاصة لهذا الجزء، بدلاً من المرور على كل أجزاء الصفحة وحتى الوصول للجزء المطلوب، ويتم ذلك عن طريق إنشاء anchors ، وهي روابط تشعبية ترتبط بأجزاء الصفحة المراد الانتقال إليها، عن طريق تعيين خاصية id للجزء المراد الانتقال إليه، ثم إسناد قيمة هذه الخاصية يسبقها الرمز "#" ، كقيمة للخاصية href الخاصة بالعنصر <a> الذي سنضغط عليه لينقلنا إلى هذا الجزء، أو غيره من الصفحة كما يلي:

```
<p><a href="#contacts">Go to contacts</a></p>
Lorem ipsum dolor sit amet,<br>
consectetur adipiscing elit,<br>
sed do eiusmod tempor incididunt<br>
ut labore et dolore magna aliqua.<br>
Ut enim ad minim veniam, quis<br>
nostrud exercitation ullamco<br>
laboris nisi ut aliquip ex ea<br>
commodo consequat.<hr>
<div id="contacts">Contacts:</div>
<div>Tel: 01254526321</div>
<div>email: someone@somesite.com</div>
<div>facebook: facebook.com/someone</div>
```

الشكل التالي هو نموذج مصغر، لما يكون عليه الوضع مع الصفحات الكبيرة، حيث إنه عندما يريد المستخدم الانتقال إلى الجزء Contacts مباشرة (بافتراض أن الصفحة ذات ارتفاع كبير وهذا الجزء غير مرئي للمستخدم)، نقوم بتعيين الخاصية id للعنصر ، الذي نريد الانتقال إليه، ونقوم بإسناد قيمة هذه الخاصية إلى الخاصية href ، الخاصة بالرابط

التشعبي في أول الصفحة، يسبقها الرمز "#" .

Go to contacts

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Contacts:
Tel: 01254526321
email: someone@somesite.com
facebook: facebook.com/someone

ربط رقم هاتف Linking to a telephone number

يمكن ربط العنصر <a> برقم هاتف، عن طريق إضافة الكلمة tel: ثم يتبعها رقم الهاتف المطلوب الاتصال به، كقيمة للخاصية href .

- فإذا كانت الصفحة الحالية مفتوحة عن طريق هاتف خلوي مثلاً، فعند الضغط على الرابط يتم الاتصال برقم الهاتف.

- أما إذا تم فتح الصفحة من حاسب المستخدم، فسيقوم برنامج إجراء المكالمات مثل skype أو FaceTime ، بالاتصال بالهاتف المطلوب.

```
<a href="tel:+340897678930"> Phone</a>
```

التنسيق الافتراضي:

```
a {
   display: inline;
   text-decoration: underline;
   cursor: auto;
   }
```

عناصر القوائم List Elements

عناصر مخصصة لعرض بيانات معينة، على هيئة قوائم رئيسية تحتوي بنوداً فرعية، وهذه القوائم الرئيسية نوعان: القوائم المرتبة أو ordered list ، ويمثلها العنصر <ol> ، أو القوائم غير المرتبة أو unordered list ، ويمثلها العنصر <ul> .

....

عنصر مخصص لإنشاء القوائم المرتبة عددياً أو هجائياً، وتضم هذه القائمة بنداً أو أكثر، ويمثل هذه البنود العنصر <li> أو list item .

```
<h4>Programming Applications</h4>
<ol>
   <li>Web applications</li>
   <li>Desktop applications</li>
   <li>Mobile applications</li>
</ol>
```

Programming Applications

1. Web applications
2. Desktop applications
3. Mobile applications

والعنصر الفرعي <li> يمكن أن يتفرع هو نفسه، لعدة قوائم فرعية أصغر:

```
<h4>Artificial intelligence</h4>
      <ol>
         <li>Machine learning
            <ol>
               <li>C++</li>
               <li>Python</li>
               <li>R language</li>
            </ol>
         </li>
```

```
        <li>Data analysis
          <ol>
            <li>Python</li>
            <li>R Language</li>
              <li>Tableau</li>
            </ol>
        </li>
        <li>Deep learning
          <ol>
            <li>Python</li>
            <li>R Language</li>
          </ol>
        </li>
      </ol>
```

Artificial intelligence

1. Machine learning
    1. C++
    2. Python
    3. R language
2. Data analysis
    1. Python
    2. R Language
    3. Tableau
3. Deep learning
    1. Python
    2. R Language

لاحظ أن كل بند في القائمة الرئيسية انقسم إلى عدة بنود فرعية أخرى، فأصبح البند كأنه قائمة فرعية من القائمة الرئيسية، ولها بنود فرعية أخرى، وهذا السلوك يسمى nesting .

start

خاصية يتم تعيينها للعنصر <ol> ، تحدد القيمة أو العدد الذي يبدأ منه ترقيم بنود القائمة، ثم تزيد بعد ذلك بالترتيب.

```
<h4>Mobile applications</h4>
    <ol start="11">
      <li>Android
        <ol><li>Java</li><li>Kotlin</li><li>C#</li></ol>
      </li>
      <li>ios
        <ol> <li>Objective-C</li> <li>Swift</li></ol>
      </li>
    </ol>
```

Mobile applications

11. Android
    1. Java
    2. Kotlin
    3. C#
12. ios
    1. Objective-C
    2. Swift

لاحظ أن ترقيم البنود في القائمة الرئيسية بدأ من القيمة 11 وهي قيمة الخاصية start في العنصر <ol> ، بينما ترقيم بنود القائمة الفرعية بدأ بالشكل الافتراضي من الرقم 1 .

value

خاصية للعنصر <li> تحدد القيمة أو العدد الذي سيبدأ منه الترقيم.

```
<h4>Mobile applications</h4>
  <ol>
   <li value="11">Android
    <ol><li value="111">Java</li><li>Kotlin</li><li>C#</li></ol>
   </li>
    <li>ios
        <ol> <li>Objective-C</li> <li>Swift</li></ol>
     </li>
  </ol>
```

Mobile applications

11. Android
    111. Java
    112. Kotlin
    113. C#
12. ios
    1. Objective-C
    2. Swift

لاحظ أن ترقيم بنود القائمة الرئيسية بدأ من العدد 11 ، وهي قيمة الخاصية value للبند <li> الأول في القائمة الرئيسية .

بينما ترقيم القائمة الفرعية الأولى (وهي أول بند في القائمة الرئيسية) بدأ من العدد 111 ، وهي قيمة الخاصية value للبند الفرعي الأول في القائمة الفرعية.

reversed

خاصية للعنصر <ol> ، تقوم بعرض ترتيب الترقيم المستخدم بشكل معكوس، حيث يبدأ الترقيم من آخر بند في القائمة، ويزيد تصاعدياً من الأسفل إلى أعلى بند.

```
<h4>Mobile applications</h4>
  <ol reversed>
    <li>Android
      <ol reversed><li>Java</li><li>Kotlin</li><li>C#</li></ol>
    </li>
    <li>ios
      <ol> <li>Objective-C</li> <li>Swift</li></ol>
    </li>
  </ol>
```

Mobile applications

2. Android
    3. Java
    2. Kotlin
    1. C#
1. ios
    1. Objective-C
    2. Swift

لاحظ أن ترتيب الترقيم في القائمة الرئيسية وفي القائمة الفرعية (البند الأول في القائمة الرئيسية) بدأ من أسفل لأعلى، لاستخدامنا الخاصية reversed ، بينما في القائمة الفرعية الثانية (البند الثاني في القائمة الرئيسية) ظهر ترتيب الترقيم بشكله الافتراضي من أعلى إلى أسفل.

type

خاصية تحدد نوع الترقيم في القائمة، وتقبل قيمتها على هيئة رقم واحد، أو حرف واحد،

دون أي فراغات قبل أو بعد الحرف أو الرقم، وخلاف ذلك، فإن الخاصية سوف تستخدم القيمة الافتراضية وهي الأرقام.

```
<ol type="A">
   <li>Web applications</li>
   <li>Desktop applications</li>
   <li>Mobile applications</li>
</ol>
```

A. Web applications
B. Desktop applications
C. Mobile applications

```
<ol type="a">
   <li>Web applications</li>
   <li>Desktop applications</li>
   <li>Mobile applications</li>
</ol>
```

a. Web applications
b. Desktop applications
c. Mobile applications

```
<ol type="I">
   <li>Web applications</li>
   <li>Desktop applications</li>
   <li>Mobile applications</li>
</ol>
```

I. Web applications
II. Desktop applications
III. Mobile applications

```
<ol type="i">
   <li>Web applications</li>
   <li>Desktop applications</li>
   <li>Mobile applications</li>
</ol>
```

i. Web applications
ii. Desktop applications
iii. Mobile applications

التنسيق الافتراضي:

```
ol {
    display: block;
    list-style-type: decimal;
    margin: 1em   0;
    padding: 40px;
    }
```

....

كما سبق أن أوضحنا هو عنصر داخلي للعنصرين <ol> و <ul> ، وهو الذي يحتوي داخله بنود القائمة الرئيسية أو الفرعية، سواء كانت مرتبة <ol> ، أو غير مرتبة <ul> .

التنسيق الافتراضي:

```
li {
   display: list-item;
   }
```

Tag Omission

العنصر <li> يمكن أن يستغني عن وسم الإغلاق closing tag إذا تبع محتواه عنصر <li> آخر، أو كان هذا العنصر هو آخر عنصر في القائمة.

```
<ol>Programming Applications
   <li style="color:blue">Web applications
   <li style="color:red">Desktop applications</li>
        Tag Omitted
   <li style="color:green">Mobile applications
</ol>
```

Programming Applications
1. Web applications
2. Desktop applications
Tag Omitted
3. Mobile applications

لاحظ أن البند الأول <li> في القائمة لم يتم إغلاقه، ومع ذلك لم يؤثر على العناصر بعده، أو النص "Tag Omitted" الواقع خارج عناصر القائمة، وكذلك البند الثالث في القائمة لم يحتج لوسم إغلاق لأنه آخر بند في القائمة.

....

عنصر مخصص لإنشاء قوائم غير مرتبة ترتيباً معيناً، وكما شرحنا سابقاً، فكل بند من بنود القائمة، يكون داخل وسمي البداية والنهاية للعنصر <li> .

```
<h4>Programming Languages</h4>
  <ul>
    <li>Web applications</li>
    <li>Desktop applications</li>
    <li>Mobile applications</li>
  </ul>
```

Programming Languages

- Web applications
- Desktop applications
- Mobile applications

لاحظ أن كل بند من بنود القائمة ظهر قبله رمز marker ، وهو عبارة عن قرص مصمت وهو القيمة الافتراضية ، بديلاً عن الأرقام التي كانت تسبق أي بند في النوع السابق (القوائم المرتبة).

والعنصر الفرعي <li> يمكن أن يتفرع هو نفسه لعدة قوائم أصغر، وهو ما يسمى nesting ويكون لكل قائمة رمز marker مختلف عن القائمة قبلها أو بعدها كالتالي:

```
<h4>Programming Languages</h4>
<ul>
  <li>Web applications
    <ul>
      <li>Design
        <ul><li>HTML</li><li>CSS</li><li>Jscript</li></ul>
      </li>
      <li>Developing
        <ul><li>PHP</li><li>MySql</li><li>Laravel</li></ul>
      </li>
    </ul>
  </li>
  <li>Desktop applications
```

```
    <ul>
        <li>Software development
            <ul><li>C Lang</li><li>C++</li><li>C#</li></ul>
        </li>
    </ul>
  </li>
```

Programming Languages

- Web applications
  - Design
    - HTML
    - CSS
    - Jscript
  - Developing
    - PHP
    - MySql
    - Laravel
- Desktop applications
  - Software development
    - C Lang
    - C++
    - C#

لاحظ أن كل قائمة لها رمز خاص بها، مختلف عن رمز أي قائمة أخرى، ويتم تعيين هذه الرموز بشكل تلقائي، حيث تتغير مع كل قائمة تتفرع من بند معين.

تخصيص رمز القائمة "marker"

يمكن تخصيص وتغيير نوع الرمز marker عن طريق تنسيقات css باستخدام الخاصية list-style-type بعد أن أصبحت الخاصية type مهملة، ولا ينصح باستخدامها، بالرغم من أنها ما زالت تعمل على كثير من المتصفحات بطريقة صحيحة، حتى وقتنا هذا.

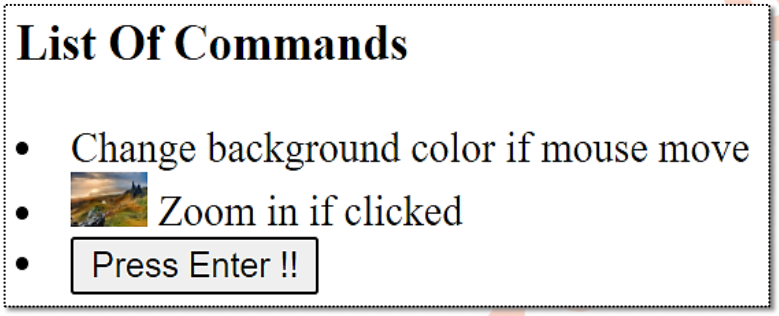
التنسيق الافتراضي:

```
ul {
    display: block;
    list-style-type: disc;
    margin: 1em  0;
    padding: 40px;
    }
```

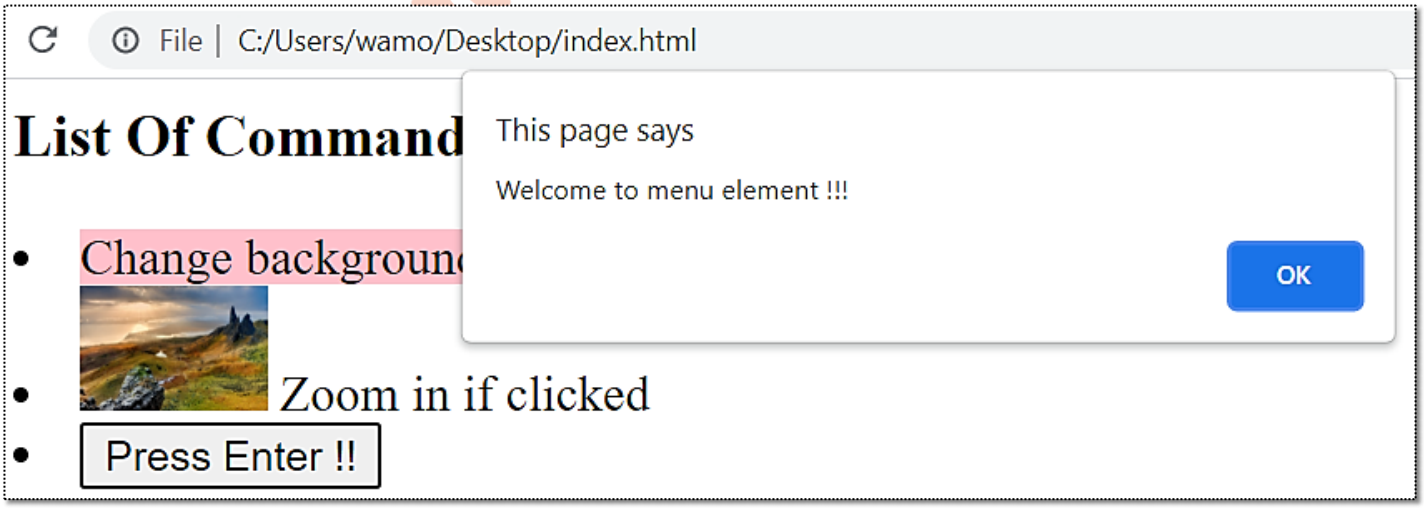
<menu>....</menu>

عنصر قائمة غير مرتبة، مثل القائمة السابقة ، حيث تحتوي بنوداً من العنصر <li> ، والاختلاف بينهما، أن القائمة <ul> تحتوي بنوداً للعرض فقط، بينما القائمة <menu> تحتوي بنوداً تفاعلية، مثل الأزرار أو أي عناصر يتفاعل معها المستخدم، عن طريق حركة الفأرة، أو لوحة المفاتيح، لتؤدي وظيفة معينة.

```
<menu> <h3>List Of Commands</h3>
<li><span onmousemove="style='background:pink;'">Change background color if mouse move</span></li>
<li><img src="mount.jpg" width="30" onclick="style='width:60px'">
Zoom in if clicked</li>
<li><button onclick="alert('Welcome to menu element !!!')"
autofocus>Press Enter !!</button> </li>
</menu>
```



القائمة تضم ثلاثة بنود، الأول تتغير خلفيته عند مرور الفأرة فوقه، والثاني صورة يتم تكبيرها بنقرها بالفأرة، والثالث زر عند نقل تركيز لوحة المفاتيح إليه عن طريق مفتاح tab ثم الضغط على enter فتظهر رسالة معينة، وتكون النتيجة كما في الشكل التالي:



<dl>....</dl>

عنصر قائمة شرح أو description list ، مخصص لانشاء قائمة لشرح مصطلح معين، وتقوم بذلك عن طريق عنصرين داخليين:

<dt>....</dt>

عنصر description term ، مخصص ليحتوي المصطلح أو العنوان المطلوب شرحه.

<dd>....</dd>

عنصر description data ، مخصص ليحتوي البيانات أو المعلومات، التي تصف أو تشرح مفهوم المصطلح المطلوب.

```
<dl>
   <dt><b>HTML</b></dt>
         <dd>Hyper Text Markup Language</dd>
   <dt><b>CSS</b></dt>
         <dd>Cascading Style Sheets</dd>
   <dt><b>XML</b></dt>
         <dd>Extensible Markup Language</dd>
</dl>
```

لاحظ أنني قد قمت بتنسيق المصطلح المطلوب شرحه <dt> بخط سميك حتى يمكن للقارئ أن يميزه عن الوصف أو الشرح.

HTML
Hyper Text Markup Language
CSS
Cascading Style Sheets
XML
Extensible Markup Language

التنسيق الافتراضي:

```
dl  { display:  block; margin:  1em    0; }
dt  { display:  block; }
dd { display:  block;  margin-left:  40px; }
```

Tag Omission

شرحنا من قبل، وجود هذه الميزة في بعض عناصر block elements ، ومن ضمن هذه العناصر التي تتأثر بهذه الميزة، العنصرين <dt> إذا أنشأ داخله عنصر <dd> ، والعنصر <dd> إذا أنشأ داخله عنصر <dt> .

```
<dl  style="width:250px;">
 <dt  style="background:red;">HTML
    <dd  style="background:gold;">Hyper Text Markup Language</dd>
    CSS
  </dt>
</dl>
```

HTML
Hyper Text Markup Language
CSS

لاحظ أن النص CSS لم يتم تنسيقه بنفس تنسيق العنصر <dt> برغم وقوعه داخله، لأن العنصر أغلق بفعل إنشاء العنصر الداخلي (الابن) <dd> .

```
<dl  style="width:250px;">
 <dt  style="background:red;">HTML</dt>
    <dd  style="background:gold;">Hyper Text Markup Language
        <dt>  CSS</dt>
              <dd>Cascading style sheets</dd>
    </dd>
</dl>
```

HTML
Hyper Text Markup Language
CSS
Cascading style sheets

لاحظ أن العنصرين <dt> و <dd> في السطرين الرابع والخامس، لم يتم تنسيقهما بنفس تنسيق العنصر الأب <dd> في السطر الثالث رغم وقوعهما داخله.

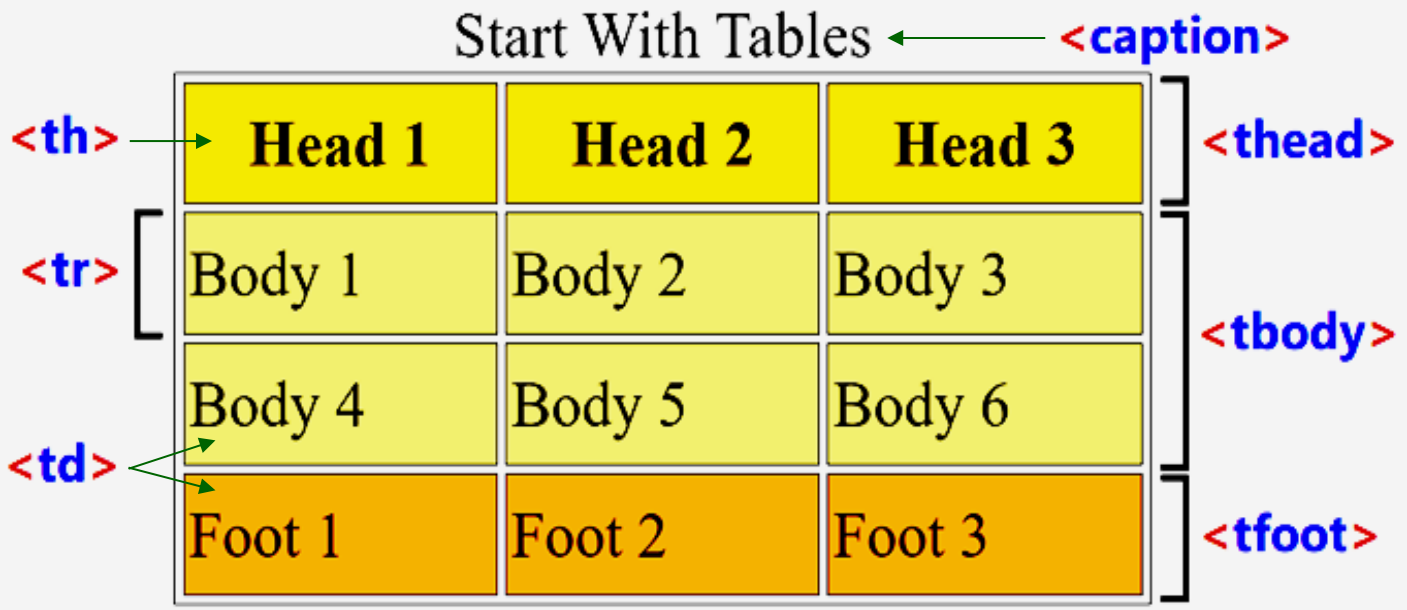
عناصر الجدول Table Elements

مجموعة عناصر تبدأ بالعنصر الرئيسي <table> ، وداخله مجموعة عناصر داخلية أخرى، مخصصة لإنشاء الجداول ذات الصفوف والأعمدة، ومنها عناصر مخصصة لإنشاء الصفوف والخلايا ورؤس الصفوف وغيرها.

<table>....</table>

العنصر الرئيسي وهو مخصص لاحتواء جميع عناصر مكونات الجدول.

```
<head>
<style> table, th, td { border: 0.5px  solid; }
        th, td { width: 80px; height: 30px; }</style>
</head>
<body>
<table><caption> Start With Tables</caption>
    <thead style="background: gold ">
            <th> Head 1</th><th> Head 2</th><th> Head 3</th>
    </thead>
    <tbody  style="background:  khaki">
          <tr>
             <td>Body 1</td><td> Body 2</td><td> Body 3</td>
          </tr>
          <tr>
             <td>Body 4</td><td> Body 5</td><td> Body 6</td>
          </tr>
    </tbody>
    <tfoot  style="background: orange">
        <tr>
            <td>Foot 1</td><td> Foot 2</td><td> Foot 3</td>
        </tr>
    </tfoot>
</table>
</body>
```



Start With Tables

Head 1	Head 2	Head 3
Body 1	Body 2	Body 3
Body 4	Body 5	Body 6
Foot 1	Foot 2	Foot 3

لاحظ أننا قمنا باستخدام العنصر <style> داخل العنصر <head> ليقوم بإعطاء حدود للجدول والخلايا، وعرض وارتفاع لتوضيح محازاة النص، واستخدمنا الخاصية style داخل العناصر <thead> و<tbody> و<tfoot> لإعطائهم لون خلفية، حتى لا يظهر الجدول كما بالشكل المقابل، وبالتالي لن تستطيع التمييز بين أجزائه.

التنسيق الا فتراضي:

```
table {
        display: table;
        border-collapse: separate;
        border-spacing: 2px;
        border-color: gray;
       }
```

<caption>....</caption>

عنصر مخصص لإنشاء عنوان للجدول، ودائماً ما يكون أول عنصر بعد وسم بداية الجدول، وموضعه الافتراضي أعلى منتصف الجدول مهما كان موضعه داخل الكود.

التنسيق الافتراضي:

```
caption {
        display: table-caption;
        text-align:  center;
        }
```

<thead>....</thead>

عنصر مخصص لاحتواء صف يكون رأساً لجميع الأعمدة، والمحازاة الرأسية للنص داخله تكون في منتصف الصف ، وهو عنصر غير مؤثر في بناء الجدول، إذ يمكن بناء الجدول بدونه، ولكنه يمكن أن يستخدم في تنسيق خلايا رأس الجدول دفعة واحدة، كما يستخدمه المتصفح للانتقال مباشرة إلى رأس الجدول، وترتيب هذا العنصر يأتي بعد العنصرين <caption> و<colgroup> ، وعموماً، فأياً كان ترتيب العنصر <thead> داخل الكود، فإن الصفوف داخله ستعرض دائماً في مقدمة الجدول.

التنسيق الافتراضي:

```
thead {
        display: table-header-group;
        vertical-align: middle;
         border-color:  inherit;
        }
```

<th>....</th>

عنصر داخلي إما للعنصر <thead> ، أو العنصر <tr> وهو مخصص لإنشاء خلايا الصفوف العلوية أو رؤوس الأعمدة، حيث يعرض النص داخله بخط سميك bold، وتكون

المحازاة الأفقية للنص داخل الخلية، في منتصف الخلية "أفقياً".

```
<style>table, th, td { border: 0.5px solid } </style>
<table>
 <tr><th>Head 1</th><th>Head 2</th><th>Head 3</th></tr>
 <tr><td>Body 1</td><td> Body 2</td><td> Body 3</td></tr>
</table>
```

Head 1	Head 2	Head 3
Body 1	Body 2	Body 3

ويتم تعيين عدة خصائص للعنصر <th> تتحكم في سلوكه:

abbr

خاصية تعطي وصفاً مختصراً لمحتوى خلية معينة، في صف عنوان الأعمدة، ولكن لن تلاحظ لها أي تأثير مرئي، ولكن يتم قراءة قيمتها عن طريق برامج قراءة الشاشة.

```
 <th abbr="First head cell of the table">Head 1</th>
```

rowspan

خاصية تسمح للخلية بالتمدد رأسياً، واحتلال مكان خليتين أو أكثر، حيث تقوم الخلية العليا بشغل العدد المطلوب من الخلايا، فتقوم بإزاحة كل خلايا العمود إلى الأسفل، بنفس عدد الخلايا التي شغلتها، لذلك يجب أن يكون عدد خلايا هذا العمود أقل من عدد خلايا باقي أعمدة الجدول، بنفس عدد الخلايا التي ستضمها إليها هذه الخلية، وقيمة هذه الخاصية تكون هي عدد الصفوف التي ستشغلها الخلية بعد ضم الخلايا الأخرى، فإذا كانت الخلية ستضم إليها خلية واحدة ، تكون قيمة الخاصية 2 .

```
<style>table, th, td {border: 0.5px   solid } </style>
<table>
 <tr>
  <th rowspan="2">Head 1</th><th>Head 2</th><th>Head 3</th>
 </tr>
 <tr><td>Body 1</td><td> Body 2</td></tr>
</table>
```

Head 1	Head 2	Head 3
Body 1	Body 2	

original

Head 1	Head 2	Head 3
	Body 1	Body 2

rowspan="2"

تجنب استخدام القيمة 0 للخاصية rowsapn لأن تأثيرها يختلف بين المتصفحات :

```
<style>table, th, td {border: 0.5px  solid } </style>
<table>
  <tr><td rowspan="0">Col</td></tr>
  <tr><td>Row 1</td><td>Row 1</td><td>Row 1</td></tr>
  <tr><td>Row 2</td><td>Row 2</td><td>Row 2</td></tr>
  <tr><td>Row 3</td><td>Row 3</td><td>Row 3</td></tr>
</table>
```

Col	Row 1	Row 1	Row 1
	Row 2	Row 2	Row 2
	Row 3	Row 3	Row 3

Google Chrome

Col		
Row 1	Row 1	Row 1
Row 2	Row 2	Row 2
Row 3	Row 3	Row 3

Mozilla Firefox

colspan

خاصية تسمح للخلية بالتمدد أفقياً واحتلال مكان خليتين أو أكثر، حيث تقوم الخلية اليسرى بالتمدد إلى اليمين، فتقوم بإزاحة كل خلايا الصف إلى اليمين بنفس عدد الخلايا التي ضمتها، لذلك يجب أن يكون عدد خلايا الصف العلوي، أقل من عدد خلايا الصف أو الصفوف السفلية بخلية واحدة أو أكثر، حسب عدد الخلايا أو الأعمدة المطلوب أن تتمدد عليها هذه الخلية، وقيمة الخاصية تكون هي عدد الخلايا التي ستشغلها الخلية، فإذا كانت هذه الخلية ستضم إليها خليتين آخريين، فتكون قيمة الخاصية هي 3 .

```
<style>table, th, td {border: 0.5px solid } </style>
<table>
 <tr><th  colspan="2">Head 1</th><th>Head 2</th></tr>
 <tr><td>Body 1</td><td> Body 2</td><td> Body 3</td></tr>
</table>
```

Head 1	Head 2	
Body 1	Body 2	Body 3

original

Head 1		Head 2
Body 1	Body 2	Body 3

colspan="2"

headers

خاصية تقوم بربط خلايا th ببعضها عن طريق، إسناد قيمة id أحدها لباقي الخلايا، وهذا السلوك ليس له أثر مرئي للمستخدم، ولكن أثره يظهر عند استخدام برامج قراءة الشاشة.

```
<table>
 <tr><th id="first_cell">Head 1</th><th>Head 2</th></tr>
 <tr><td headers="first_cell">Body 1</td><td> Body 2</td></tr>
</table>
```

scope

خاصية تحدد هل th هي رأس عمود، أم رأس صف، أم رأس مجموعة من الأعمدة، أو الصفوف، وتقبل عدة قيم وهي: col وrow وcolgroup وrowgroup ومثلها مثل الخاصية السابقة، ليس لسلوكها أثر مرئي يمكن أن يلاحظه المستخدم، وإنما أثرها يكون عند استخدام برامج قراءة الشاشة.

```
<table>
 <tr><th scope="col">Head 1</th><th>Head 2</th></tr>
 <tr><td>Body 1</td><td> Body 2</td></tr>
</table>
```

التنسيق الافتراضي:

```
th {
    display: table-cell;
     text-align:  center;
     vertical-align: inherit;
     font-weight: bold;
     }
```

<tbody>....</tbody>

عنصر مخصص لاحتواء الصفوف، التي تكوِّن خلايا جسم الجدول، والمحازاة الرأسية للنص داخله تكون في منتصف الصف، ويجب أن يحتوي داخله صف واحد على الأقل، ولكنه غير مؤثر في بناء الجدول، إذ يمكن بناء الجدول بدونه، ولكن يمكن أن يستخدم في تنسيق مجموعة صفوف جسم الجدول، كما يمكن أن يستخدمه المتصفح للانتقال مباشرة إلى جسم الجدول، وترتيب هذا العنصر يأتي بعد العنصر <caption> والعنصر <colgroup> والعنصر <thead> ، وعموماً، وأياً كان ترتيب العنصر <tbody> داخل الكود، فإن الصفوف داخله ستعرض دائماً بين العنصرين <thead> و <tfoot> .

التنسيق الافتراضي:

```
tbody {
        display:  table-row-group;
        vertical-align: middle;
         border-color:  inherit;
         }
```

<tfoot>....</tfoot>

عنصر مخصص لاحتواء آخر صف أو مجموعة صفوف في الجدول، والمحازاة الرأسية للنص داخله في منتصف الصف، وهو عنصر غير مؤثر في بناء الجدول، إذ يمكن بناء الجدول بدونه، ولكن يمكن أن يستخدم في تنسيق مجموعة صفوف ذيل الجدول، كما يمكن للمتصفح

أن يستخدمه للانتقال مباشرة إلى ذيل الجدول، وترتيب هذا العنصر يأتي بعد العناصر <caption> و<colgroup> و<thead> و <tbody> ، وعموماً، فأياً كان ترتيب العنصر <tfoot> داخل الكود، ستعرض الصفوف داخله في مؤخرة الجدول دائماً.

التنسيق الافتراضي:

```
tfoot {
        display:  table-footer-group;
        vertical-align: middle;
         border-color:  inherit;
        }
```

<tr>....</tr>

عنصر مخصص لانشاء صف في الجدول، ويحتوي داخله العنصر <td> ، أو العنصر <th> المسئولين عن إنشاء الخلايا داخل الصفوف.

التنسيق الافتراضي:

```
tr {
     display:  table-row;
    vertical-align:  inherit;
    border-color:  inherit;
   }
```

<td>....</td>

عنصر مخصص لإنشاء الخلايا في صف من الجدول، والمحازاة الأفقية للنص داخلها تكون لليسار left-aligned ، ويكون سمك الخط normal ، ويمكن أن نعين لهذا العنصر الخصائص السابق شرحها مع العنصر <th> مثل headers و colspan وrowspan .

التنسيق الافتراضي:

```
td {
    display:  table-cell;
```

```
    vertical-align:  inherit;
    text-align: left;
   }
```

<colgroup>....</colgroup>

عنصر مخصص لتنسيق عمود أو مجموعة من أعمدة الجدول بتنسيق معين، ويتم تحديد عدد الأعمدة التي سيطبق عليها هذا التنسيق، عن طريق الخاصية span ، وترتيب هذا العنصر يأتي بعد العناصر <table> و<caption> وقبل العناصر <thead> و<tbody> و<tfoot> و<tr> .

```
<style>table, td {border: 0.5px   solid } </style>
<table>
 <colgroup  span="1" style="width: 50px; background: gold;">
 </colgroup>
<tr><td>Col 1</td><td>Col 2</td><td>Col 3</td><td>Col 4
</td></tr>
<tr><td>Col 1</td><td>Col 2</td><td>Col 3</td><td>Col 4
</td></tr>
<tr><td>Col 1</td><td>Col 2</td><td>Col 3</td><td>Col 4
</td></tr>
<tr><td>Col 1</td><td>Col 2</td><td>Col 3</td><td>Col 4
</td></tr>
</table>
```

Col 1	Col 2	Col 3	Col 4
Col 1	Col 2	Col 3	Col 4
Col 1	Col 2	Col 3	Col 4
Col 1	Col 2	Col 3	Col 4

span = "1"

span

خاصية تقبل قيمتها كعدد صحيح موجب، لتحديد عدد الأعمدة التي سيطبق عليها تنسيق العنصر <colgroup> ، وإذا أسند للخاصية أي قيمة أصغر من 1 ، فإن الخاصية ستعتبرها 1 ، ويجب ألا تزيد قيمتها عن عدد أعمدة الجدول.

Col 1	Col 2	Col 3	Col 4
Col 1	Col 2	Col 3	Col 4
Col 1	Col 2	Col 3	Col 4
Col 1	Col 2	Col 3	Col 4

span = "4"

Col 1	Col 2	Col 3	Col 4
Col 1	Col 2	Col 3	Col 4
Col 1	Col 2	Col 3	Col 4
Col 1	Col 2	Col 3	Col 4

span = "2"

التنسيق الافتراضي:

```
colgroup {
            display:  table-column-group;
           }
```

<col>

عنصر empty element داخلي للعنصر <colgroup> ينسق مجموعة من الأعمدة، بأكثر من تنسيق مختلف، باستخدام قيمة الخاصية span .

```
<style>table, td {border: 0.5px solid } </style>
<table>
 <colgroup>
    <col span="1" style="width: 40px; background: gold;">
    <col span="2" style="width: 50px; background: orange;">
    <col span="1" style="width: 60px; background: tomato;">
</colgroup>
<tr><td>Col 1</td><td>Col 2</td><td>Col 3</td><td>Col 4
</td></tr>
<tr><td>Col 1</td><td>Col 2</td><td>Col 3</td><td>Col 4
</td></tr>
<tr><td>Col 1</td><td>Col 2</td><td>Col 3</td><td>Col 4
</td></tr>
<tr><td>Col 1</td><td>Col 2</td><td>Col 3</td><td>Col 4
```

```
</td></tr>
</table>
```

Col 1	Col 2	Col 3	Col 4
Col 1	Col 2	Col 3	Col 4
Col 1	Col 2	Col 3	Col 4
Col 1	Col 2	Col 3	Col 4

لاحظ أن قيمة الخاصية span لأول عنصر <col> كانت 1 فتم تطبيق التنسيق على أول عمود، وكانت قيمتها للعنصر الثاني هي 2 ، فتم تطبيقها على العمودين الثاني والثالث، وكانت قيمتها للعنصر الثالث هي 1 ، فتم تطبيقها على العمود الرابع.

إذا احتوى العنصر <colgroup> على تنسيقات وكان داخله ولو عنصر <col> واحد فقط، ويستهدف ولو عمود واحد فقط، فإنه يلغي العنصر <colgroup> وتنسيقاته لكل الأعمدة وهذا ما يسمى Tag Omission ، حيث ألغي العنصر الابن وسم بداية العنصر الأب.

إذا تعددت العناصر <colgroup> داخل الجدول، فإن أولوية التنسيق تكون للعنصر الأول، ثم يستكمل العنصر الثاني تنسيق الأعمدة التي لم يستكملها العنصر الأول بالترتيب، ثم الثالث والرابع وحتى آخر عنصر.

يمكن استخدام العنصر <col> في تنسيق الأعمدة كعنصر مستقل، دون أن يكون داخل عنصر أب من النوع <colgroup> .

التنسيق الافتراضي:

```
col {
    display:  table-column;
    }
```

في بعض الأكواد السابقة قمت بوضع العنصر <style> قبل العنصر <table> مباشرة، لتقليل مساحة الكود ولعدم إصابة القارئ بالملل من زيادة عدد أسطر الكود، ولكن الأصل، أن العنصر <style> يجب أن يتم إنشاؤه دائماً داخل العنصر <head> ، حيث إنه من عناصر metadata ، والتي سنتناولها لاحقاً بالشرح.

Tag Omission

في كثير من عناصر الجدول يمكن الاستغناء عن وسم الإغلاق ، كما أن العناصر الأبناء تقوم بإغلاق العنصر الأب، وقد سبق أن شرحنا كيفية حدوث ذلك على بعض العناصر السابقة، ولا حاجة لشرح أمثلة جديدة، وسوف نشير فقط، إلى هذه العناصر وطريقة عملها وعليك أن تتوقع نفس سلوك والتأثير المشابه للعناصر السابق شرحها.

1- العنصر <caption> : يمكن الاستغناء عن وسم الإغلاق closing tag ، فبمجرد إنشاء عنصر آخر بعد محتوى هذا العنصر، يتم إغلاق العنصر <caption> تلقائياً.

2- العنصر <tbody> : ليس عنصراً أساسياً في بناء الجدول، وهو مخصص ليحتوي صفوف جسم الجدول، ولكن في حالة وجود صفوف من النوع <tr> كأبناء مباشرة للجدول الأب، فلا يجب أن يتواجد عنصر <tbody> في بناء الجدول.

3- العنصر <thead> : يمكن الاستغناء عن وسم الإغلاق، إذا تبع محتواه عنصر <tbody> أو عنصر <tfoot> .

4- العنصر <tfoot> : يمكن الاستغناء عن وسم الإغلاق، إذا كان آخر عنصر في الجدول.

5- العنصر <tr> : يمكن الاستغناء عن وسم الإغلاق، إذا تبع محتواه عنصر <tr> آخر، أو كان هو آخر عنصر ابن داخل العنصر الأب، سواء كان العنصر الأب هو <tbody> أو <thead> أو <tfoot> .

6- العنصر <th> : يمكن الاستغناء عن وسم الإغلاق، إذا تبع محتواه عنصر <th> آخر، أو عنصر <td> ، أو كان هو آخر عنصر ابن داخل العنصر الأب.

7- العنصر <td> : يمكن الاستغناء عن وسم الإغلاق، إذا تبع محتواه عنصر <td> آخر، أو عنصر <th> ، أو كان هو آخر عنصر ابن داخل العنصر الأب.

wailmonir0@protonmail.com

عناصر النموذج Form Elements

هي العناصر التي توضع داخل عنصر النموذج أو <form> ، وتستخدم للتفاعل مع المستخدم، عن طريق إدخال بعض البيانات، ليتم استخدامها في موضع آخر عند اعتماد النموذج أو submit ، حيث يتم إرسالها إلى صفحة أخرى أو إلى خادم server ، ليقوم بمعالجتها، لاستخراج معلومات يطلبها المستخدم الحالي أو مستخدمين آخرين.

وسنبدأ بالعنصر الرئيسي الذي يحتوي جميع العناصر وهو العنصر <form> .

<form>....</form>

العنصر الأب الذي يحتوي كل عناصر النموذج الأخرى، وهو عنصر غير مرئي إلا إذا تم تنسيقه بال css ، وبالرغم من أن العناصر الداخلية لهذا العنصر يمكن أن تظهر للمستخدم حتى ولو لم توضع داخله، إلا أن هذه العناصر لا يتم عمل submit لها ، أي لا يتم إرسال بياناتها إلى الخادم أو صفحة أخرى، إلا إذا كانت داخل عنصر <form> .

التنسيق الافتراضي:

```
form {
        display:  block;
        margin-top:  0em ;
       }
```

ويتم تعيين العديد من الخصائص للعنصر <form> :

accept-charset

خاصية تحدد نوع ترميز الحروف الذي سيسخدمه عنصر <form> أو النموذج ، حتى يفهم المتصفح اللغة المكتوبة، والتي يستخدمها عنصر <form> في اعتماد أو عمل submit للبيانات التي تم إدخالها.

ويوجد أنواع عديدة من الترميز أشهرها:

- ***utf-8*** : ويكون الترميز حسب المعايير الموحدة unicode ، التي تقوم بترميز كل الحروف والرموز في كل اللغات تقريباً.
- ***ISO-8859-1*** : ويكون الترميز حسب معايير الأحرف اللاتينية، حيث يتم ترميز حروف

ascii (وهي أغلب الحروف والرموز المستخدمة في اللغة الإنجليزية) بالإضافة لترميز عدد من الرموز اللاتينية، مثل رمز الجنية الاسترليني ورمز السنت الأمريكي وغيرها.

action

خاصية تقوم بتحديد الصفحة أو الخادم، الذي سيستقبل البيانات المرسلة من <form> بعد اعتمادها، ليتم معالجتها والتعامل معها بعد ذلك.

autocomplete

خاصية تتيح للمتصفح اقتراح استكمال بيانات النموذج أثناء إدخالها، عن طريق قائمة منسدلة، تقترح بيانات أخرى قد تم إدخالها سابقاً، ويسند لها قيمتين:

- ***on*** : قيمة تعني أن المتصفح يستطيع اقتراح بيانات لاستكمال عملية الإدخال.
- ***off*** : قيمة تعني منع المتصفح من تقديم أي اقتراح لاستكمال البيانات.

```
<form  autocomplete="on">
<input  type="text"  name="nm" value="" autocomplete="off">
<input  type="email"   name="ml" value="">
<input type="submit" value="Send">
</form>
```

method

خاصية تحدد طريقة إرسال بيانات النموذج إلى السيرفر أو صفحة أخرى بعد اعتمادها، وتسند لها قيمتين هما get و post وشرح مدلول القيمتين في ملحق 1 .

enctype

طريقة تشفير بيانات النموذج عند دخولها إلى السيرفر بعد اعتمادها، ويسند لها عدة قيم، والقيم ومدلولها ملحق 1 .

multiple

خاصية تسمح للعنصر بإدخال أكثر من قيمة، أو أكثر من اختيار، مثل النوع email للعنصر <input> حيث تسمح بإدخال أكثر من بريد الكتروني، ومثل العنصر <select> حيث تسمح باختيار أكثر من واحد من قائمة اختيارات هذا العنصر.

name

خاصية تقوم بتحديد اسم للنموذج الذي نتعامل معه.

novalidate

خاصية تعني عدم خضوع بيانات حقول النموذج، لعملية مراجعة ومطابقة للمعايير الخاصة بكل عنصر دا خلي، قبل اعتمادها.

rel

خاصية تحدد نوع العلاقة بين النموذج والصفحة أو السيرفر الذي ستنتقل إليه البيانات بعد اعتمادها، وشرح جميع قيم هذه الخاصية ملحق 1 .

target

خاصية تحدد موضع فتح الصفحة أو النموذج، الذي سيقوم النموذج بالانتقال إليه بعد اعتماده، ويسند لها أي من القيم التالية: _blank و _parent و _self و _top وframename ، وشرح مدلول هذه القيم ملحق 1 .

وسوف نستعرض العناصر الداخلية، التي يتم إنشاؤها داخل النموذج أو عنصر <form> الأب، والتي تم تخصيصها للتعامل والتفاعل مع المستخدم بطرق عديدة، لاستقبال البيانات منه .

wailmonir0@protonmail.com

<input>

عنصر empty element مخصص لإدخال واستقبال البيانات من المستخدم، وهذا العنصر له عدة خصائص، وبعضها يتشابه مع مثيلاتها في النموذج أو عنصر <form> الأب، الذي يحتوي هذا العنصر الداخلي، ويختلف عنها في أن قيم الخصائص لعنصر النموذج الأب يتم تطبيقها على كافة عناصر النموذج، أما القيم التي يتم إسنادها إلى العنصر الداخلي ومنها العنصر <input> ، يتم تطبيقها على هذا العنصر فقط، ويتم إيقاف تأثير قيمة الخاصية في النموذج الأب على هذا العنصر، ولاحظ تأثير الخاصية autocomplete السابق شرحها على العنصر <form> والعنصرين الداخليين <input> في المثال المرفق ص 105 .

ويمكن تعيين عدة خصائص للعنصر <input> :

type

خاصية تحدد نوع عنصر الإدخال، ويسند لها عدة أنواع من القيم:

button : زر.

Login

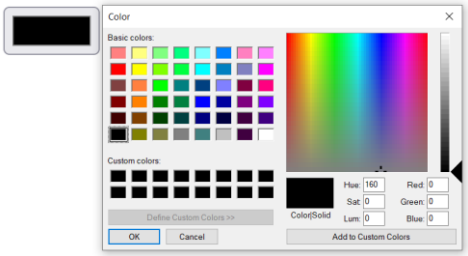
```
<input type="button" value="Login">
```

checkbox : حقل اختيار أو تأشير.

Watched

```
<input type="checkbox" checked> Watched
```

color : حقل اختيار الألوان.




```
<input type="color" value="#4d5fae" name="clr">
```

file : حقل اختيار ملف للتحميل.

Choose File No file chosen

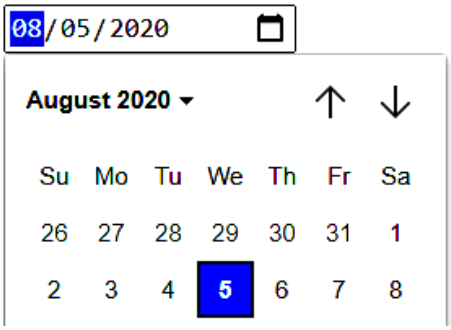
```
<input type="file" accept="application/pdf,video/mp4" multiple>
```

email : حقل إدخال البريد الإلكتروني.

someone@mail.com

```
<input type="email" value="someone@mail.com" readonly>
```

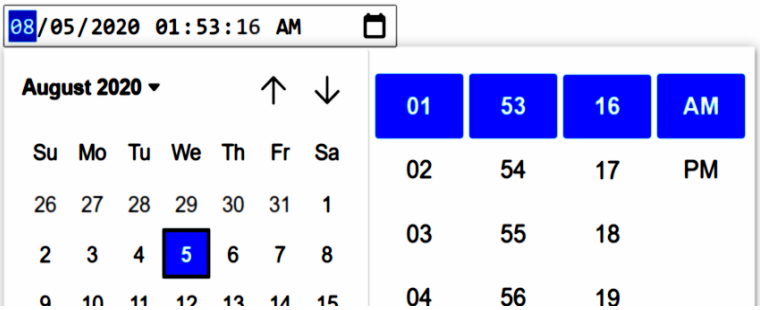
date : حقل إدخال التاريخ.




```
<input type="date" value="2020-08-05" name="dt" >
```

datetime-local :

حقل إدخال التاريخ والوقت.




```
<input type="datetime-local" value="2020-08-05T01:53:16" >
```

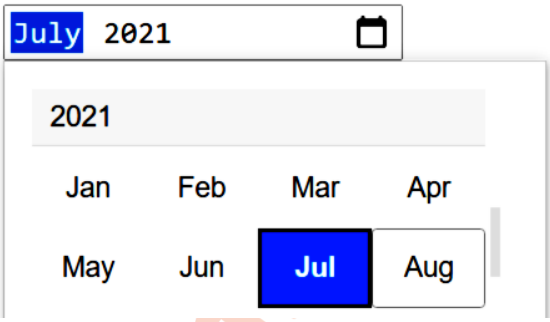
hidden : حقل مخفي.

```
<input type="hidden" id="pswrd">
```

image : زر على هيئة صورة.

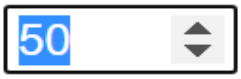
```
<input type="image" src="mount.jpg" alt="nice_view">
```

month : حقل إدخال الشهر.




```
<input type="month" value="2021-07" name="mnth">
```

number : حقل إدخال عدد معين.




```
<input type="number" min="20" max="50" step="5">
```

password : حقل إدخال كلمة سر.

●●●●●

```
<input type="password" name="pswrd">
```

radio : زر راديو.

Yes No

```
<input type="radio" name="yes_no" checked>Yes
<input type="radio" name="yes_no">No
```

range : شريط منزلق.

```
<input type="range" value="50" max="100" step="5">
```

reset : زر مسح البيانات وإعادة الوضع الأصلي.

Start Again

```
<input type="reset" value="Start Again">
```

submit : زر اعتماد وإرسال البيانات.

Send

```
<input type="submit" value="Send">
```

search : حقل بحث.

Find Something ✕

```
Find <input type="search" value="Something">
```

tel : حقل إدخال رقم تليفون.

Enter your phone no.

```
<input type="tel" placeholder="Enter your phone no.">
```

text : حقل إدخال نص.

Name: Enter your name

```
Name: <input type="text" placeholder="Enter your name">
```

url : حقل إدخال عنوان موقع إلكتروني

Website: www.somesite.com

```
Website: <input type="url" placeholder="www.somesite.com">
```

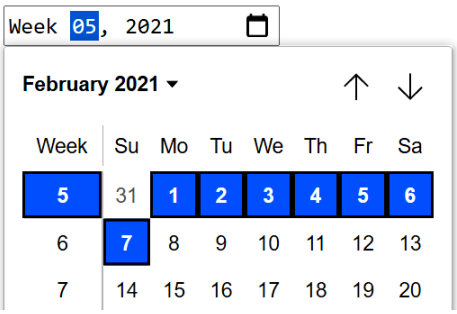
time : حقل إدخال الوقت.

07:30:45 PM

07	30	45	PM
08	31	46	AM
09	32	47	
10	33	48	

```
<input type="time" name="tm" value="19:30:45" required>
```

week : حقل إدخال الأسبوع.




```
<input  type="week"   value="2021-W07" name="wk">
```

نظرة سريعة على باقي خصائص العنصر <input>

accept

خاصية للنوع file ، حيث تحدد أنوع الملفات التي يتم قبول تحميلها.

alt

خاصية تعرض نص عند حدوث خطأ في عرض الصورة، في النوع image.

autocapitalize

خاصية تقوم بتحويل الحروف التي يتم إدخالها إلى حروف كبيرة، ولا يظهر تأثير هذه الخاصية على لوحة المفاتيح المادية (الخاصية بالحاسب)، ولكن تأثيرها يكون على لوحة المفاتيح الافتراضية للأجهزة المحمولة (الهاتف النقال أو التابلت)، ولا يظهر تأثيرها على النوع password أو النوع email أو النوع url، وتقبل عدة قيم:

- ***off | none*** : قيمة تعني عدم تفعيل الخاصية.
- ***on | sentence*** : قيمة تعني تطبيق الخاصية على أول حرف في الجملة.
- ***words*** : قيمة تعني تطبيق الخاصية على أول حرف في كل كلمة.
- ***characters*** : قيمة تعني أن كل الحروف يتم إدخالها كحروف كبيرة.

autocomplete

خاصية تسمح للمتصفح باقتراح استكمال النص الذي يتم إدخاله، وسبق شرحها ص 105 .

autofocus

خاصية boolean attribute ، ووجودها يعني تحول التركيز التلقائي للوحة المفاتيح، على

wailmonir0@protonmail.com

العنصر في بداية تحميل الصفحة أو عند الانتقال للنموذج.

checked

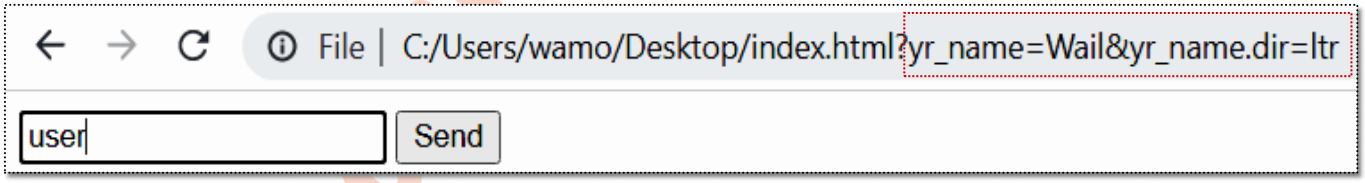
خاصية للنوع checkbox والنوع radio ، تعرض العنصر وقد تم اختياره تلقائياً عند تحميل الصفحة .

dirname

خاصية تحدد اتجاه عنصر <input> عند اعتماده ويتم ذلك بتعيين الخاصية dirname للنوع text أو النوع search ويسند لها قيمة عبارة عن اسم العنصر (وهو قيمة الخاصية name) متبوعاً باللاحقة dir، فإذا كان اسم العنصر yr_name فتكون قيمة الخاصية dirname هي yr_name.dir كما بالكود التالي:

```
<form>
<input  type="text"   name="yr_name" value="user"
dirname="yr_name.dir">
<input type="submit" value="Send">
</form>
```

وعند اعتماد النموذج يتم قبول البيانات على جزئين، بينهما "&" ، ويكون الجزء الأول هو قيمة الخاصية name ، مسند إليها قيمة الخاصية value ، والجزء الثاني هو قيمة الخاصية dirname ، مسند إليها اتجاه العنصر من النوعين text أو search ، كما بالشكل التالي:

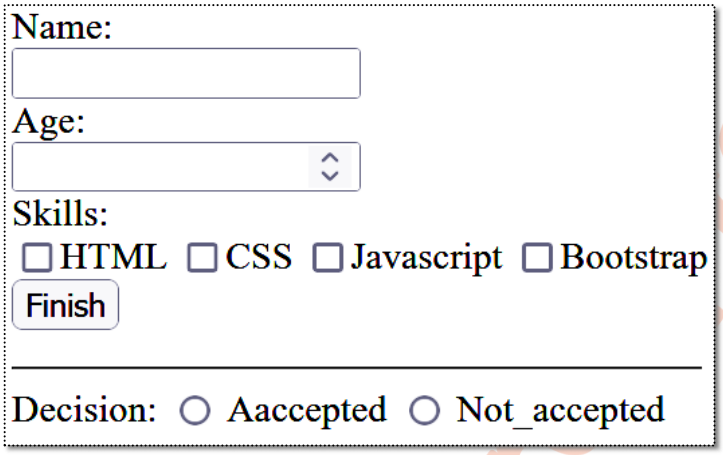


لاحظ شريط العنوان، وكيفية قبول البيانات بعد اعتمادها، ولاحظ أن حقل الاسم تحول إلى القيمة الافتراضية، بعد اعتماد البيانات التي تم إدخالها.

form

خاصية تسمح لعنصر موجود خارج النموذج الحالي وليكن A ، أو داخل نموذج آخر وليكن B ، بالارتباط بالنموذج A عن طريق إسناد قيمة المعرف الفريد id للنموذج A ، كقيمة للخاصية form التي تم تعيينها لهذا العنصر، وبالتالي عند اعتماد هذا النموذج A ، يتم قبول مدخلات هذا العنصر ، كما لو كان عنصراً من عناصر النموذج A .

```
<form id="user_info">
Name:<br><input type="text" name="user_name"><br>
Age:<br><input type="number"><br>
Skills:<br><input type="checkbox">HTML
          <input type="checkbox">CSS
          <input type="checkbox">Javascript
          <input type="checkbox">Bootstrap<br>
<input type="submit" value="Finish" >
</form><hr>
Decision:
<input type="radio" name="dec" form="user_info"> Aaccepted
<input type="radio" name="dec" form="user_info"> Not_accepted
```

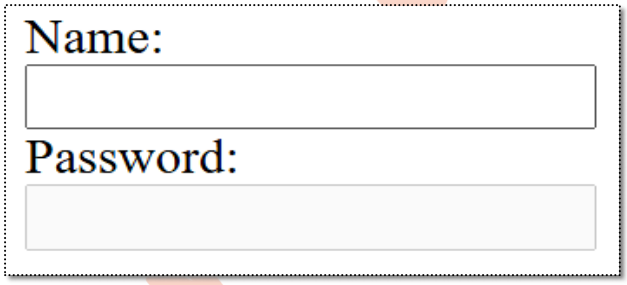


لاحظ أن الحقول بعد وسم إغلاق العنصر <form> ، يتم اعتمادها عند اعتماد بيانات النموذج، بالرغم من وقوعها خارجه، وذلك لارتباطها به عن طريق الخاصية form .

disabled

خاصية تجعل العنصر غير فعال وبالتالي لا يستطيع المستخدم التعامل معه.

```
Name:<br><input type="text" name="user_name"><br>
Password:<br><input type="password" disabled >
```



لاحظ أن العنصر <input> من النوع password ظهر باهتاً وغير فعال، ولا يمكن للمستخدم التعامل معه.

formaction

وهي نفس الخاصية action التي تم شرحها في العنصر <form> وتختص بالنوعين image و submit .

formenctype

وهي نفس الخاصية enctype التي تم شرحها في العنصر <form> ، وتختص بالنوعين image و submit .

formnovalidate

وهي نفس الخاصية novalidate التي تم شرحها في العنصر <form> ، وتختص بالنوعين image و submit .

formmethod

وهي نفس الخاصية method التي تم شرحها في العنصر <form> ، وتختص بالنوعين image و submit.

formtarget

وهي نفس الخاصية target التي تم شرحها في العنصر <form> ، وتختص بالنوعين image و submit.

height

خاصية تحدد ارتفاع العنصر وتستخدم مع النوع image .

list

خاصية تربط بين عنصر إدخال وقائمة منسدلة من العنصر <datalist> ، فعند وضع المؤشر داخل العنصر، تفتح القائمة لاختيار أحد بنودها، وذلك عن طريق إسناد قيمة المعرف الفريد لهذه القائمة، كقيمة للخاصية list التي يتم تعيينها لعنصر الإدخال.

max

خاصية تحدد أعلى قيمة يتم إسنادها لنوع معين، مثل number و range .

min

خاصية تحدد أقل قيمة يتم إسنادها لنوع معين، مثل number وrange .

wailmonir0@protonmail.com

maxlength

خاصية تحدد أقصى عدد ممكن من الحروف التي سيتم إدخالها إلى العنصر.

minlength

خاصية تحدد أقل عدد ممكن من الحروف التي سيتم إدخالها إلى العنصر .

multiple

خاصية تسمح للعنصر من النوع file أن يقوم باختيار أكثر من ملف للتحميل، وتسمح للنوع email بإدخال أكثر من بريد إلكتروني.

name

خاصية تحدد اسم النوع، وترجع أهمية هذه الخاصية لاستخدام قيمتها في اعتماد النموذج، بالإضافة إلى أنها تربط عناصر النوع radio في مجموعة واحدة.

pattern

خاصية تسند إليها قيمة عبارة عن عينة regular expression تستخدم لمطابقة مدخلات عنصر معين مع هذه العينة، مثل عينة محددة لكلمة السر أو البريد الإلكتروني.

placeholder

خاصية إدخال تلميح يوضح للمستخدم طبيعة البيانات المطلوب إدخالها.

readonly

خاصية تجعل العنصر معروضاً للقراءة فقط، ولا يمكن التعديل عليه بأي وسيلة، لذلك يجب أن يتم تعيين الخاصية value للعنصر، حتى تعرض قيمتها داخله.

required

خاصية تفيد أن العنصر مطلوب إدخاله، وبدونه لن يتم اعتماد أو عمل submit للنموذج، الذي يحتوي هذا العنصر.

size

خاصية تحدد عرض العنصر، مقدراً بعدد الأحرف التي ستظهر من المدخلات.

wailmonir0@protonmail.com

spellcheck

خاصية تحدد إمكانية تحديد الكلمات التي بها خطأ هجائي، ويسند لها ثلاث قيم:

- ***true*** : قيمة تعني إمكانية فحص النص وإظهار مواضع الأخطاء.
- ***false*** : قيمة تعني عدم فحص النص وبالتالي تجاهل مواضع الأخطاء.
- ***default*** : قيمة تعني أن الخاصية يسند إليها نفس قيمة الخاصية للعنصر الأب.

src

خاصية تحدد مسار أو مصدر الصورة للعنصر من النوع image .

step

خاصية تحدد حجم خطوة الانتقال لأعلى أو لأسفل، بين قيم العنصر من النوع number أو range أو date أو time أو month أو week .

value

خاصية تسند للعنصر قيمة مبدأية عند تحميل الصفحة، تظهر داخل العنصر إذا كان نوع العنصر مربع إدخال، أو تظهر القيمة مكتوبة فوقه، إذا كان النوع زراً .

width

خاصية تحدد عرض الصورة المستخدمة في النوع image .

وسوف ندرس هذه الخصائص مرة أخرى بشيء من التفصيل، ونعاين تأثير ها على الأنواع المختلفة، عندما نشرح أنواع العنصر <input> بعد دراستنا للعنصر <label> .

wailmonir0@protonmail.com

<label>....</label>

عنصر يقوم بإضافة نص بعد أو قبل بعض عناصر النموذج <form> ، وبالأخص أنواع العنصر <input> ، وهو يساعد برامج قراءة الشاشة في التعرف على عنصر <form> ، الذي انتقل إليه المستخدم، كما أنه يزيد مساحة نشاط العنصر ، حيث إن النقر عليه ينشط العنصر المرتبط به.

```
<form>
<label for="nm">Name<br></label><input type="text" id="nm"> <br>
<label>Password<br><input type="password" name="pss"> </label>
</form>
```

Name

Password

for

خاصية تربط بين العنصر <label> وعناصر النموذج <form> مثل أنواع العنصر <input> ، وذلك عن طريق إسناد قيمة الخاصية id لهذا العنصر كقيمة للخاصية for داخل العنصر <label> .

الربط بين العنصر <label> وعناصر <form> مثل العنصر <input>

يتم الربط بين العنصر <label> و العنصر <input> - على سبيل المثال - بطريقتين:

الأولى: يتم تعيين الخاصية for للعنصر <label> ، ويسند لها قيمة خاصية المعرف الفريد id للعنصر <input> ، كما في النوع text في المثال السابق.

الثانية: يتم وضع عنصر النموذج أو العنصر <input> بين وسمي البداية والنهاية للعنصر <label> ، كما في النوع password في المثال السابق أيضاً.

ملاحظات على استخدام العنصر <label>

- لا تضع أي عناصر تفاعليه (مثل الروابط التشعبية أو الأزرار) داخل وسمي العنصر <label> ، لأن انتقال تركيز لوحة المفاتيح focus إلى العنصر <label> سينقل أيضاً التركيز إلى تلك العناصر، بالإضافة إلى العنصر <input> وهو المطلوب نقل التركيز إليه، وبالتالي سيصعب التعامل مع كل هذه العناصر النشطة دفعة واحدة.

- لا تضع أياً من عناصر headings داخل وسمي العنصر <label> ، لأنه يستخدم كمساعد في التنقل بين أجزاء الصفحة في برامج قراءة الشاشة.

- النوع button أو submit أو reset لا تحتاج أن توضع داخل وسمي عنصر <label> لأن قيمتها توضع فوقها، وهي قيمة الخاصية value .

- العنصر <button> والذي سنتناوله لاحقاً ، لا يحتاج أن يوضع داخل وسمي عنصر <label> لأن قيمة العنصر <button> توضع بين وسمي البداية والنهاية الخاصين به.

wailmonir0@protonmail.com

أنواع العنصر <input>

لقد درسنا أنواع العنصر <input> بشكل مختصر، والآن سوف نتناول هذه الأنواع بشيء من التفصيل.

button

نوع زر ، يقوم بتنفيذ وظيفة معينة، أو تنفيذ خطوات برمجية معينة، عند الضغط عليه، وقيمة الخاصية value هي التي تظهر على الزر .

```
<form>
Change color: <input type="button" value="Change color"
onclick="style='color: yellow; background-color: blue'"><br><br>
Hide button: <input type="button" value="Hide me"
onclick="style='display: none'">
</form>
```

Change color : Change color

Hide button : Hide me

لاحظ أن القيمة التي ظهرت على الزرين، هي قيمة الخاصية value لكل منهما.

لاحظ أننا قمنا بتعيين الخاصية onclick لكل زر ، وهي خاصية تقوم بتنفيذ وظيفة معينة عند الضغط على الزر ، وهي إحدى خصائص الحدث event attibutes .

والوظيفة التي أسندناها للخاصية onclick ، هي تغيير تنسيق العنصر الأول، وإخفاء العنصر الثاني، باستخدام css .

قم بتنفيذ الكود كما هو، ولا تنشغل بفهم محتواه حالياً، عندما تدرس css سوف تستوعبه بسهولة، عليك فقط أن تدرك طبيعة وظيفة النوع button من العنصر <input> ، وهي تنفيذ خطوات أو وظيفة معينة عند الضغط عليه.

checkbox

مربع اختيار، يكون الإدخال فيه عن طريق التأشير داخله، ويتيح هذا النوع التأشير على اختيار واحد، أو التأشير على أكثر من اختيار.

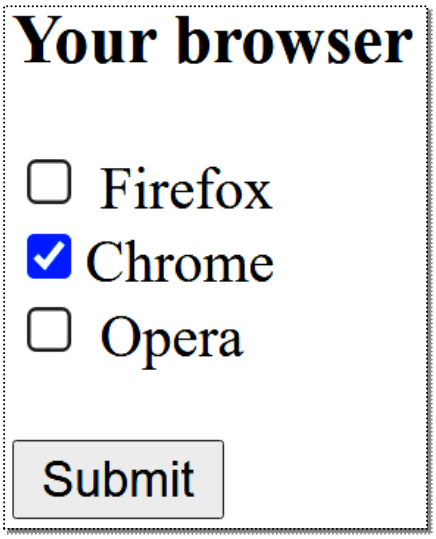
```
<form>
<h3>Your browser</h3>
<input type="checkbox" id="ffx" name="browser" value="first">
<label for="ffx"> Firefox</label><br>
<input type="checkbox" id="chm" name="browser" value="second"
checked><label for="chm">Chrome</label><br>
<input type="checkbox" id="opr" name=" browser " value="third"> <label
for="opr"> Opera</label><br><br>
<input  type="submit"  value="Submit">
</form>
```

checked

خاصية تقوم بعرض عنصر أو أكثر ، وقد تم اختيارهم عند تحميل الصفحة، لاحظ أن العنصر Chrome تم اختياره تلقائياً، نظراً لتعيين الخاصية checked له.

Before checking

Checking



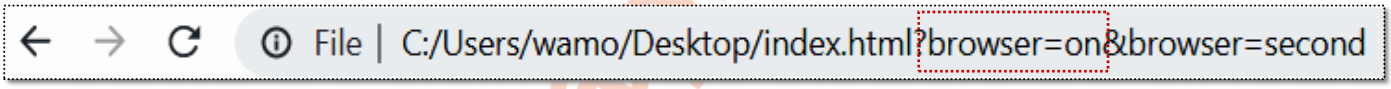
After submit

عند عمل submit لل form ، يتعامل السيرفر مع كل اختيار له قيمة للخاصية name وقيمة للخاصية value ، وفي حالة الاختيار الأول على سبيل المثال، يكون الاسم والقيمة كالآتي browser=first .

Before submitting

After submitting

ويعتمد السيرفر الاسم والقيمة للاختيارات المنتقاة، أي checked فقط، وفي حالة عدم تعيين الخاصية value داخل العنصر الذي تم اختياره، تكون القيمة الافتراضية للخاصية value لهذا العنصر، هي on .



first item checked but has no value attribute

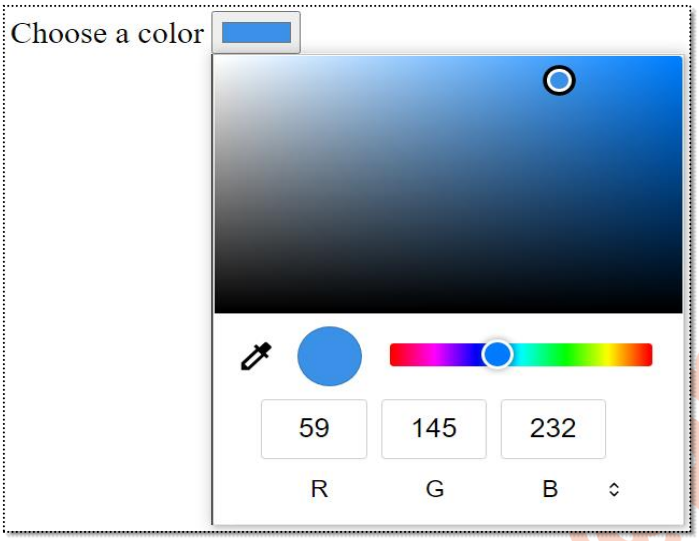
wailmonir0@protonmail.com

color

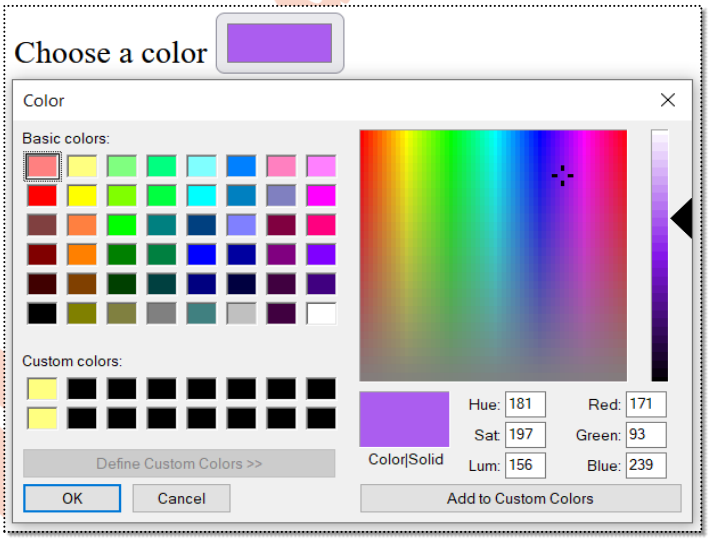
مربع اختيار الألوان.

```
<form>
<label for="clr"> Choose a color </label>
 <input type="color"  id="clr"  value="#AB5D9F" >
</form>
```

لاحظ أن مربع الألوان، يختلف في الشكل من متصفح لآخر.



Google chrome



Mozilla Firefox

تعيين الخاصية value للنوع color

value

خاصية تقوم باختيار اللون الافتراضي لمربع الألوان، وتقبل قيمتها بالصيغة السداس عشرية فقط، وهي قيمة تمثل فيها الألوان الثلاثة rgb بقيم سداس عشرية، حيث يمثل كل لون برقمين من 0 - 9 أو حرفين من A – F أو حرف ورقم، ليكون لدينا تركيبة لون من ستة حروف وأرقام، على أن يسبق الحروف والأرقام الستة رمز "#" .

اختيار اللون من مربع الألوان

يتم اختيار اللون من خلال مربع الألوان، بإدخال القيم بعدة طرق:

- بسحب المنزلق على شريط الألوان لاختيار اللون، ثم النقر بمفتاح الفأرة الأيسر على

درجة اللون المطلوبة في مربع الألوان.

- ادخال القيم العددية للألوان الثلاثة (0-255, 0-255, 0-255)rgb .
- إدخال القيم السداس عشرية للألوان الثلاثة (0-9 & A-F) .
- إدخال قيم المكونات الثلاث (0-255,0-100,0-100) hsl .

إذا كان لديك معرفة مسبقة بنظام الألوان في css ، فعليك أن تعلم أن نظام الألوان في html يختلف عنه، فلا يتم استخدام درجة الشفافية alpha في html ، حتى لا تصبح القيمة غير صحيحة، وفي حالة إدخال اللون بطريقة غير صحيحة، سيؤدي ذلك لظهور اللون الأسود بديلاً عن اللون الذي تم اختياره.

- alpha أو الشفافية هي قيمة رابعة تضاف بعد قيم الألوان الثلاثة في css .

wailmonir0@protonmail.com

date

مربع إدخال التاريخ، ويتكون من اليوم والشهر والعام.

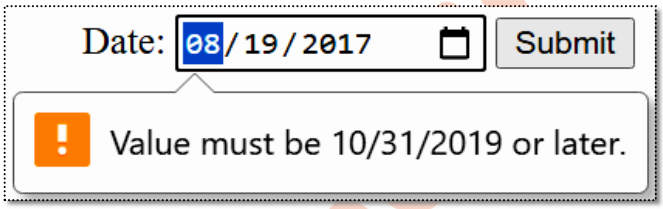
```
<form>
<label for="today">Date: </label>
<input type="date" id="today" name="today" step="5"
min="2019-10-31" max="2022-12-31" value="2020-08-19">
<input type="submit">
</form>
```

لاحظ أن قيم التواريخ تبدأ من اليسار، (yyyy-mm-dd) بالعام ثم الشهر ثم اليوم، على أن يتكون العام من أربعة أرقام، والشهر واليوم من رقمين، وفي حالة وجود رقم واحد (آحاد)، يتم إضافة 0 إلى يسار هذا اليوم أو الشهر، وأي قيمة خلاف ذلك، ستصبح القيم غير صحيحة، ولن تعرض قيمة الخاصية value الافتراضية في مربع الإدخال، أو لن يكون للتاريخ قيمة صغرى أو قيمة كبرى أو خطوة تنقل step .

لاحظ أن عرض التاريخ يختلف عند العرض في المتصفح، حيث تبدأ القيم من اليسار بالشهر ثم اليوم ثم العام (mm-dd-yyyy).

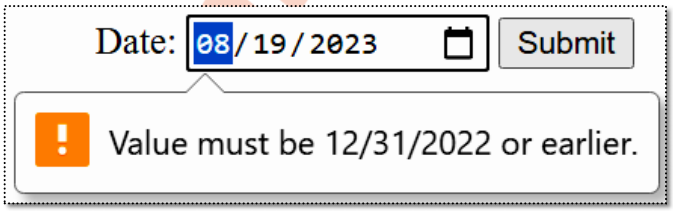
ويمكن تعيين عدة خصائص للنوع date :

value



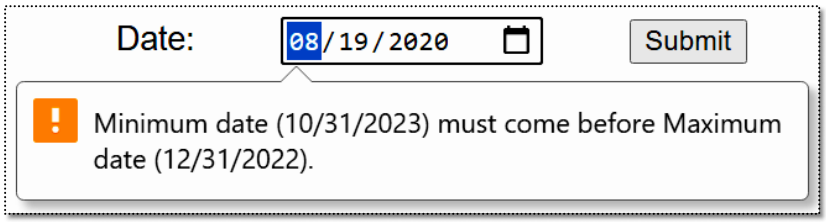
خاصية تحدد القيمة الافتراضية التي تظهر عند فتح مربع التاريخ، ويجب ألا تتخطى قيمة الخاصية max ، ولا تقل عن قيمة الخاصية min ، وإلا فإنه لن يتم اعتماد النموذج وستظهر رسالة تنبيه بهذا الخطأ.

max



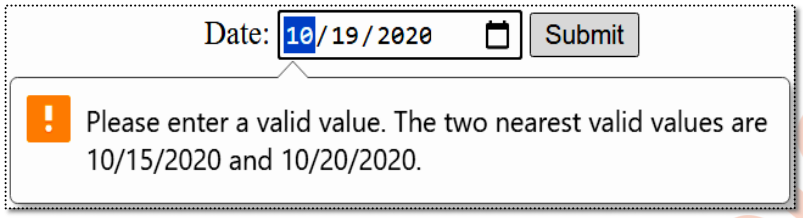
خاصية تحدد أقصى تاريخ لا يمكن تخطيه، أو إدخال أي تاريخ بعده، ويجب ألا تقل قيمة هذه الخاصية عن قيمة الخاصية min ، وإلا فإنه لن يتم اعتماد النموذج، وستظهر رسالة تنبيه بهذا الخطأ.

min

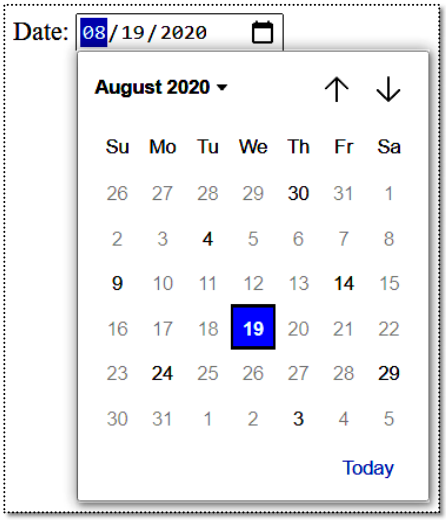


خاصية تحدد أدنى تاريخ لا يمكن النزول عنه، أو إدخال أي تاريخ قبله، ويجب ألا تزيد قيمتها عن قيمة الخاصية max ، وإلا فإنه لن يتم اعتماد النموذج وستظهر رسالة تنبيه بهذا الخطأ.

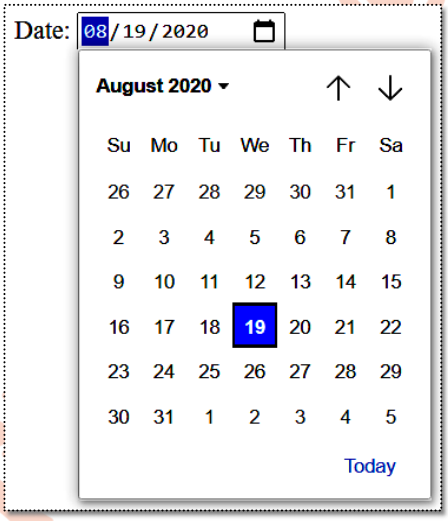
step



خاصية تحدد خطوة التنقل بين الأيام النشطة، حيث تظهر باقي الأيام غير فعالة، ولا يمكن إدخالها، وقيمة الخاصية تكون رقماً يمثل الفترة بالأيام، وإذا تم إدخال أي قيمة بينية غير فعالة عن طريق لوحة المفاتيح، فلن يتم اعتماد النموذج وستظهر رسالة تنبيه بهذا الخطأ، وقد تقبل الخاصية القيمة any وتعني أن كل الأيام نشطة ومتاح التعامل معها.



step



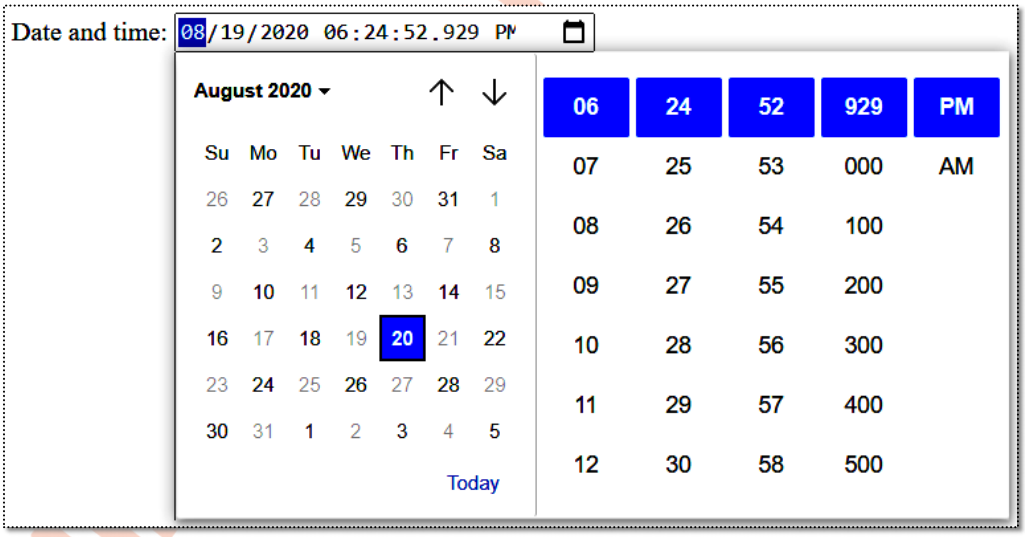
value

لاحظ في الشكل الأول على اليسار، وقد ظهرت فيه القيمة الافتراضية عند فتح مربع التاريخ، وكل الأيام نشطة، حيث يمكن اختيار أي منها، في حين أن الشكل الثاني على اليمين، قد بدت الأيام نشطة كل خمسة أيام، وهي قيمة الخاصية step .

datetime-local

نوع يقوم بإدخال التاريخ والوقت، على أن يكتب التاريخ بنفس الصيغة السابق شرحها في النوع date ، ثم يفصل بينه وبين الوقت بحرف "T" كبير، ثم الساعة تليها الدقائق ثم الثواني على أن يفصل بينها ":" ثم أجزاء الثانية ويفصل بينها وبين الثواني "." على أن تكتب الساعات والدقائق والثواني من رقمين، وأجزاء الثانية من رقم إلى ثلاثة، ويعرض على هيئة ثلاثة أرقام، على أن يعرض التاريخ بنفس ترتيب النوع السابق date يلحقه حرف "T" ، ثم الساعات ثم الدقائق ثم الثواني ثم أجزاء الثانية، والمسافات الفارغة غير مسموح بها بين أجزاء القيمة، ويمكن الاستغناء عن الثواني وأجزاء الثواني.

```
<form>
<label for="dtloacl"> Date and time:  </label>
<input type="datetime-local"  id="dtlocal"   step="172800"
max="2022-10-21T14:30:45.929"  value="2020-08-19T18:24:52.929">
</form>
```



ويمكن تعيين نفس خصائص النوع date لهذا النوع وهي value و max و min و step ، مع العلم أن تأثير الخاصيتين min و max يظهر على التاريخ دون الوقت، والخاصية step تقبل قيمتها، على هيئة عدد الثواني فالقيمة 172800 تمثل يومين.

هذا النوع غير مدعوم في متصفح Firefox .

email

نوع مخصص لإدخال نص مطابق لمعايير البريد الإلكتروني.

```
<form>
<label for="mail">E-mail:  </label>
<input type="email" id="mail" name="mail"  size="30" multiple
placeholder=" your email"   maxlength="25" minlength="15">
<input type="submit" value="Submit email">
</form>
```

E-mail: your email Submit email

بعد إدخال النص يقوم المتصفح بالتحقق من مطابقته للمعايير، وإذا لم يكن مستوفياً لها، تظهر رسالة تنبيه، والمطابقة للمعايير لا تعني أن البريد الإلكتروني صحيح أو موجود بالفعل، إنما هي مجرد مطابقة للتنسيقات والمواصفات الشكلية فقط.

E-mail: aaaaaaaa.fffffff Submit email

Please include an '@' in the email address.
'aaaaaaaa.fffffff' is missing an '@'.

والنوع email يقبل عدة أنواع من الخصائص:

minlength

خاصية تحدد أقل عدد من الحروف، التي يمكن للمستخدم إدخالها، وتكون القيمة عدداً صحيحاً موجباً، وإدخال عدد حروف أقل، لن يسمح باعتماد النموذج، وستظهر رسالة تنبيه بهذا الخطأ.

E-mail: aa@ddd.com Submit email

Please lengthen this text to 15 characters or more (you are currently using 10 characters).

maxlength

خاصية تحدد أقصى عدد من الحروف التي يمكن للمستخدم إدخالها وتكون القيمة عدداً

صحيحاً موجباً، ولن يسمح بإدخال عدد حروف أكبر من قيمة الخاصية.

يجب أن تكون قيمة الخاصية maxlength ، أكبر من أو تساوي قيمة الخاصية minlength ، وبالتالي، فقيمة الخاصية minlength يجب أن تكون أصغر من أو تساوي قيمة الخاصية maxlength ، وإلا فإن المدخلات لن يتم اعتمادها، لأنك لن تستطيع إدخال قيمة أكبر من قيمة maxlength ، لتستوفي قيمة الخاصية minlength ، ولن تسطيع إدخال قيمة أقل من قيمة minlength ، لتستوفي قيمة الخاصية maxlength .

size

خاصية تحدد عرض عنصر الإدخال مقدراً بعدد الحروف، التي ستظهر من المدخلات، ويمكن لقيمة الخاصية maxlength أن تكون أكبر من قيمة هذه الخاصية، لأن مفهوم الخاصية size لا يعني أنه لا يمكن إدخال عدد حروف أكبر من قيمة الخاصية maxlength ، وإنما تعني أنها ستعرض عدداً معيناً من الحروف، سواء كان أكبر أو أصغر من قيمتها، فإذا كانت قيمة الخاصية size أكبر من قيمة الخاصية maxlength فستقوم بعرض المدخل كاملاً، وتعرض باقي مساحة العنصر فارغة، وإذا كانت قيمتها أقل، فستعرض عدداً من حروف المدخل يساوي قيمتها، ثم تقوم بإخفاء باقي الحروف.

E-mail: neomemail@somemailserverforall.co Submit email

لاحظ اختفاء جزء من النص لأن عدد حروفه أكبر من قيمة الخاصية size .

multiple

خاصية تسمح بإدخال أكثر من بريد إلكتروني بينها فاصلة.

E-mail: one@email.com, two@email.com Submit email

placeholder

خاصية تقوم بكتابة تلميح داخل العنصر، لتوضيح نوع البيانات المطلوب إدخالها في هذا الحقل، على أن يختفي هذا التلميح عند وضع المؤشر داخل العنصر.

الخاصية غير محسوسة لبرامج قراءة الشاشة وقد تتجاهلها برامج الترجمة الفورية.

wailmonir@protonmail

pattern

خاصية تحدد عينة ذات تنسيق معين، يجب أن يتطابق معها النص الذي يتم إدخاله في العنصر، وهذه العملية تعرف على نطاق واسع، في لغات البرمجة باسم regular expressions حيث إن لها أسساً وقواعد تنظمها، وتتشابه هذه القواعد في الكثير من لغات البرمجة، وتختلف فيما بينها اختلافات بسيطة.

```
<form>
<label for="mail">E-mail:  </label>
<input  type="email" id="mail" name="mail"
pattern=".+@.+\.(com|org|net)">
<input type="submit" value="Submit email">
</form>
```

لاحظ أن المتصفح يعرض رسالة تنبيه، توضح عدم مطابقة النص الذي تم إدخاله للعينة التي تم وضعها للمطابقة.

توضيح مدلول العينة ".+@.+\.(com|org|net)"

"." النقطة منفردة لا تسبقها شرطة مائلة تعني حرفاً واحداً أو رقماً واحداً.

"+" تعني واحداً أو أكثر من الرمز قبلها وهو "." .

"@" رمز مطلوب إدخاله بعد الحروف أو الأرقام أو خليط منهما قبله، ثم يتم إدخال عدد آخر من الحروف أوالأرقام أو مزيج منهما بعده.

"\." للتعبير عن النقطة كنقطة وليس أي حرف أو رقم.

(com|org|net) أي يجب أن تكون نهاية قيمة المدخل قيمة اختيارية من أي منهم.

readonly

خاصية لا تسمح بإدخال أي نص في العنصر، وإنما تسمح فقط بقراءة نص تم إدخاله تلقائياً، عن طريق قيمة الخاصية value ، وفي حالة وجود هذه الخاصية فلا حاجة ولا معنى، لتعيين أي من الخصائص السابقة.

```
<form>
<label>E-mail:  <input type="email" id="mail" name="mail"  readonly
value="someone@email.com"></label>
<input type="submit" value="Submit email">
</form>
```

E-mail: someone@email.com Submit email

wailmonir0@protonmail.com

file

نوع مخصص لتحميل ملف أو عدة ملفات من حاسوبك، إلى خادم أو server معين أو إجراء معالجة خاصة لهذه الملفات، بواسطة نص برمجي معين، عند اعتماد النموذج.

```
<form>
<label for="fl">upload:  </label>
<input  type="file"   accept="image/png, video/*"   multiple>
</form>
```

upload: Choose Files No file chosen

ويمكن تعيين عدة خصائص للنوع file وهي:

accept

خاصية تحدد أنواع الملفات التي يمكن تحميلها باستخدام النوع file ، وفي حالة تعيين الخاصية multiple ، يمكن تحديد أكثر من نوع، يفصل بينهم "," .

كيفية تحديد أنوع الملفات

- تحديد نوع ملفات ذات امتداد معين، أياً كان اسم الملف، وذلك بكتابة نقطة يتبعها امتداد الملف، ويكون بهذا الشكل: ".doc, .pdf, .mp4, .mp3, .jpg"

- تحديد نوع معين من الملفات، بامتداد معين، يفصل بينهما شرطة مائلة بهذا الشكل: "audio/mp3, video/mp4, applicatin/pdf, image/png".

- تحديد نوع معين من الملفات، وأي امتداد لهذا النوع بهذا الشكل: "image/*, audio/*, video/*, application/*".

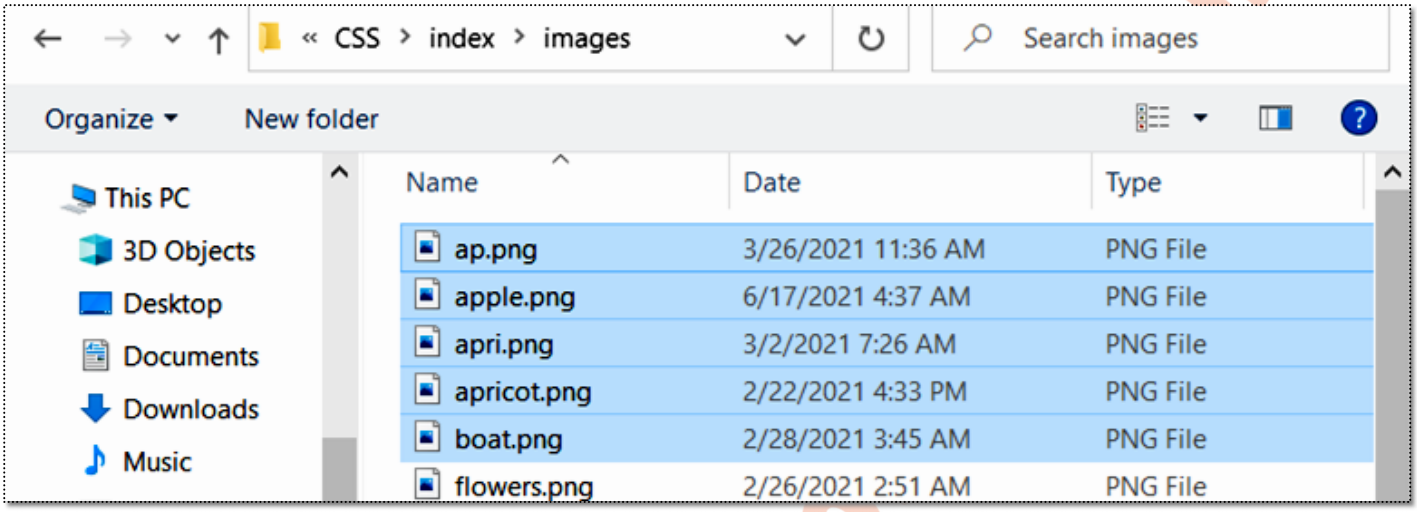
ومن أنواع الملفات التي تهم مطوري الويب: "text/html, text/css, text/javascript".

capture

خاصية تحدد الكاميرا أو الميكروفون، الذين يستخدمان في التقاط الصورة، أو ملف الفيديو، أو ملف الصوت لتحميله، إذا كان نوع الملف من الأنواع المحددة في قيمة الخاصية accept .

multiple

خاصية تسمح للنوع file باختيار وتحديد أكثر من ملف لتحميلهم سوياً.

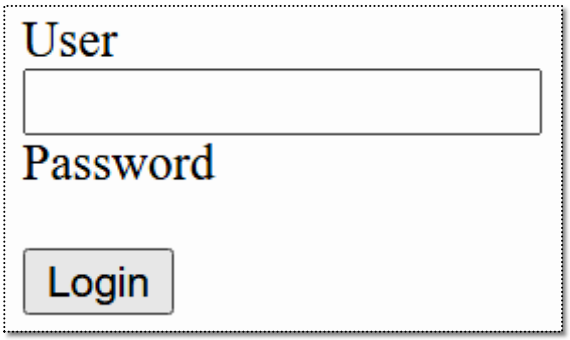


wailmonir0@proto

hidden

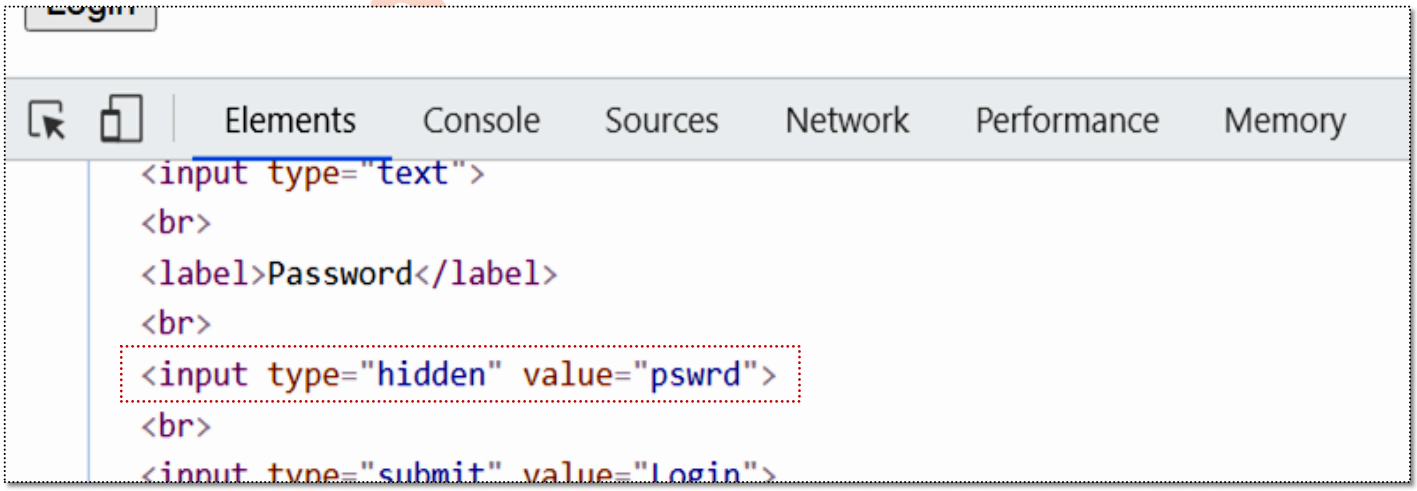
نوع مخصص لإرسال قيمة مخفاة عن المستخدم، للخادم أو server أو لصفحة أخرى، بعد اعتماد النموذج الذي يحتوي هذا النوع.

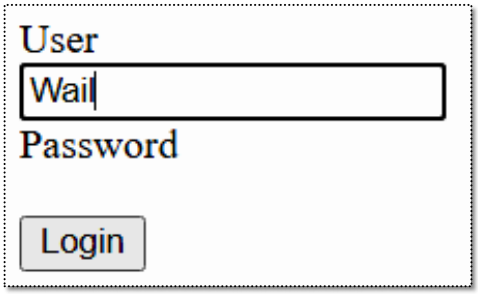
```
<form>
<label>User</label><br><input type="text" name="usr"><br>
<label>Password</label><br>
<input type="hidden" value="pswrd" name="pass"><br>
<input type="submit" value="Login">
</form>
```



لاحظ أن الحقل User ظاهر للمستخدم لأنه من النوع text بينما الحقل Password مخفي لأنه من النوع hidden .

وإن كانت القيمة التي يتم إرسالها مخفاة عن المستخدم، ولا تظهر في جسم الصفحة، إلا أنه يمكن معاينتها، باستخدام أدوات التطوير التي يستخدمها كل مطوري الويب، وأبسطها، الضغط على الصفحة بالزر الأيمن للفأرة، ثم الضغط على الأمر inspect وتحت التبويب elements ستجد كود html الخاص بالصفحة، وبداخله العنصر hidden ، وقيمته التي تم إخفاؤها عن المستخدم، كما بالشكل التالي:





طريقة أخرى للتعرف على قيمة الحقل المخفي، إذا كان الحقل Password قد تم تعيين الخاصيتين name و value له كما في المثال السابق، فعند ملء الحقل User ثم الضغط على الزر Login يظهر في شريط العنوان address bar ، أسماء وقيم العناصر التي تم اعتمادها، وهي قيمة الخاصية name مسنداً إليها قيمة الخاصية value ، حتى وإن كان هذا العنصر من النوع hidden ، طالما تم تعيين الخاصيتين له كما في الشكل التالي:

هذه الطريقة لا تصلح دائماً، لأن ظهور القيم في شريط العنوان من عدمه، يعتمد على قيمة الخاصية method للنموذج، أو الخاصية formmethod لنوع عنصر الإدخال <input> وشرح قيم الخاصية ملحق 1 .

استخدامات النوع hidden

ويستخدم النوع hidden لتحسين أمان المواقع، حيث يتم تخزين رموز خاصة بالموقع تمنع المخترقين من تقليده، فربما يكون هذا الموقع يستخدم خاصية الدفع الآلي، فيقوم المخترقون بخداع العملاء بموقع مزيف لإدخال بياناتهم السرية، ولكن هذه الحقول المخفاة في الموقع الحقيقي لا يراها المخترقون، وهي ترسل معلوماتها إلى الخادم، ومنه إلى العملاء، في حين، أن الصفحة المزيفة لن ترسل هذه المعلومات للعملاء، فيتبين العملاء أن الموقع مزيف.

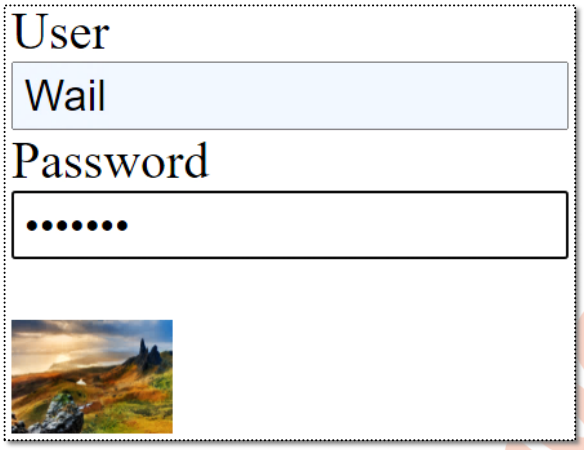
كما يستخدم النوع hidden في إخفاء التعديل، على عنصر معين وعرض العنصر مباشرة بعد التعديل، مثل بوست أو مقال معين عند التعديل عليه، فلا يشاهده المستخدمون أثناء التعديل، ولكن يتم مشاهدته بعد إتمام التعديل.

wailmonir0@protonmail.com

image

نوع يعمل كزر رسومي بدلاً من الأزرار العادية، حيث يقوم بعمل اعتماد أو submit للنموذج عند الضغط عليه، وإرسال بيانات النموذج إلى الخادم أو إلى صفحة أخرى.

```
<form method="post">
<label>User</label><br><input  type="text" name="usr"><br>
<label>Password</label><br>
<input type="password"  value="" name="pass"><br><br>
<input type="image"   id="img"   name="logo" width="50" height="35"
alt="mountain"  src="images/mount.jpg" formtarget="_blank"
formmethod="get">
</form>
```



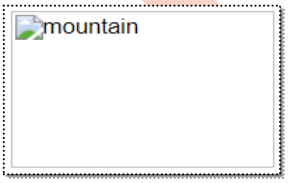
لاحظ أن الضغط على الزر image سيعتمد البيانات التي يتم إدخالها، وعند فحص شريط العنوان لمعرفة البيانات التي تم اعتمادها نلاحظ وجود قمتين أخريين غير قيمتي الحقلين User وPassword ، وهما الإحداثي الأفقي logo.x والإحداثي الرأسي logo.y لموضع الضغط بالفأرة على الصورة أو الزر.

النوع image لا يقبل تعيين الخاصية value ، لأن القيمة لن تظهر على الصورة ولكنه يقبل تعيين خصائص عديدة:

src

خاصية تحدد مسار الصورة على حاسوبك، أو على أحد مواقع الانترنت، ويتم تحديد المسار بعدة طرق، انظر ص 171 .

alt



خاصية تحدد النص الذي سيظهر عند حدوث خطأ في مسار الصورة.

height & width

خاصيتان تحددان ارتفاع وعرض الصورة أو الزر.

formtarget

خاصية تحدد الموضع الذي ستفتح فيه الصفحة الجديدة، بعد اعتماد النموذج، وتقبل عدة قيم، هي نفس قيم الخاصية target ، وشرح مدلول كل قيمة ملحق 1 .

formaction

خاصية تحدد عنوان url الذي سيتم إرسال بيانات النموذج له بعد اعتماد النموذج.

formmethod

خاصية تحدد الطريقة التي يتم بها إرسال بيانات النموذج إلى السيرفر، وتقبل قيمتين هما get و post وشرح مدلول كل قيمة ملحق 1 .

formenctype

خاصية تحدد طريقة تشفير بيانات النموذج بعد اعتمادها وأثناء دخولها للسيرفر، وهي تقبل عدة قيم، والقيم وشرح مدلول كل قيمة ملحق 1 .

formnovalidate

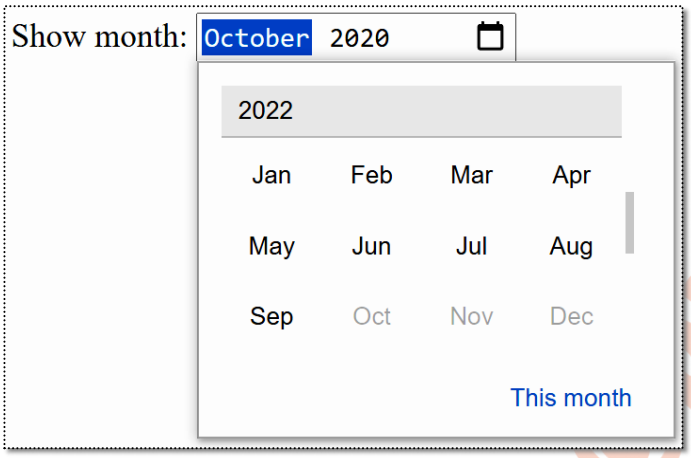
خاصية تسمح بعدم التحقق من صحة بيانات النموذج، أو موافقتها للمعايير عند اعتمادها وقبل إرسالها للخادم.

wailmonir0@protonmail.com

month

نوع مخصص لإدخال شهر معين، يتبعه العام من التاريخ، ويتم إدخال العام أولاً مكوناً من أربعة أرقام، ثم إدخال الشهر مكوناً من رقمين، ويكون الإدخال بنفس التنسيق "yyyy-mm".

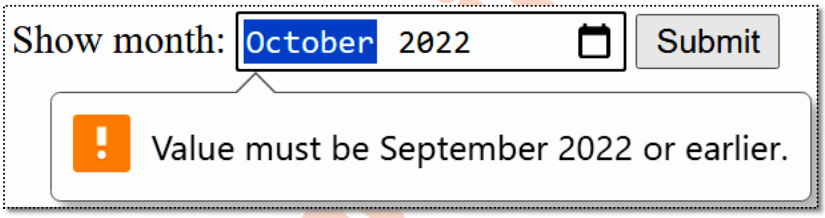
```
<form>
<label for=" mnth ">Show month:  </label>
<input  type="month"  id="mnth" name="mnth"  min="2018-04"
max="2022-09"  value="2020-10" step="3">
<input type="submit">
</form>
```



لاحظ أن مربع التاريخ يعرض الشهر أولاً ثم العام، على هيئة Month yyyy وهو ليس اتجاه أو تنسيق التاريخ الذي تم إدخاله، لأن المتصفح يستخدم الإعدادات المحلية في عرض التاريخ.

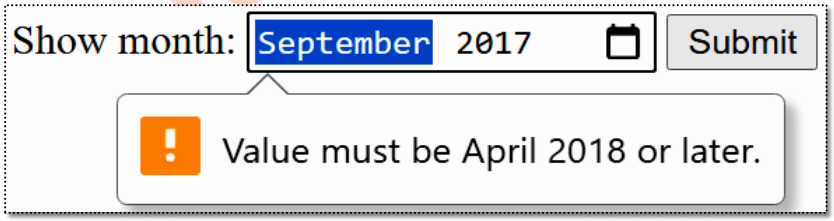
ويمكن تعيين نفس خصائص النوع date للنوع month وهي كالتالي:

max



خاصية تحدد أقصى شهر لا يمكن تخطيه أو إدخال أي قيمة أكبر منه، ولا يجب أن تقل قيمة هذه الخاصية عن قيمة الخاصية min ، وإذا حدث خلاف ذلك لا يتم اعتماد النموذج، وتظهر رسالة تنبيه بهذا الخطأ.

min



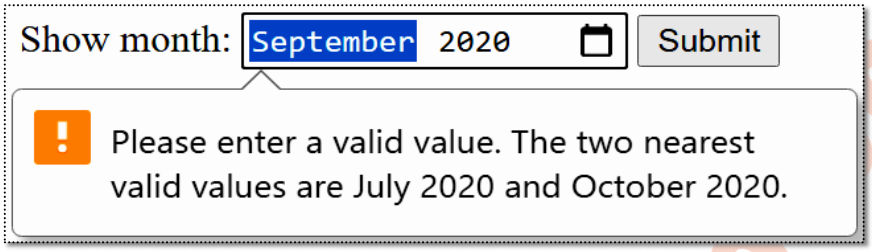
خاصية تحدد أقل عدد من الأشهر لا يمكن النزول عنه، أو إدخال أي عدد أقل منه، ولا يجب أن تزيد

قيمة هذه الخاصية عن قيمة الخاصية max، وإذا حدث خلاف ذلك، لا يتم اعتماد النموذج، وتظهر رسالة تنبيه بهذا الخطأ.

value

خاصية تحدد القيمة الافتراضية التي تظهر عند فتح مربع الأشهر، ويجب ألا تتخطى قيمة الخاصية max ، ولا تقل عن قيمة الخاصية min ، وإذا حدث خلاف ذلك لا يتم اعتماد النموذج، وتظهر رسالة تنبيه بهذا الخطأ، كما بالشكلين السابقين.

step



خاصية قيمة القفزة بين القيم المدخلة، والتي تحدد الفترة النشطة مقدرة بالأشهر، حيث تظهر باقي الأشهر غير فعالة، وفي حالة إدخال عدد أشهر لا يتماشى مع قيمة هذه الخاصية، لا يتم اعتماد النموذج، وتظهر رسالة تنبيه بهذا الخطأ.

ويمكن إسناد القيمة any لهذه الخاصية، وتعني أن كل الأشهر نشطة ولا يوجد فترة زمنية غير فعالة بينها.

ويوجد خصائص أخرى يمكن تعيينها للنوع month ، وقد سبق شرحها سابقاً في مواضع عديدة مثل الخاصية readonly والخاصية required ، ويمكن الرجوع إليها عند الحاجة.

والنوع month غير مدعوم في متصفح Firefox .

wailmonir0@protonmail.com

number

نوع مخصص لإدخال الأعداد دون الحروف، ويكون للحقل number أسهماً لزيادة وتقليل قيمة العدد، بالضغط بالفأرة على الأسهم لأعلى أو لأسفل، أو باستخدام أسهم لوحة المفاتيح.

```
<form>
<label for="num">Number: </label>
<input  type="number"  id="num"  name="num"  min="10" max="100"
step="5"  value="45">
<input type="submit" value="Submit">
</form>
```

ويمكن تعيين عدة خصائص للنوع number :

Number: 45

max

خاصية تحدد أكبر عدد يمكن إدخاله ولا يمكن للمستخدم تخطيه أو إدخال أي عدد أكبر منه، ولا يجب أن تقل قيمة هذه الخاصية عن قيمة الخاصية min وإذا حدث خلاف ذلك، لا يتم اعتماد النموذج، وتظهر رسالة تنبيه بهذا الخطأ.

Number: 200 Submit

Value must be less than or equal to 100.

min

خاصية تحدد أقل عدد يمكن إدخاله ولا يمكن للمستخدم النزول عنه، ولا يجب أن تزيد قيمة هذه الخاصية عن قيمة الخاصية max ، وإذا حدث خلاف ذلك، لا يتم اعتماد النموذج، وتظهر رسالة تنبيه بهذا الخطأ.

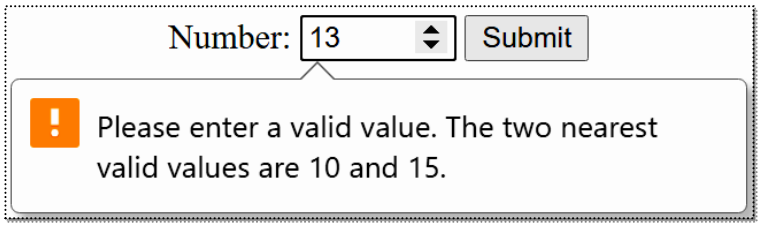
Number: 5 Submit

Value must be greater than or equal to 10.

step

خاصية تحدد قيمة القفزة بين الأعداد المدخلة، عند استخدام أسهم مربع الإدخال، أو عند استخدام أسهم لوحة المفاتيح، وفي حالة إدخال العدد عن طريق لوحة المفاتيح، وإدخال

عدد لا تنطبق عليه الشروط السابقة، لا يتم اعتماد النموذج وتظهر رسالة تنبيه بهذا الخطأ.



value

خاصية تحدد القيمة الافتراضية التي تظهر عند تحميل الصفحة، ويجب ألا تتخطى قيمة الخاصية max ، ولا تقل عن قيمة الخاصية min ، مع مراعاة قيمة القفزة المحددة في الخاصية step ، وإذا حدث خلاف ذلك لا يتم اعتماد النموذج، وتظهر رسالة تنبيه تحذر من هذه الأخطاء.

wailmonir0@protonmail.com

password

نوع مخصص لإدخال كلمة سر، ولا يتم عرض محتوى الحقل للمستخدم على هيئة الحروف والأرقام والحروف الخاصة، ولكن يتم عرض المدخلات على هيئة رموز، إما "*" أو "•" ، من أجل توفير أكبر قدر من الأمان والحماية.

```
<form>
<label for="psswrd">Password: </label>
<input  type="password"  id="psswrd" size="20" name="psswrd"
maxlength="30" minlength="9" pattern=".{9-30}" placeholder="Enter Your
password" required>
<input type="submit" value="Submit">
</form>
```

Password: Enter Your password Submit

ويمكن تعيين عدة خصائص للنوع **password** :

required

خاصية تعني أن هذا الحقل مطلوب، أي لا يمكن اعتماد النموذج إذا كان هذا الحقل فارغاً، ولم يتم ملؤه بنوع البيانات المتوقع، وتظهر رسالة تنبيه بهذا الخطأ.

Password: Enter Your password Submit

Please fill out this field.

pattern

عينة **regular expression** ، التي يجب أن تتطابق معها مدخلات كلمة السر، حيث يمكن التحكم في نوع المدخلات، وعدد ومكان إدخال كل نوع، وعند عدم مطابقتها تظهر رسالة تنبيه بهذا الخطأ، انظر ص 128 .

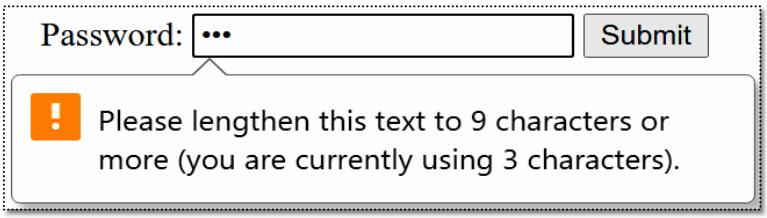
maxlength

خاصية تحدد أقصى عدد من الحروف والأرقام والرموز، يمكن إدخالها في حقل كلمة السر،

ولا يتم السماح بتجاوز قيمة هذه الخاصية لعدد الحروف المدخلة، ويتوقف الإدخال عند إدخال العدد الأقصى من الحروف المحدد في قيمة هذه الخاصية.

minlength

خاصية تحدد أقل عدد من الحروف والأرقام والرموز، يمكن إدخاله في حقل كلمة السر، وتظهر رسالة خطأ في حالة اعتماد النموذج، وكان عدد الحروف المدخلة أقل من قيمة هذه الخاصية.



وتنطبق على هذه الخاصية والخاصية السابقة نفس شروط الخاصيتين maxlength وminlength ص 127 ، من حيث عدم إمكانية تجاوز أي منهما لقيمة الخاصية الأخرى.

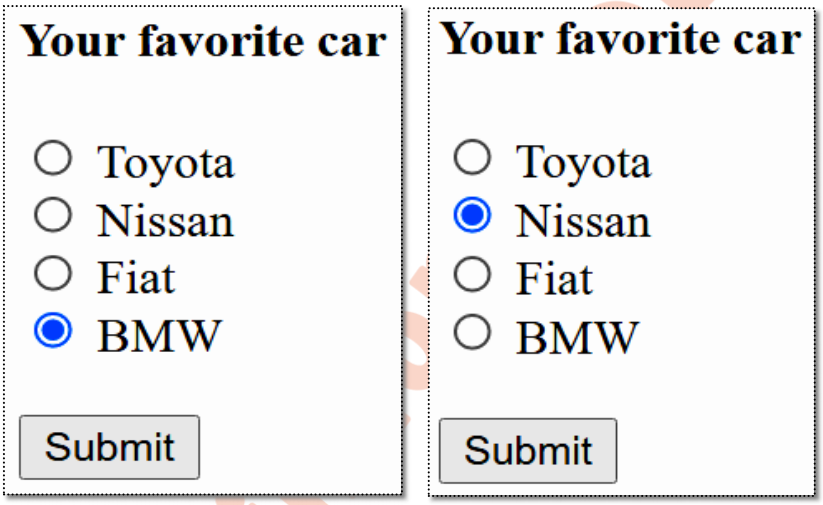
وباقي الخصائص size و placeholder وغيرها والتي وردت بالكود السابق، قد سبق شرحها في مواضع كثيرة، ويمكن الرجوع إلى كل منها في أحد هذه المواضع.

wailmonir0@protonmail.com

radio

نوع مخصص ليُكوّن مجموعة من الأزرار، تشبه أزرار الراديو، كل زر يمثل بنداً معيناً، ويجب اختيار واحد منها فقط، وهذا هو وجه الاختلاف بين هذا النوع وبين مربعات الاختيار checkbox، حيث يمكن اختيار أكثر من بند من مربعات الاختيار.

```
<form>
<h4>Your favorite car</h4>
<input  type="radio"  id="rdo" name="car" value="toyota">
<label for="rdo"> Toyota</label><br>
<input  type="radio"  id="nsn" name="car" value="nissan">
<label for="nsn"> Nissan</label><br>
<input  type="radio"  id="ft" name="car" value="fiat">
<label for="ft"> Fiat</label><br>
<input  type="radio"  id="bm" name="car" value="bmw" checked>
<label for="bm"> BMW</label><br><br>
<input type="submit" value="Submit">
</form>
```



لاحظ أن الاختيار BMW تم اختياره افتراضياً عند تحميل الصفحة أو النموذج نظراً لتعيين الخاصية chedcked له، كما أنه عند التأشير على أحد الاختيارات يتم إلغاء التأشيرة السابقة، وعند الضغط على زر submit ، يظهر في شريط العنوان اسم الاختيار، وهو قيمة الخاصية name ، مسنداً لها قيمة الخاصية value ، للاختيار الذي تم التأشير عليه من قبل المستخدم.

ويمكن تعيين عدة خصائص للنوع radio :

checked

كما ذكرنا سابقاً، هي خاصية تعرض أحد الاختيارات، وقد تم التأشير عليه عند تحميل الصفحة أو النموذج، مثل الاختيار BMW في المثال السابق، الذي ظهر مختاراً عند بداية تحميل الصفحة.

name

خاصية تمنح النوع اسماً معيناً، وهذه الخاصية لها ميزة إضافية في هذا النوع عن باقي الأنواع والعناصر، حيث إن اشتراك عدة عناصر من نوع زر الراديو radio في اسم واحد، يجعل هذه الاختيارات مجموعة واحدة، مما يعني أنه لا يمكن التأشير إلا على اختيار واحد منها فقط، وفي حالة عدم تعيين هذه الخاصية، أو في حالة عدم تطابق قيمتها في كل العناصر، فلن تتكون المجموعة، ويمكن للمستخدم عندئذ، أن يختار أكثر من اختيار، وبالتالي، يفقد النوع radio ميزته التي تميزه عن النوع checkbox .

wailmonir0@protonmail.com

range

نوع يقوم بإدخال عدد معين، ولكن عن طريق شريط تمرير مثبت به منزلق، ويتم الإدخال عن طريق تحريك المنزلق على الشريط، يميناً ويساراً حسب الرغبة، سواء بالزيادة أو بتقليل قيمة العدد.

ونظراً لأن قيمة النوع range تكون غير دقيقة، فلا ينصح باستخدام هذا النوع ما لم تكن القيمة التي يقوم المستخدم بإدخالها، تتطلب قيمة تقريبية غير محددة تماماً.

ومن استخدامات النوع range ، ضبط الصوت والتحكم فيه، أو ضبط الألوان ودرجاتها ونسبة كل لون، أو ضبط مستوى صعوبة لعبة معينة، وغيرها من الاستخدامات.

```
<form>
<label for="rng"> Range: </label>
<input type="range" id="rng" name="rng" value="50" min="0" max="100" step="2">
<input type="submit" value="Submit">
</form>
```

Range: Submit

ويمكن تعيين العديد من الخصائص للنوع range كالتالي:

max

خاصية تحدد أكبر عدد يمكن إدخاله، وتكون قيمتها هي نهاية شريط التمرير، فلا يمكن للمستخدم تخطيها، أو إدخال أي قيمة أكبر منها، ولا يجب أن تقل قيمة هذه الخاصية عن قيمة الخاصية min ، وفي حالة نقصانها يتم تثبيت المنزلق عند بداية الشريط، ولن يتم سحبه على الشريط.

min

خاصية تحدد أقل عدد يمكن إدخاله، وتكون قيمتها هي بداية شريط التمرير، فلا يمكن للمستخدم النزول عنها، أو إدخال أي عدد أصغر منها، ولا يجب أن تزيد قيمة هذه الخاصية عن قيمة الخاصية max ، وفي حالة زيادتها يتم تثبيت المنزلق عند بداية الشريط، ولن يتم

سحبه على الشريط.

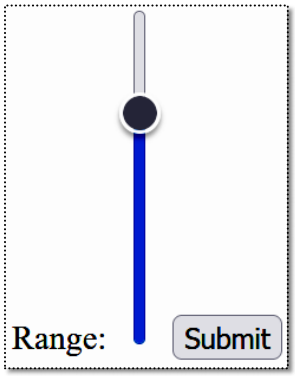
step

خاصية تحدد الفرق بين قيمة كل عملية تمرير للمنزلق على الشريط، سواء لليمين أو لليسار، أي هي قيمة القفزة أو الخطوة، سواء إلى الأمام بالزيادة، أو إلى الخلف بالنقصان.

value

خاصية تحدد القيمة الافتراضية، التي يقف عندها المنزلق على شريط التمرير عند تحميل الصفحة أو النموذج، ويجب ألا تتخطى قيمة هذه الخاصية قيمة الخاصية max ، ولا تقل عن قيمة الخاصية min ، وفي حالة زيادتها عن قيمة الخاصية max ، يظهر المنزلق في آخر الشريط، وفي حالة عدم تغييرها سيتم اعتمادها بنفس قيمة الخاصية max ، وفي حالة نقصانها عن قيمة الخاصية min ، سيظهر المنزلق في أول الشريط، وفي حالة عدم تغييرها سيتم اعتمادها بنفس قيمة الخاصية min .

orient



خاصية مدعومة في متصفح Firefox فقط، تتحكم في وضع شريط التمرير سواء كان أفقياً أم رأسياً، ولذلك تقبل هذه الخاصية قيمتين هما horizontal ، وهي القيمة الافتراضية، والقيمة الثانية هي vertical .

wailmonir@protonmail.com

reset

نوع يعمل كزر يقوم بمسح جميع المدخلات داخل النموذج، ويعيد جميع الحقول إلى قيمها الافتراضية عند بداية تحميل الصفحة أو النموذج، لذلك يفضل عدم استخدام هذا النوع، لأنه عند أي ضغط عليه بطريق الخطأ ، يقوم بمسح كل المدخلات، والتي يمكن أن تحتاج إلى بعض الوقت لإعادة ملء الحقول التي تم مسح قيمتها.

والنوع reset لأنه يقوم بوظيفة زر، فهو لا يحتاج أن يوضع قبله عنصر <label> ، لأن قيمته توضع فوقه، وبالتالي يكون معروفاً للمستخدم طبيعة استخدام هذا الزر.

ويمكن تعيين خاصية value للنوع reset ، في حالة الرغبة في تغيير الاسم الافتراضي للنوع، حيث إن هذه الخاصية مسئولة عن كتابة اسم الزر فوقه، لتوضيح طبيعة عمله كما ذكرنا.

ويمكن تعطيل الزر reset بتعيين الخاصية disabled له، فلا يتفاعل مع المستخدم.

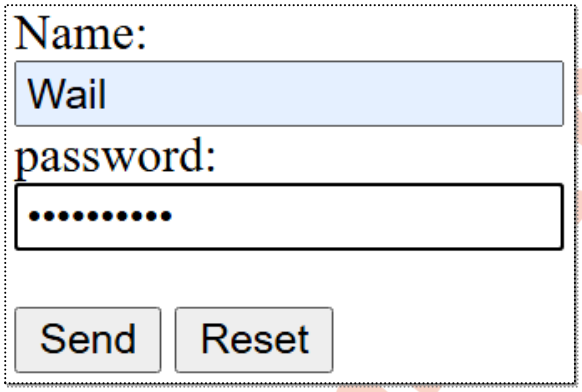
وسوف نستعرض عند دراسة النوع submit التالي، مثالاً يوضح طبيعة عمل النوع reset والنوع submit ، ليتضح للمستخدم الفرق في طبيعة عمل كل منهما.

wailmonir0@protonmail.com

submit

نوع يعمل كزر، يقوم بعمل اعتماد submit للنموذج <form> ، حيث يقوم بإرسال بيانات النموذج إلى صفحة أخرى، أو إلى خادم server ، ليقوم بمعالجتها وفقاً للهدف من إنشاء هذا النموذج، وعند تعيين الخاصية action للنموذج أو الخاصية formaction للزر، فإن قيمة أي منهما تكون هي وجهة بيانات النموذج بعد اعتمادها .

```
<form>
<label for="user"> Name:</label><br>
<input   type="text"  id="txt" name="user"><br>
<label for="pswrd"> password:</label><br>
<input  type="password"  id="pswrd" name="pswrd"><br><br>
<input type="submit" value="Send"  formaction="page.php">
<input type="reset">
</form>
```



submit

reset

لاحظ أننا لم نعين الخاصية value للنوع reset فظهر عليه الاسم الافتراضي، بينما استخدمنا نفس الخاصية، لتغيير الاسم الافتراضي للنوع submit إلى Send .

لاحظ طريقة انتقال بيانات النموذج بعد اعتماده، كما تظهر في شريط العنوان، حيث يظهر اسم الحقل، وهو قيمة الخاصية name ، مسنداً إليه القيمة التي تم إدخالها من قبل المستخدم، في النوع text والنوع password .

والنوع submit يحتاج لتعيين الخاصية value ، في حالة الرغبة في تغيير الاسم الافتراضي الذي يظهر على الزر، لأنها تمثل اسم الزر الذي سيظهر للمستخدم.

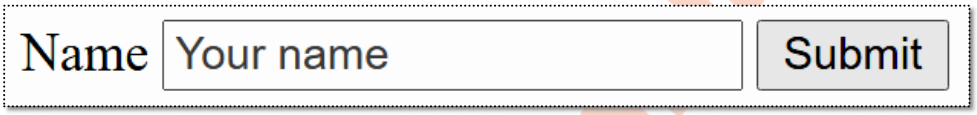
بما أن هذا النوع هو زر يقوم باعتماد بيانات النموذج، فلاحظ أن هذا الاعتماد يحتاج إلى عدد من الخصائص مثل formaction, formmnovalidate اللتين سبق شرحهما من قبل، بالإضافة إلى الخصائص formtarget, formenctype, formmethod ، وتم شرح طبيعة عمل هذه الخصائص وتأثيرها في ملحق 1 .

wailmonir0@protonmail.com

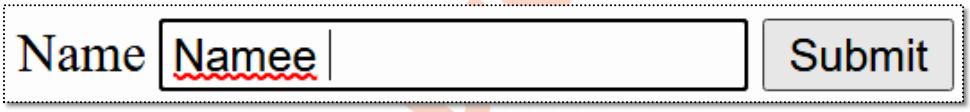
text

نوع مخصص لإدخال نص من سطر واحد، وهو القيمة الافتراضية، في حالة عدم تعيين أي قيمة للخاصية type للعنصر <input> .

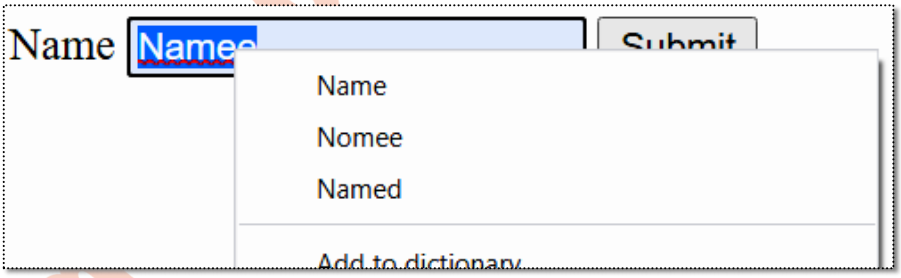
```
<form>
<label for="txt"> Name </label>
<input type="text"  id="txt"  name="text"  size="20"  minlength="11"
maxlength="100"  spellcheck="true" placeholder="Your name">
<input type="submit">
</form>
```



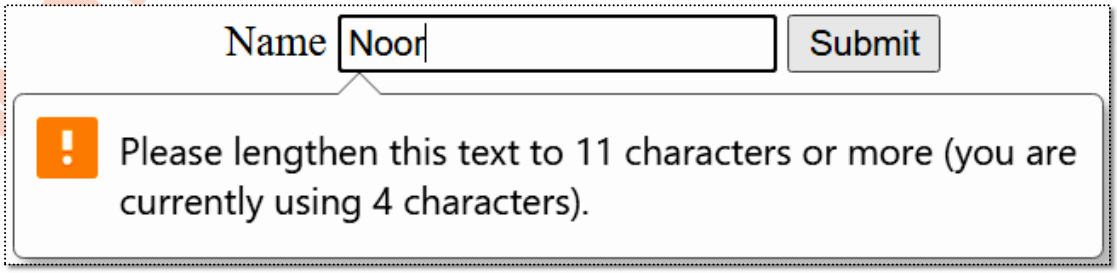
لاحظ أن تعيين الخاصية spellcheck وإسناد القيمة true لها، سيقوم بتنبيهك عند وجود خطأ هجائي عند إدخال النص، ولكن ذلك الخطأ لن يمنع اعتماد النموذج.



وعند الضغط على الزر الأيمن للفأرة، تظهر قائمة تعرض عليك مجموعة من الكلمات المقترحة، لاستبدال الكلمة الغير صحيحة.



وكما سبق ودرسنا من قبل، فإن الخاصيتين maxlength و minlength لا يتم اعتماد النموذج عند حدوث أي خطأ في قيمة أي منهما:



```
<form>
<label for="txt"> Name </label>
<input  type="text"  id="txt"  name="text"  value="Wail Monir"
readonly>
<input type="submit">
</form>
```

لاحظ أن الحقل Name تظهر به قيمة الخاصية value ، وقد قامت الخاصية readonly بجعل الحقل غير قابل للتعديل، وبالتالي لم يعد الحقل في حاجة لمعظم الخصائص السابقة.

Name Wail Monir Submit

ويمكن تعيين عدة خصائص للنوع text ، وقد سبق شرح كل الخصائص بالكود السابق في مواضع كثيرة، ويمكن الرجوع إليها في أحد المواضع السابقة، وهي:

value, name, minlength, maxlength, placeholder, spellcheck, pattern, readonly, required, size.

wailmonir0@protonmail.com

search

نوع مخصص لإدخال نص معين، بغرض البحث والاستعلام عن موضوع يتعلق بهذا النص، ويمكن تعيين نفس خصائص النوع **text** لهذا النوع، وفي الغالب لن يكون من بينها الخاصية **readonly** أو الخاصية **value** ، حيث إن وجودهما يخالف طبيعة عمل هذا النوع، حيث إنه يتطلب دائماً إدخال نص جديد بهدف البحث عنه.

```
<form>
<label for="srch"> Web Search </label>
<input  type="search"  id="srch" name="srch" maxlength="100"
minlength="10" size="20" placeholder="Search what ?" spellcheck="true">
<input type="submit" value="Go">
</form>
```

Web Search [Search what ?] [Go]

ويمكن تعيين الكثير من خصائص العنصر **<input>** للنوع **search** ، وكل الخصائص في الكود السابق، قد مررنا بها من قبل في مواضع كثيرة، ويمكنك الرجوع إليها في أي موضع منها، وعليك أن تتوقع نفس السلوك والتأثير لهذه الخصائص على هذا النوع.

wailmonir0@protonmail.com

tel

نوع مخصص لإدخال رقم هاتف، ولا يتم التحقق من صحة إدخال رقم الهاتف، مثل البريد الإلكتروني، أو مواقع الويب، لأن تنسيق أرقام الهاتف يختلف من مكان إلى مكان حول العالم.

والنوع tel يعامل معاملة النوع text ، لذلك لن يحدث أي خطأ أو اعتراض، إذا أدخلنا حروفاً بدلاً من الأرقام، بل إن النموذج سيتم اعتماده بطريقة عادية، وسيظهر اسم الحقل، وقد أسند إليه القيمة النصية كالتالي:

Phone no. Wael Monier Send

File | C:/Users/wamo/Desktop/index.html?phone=Wael+Monier

لذلك ينصح بتعيين الخاصية pattern لهذا النوع، ووضع القيود اللازمة للتحقق من صحة تنسيق رقم الهاتف، حسب تنسيق المنطقة من العالم المطلوب إدخال رقم هاتف خاص بها.

والكود التالي يوضح عينة بسيطة للخاصية pattern ، للتاكد من إدخال أرقام بتنسيق معين، وأي مخالفة لذلك، ستظهر رسالة تنبيه بهذا الخطأ عند اعتماد النموذج:

```
<form>
<label for="phone"> Phone no. </label>
<input type="tel"  id="phone"  name="phone"  size="20"
maxlength="100"  minlength="11"
placeholder="002########"  pattern="002[0-9]{8}">
<input type="submit" value="Send">
</form>
```

Phone no. 0021598756 Send

Please match the requested format.

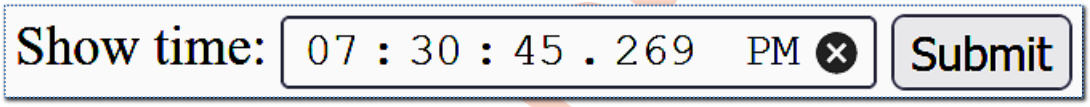
لاحظ ظهور رسالة تنبيه عند عدم استيفاء متطلبات عينة أو نموذج pattern .

والنوع tel يقبل الكثير من خصائص العنصر <input> ، وكل الخصائص التي وردت بالكود السابق ، تم شرحها في مواضع كثيرة، ويمكن الرجوع إلى كل منها في أحد هذه المواضع.

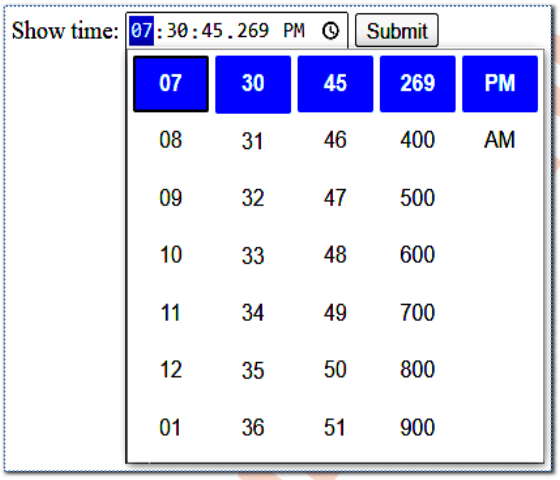
time

نوع مخصص لإدخال الوقت، بالساعة والدقائق والثواني وأجزاء الثانية، ولكن إدخال الثواني وأجزائها يكون اختيارياً، ويمكن الاستغناء عنهما، كما أن مربع الوقت يعرض الفترة الزمنية سواء كانت صباحا "AM" أو مساءً "PM" إذا كان التوقيت المستخدم 12 ساعة، ولا يعرض رمز الفترة الزمنية سواء كانت صباحية أم مسائية إذا كان التوقيت 24 ساعة (وذلك حسب توقيت نظام التشغيل)، ويختلف شكل مربع الوقت من متصفح إلى آخر.

```
<form>
<label for="tm">Show time:  </label>
<input  type="time"  id="tm" name="tm"  min="15:45:30"
max="23:20:55"  value="19:30:45.269">
<input type="submit" value="Submit">
</form>
```



Firefox

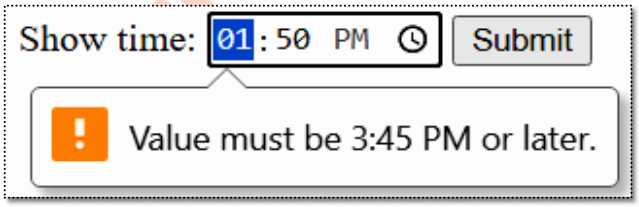


Chrome

لاحظ أن متصفح Firefox يعرض النوع time على هيئة سطر واحد، ويتم إدخال قيمة أخرى للوقت عن طريق الإدخال المباشر، باستخدام لوحة المفاتيح، أو باستخدام أسهم لوحة المفاتيح، أما متصفح Chrome فبجانب هذه الطرق، فإنه يفتح مربعاً، لاختيار الساعة والدقائق والثواني وأجزائها، بالنقر بالفأرة.

ويمكن تعيين عدة خصائص للنوع time :

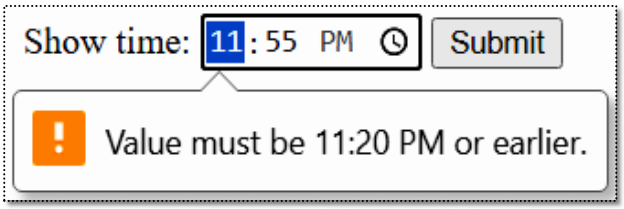
value



خاصية تحدد القيمة الافتراضية، التي تظهر عند فتح مربع الوقت، ويجب ألا تتخطى قيمة الخاصية max ، ولا تقل عن قيمة الخاصية

min ، وإذا حدث خلاف ذلك، لا يتم اعتماد النموذج، وتظهر رسالة تنبيه بهذا الخطأ.

max



خاصية تحدد أقصى وقت لا يمكن تخطيه، أو إدخال أي وقت بعده، ويجب ألا تقل قيمتها عن قيمة الخاصية min ، وإذا حدث خلاف ذلك، لا يتم اعتماد النموذج، وتظهر رسالة تنبيه بهذا الخطأ.

min

خاصية تحدد أدنى وقت لا يمكن النزول عنه، أو إدخال أي وقت قبله، ولا يجب أن تزيد قيمتها عن قيمة الخاصية max ، وإذا حدث خلاف ذلك، لا يتم اعتماد النموذج، وتظهر رسالة تنبيه بهذا الخطأ، نفس الشكل بالخاصية value .

wailmonir0@protonmail.com

url

نوع مخصص لإدخال عنوان موقع الكتروني.

```
<form>
<label for="url">URL:  </label>
<input   type="url"  id="url" name="url" maxlength="30" minlength="10"
placeholder="https://www.example.com" required
size="20" pattern="https://w{3}\..+\.(com|org)" spellcheck="true">
<input type="submit" value="Submit">
</form>
```

حقل إدخال موقع إلكتروني.

وتقوم عملية submit بالتحقق من أن الحقل ليس خالياً.

كما تتحقق من صحة تنسيق عنوان الموقع ولكن لا يتم التأكد من صحة وجود الموقع ، أي أن التحقق يكون شكلياً فقط.

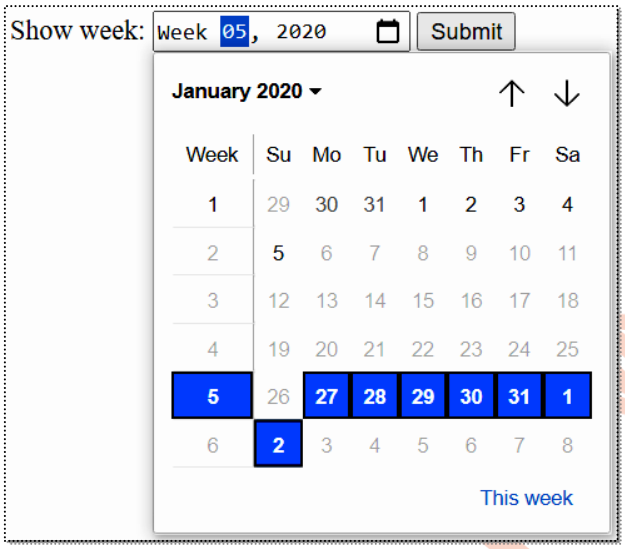
ويمكن تعيين عدة خصائص للنوع url ، وقد سبق شرح كل الخصائص التي وردت بالكود السابق في مواضع عديدة، ويمكن الرجوع إليها في هذه المواضع وهي:

value, name, minlength, maxlength, placeholder, spellcheck, readonly, required, size, pattern.

week

نوع مخصص لإدخال رقم الأسبوع من العام، أي من 1 إلى 52 أو 53 ، ويتم تنسيق المدخلات بإدخال العام مكوناً من أربعة أرقام، ثم يتبعه عدد الأسابيع مكوناً من رقمين مسبوقين بحرف W كبير، ويفصل بين العام وعدد الأسابيع "-".

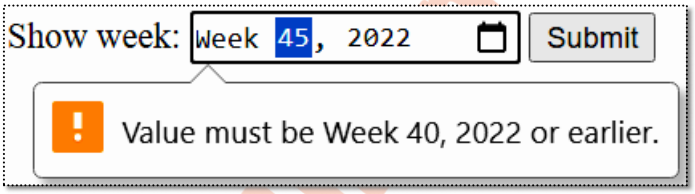
```
<form>
<label for="wk">Show week:  </label>
<input   type="week"  id="wk" name="wk" step="4"
 value="2020-W05" min="2018-W25" max="2022-W40">
<input type="submit" value="Submit">
</form>
```



لاحظ أن مربع الأسبوع قد ظهر به الأسبوع الخامس من العام 2020 (وهو قيمة الخاصية value) باللون الأزرق، وظهر الأسبوع الأول قبله بأربعة أسابيع نشطاً، وهي الفترة الزمنية في قيمة الخاصية step .

ويمكن تعيين عدد من الخصائص لهذا النوع:

max



خاصية تحدد أقصى عدد من الأسابيع، لا يمكن تخطيه أو إدخال أي عدد أكبر منه، ولا يجب أن تقل قيمة هذه الخاصية عن قيمة الخاصية min ، وإذا حدث خلاف ذلك، لا يتم اعتماد النموذج، وتظهر رسالة تنبيه بهذا الخطأ.

min

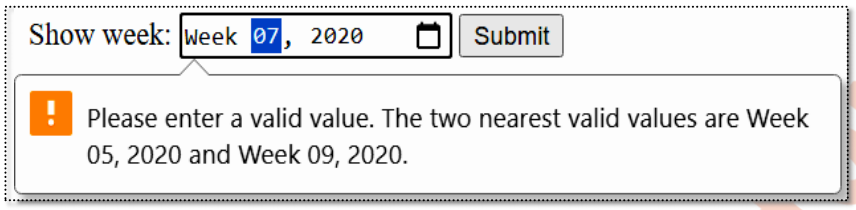
خاصية تحدد أقل عدد من الأسابيع لا يمكن النزول عنه أو إدخال أي عدد أقل منه، ولا يجب أن تزيد قيمة هذه الخاصية عن

قيمة الخاصية max ، وإذا حدث خلاف ذلك، لا يتم اعتماد النموذج، وتظهر رسالة تنبيه بهذا الخطأ.

value

خاصية تحدد القيمة الافتراضية، التي تظهر عند فتح مربع الأسبوع، ويجب ألا تتخطى قيمة الخاصية max ، ولا تقل عن قيمة الخاصية min ، وإذا حدث خلاف ذلك، لا يتم اعتماد النموذج، وتظهر رسالة تنبيه بهذا الخطأ كما بالشكلين السابقين.

step



خاصية القفزة بين القيم المدخلة، والتي تحدد الفترة غير النشطة بين المدخلات مقدرة بالأسابيع، حيث تظهر الأسابيع بين القيم النشطة غير فعالة، وفي حالة إدخال عدد أسابيع لا يتماشى مع قيمة هذه الخاصية، لا يتم اعتماد النموذج، وتظهر رسالة تنبيه بهذا الخطأ، ويمكن إسناد القيمة any لهذه الخاصية، وتعني أن كل الأسابيع نشطة، ولا يوجد فترة زمنية غير فعالة بينها.

📌 لاحظ أن كتابة عدد الأسابيع من رقم واحد كقيمة لأي خاصية، أو كتابة القيمة من رقمين ولكن يسبقهما حرف w صغير، سيجعل القيمة غير صحيحة، ولن تؤخذ هذه القيمة في الاعتبار.

📌 النوع week غير مدعوم في متصفح Firefox .

<textarea>...</textarea>

عنصر مخصص لإدخال نص متعدد الأسطر، ليستخدم في كتابة منشور أو تعليق.

```
<form>
<textarea  name="opn" id="opn" cols="20" rows="5" wrap="hard"
autocomplete  autofocus  placeholder="What you think ??? "
spellcheck="true"  required ></textarea><br><br>
<input type="submit" value="Submit">
</form>
```

What you think ???

Submit

لاحظ أن العنصر <textarea> قد ظهر بداخله تلميح عن نوع المدخل، وهذا التلميح يختفي عند وضع المؤشر داخله، ولكن إذا أردت أن يحتوي على قيمة افتراضية، فعليك أن تكتبها بين وسمي البداية والنهاية، لأن العنصر <textarea> لا يدعم الخاصية value .

ويمكن تعيين عدة خصائص للعنصر <textarea> :

maxlength

خاصية تحدد أقصى عدد من الحروف، يمكن للمستخدم إدخالها، ويجب أن تكون قيمتها عدداً صحيحاً موجباً. وتنطبق عليها نفس الشروط السابق شرحها ص 127 .

minlength

خاصية تحدد أقل عدد من الحروف، يمكن للمستخدم إدخالها، ويجب أن تكون قيمتها عدداً صحيحاً موجباً، وتنطبق عليها نفس الشروط السابق شرحها ص 127 .

cols

خاصية تحدد عرض العنصر مقدراً بعدد الحروف التي ستظهر للمستخدم أفقياً، والقيمة الافتراضية هي 20 حرف.

rows

خاصية تحدد ارتفاع العنصر، مقدراً بعدد الصفوف التي ستظهر للمستخدم رأسياً.

wailmonir0@protonmail.com

autofocus

خاصية عند تعيينها للعنصر، تجعله موضع التركيز التلقائي للوحة المفاتيح، من بين كل عناصر النموذج، أي أن لوحة المفاتيح ستستهدف هذا العنصر، عند الانتقال من نمودج آخر، أو من جزء آخر في الصفحة، إلى النموذج الذي يحتوي هذا العنصر، دون النقر بالمؤشر على العنصر، أو دون التنقل بين عناصر النموذج، عن طريق الضغط على مفتاح tab .

wrap

خاصية تحدد إمكانية كسر سطر الإدخال، والتفاف النص إلى سطر جديد، وهذه الخاصية لن يظهر أي فارق بين قيمها في المتصفح، ولكن الفارق يكون عند اعتماد النموذج، ودخول البيانات إلى السيرفر، وتقبل هذه الخاصية قيمتين:

- ***soft*** : القيمة الافتراضية، وتعني أنه لن يحدث كسر للسطر ، ولا التفاف للنص عند اعتماد النموذج
- ***hard*** : قيمة تعني حدوث انكسار للسطر، والتفاف النص لسطر جديد، عند اعتماد النموذج ويكون عرض السطر هو قيمة الخاصية cols .

ويمكن تعيين عدد آخر من الخصائص للعنصر <textarea> ، وقد سبق دراستها في مواضع كثيرة، ويمكن الرجوع إليها في مواضعها السابقة، وهي :

autocomplete , placeholder, spellcheck , readonly , required.

wailmonir0@protonmail.com

<button>...</button>

عنصر على هيئة زر، مخصص لأداء وظيفة معينة عند النقر عليه، ويختلف عن النوع button في العنصر <input> ، بأن قيمته تكتب بين وسمي البداية والنهاية، وليست قيمة للخاصية value، كما أنه يمكن أن يحتوي داخله على أي عنصر من عناصر html ، مثل <i> أو <b> أو <u> أو <img> وغيرها، في حين أن النوع button لا يقبل ذلك، لأن العنصر <input> هو عنصر فارغ empty element أي ليس له وسم نهاية، وبالتالي لا يمكن أن يحتوي أي عناصر داخله.

```
<form id="buttons_form">
<label>Name:</label><br><input type="text" name="user"><br><br>
<label>Password:</label><br><input type="password"
name="pass"><br><br>
<button type="submit" name="sbmt" value="submit"
autofocus><strong><i>Submit</i></strong></button>
<button type="reset" name="rst"><b><u>Reset</u></b>
</button>
<button type="button" name="btn"><img src="pic.png" width="30">
</button>
</form>
```

لاحظ أن أسماء الأزرار مكتوبة بين وسمي البداية والنهاية، وأن الزر الأول Submit ظهر بخط سميك ومائل، لوجود محتواه بين العنصرين و<i> ، وظهر الزر Reset بخط سميك وتحته خط، لوجود محتواه بين العنصرين <b> و <u> ، بينما ظهر الزر الثالث داخله صورة، لوجود العنصر <img> في محتواه، كما ظهر الزر Sumbit محاطاً بحدود خارجية outline نظراً لتعيين الخاصية autofocus له، فانتقل إليه، التركيز التلقائي للوحة المفاتيح.

والعنصر <button> له عدة أنواع تحددها الخاصية type .

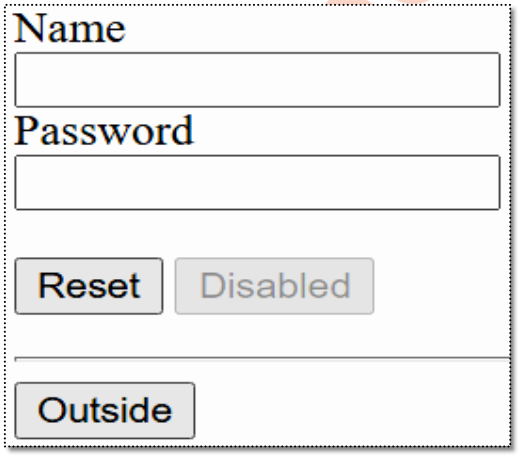
type

خاصية تحدد وظيفة العنصر <button> ويسند لها عدة قيم:

- ***submit*** : نوع يقوم باعتماد النموذج، لإرسال بياناته إلى السيرفر.
- ***reset*** : نوع يقوم بمسح البيانات التي تم إدخالها في النموذج، وإعادة الحقول إلى قيمها الافتراضية عند بداية تحميل الصفحة أو النموذج.
- ***button*** : نوع يقوم بتنفيذ وظيفة معينة، أو كود برمجي معين، عند الضغط عليه، مما يجعل سلوكه مشابهاً لسلوك النوع button في العنصر <input> .

بما أن النوع submit للعنصر <button> يقوم باعتماد بيانات النموذج، فلاحظ أن هذا الاعتماد يحتاج إلى عدد من الخصائص مثل formaction, formnovalidate اللتين سبق شرحهما من قبل، بالإضافة إلى الخصائص formtarget, formenctype, formmethod ، وتم شرح طبيعة عمل هذه الخصائص وتأثيرها في ملحق 1 .

```
<form id="buttons_form">
<label>Name</label><br><input type="text" name="user"><br>
<label>Password</label><br><input type="password"
name="pass"><br><br>
<button type="reset" name="rst" value="reset">Reset</button>
<button type="button" name="dsbl" value="button" disabled>
Disabled</button>
</form><hr>
<button type="submit"  form="buttons_form">Outside</button>
```

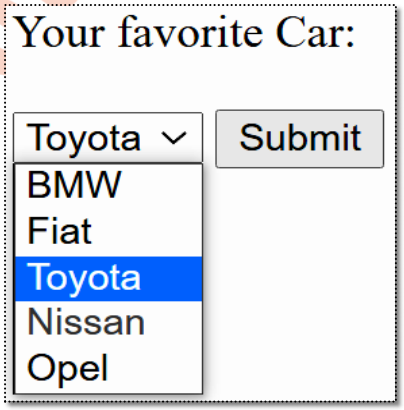


لاحظ أن الزر Outside يقوم باعتماد النموذج بالرغم من وجوده خارجه، لأنه مرتبط بالنموذج عن طريق الخاصية form ، التي أسند لها قيمة المعرف الفريد id للنموذج، كما أن الزر Disabled ظهر باهتاً، لأنه غير فعال، نظراً لتعيين الخاصية disabled له.

<select>...</select>

عنصر مخصص لإنشاء قائمة، تضم مجموعة بنود، لاختيار بند واحد منها أو أكثر .

```
<form>
<label  for="car">Your favorite Car:</label><br><br>
<select  id="car"  name="car" >
<option   value="bm">BMW</option>
<option   value="ft">Fiat</option>
<option   value="tt"  selected>Toyota</option>
<option   value="nsn"  disabled>Nissan</option>
<option   value="opl">Opel</option>
</select>
<button type="submit" >Submit</button>
</form>
```



ويمكن تعيين عدد من الخصائص للعنصر <select> :

id

خاصية تحدد المعرف الفريد، الذي سيربط العنصر <select> بالعنصر <label> ، ليسهل عملية البحث والانتقال إلى العنصر.

name

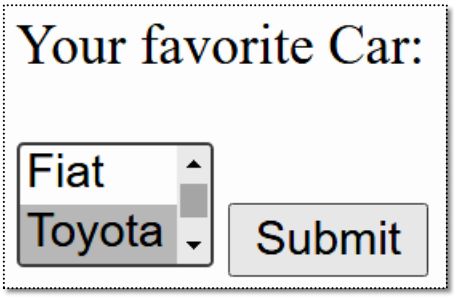
خاصية تحدد اسم العنصر ، والذي سيستخدم عند اعتماد النموذج.

size

خاصية تحدد عدد الصفوف (الاختيارات)، التي سيتم عرضها في واجهة القائمة <select>

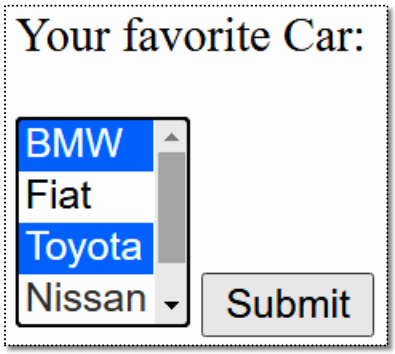
والقيمة تكون عدداً صحيحاً موجباً، لإظهار عدد الصفوف المطلوبة، والقيمة الافتراضية هي 0 وتعني عرض الاختيار الأول في القائمة، عند بداية تحميل الصفحة أو النموذج، أو عرض الاختيار الذي يتم تعيين الخاصية selected له، وإذا تم تعيين الخاصية لأكثر من اختيار ، يتم عرض آخر اختيار تم تعيين الخاصية له.

```
<select id="car" name="car" size="2">
```



لاحظ أنه تم عرض صفين من الاختيارات نتيجة تعيين الخاصية size للقائمة، وإسناد القيمة 2 لها.

multiple



خاصية تسمح باختيار أكثر من عنصر، على أن تكون القيمة size أكبر من 1 ، ويتم الاختيار في حالة اختيار عناصر متتالية، عن طريق الضغط على مفتاح Shift ، والنقر بزر الفأرة الأيسر على العناصر المختارة، أو باستخدام أسهم لوحة المفاتيح لأعلى أو لأسفل، وفي حالة اختيار عناصر متباعدة، يكون الاختيار عن طريق الضغط على مفتاح Ctl ، والنقر بالزر الأيسر للفأرة على الاختيارات المطلوبة.

```
<select id="car" name="car" size="4"  multiple>
```

ويمكن تعيين خصائص أخرى للعنصر <select> ، وقد تم شرحها من قبل على العديد من العناصر، مثل الخصائص autofocus و required و disabled ، ويمكن الرجوع إليها في أي من مواضعها السابقة.

ويكون كل عنصر في القائمة عبارة عن عنصر <option> .

wailmonir0@protonmail.com

<option>...</option>

عنصر داخلي مخصص ليكون أحد اختيارات القائمة، ويتم تعيين عدة خصائص له:

selected

خاصية تجعل الاختيار يظهر في واجهة القائمة، عند بداية تحميل الصفحة أو النموذج، بغض النظر عن ترتيبه داخل القائمة (مثل العنصر Toyota في المثال السابق).

disabled

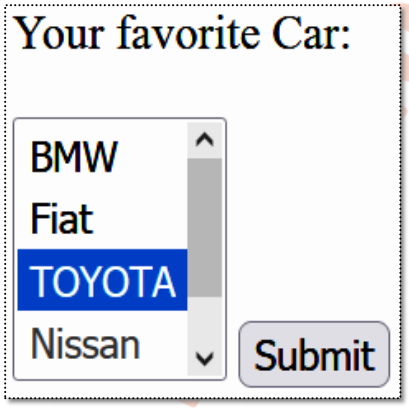
خاصية تجعل العنصر غير فعال داخل القائمة، وبالتالي لا يمكن اختياره (مثل العنصر Nissan في المثال السابق).

value

خاصية يتم تعيينها لكل عنصر، حيث إن قيمة هذه الخاصية، سوف تسند للخاصية name في العنصر <select> عند اعتماد النموذج، لتكون القيمة عند اعتماد النموذج بهذا الشكل name(select name)=value(option value) .

لاحظ أن الاسم الذي يظهر للمستخدم، هو النص بين وسمي البداية والنهاية لهذا العنصر، وهناك طريقة أخرى لوضع هذا الاسم، وهي الخاصية label .

label



خاصية تحدد اسم العنصر الذي سيعرض للمستخدم في القائمة، بدلاً من وضع الاسم بين وسمي البداية والنهاية للعنصر <option> ، وفي حالة تعيينها واختلاف قيمتها، عن قيمة الاسم بين وسمي البداية والنهاية، تكون السيادة لقيمة الخاصية label ، وإهمال النص بين وسمي العنصر.

```
<option  value="bm"  label="TOYOTA">Toyota</option>
```

لقد وجدت بالتجربة، أن قيمة الخاصية label لا تظهر في متصفح chrome عند تعيينها للعنصر <option> ، وتعمل بشكل جيد مع متصفح Firefox ، في حين أنها تعمل بشكل جيد على كل المتصفحات مع العنصر <optgruop> والذي سندرسه بعد هذا العنصر ، فإذا صادفتك نفس المشكلة مع العنصر <option> ، فالأفضل أن تستخدم دائماً النص بين وسمي البداية والنهاية لهذا العنصر، ليكون هو الاختيار الظاهر للمستخدم.

Tag Omission

يمكن الاستغناء عن وسم النهاية للعنصر <option> إذا تبع محتواه عنصر <option> آخر، أو كان هو آخر عنصر في قائمة الاختيارات، وقد سبق شرح ذلك في عدة مواضع سابقة، وعليك أن تتوقع نفس السلوك مع العنصر <option> .

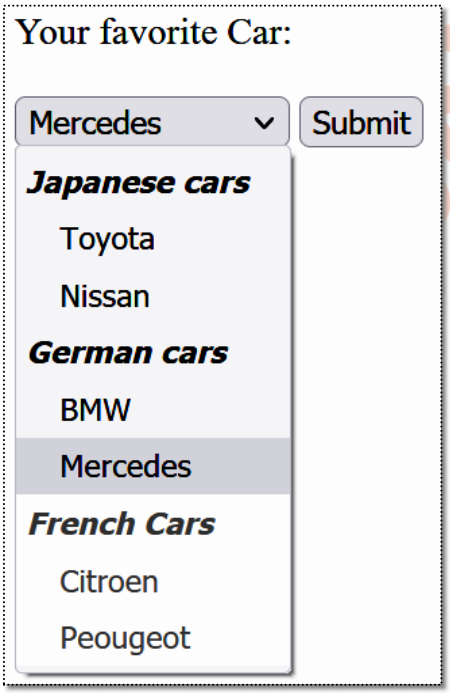
يمكن وضع عدد من الاختيارات تحت مجموعة واحدة باستخدام العنصر <optgroup> .

wailmonir0@protonmail.

<optgroup>...</optgroup>

عنصر يقوم بدور العنصر الأب لمجموعة من الاختيارات، بوضعها داخله، فيمكن تمييز الاختيارات عن بعضها، أو عدم تفعيل مجموعة معينة من الاختيارات دفعة واحدة:

```
<form><label for="car">Your favorite Car:</label><br><br>
<select id="car" name="car" >
<optgroup label="Japanese cars">
<option  value="ft" label="Toyota"></option>
<option  value="nsn" label="Nissan"></option>
</optgroup>
<optgroup label="German cars">
<option  value="bm" label="BMW"></option>
<option  value="tt" label="Mercedes"></option>
</optgroup>
<optgroup label="French Cars" disabled>
<option  value="cit" label="Citroen"></option>
<option  value="pgt" label="Peougeot"></option>
</optgroup>
</select>
<button type="submit" >Submit</button></form>
```



لاحظ أن قيمة الخاصية label للعنصر <optgroup> ظهرت أعلى مجموعة الاختيارات تحتها، وكذلك، لاحظ أن كل اختيارات المجموعة French cars أصبحت غير فعالة، نظراً لتعيين الخاصية disabled للعنصر <optgroup> ، الذي يضم هذه المجموعة من السيارات.

Tag Omission

يمكن الاستغناء عن وسم النهاية للعنصر <optgroup> إذا تبع محتواه عنصر <optgroup> آخر، أو كان هو آخر عنصر ابن، وقد سبق شرح ذلك في عدة مواضع سابقة، وعليك أن تتوقع نفس السلوك مع العنصر <optgroup> .

<fieldset>....</fieldset>

عنصر مجمع للحقول، مخصص ليجمع عدداً من عناصر النموذج <form> ، بينهم علاقة أو وظيفة معينة في إطار واحد، حيث يمكن التحكم فيهم دفعة واحدة، مثل جعل هذه العناصر غير فعالة، فلا يستطيع المستخدم التعامل معها عن طريق تعيين الخاصية disabled للمجمع، أو جعل كل العناصر غير قابلة للتعديل، عن طريق تعيين الخاصية readonly للمجمع، أو جعل ملء هذه العناصر ضروري لاعتماد النموذج، عن طريق تعيين الخاصية required للمجمع.

<legend>....</legend>

عنصر داخلي للمجمع <fieldset> مخصص لإنشاء عنوان للمجمع، ويمكن إنشاؤه في أي موضع داخل المجمع، وسوف يتم التعرف عليه على أنه عنوان للمجمع، ولكن الممارسة والتطبيق الأمثل أن يتم إنشاؤه كأول عنصر داخل المجمع.

```
<form><h3>Login</h3>
<label for="myname">Name<br></label><input type="text"
id="myname" name="user"><br>
<label>Password<br> <input type="password"
name="pass"></label><br><br>
<button type="submit" style="width: 85px">Ok</button>
<button type="button" style="width: 85px"> Cancel</button><br><br>
<fieldset name="frame" style="width: 180px;" disabled>
<legend>Sign up</legend>
<label for="mail">Email </label><input type="email" id="mymail"
name="ml"><br>
<label for="mail">Phone </label><input type="tel" id="mytel"
name="phn"><br>
</fieldset>
</form>
```

لاحظ أن المجمع اتخذ عنواناً وهو Sing up وهذا هو النص بين وسمي البداية والنهاية للعنصر <legend> .

لاحظ أن العناصر داخل المجمع ظهرت باهتة وغير فعالة، نتيجة تعيين الخاصية disabled للمجمع، فتم تطبيقها على جميع العناصر داخله.

لاحظ في الكود السابق، أن المجمع يقع داخل النموذج، وبالتالي، فعند تنشيطه سيتم إرسال بيانات العناصر داخله بعد ملئها، عند اعتماد النموذج، وإن كانت عناصر المجمع خارج النموذج، فيمكن الربط بينها وبين النموذج، عن طريق تعيين الخاصية form، للعنصر المطلوب ربطه بالقائمة من داخل المجمع، على أن يسند له قيمة المعرف الفريد للنموذج id .

Login

Name

Password

Ok Cancel

Sign up

Email

Phone

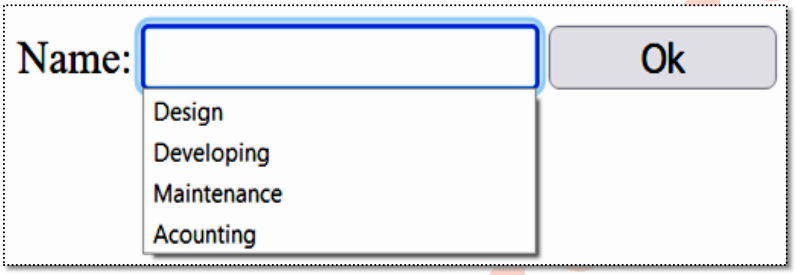
لاحظ أنني أطلقت على عنصر <fieldset> اسم "المجمع" ، حيث إن وظيفته هي تجميع عدد من عناصر النموذج في مجموعة واحدة، وذلك حتى لا يحدث لديك خلط بينه وبين عنصر الإطار <iframe> ، والذي يقوم بدمج مصادر خارجية إلى الصفحة، وسوف ندرسه ضمن العناصر المدمجة، في الجزء القادم من هذا المرجع.

wailmonir@protonmail.com

<datalist>....</datalist>

عنصر مخصص لإنشاء قائمة اختيارات، ترتبط بعنصر <input> حيث يتم إدخال أحد خياراتها إلى العنصر<input> ، وأي خيار في القائمة يكون بين وسمي عنصر <option> .

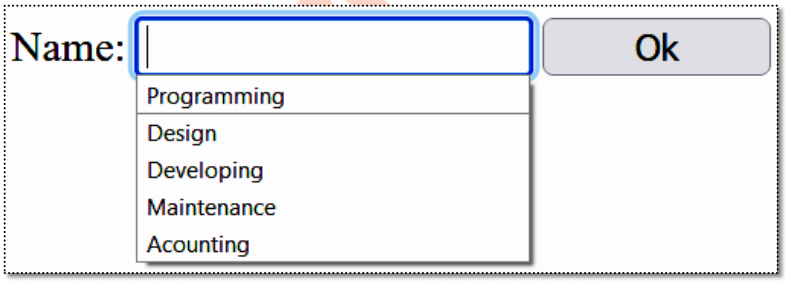
```
<form>
<label for="jobs">Name: </label>
<input id="jobs" name="jobs" list="jbs">
<datalist id="jbs">
<option  value="Design"></option>
<option  value="Developing"></option>
<option  value="Maintenance"></option>
<option  value="Acounting"></option>
</datalist>
<button type="submit" style="width: 85px">Ok</button>
</form>
```



لاحظ أن القائمة تنسدل عند الضغط على العنصر <input> وتظهر كجزء منه.

list

خاصية تربط بين عنصر القائمة <datalist> والعنصر <input> ، عن طريق تعيينها للعنصر <input> ، وتسند لها قيمة المعرف الفريد id لعنصر القائمة.



اعلم أنك تستطيع أن تضيف خيارات أخرى لهذه القائمة في حالة الرغبة في إدخال عنصر غير موجود في القائمة، فيمكن إدخاله عن طريق لوحة المفاتيح، وسيتم إدراج العنصر الجديد في القائمة عند فتحها مرة أخرى.

<output>....</output>

عنصر يعرض مخرجات كود برمجي معين، ولكن لا يتم اعتماد قيمته عند اعتماد النموذج، أي أنه مخصص للعرض فقط.

```
<form
onchange="result.value=Number(first.value)+Number(second.value)">
<input type="number"  id="frst" name="first" value="20">
<input type="number"  id="scnd" name="second" value="30">
<output name="result" for="first  second">0</output>
</form>
```

20 30 50

يوجد لدينا عنصري input من النوع number ، الأول first وأسندت له قيمة للخاصية value وهي 20 ، والثاني second وأسندت له القمية 30 ، ولدينا عنصر <output> هو result ، ونسند له أي قيمة بين وسمي البداية والنهاية، ولتكن 0 .

المطلوب إدخال قيمتين جديدتين إلى عنصري <input> أي first و second ، وجمع القيمتين first+second ، ووضع ناتج الجمع في العنصر <output> .

- نستخدم الخاصية onchange ، وهي تنفذ خطوات معينة عند حدوث أي تغير في القيمة، داخل أي حقل من حقول الإدخال <input> .
- نستخدم الخاصية value للحصول على القيمة داخل كل حقل.
- نستخدم الدالة Number لتحويل قيم المدخلات والمخرجات إلى أرقام، حيث إن مربعات الإدخال للعنصر <input> تقبل قيمتها على هيئة نص، ويجب تحويله لرقم، حتى يمكن إجراء أي عملية حسابية عليه.

for

خاصية تربط بين حقلي الإدخال <input> وحقل الإخراج <output> حيث يتم تعيينها للعنصر <output> ويسند لها قيمة الخاصية name لحقلي الإدخال.

عناصر الدمج Embedding Elements

مجموعة من العناصر لا تقوم بإنشاء عنصر جديد، ولكن تقوم بدمج مصادر خارجية في الصفحة، عن طريق تتبع المسار الموجود به المصدر الخارجي، وإسناده كقيمة للخاصية src ، أو أي خاصية أخرى تقوم بتتبع المسار في عنصر الدمج، ويكون دور هذه المجموعة من العناصر هو احتواء المصادر الخارجية وضبط خصائصها، وهذه المصادر الخارجية قد تكون صوراً أو ملفات نصية أو ملفات صوت أو فيديو أو صفحات ومواقع ويب وغيرها.

عنصر empty element يقوم بإدخال الصور بأنواعها إلى الصفحة.

```
<img src="road.jpg" alt="nice_view" width="200" height="120">
```

لاحظ أنه تم إدخال الصورة إلى الصفحة، وتم ضبط ارتفاعها وعرضها حسب المطلوب.

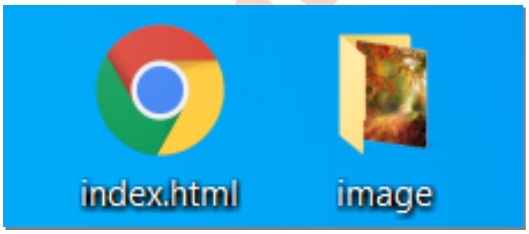
ويمكن تعيين عدة خصائص للعنصر <img> :

src

خاصية يسند إليها مسار الصورة المطلوب دمجها في الصفحة، ويتم الحصول على هذا المسار بعدة طرق:

الطريقة الأولى: وتسمى المسار النسبي أو relative path

1- إذا كانت الصورة مخزنة في نفس المكان (الفولدر أو المخزن فيه الصفحة، نضع اسم الصورة (directory مباشرة كقيمة للخاصية src ، كما في المثال السابق "road.jpg" .

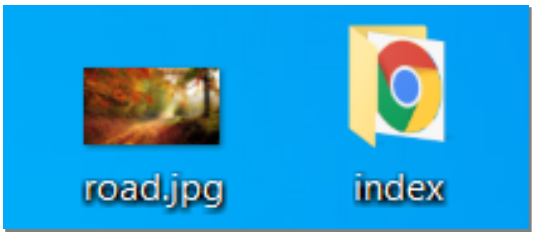


2- إذا كانت الصورة مخزنة داخل فولدر (directory) بجانب الصفحة، نكتب اسم الفولدر ثم اسم الصورة ويفصل بينهما "/" .

```
<img src="image/road.jpg " width="200" height="120">
```

وإذا كانت الصورة في فولدر داخل فولدر، نقوم بإضافة اسم الفولدر الابن الداخلي، بعد اسم الفولدر الأب الخارجي ويفصل بينهما "/" .

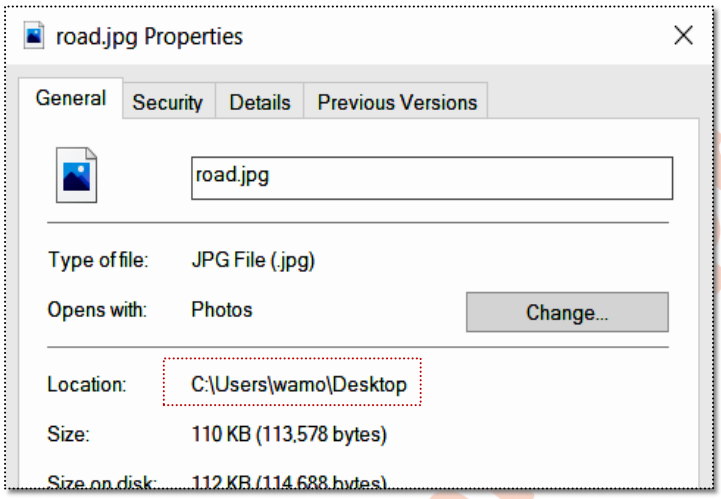
```
<img src="image/child/road.jpg " width="200" height="120">
```



3- إذا حدث العكس وكانت الصفحة مخزنة في الفولدر، والصورة خارج الفولدر نكتب نقطتين وشرطة مائلة قبل اسم الصورة، كدليل على أن مسار الصورة سوف يكون للخارج خطوة واحدة، وإذا خرجت الصورة أكثر من خطوة، نكرر وضع النقطتين والشرطة المائلة.

```
<img src="../road.jpg " width="200" height="120">
```

الطريقة الثانية، وتسمى المسار المطلق absolute path



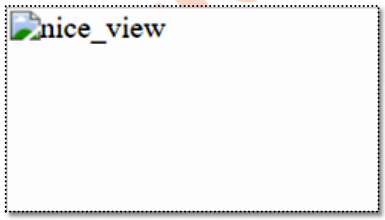
وتبدأ هذه الطريقة بأن تقف على الصورة بالفأرة، وتضغط بمفتاح الفأرة الأيمن وتختار البند properties من القائمة، فيفتح لك مربع الخصائص، انسخ منه المسار أمام location ، ثم ضع بعده شرطة مائلة، ثم اسم الصورة كقيمة للخاصية src .

```
<img src="C:\Users\wamo\Desktop\road.jpg"  width="200">
```

الطريقة الثالثة، الحصول على صورة من موقع ويب

إذا كانت الصورة موجودة على موقع ويب، نضع رابط الصورة كقيمة للخاصية src

```
<img src="https://cdn.pixabay.com/photo/2017/11/15/13/27/river-2951997_960_720.jpg"  alt="nice_view" width="200" height="120">
```



alt

خاصية تعرض نص يوضح مضمون الصورة، في حالة حدوث خطأ في مسار الصورة، وعدم ظهورها في المتصفح.

width

خاصية توضح قيمة عرض الصورة الذي ستظهر به في المتصفح، ويمكن أن تعطى قيمة مطلقة مقدرة بالـ pixels أو px، ولكن تكتب القيمة على هيئة رقم بدون وحدة بعدها (100) ، ويمكن أن تكون نسبة من عرض العنصر الأب الذي يحتوي الصورة، وإذا كانت بدون عنصر أب، فتكون قيمة عرض الصورة نسبة من عرض الصفحة (مثلاً 30%).

```
<img src="road.jpg" alt="nice_view" width="30%">
```

height

خاصية توضح قيمة ارتفاع الصورة الذي ستظهر به في المتصفح، ويمكن أن تعطى قيمة مطلقة مقدرة بالـ pixels ، ولكن تكتب القيمة على هيئة رقم بدون وحدة بعدها (100) ، ويمكن أن تكون نسبة من ارتفاع العنصر الأب الذي يحتوي الصورة، وإذا كانت بدون عنصر أب، فتكون قيمة ارتفاع الصورة نسبة من ارتفاع الصفحة (مثلاً 15%).

```
<img  src="road.jpg"  alt="nice_view"  height="15%">
```

لاحظ أن تعيين العرض أو الارتفاع كنسب مئوية، سيجعل حجم الصورة متغيراً حسب تغير أبعادهما، لذلك يجب عند تحديد العرض والارتفاع كنسب مئوية، أن تعين خاصية واحدة فقط منهما، لتحافظ على النسبة بين العرض والارتفاع ولا تتشوه الصورة.

decoding

خاصية تعطي المتصفح تلميحاً عن طريقة فك تشفير الصورة، ولها ثلاث قيم:

- ***sync*** : قيمة تعني أن فك تشفير الصورة يكون بشكل متزامن مع باقي المهام، أي أن كل مهمة يتم تنفيذها على حدة، واحدة تلو الأخرى.
- ***async*** : قيمة تعني أن فك تشفير الصورة، يكون بشكل غير متزامن مع باقي المهام، أي أنه يتم أثناء أداء مهام أخرى، أي بالتوازي معها، لتسريع عملية عرض باقي العناصر.
- ***auto*** : لا يوجد طريقة محددة لفك التشفير، ويترك للمتصفح تحديد الطريقة الأنسب للمستخدم من الطريقتين السابقتين.

crossorigin

خاصية توضح أنه في حالة دمج صورة، من موقع أو صفحة معينة إلى صفحتنا، فيمكن

إرسال معلومات عن صفحتنا (basic authentication, cookies, ...) إلى هذا الموقع، وهذه الخاصية لها قيمتان anonymous و use-credentials وشرح مدلول كل منهما ملحق 1 .

usemap

خاصية تقوم بتخصيص جزء معين من الصورة على شكل هندسي، لتكون clickable link area ، أي منطقة حساسة للنقر بزر الفأرة الأيسر، ومرتبطة بصفحة أو ملف أو صورة أو رابط أو وثيقة أو غيره.

ismap

خاصية تعني أن إرسال إحداثيات النقر على المنطقة الحساسة إلى السيرفر، إذا كانت الصورة محتواة داخل العنصر <a> .

loading

خاصية تحدد طريقة تحميل الصورة، ولها قيمتان:

- *eager* : القيمة الافتراضية، وتعني أن تحميل الصورة يتم فوراً عند بداية تحميل الصفحة، حتى وإن كانت الصورة خارج شاشة العرض (أي لم ينتقل إليها المستخدم أو لم يشاهدها بعد).

- *lazy* : تعني أن تحميل الصورة لا يتم، إلا عندما تقترب الصورة من شاشة العرض بمسافة معينة، لتكون جاهزة للعرض.

srcset

خاصية تتحكم في إظهار الصورة بالجودة المناسبة للجهاز الذي تعرض عليه، فالأجهزة الصغيرة مثل الهواتف النقالة لا تحتاج إلى صورة عالية الجودة مثل شاشة الحاسب، والعكس صحيح، فلا يمكن استخدام الصورة منخفضة الجودة الخاصة بالهواتف النقالة مع الأجهزة الكبيرة، لأن الصور سوف تظهر منقطة ومشوهة.

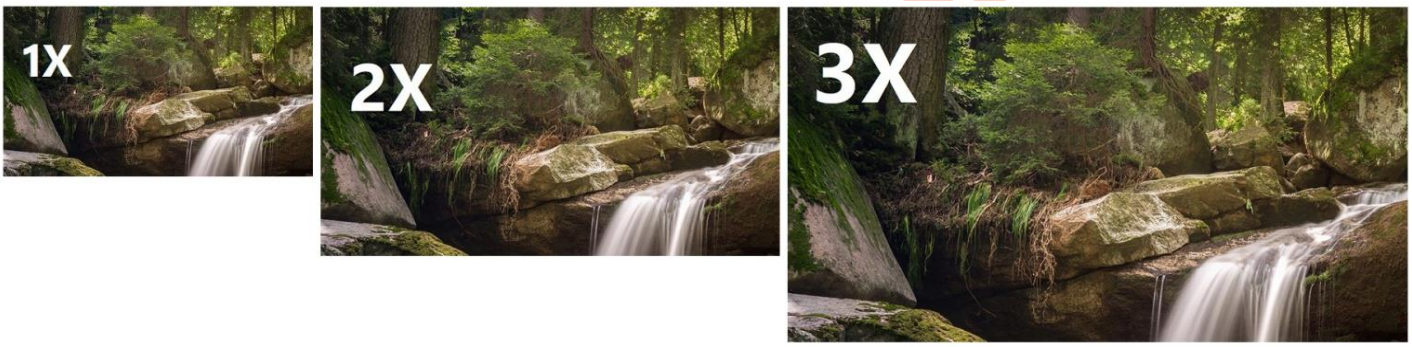
لذلك، فهذه الخاصية تحتوي قيمتها على صورة واحدة، ولكن لها أكثر من جودة، ونكتب بجوار اسم كل صورة حجم شاشة الجهاز الذي ستعرض عليه، ويفصل بين كل جودة والتي تليها بفاصلة بهذا الشكل "river-2X.jpg 2x , river-1X.jpg 1x" ، حيث تمثل القيمة 1x أقل حجم للشاشة وبالتالي أقل جودة، ثم القيمة 2x تمثل شاشة أكبر وجودة أعلى، وكلما زاد

حجم الشاشة زادت القيمة إلى 3x وهكذا، وعند عرض الصورة على أي جهاز، يتم اختيار الجودة المناسبة لهذا الجهاز، ويمكن الاستغناء عن القيمة 1x الخاصة بالجودة الأقل، حيث سيدرك المتصفح تلقائياً، أن أول قيمة هي أقل جودة:

```
<img src="river-1X.jpg" alt="river"
 srcset="river-1X.jpg,  river-2X.jpg 2x,  river-3X.jpg 3x">
```

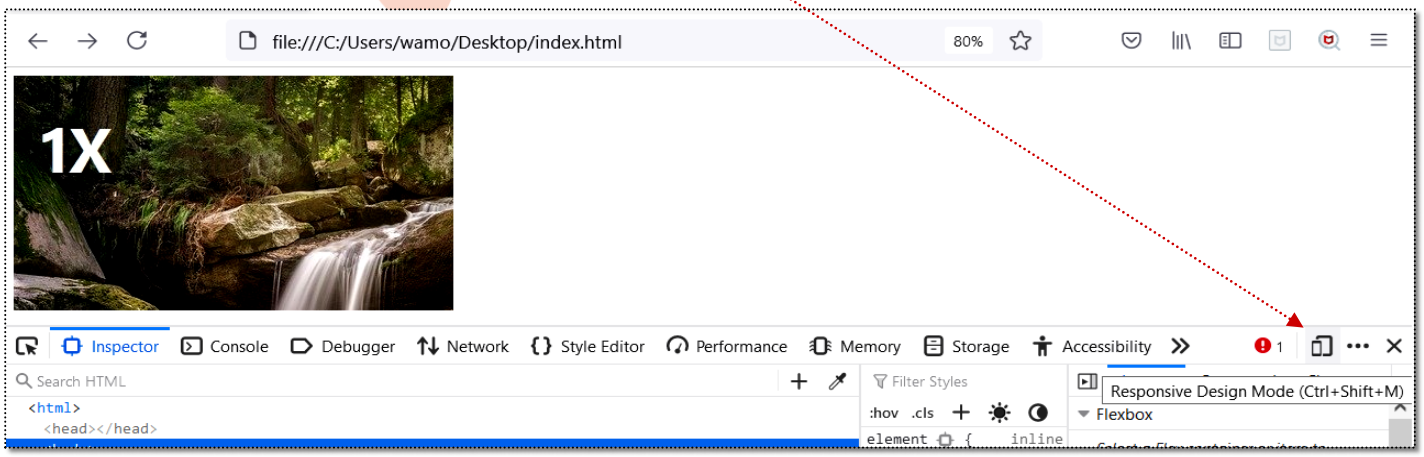
التطبيق:

ولتطبيق الخاصية السابقة، قمنا باختيار ثلاث نسخ من صورة واحدة، حيث تمثل هذه النسخ ثلاثة أنواع من الجودة، جودة منخفضة وجودة متوسطة وجودة عالية.

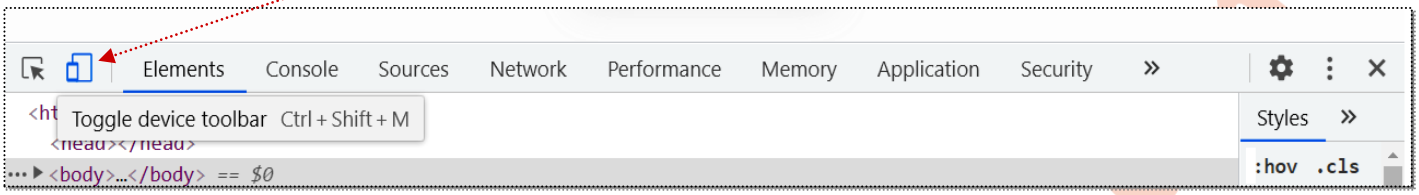


river-1X river-2X river-3X

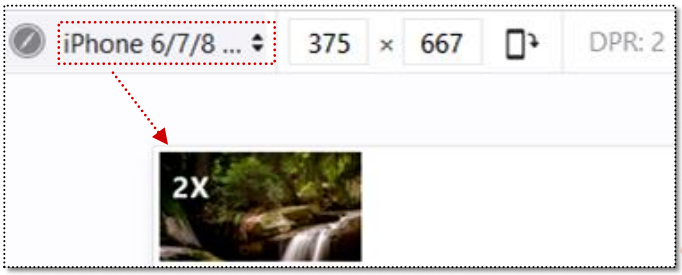
وأسندنا أسماء هذه النسخ مع جودتها كقيمة للخاصية srcset ، ويفصل بين كل نسخة والأخرى بفاصلة، وبعد إعداد الكود كما بالشكل السابق، وحفظه وفتحه في المتصفح، نضغط مفتاح F12 ، أو نضغط على مفاتيح Ctl+Shift+i ، أو نضغط مفتاح الفأرة الأيمن، ونختار من القائمة المنسدلة الأمر inspect ، ليفتح لنا المتصفح تبويبات أدوات المطور developer tools ونختار منها reponsive design mode كما في متصفح Firefox :

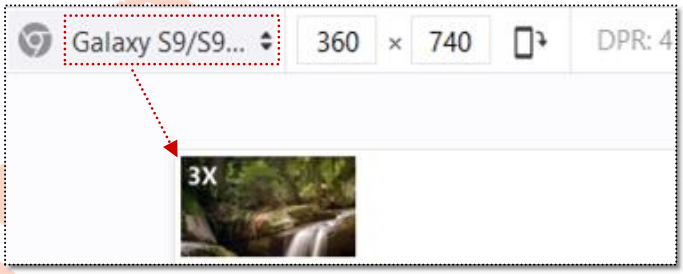


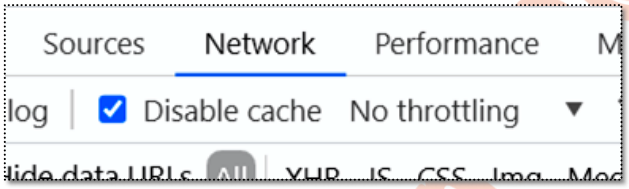
السهم السابق يشير إلى مكان أيقونة reponsive design mode في متصفح Firefox ، أما متصفح Chrome فتكون على يسار المستخدم، باسم toggle device toolbar .



لاحظ في الشريط العلوي أسماء أجهزة ذات شاشات ودقة عرض مختلفة، اختر بعضها واحداً تلو الآخر ، ولاحظ التغير في جودة الصورة، مع تغيير الجهاز.







لاحظ أن متصفح Chrome لن يقوم بعرض النتيجة فوراً مثل متصفح Firefox لأنه يحتاج لتفعيل الخاصية Disable cashe تحت التبويب Network في تبويبات Developer tools كما بالشكل المقابل، ثم عمل refresh للصفحة مع كل اختيار لجهاز جديد.

القيمة w

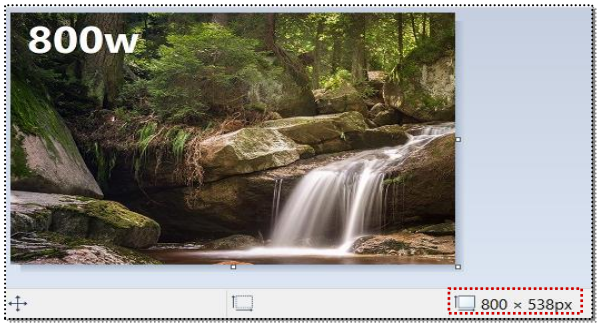
يوجد قيمة أخرى للتعبير عن جودة الصورة وهي القيمة w ، حيث يتم كتابة عرض الصورة متبوعاً بالحرف w مثل"river-1X.jpg 400w, river-2X.jpg 700w" مما يعني أن كل نسخة من الصورة، بدلاً من أن نحدد جودتها عن طريق الرمز x ، قمنا بتحديد جودتها باستخدام عرض الصورة عن طريق الرمز w، وحتى يدرك المتصفح متى يستخدم أي نسخة من الصورة، نقوم بتعيين الخاصية sizes .

sizes

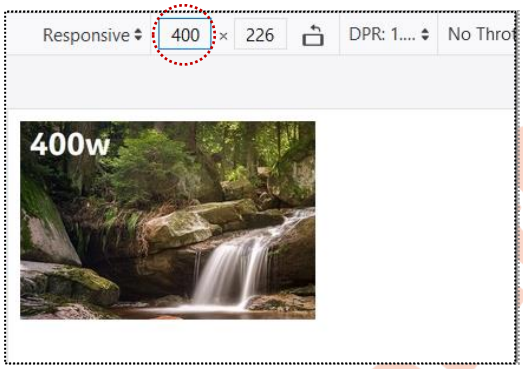
خاصية تقوم بتحديد عرض الشاشة، يتبعه عرض الصورة الذي ستظهر به:

```
<img src="river-1X.jpg"
srcset="river-1X.jpg  400w, river-2X.jpg  590w,river-3X.jpg 800w"
sizes="(max-width:400px) 200px, (max-width:590px) 400px, (max-width:
800px) 600px">
```

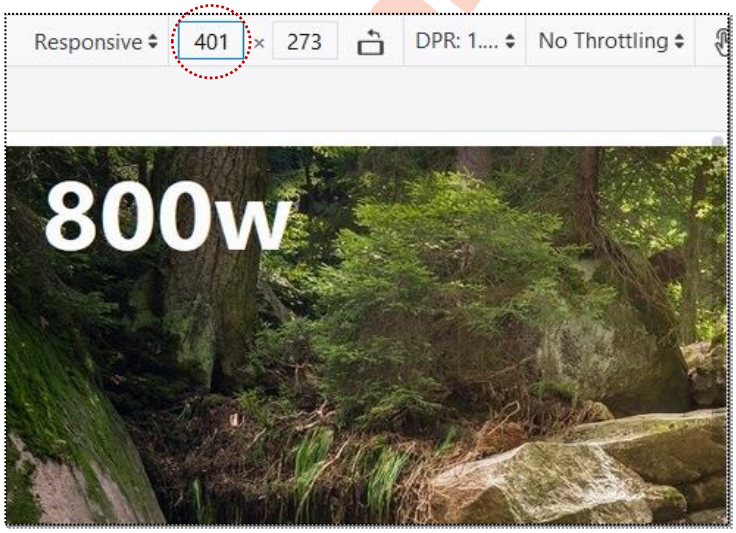
ولتطبيق هذه الخاصية قمنا باختيار نفس الصورة السابقة، بنسخها الثلاث ذات أنواع الجودة المختلفة، وأسندنا أسماء هذه الصورة للخاصية srcset ، تماماً كما فعلنا في المرة السابقة، ولكننا هذه المرة أعقبنا اسم كل نسخة من الصورة بعرض هذه الصورة متبوعاً بالحرف w .



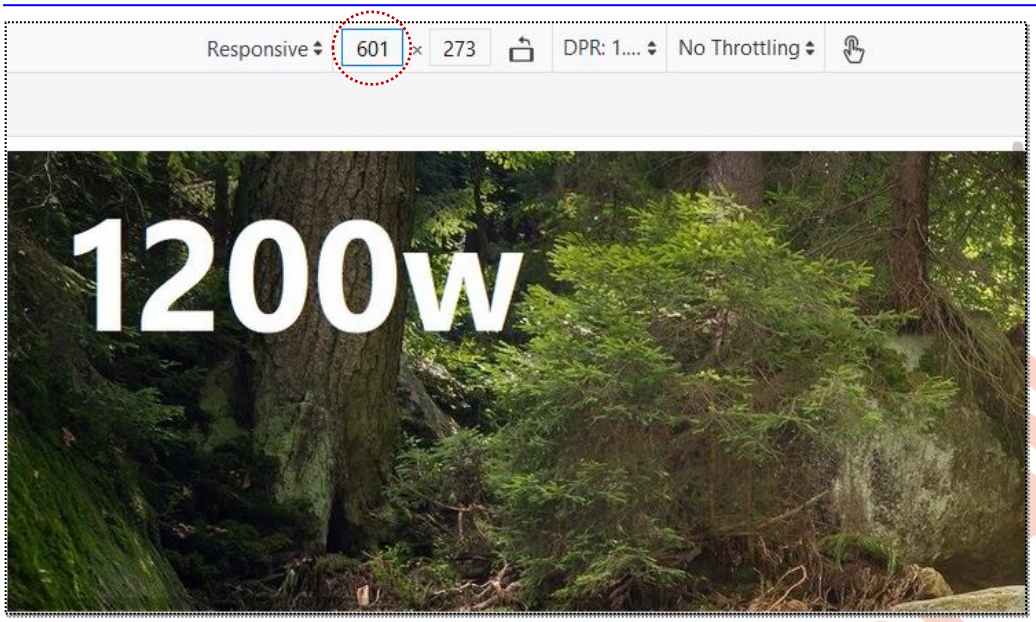
وهناك عدة طرق لمعرفة عرض الصورة أبسطها أن تقوم بفتح الصورة عن طريق برنامج الرسام، الذي يأتي مع نظام الويندوز، فستجد أن أبعاد الصورة في الشريط السفلي للبرنامج:



افتح صفحة الويب في المتصفح، وقم بفتح responsive desgin mode كما تعلمنا سابقاً، وقم بسحب الحد الأيمن، ولاحظ تغير نسخة الصورة، وتغير عرضها، بتغير عرض الصفحة التي تقوم بسحبها، كما بالشكل التالي:



لاحظ أن الصورة ظلت محتفظة بجودتها المنخفضة (400w) وعرض قدره 200px طالما بقى عرض الصفحة 400px أو أقل، فإذا زاد عرض الصفحة عن 400px تغيرت الصورة للنسخة ذات الجودة المتوسطة 800w وزاد عرضها إلى 400px طالما بقى عرض الصفحة أقل من 600px .



فإذا زاد عرض الصفحة عن 600px تغيرت الصورة للنسخة عالية الجودة 1200w.

لاحظ أن المتصفحات تعرض النتائج بدقة (وأحياناً مع هامش بسيط بالزيادة أو النقصان في عرض الصفحة) ، ويتبع التغيير في حجم الصفحة، تغيير نسخة الصورة وتغيير عرضها، في الحدود التي تم تعريفها في الخاصية sizes .

لاحظ أن متصفح Firefox يمكنه زيادة أو تقليل عرض الصفحة لأي عدد من المرات، وسوف تتغير نسخة الصورة حسب معايير الخاصية sizes ، مع كل زيادة أو نقصان في عرض الصفحة، في حين أن متصفح Chrome لا يعيد النسخة الأقل جودة من الصورة مهما أنقصنا من عرض الصفحة، طالما قمنا بزيادة عرض الصفحة ولو مرة واحدة، وتغيرت النسخة إلى النسخة الأعلى جودة، فإنه يظل محتفظاً بالنسخة الأعلى جودة، حتى تقوم بعمل refresh للصفحة.

التنسيق الافتراضي:

```
img{
    display: inline-block;
}
```

<map>...</map>

عنصر يستخدم لتخصيص منطقة من صورة معينة، لتكون منطقة حساسة للنقر بزر الفأرة الأيسر clickable link area ، ولها رابط بصفحة أو صورة أو ملف أو جزء من نفس الصفحة، ويتم تعيين الخاصية name لهذا العنصر، ليتم ربطه بالصورة المطلوبة، عن طريق إسناد قيمة الخاصية name يسبقها الرمز "#" كقيمة للخاصية usemap، التي يتم تعيينها لهذه الصورة.

التنسيق الافتراضي:

```
map{
    display: inline;
}
```

ويتم تحديد المنطقة الحساسة للنقر عن طريق العنصر <area> .

<area>

عنصر empty element ، يقوم بتحديد منطقة النقر، ويمكن أن نعين له مجموعة من الخصائص:

shape

خاصية تحدد شكل المنطقة الحساسة للنقر بالمؤشر، وتقبل أربع قيم:

- ***default*** : تكون الصورة كلها منطقة حساسة للنقر.
- ***rect*** : يكون شكل المنطقة الحساسة للنقر مستطيلاً.
- ***circle*** : يكون شكل المنطقة الحساسة للنقر دائرة.
- ***poly*** : ويكون شكل المنطقة الحساسة للنقر متعدد الأضلاع (ثلاثة أضلاع أو أكثر).

coords

خاصية تحدد إحداثيات الشكل الذي يتم اختياره في الخاصية السابقة:

في حالة المستطيل rect تكون الإحداثيات عبارة عن أربعة أعداد، يفصل بينها بفاصلة، حيث يمثل العددان الأول والثاني، نقطة بداية رسم المستطيل على المحورين الأفقي

والرأسي، ثم العدد الثالث يمثل عرض المستطيل، وأخيراً العدد الرابع ويمثل ارتفاع المستطيل، لتكون القيمة بهذا الشكل "0,0,200,75".

في حالة الدائرة circle تكون الإحداثيات عبارة عن ثلاثة أعداد، يفصل بينها بفاصلة، حيث يمثل العددين الأول والثاني، الإحداثيين الأفقي والرأسي لنقطة مركز الدائرة، والعدد الثالث يمثل نصف قطر الدائرة، لتكون القيمة بهذا الشكل "150,100,75".

في حالة الشكل متعدد الأضلاع poly تكون الإحداثيات على حسب عدد الزوايا بين كل ضلعين، فإذا كان شكل المنطقة مثلثاً، فيكون لدينا ثلاث زوايا، ولكل زاوية إحداثيان أفقي ورأسي، لتكون القيمة هي : "0,200 100,0 200,200".

href

خاصية تحدد رابط الصورة أو الملف أو الصفحة أو الجزء من الصفحة، الذي سيتم الانتقال إليه، عند الضغط على المنطقة الحساسة للنقر.

alt

خاصية تحدد النص الذي سيتم عرضه، عند وجود خطأ في مسار الصورة التي سيتم الانتقال إليها، عند النقر على هذه المنطقة.

hreflang

خاصية توضح لغة الصفحة التي سيتم الانتقال إليها، عند النقر على هذه المنطقة.

target

سبق شرحها، وقيمها ومدلولها ملحق 1 .

rel

سبق شرحها، وقيمها ومدلولها ملحق 1 .

download

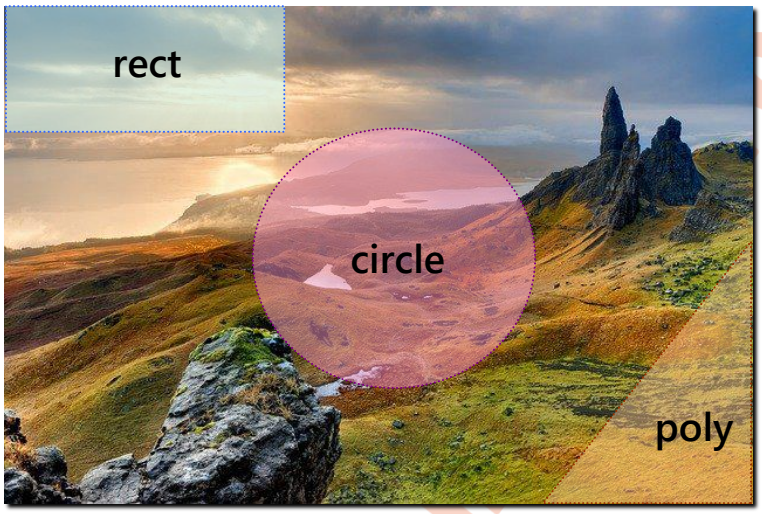
خاصية توضح أنه سيتم تحميل رابط المنطقة الحساسة إلى جهاز المستخدم، عند النقر على هذه المنطقة، على أن يكون على نفس origin (لتعريف origin انظر ص 282) واستعراض الصفحة أو الموقع من خلال السيرفر المحلي.

wailmonir0@protonmail.com

ping

خاصية تحدد قائمة بالمواقع، التي سيتم تتبعها للوصول إلى الصفحة التي سيتم الانتقال إليها، عند النقر على المنطقة المحددة.

```
<img src="mount.jpg" alt="view" width="300" usemap="#mnt">
<map name="mnt">
<area shape="rect" coords="0,0,100,50" href="#ftr">
<area shape="circle" coords="150,100,50" href="horse.jpg">
<area shape="poly" coords="300,100 300,200 200,200"
href="https://google.com">
</map>
<div style="height:1000px"></div>
<footer id="ftr">The magic of nature</footer>
```



لاحظ أنه عند الضغط بالفأرة على المنطقة rect ، يتم الانتقال إلى جزء الصفحة السفلي، وأن الضغط على المنطقة circle يؤدي إلى الانتقال إلى صورة على جهازك (ضع رابط لها)، وأن الصغط على المنطقة poly يؤدي إلى الانتقال إلى موقع google .

لقد قمت برسم هذه الأشكال التوضيحية للعنصر <area> لتتخيل شكلها فقط، ولكنها بالطبع لن تظهر عند تطبيق الكود السابق.

التنسيق الافتراضي:

```
area{
    display: none;
}
```

<picture>...</picture>

عنصر مخصص لاحتواء نسخ مختلفة، وجودة مختلفة، وأبعاد مختلفة، من صورة واحدة أو أكثر من صورة، بحيث تكون كل نسخة من صورة معينة ملائمة للعرض في بيئة معينة، مثل الهاتف النقال أو التابلت أو الحاسب، ويقوم المتصفح باختيار النسخة الأفضل للعرض، حسب البيئة التي يُستخدم فيها هذا المتصفح.

ويقوم العنصر <picture> بهذه الوظيفة من خلال العنصر الداخلي <source> .

<srource>

عنصر empty element مخصص لتحديد البيئة المطلوب عرض الصورة فيها، وكذلك تحديد النسخ المختلفة من الصورة، ويتم ذلك عن طريق الخاصيتين media وsrcset .

media

خاصية تقوم بتحديد نوع بيئة العرض، المطلوب عرض الصورة فيها.

srcset

خاصية تحدد نسخ الصورة ذات الأبعاد المختلفة، أو الجودة المختلفة، والتي سيختار المتصفح أحدها للعرض في البيئة التي سبق تعريفها في الخاصية media ، وتكون قيمتها عبارة عن اسم نسخة الصورة، يتبعها عرض هذه النسخة، متبوعاً بالحرف w مثل "pic.jpg 460w" ، أو اسم الصورة متبوعاً بدقة الصورة، ويرمز لها بالحرف x مثل "pic.jpg 2x" ، حيث إن الرقم قبل الحرف x يعني مقدار التكبير، منسوباً لأصغر دقة للصورة، أي أن القيمة 2x تعني نسخة من الصورة لها ضعف جودة أصغر جودة.

ويمكنك الاستغناء عن وضع حجم الصورة، أو مقدار الدقة بعد اسم الصورة، لإجبار المتصفح على الاستعانة بهذه النسخة من الصورة، في البيئة المعرفة في الخاصية media أياً كان حجمها، أو أياً كانت دقتها وجودتها.

type

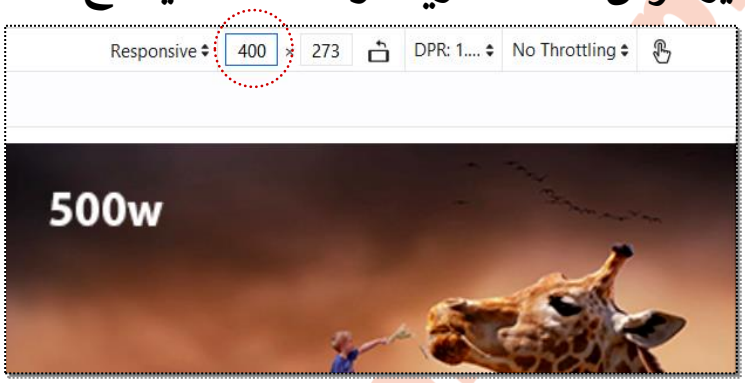
خاصية إضافية تحدد نوع الصورة المطلوب عرضها.

wailmonir0@protonmail.com

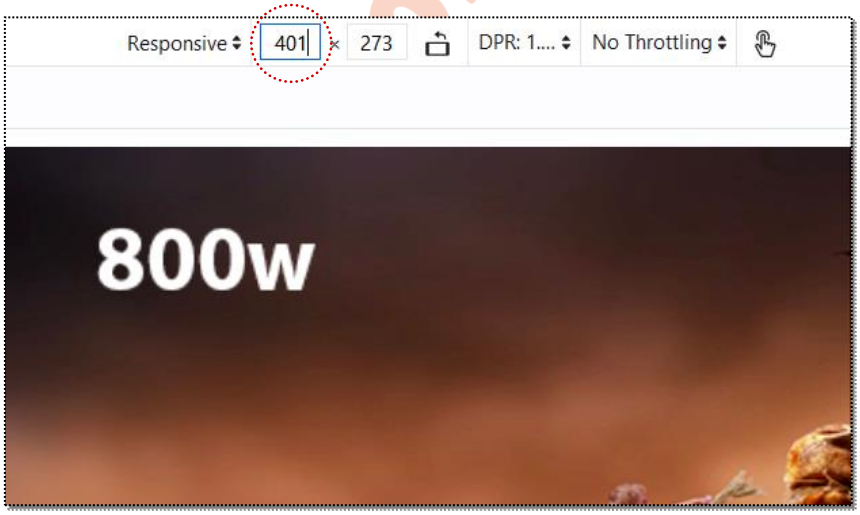
في حالة فشل المتصفح في اختيار نسخة الصورة المناسبة، أو كان المتصفح لا يدعم العنصر <picture> ، فيقوم العنصر <img> داخل العنصر <picture> بتحديد مصدر صورة أخرى، بشكل احترازي، لعرضها في حالة عدم عرض أي نسخة في البيئة المطلوبة.

```
<picture>
<source  media="(max-width: 400px)"  srcset="giraffe-500.jpg">
<source  media="(max-width: 600px)"  srcset="giraffe-800.jpg">
<source  media="(max-width: 800px)"  srcset="giraffe-1000.jpg">
<img src="paint.jpg">
</picture>
```

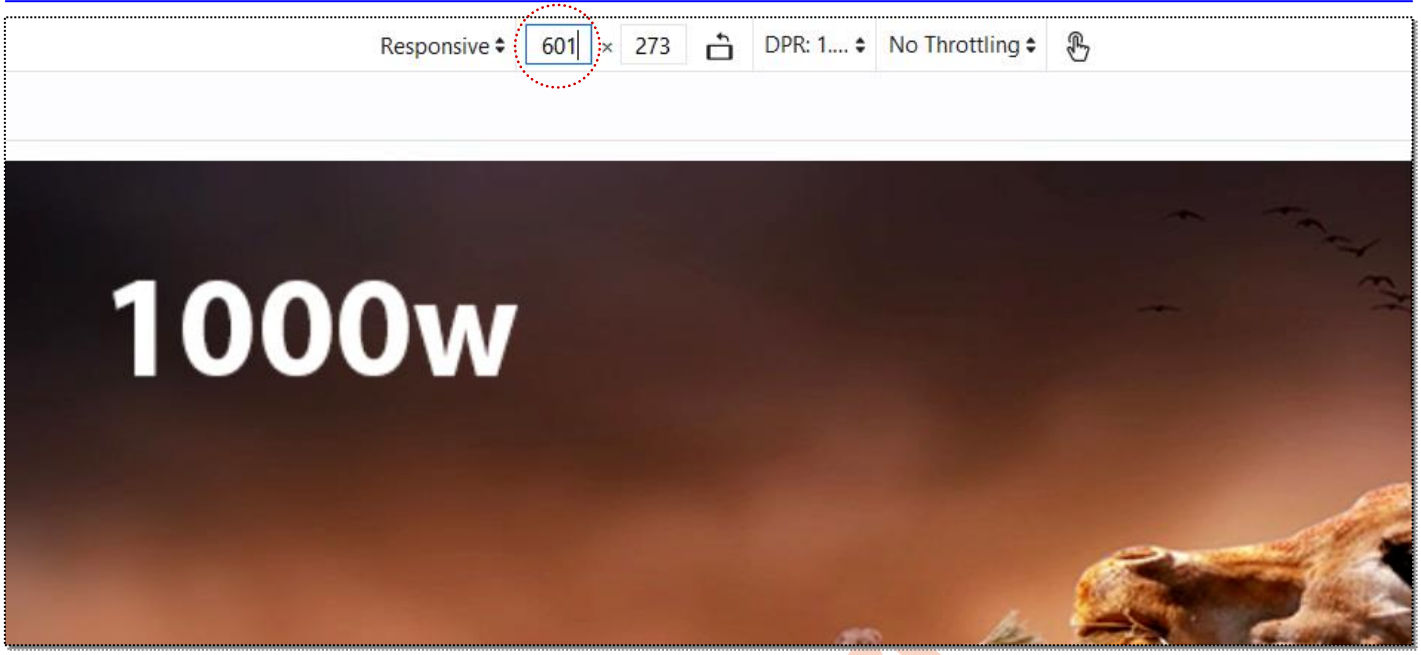
ولمشاهدة تطبيق الكود السابق على المتصفح، قم بتجهيز ثلاث نسخ من الصورة المطلوبة مختلفة الحجم والجودة، واجعلهم قيمة للخاصية srcset ، ثم قم بفتح responsive design mode كما تعلمنا سابقاً في العنصر <img> ، وقم بسحب الحد الأيمن للصفحة، ولاحظ تغير نسخة الصورة كلما تغير عرض الصفحة زيادة ونقصاناً، كما يتضح ذلك في الأشكال المقابلة.



لاحظ في الشكل الأول أن الصفحة ظلت محتفظة بالنسخة الأقل جودة 500w ، طالما كان العرض 400px أو أقل، وهو العرض الذي تم تحديده في الخاصية media أو بيئة العرض الأولى.



ولما زاد عرض الصفحة عن 400px انتقلت الصفحة إلى النسخة متوسطة الجودة 800w ، التي تم تحديدها في الخاصية media أو بيئة العرض الثانية.



ظلت الصورة محتفظة بجودتها المتوسطة حتى يصل عرض الصفحة إلى 600px ، فلما زاد عرض الصفحة عن 600px ، انتقلت الصورة إلى النسخة عالية الجودة 1000w ، التي تم تحديدها في media بيئة العرض الثالثة.

wailmonir0@proton

<figure>...</figure>

عنصر مخصص لاحتواء عنصر أو أكثر داخله، وقد تكون هذه العناصر رسومية، مثل صورة أو رسم بياني أو رسم تخطيطي، وقد تكون عناصر نصية، مثل مقالة أو بوست، ويتميز العنصر <figure> بأن محتواه بالرغم من أنه مرتبط بتدفق الصفحة، إلا أن موضعه مستقل عن تدفق الصفحة، فإذا تم إزالة المحتوى لا يتأثر التدفق في الصفحة، كما أنه يتميز بوضع عنوان لمحتوياته، عن طريق العنصر <figcaption> .

التنسيق الافتراضي:

```
figure{
        display: block;
        margin: 1em   40px;
        }
```

<figcaption>...</figcaption>

عنصر داخلي للعنصر <figure> يقوم بإنشاء عنوان لمحتويات هذا العنصر، ويمكن أن يكون أول أو آخر عنصر داخل العنصر <figure> حسب المكان المطلوب وضع العنوان فيه.

```
<figure >
<img src="colors.jpg" alt="colors" height="150">
<img src="road.jpg" alt="colors" height="150">
<figcaptin>Amazing Colors</figcaption>
</figure>
```

لاحظ أن عنوان الصور ظهر أسفل الصور، لأن العنصر <figcaption> آخر عنصر داخل

العنصر <figure> ، بينما في الشكل التالي ظهر العنوان أعلى الصور، لأن العنصر <figcaption> أول عنصر داخل العنصر <figure> .

```
<figure >
<figcaptin>Amazing Colors</figcaption>
<img src="colors.jpg" alt="yellow roses" height="150">
<img src="road.jpg" alt="yellow roses" height="150">
</figure>
```

التنسيق الافتراضي:

```
figcaption{
        display: block;
        }
```

<canvas>...</canvas>

عنصر مخصص لتنفيذ سكريبتات (أكواد برمجية) تقوم بعمل رسوم وأشكال وحركة باستخدام لغة الجافاسكريبت Javascript ، ولا مجال لدراسة هذا العنصر بالتفصيل حاليا، حيث تستلزم دراسته معرفة جيدة بهذه اللغة، وسوف ندرس مثال بسيط على ما يمكن أن يقوم به هذا العنصر.

ويمكن تعيين خاصيتين لهذا العنصر:

width

خاصية تحدد عرض مساحة الرسم للعنصر <canvas> .

height

خاصية تحدد ارتفاع مساحة الرسم للعنصر <canvas> .

```
<canvas id="canv"  width="200"  height="200"> </canvas>
<script>
const canvas = document.getElementById('canv');
const ctx = canvas.getContext('2d');
ctx.fillStyle = 'blue';
ctx.fillRect(10, 10, 200, 100);
</script>
```

لاحظ أن المعرف الفريد id للعنصر <canvas> يربط بينه وبين العنصر الذي تم إنشاؤه داخل العنصر <script> .

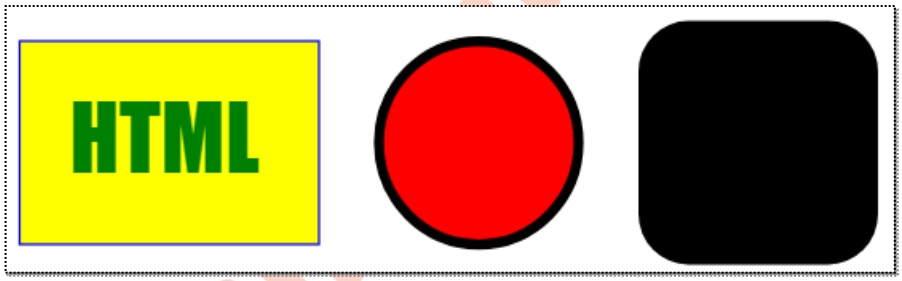
<svg>...</svg>

عنصر مخصص لاحتواء عناصر رسومية متجهة vector graphics ، وهذا العنصر يحتوي الكثير والكثير من العناصر، التي تستخدم في رسم العديد من الرسوم والأشكال، وهذه العناصر بدورها تحتوي على العديد من الخصائص، التي يسند لها العديد من القيم، والتي تمكنها من التحكم في شكل وسلوك العنصر الرسومي.

ونظراً لأهمية هذا العنصر وأهمية عناصره الداخلية، ونظراً لضخامة المادة العلمية اللازمة لتغطية هذه العناصر، فقد خصصت الإصدار التالي في هذه السلسلة، لتكون مرجعاً كاملاً لكل عناصر <svg> وخصائصه وقيمها.

وسوف نتناول مثالاً بسيطاً يوضح كيفية إنشاء بعض عناصر <svg> :

```
<svg  width="500"  height="500">
<circle  cx="240" cy="70" r="50" stroke="black" stroke-width="5"
fill="red">
<rect x="10" y="20" width="150" height="100" fill=yellow  stroke= blue
stroke-width= "5">
<rect x="320" y="10" width="120" height="120" rx="25" ry="25">
<text fill="green" font-size="45" font-family="Impact" x="35" y="85">
HTML</text>
</svg>
```



لاحظ أن العنصر <svg> احتاج لخاصيتي width وheight ليكونا أبعاداً لهذا العنصر الذي يعد لوحة رسم للعناصر المحتواة داخله، أما العناصر الداخلية فتحتاج خاصيتين لتحديد نقطتي بداية الرسم على المحورين الأفقي والرأسي (x, y) ، وخاصية أو أكثر لتحديد أبعاد هذا الشكل، سواء عرض width وارتفاع height أو نصف قطر (r)، ثم خاصية لتحديد لون ملء fill ، وخاصية لتحديد لون الخط الخارجي للعنصر stroke ، وخاصية لتحديد عرض الخط الخارجي stroke-width وغيرها.

فعلى سبيل المثال، في الكود السابق، هذه هي الخصائص الأساسية التي احتاجها عنصر الدائرة <circle> لإنشائه بالشكل السابق:

cx خاصية تحدد مقدار إزاحة نقطة مركز العنصر على المحور الأفقي x axis .

cy خاصية تحدد مقدار إزاحة نقطة مركز العنصر على المحور الرأسي y axis .

r خاصية تحدد طول نصف قطر الدائرة radius .

fill خاصية تحدد لون ملء العنصر.

stroke خاصية تحدد لون الخط الخارجي المحيط بالعنصر.

stroke-width خاصية تحدد عرض الخط الخارجي المحيط بالعنصر.

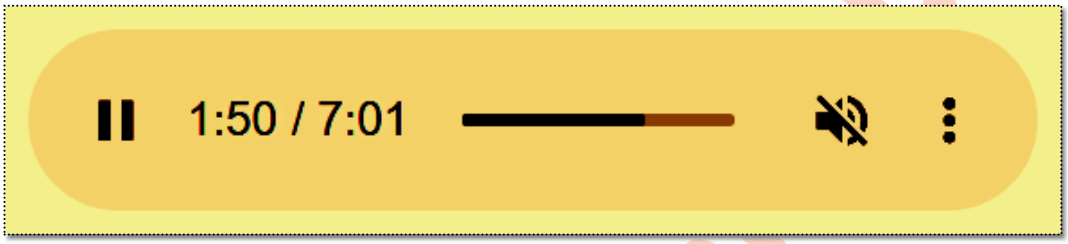
وهناك العديد والعديد من الخصائص، التي تتحكم في شكل الخط الخارجي ولونه وطريقة الملء وتشكيلها، وعناصر وخصائص الحركة، والفلاتر وغيرها من الخصائص، التي سأتناولها بالتفصيل في مرجعي القادم إن شاء الله.

wailmonir0@protonmail.com

<audio>...</audio>

عنصر مخصص لإدخال ملفات الصوت للصفحة والتحكم في وظائفها وسلوكها.

```
<audio controls muted autoplay loop preload="auto">
<source src="rabit.mp3" type="audio/mp3">
</audio>
```



والعنصر <audio> يجب أن يحتوي داخله على العنصر <source> .

<source>

عنصر empty element داخلي، يحدد موقع ملف الصوت الذي سيتم إدخاله للصفحة ونوعه، وذلك عن طريق تعيين خاصيتين لهذا العنصر :

src

خاصية تحدد مسار واسم الملف الذي سيتم إدخاله إلى الصفحة، سواء كان الملف محفوظاً على الحاسب، أو كان على موقع أو صفحة ما.

type

خاصية تحدد نوع أو تنسيق ملف الصوت، الذي سيتم إدخاله للصفحة، مثل audio/mp3 أو audio/wav .

ويمكن تعيين عدة خصائص للعنصر <audio> :

controls

خاصية تسمح بعرض الأزرار التي تتحكم في تشغيل الملف وإيقافه، وكذلك حِدة الصوت، وغيرها، وعدم وجود هذه الخاصية، يعني أن ملف الصوت لن يظهر في صفحتك.

muted

خاصية تعرض ملف الصوت مغلقاً عند تحميل الصفحة، وعلى المستخدم تفعيل الصوت

wailmonir0@protonmail.com

عند تشغيل الملف.

autoplay

خاصية تجعل ملف الصوت يعمل تلقائياً، عندما يكون جاهزاً بدون انتظار تحميل باقي أجزائه، ويجب تجنب استخدام هذه الخاصية، حتى لا يفاجأ المستخدم بوجود صوت عند تحميل الصفحة، وتترك له حرية تشغيل ملف الصوت من عدمه.

loop

خاصية تعمل على إعادة تشغيل ملف الصوت بعد انتهائه مرة تلو الأخرى دون توقف.

preload

خاصية تحدد طريقة تحميل ملف الصوت إلى الصفحة، ويسند إليها عدة قيم:

- ***none*** : قيمة تعني عدم تحميل ملفات الصوت مسبقاً عند تحميل الصفحة.
- ***auto*** : قيمة تعني أن يقوم المتصفح بتحميل كل ملفات الصوت، وإن لم يكن من المتوقع أن يتم استخدامها.
- ***metadata*** : قيمة تعني تحميل أي ملفات تصف ملف الصوت، مثل مدة تشغيله.

يتم تجاهل هذه الخاصية أحياناً، ودائماً في حالة وجود الخاصية autoplay .

crossorigin

في حالة الحصول على ملف الصوت من صفحة أو موقع آخر، فإن هذه الخاصية تحدد إمكانية إرسال معلومات، عن صفحتنا إلى الصفحة التي تحصلنا منها على ملف الصوت، ونوعية هذه المعلومات، وشرح قيم هذه الخاصية، ملحق 1 .

wailmonir0@protonmail.com

<video>...</video>

عنصر يقوم بإدخال ملفات فيديو إلى الصفحة.

```
<video controls muted width="250" height="150"
poster="masha_bear.jpg">
<source src="masha.mp4" type="video/mp4" >
<track src="masha_en.vtt" srclang="en" label="Playing football "
kind="subtitles" default>
<track src="masha_ar.vtt" srclang="ar" label="كرة القدم" kind="subtitles">
</video>
```

ويمكن تعيين نفس خصائص العنصر <audio> لهذا العنصر بنفس القيم وبنفس السلوك، ما عدا الخاصية controls في حالة عدم تعيينها سيظهر ملف الفيديو ولكن لن تستطيع تشغيله.

ويوجد خصائص أخرى يمكن تعيينها للعنصر <video> :

width

خاصية تحدد عرض ملف الفيديو الذي سيتم إدخاله إلى الصفحة.

height

خاصية تحدد ارتفاع ملف الفيديو الذي سيتم إدخاله إلى الصفحة.

poster

خاصية تحدد مسار صورة معينة، لوضعها في مقدمة ملف الفيديو، عند تحميل الصفحة قبل أن يبدأ المستخدم تشغيله.

playsinline

خاصية تعني أن الفيديو سيعمل في موضعه، ولن يملأ الشاشة عند بدء تشغيله، وغياب

wailmonir0@protonmail.com

هذه الخاصية، لا يعني أن الفيديو سيعمل بملء الشاشة تلقائياً.

والعنصر <video> يحتوي عنصرين داخليين:

<source>

عنصر empty element داخلي، يقوم بتحديد مسار ملف الفيديو الذي سيتم إدخاله عن طريق الخاصية src ، ويحدد نوع الملف عن طريق الخاصية type ، كما سبق أن أوضحنا.

<track>

عنصر empty element ، مخصص لإنشاء ملف ترجمة لملف الفيديو، أو عنوان، أو شرح لمادة مصورة، وسوف نطلق عليه مجازاً "ملف الترجمة" ، نظراً لأن هذا هو الاستخدام الشائع لهذا النوع من الملفات.

ويمكن تحديد عدة خصائص لهذا العنصر:

src

خاصية تحدد مسار الملف المصاحب (الترجمة) الذي سيتم ربطه بملف الفيديو.

srclang

خاصية تحدد لغة ملف الترجمة، ويجب أن يتم تعيين هذه الخاصية، إذا كان نوع الملف أو قيمة الخاصية kind هي subtitles .

default

خاصية تحدد ملف الترجمة الافتراضي، في حالة وجود أكثر من عنصر <track> ، وكل عنصر يحتوي ملف ترجمة خاص به، وفي حالة عدم تعيين هذه الخاصية لأي عنصر من عناصر <track> ، فلن يعمل أي ملف ترجمة منهم.

kind

خاصية تحدد نوع الملف المصاحب لملف الفيديو، أو الوظيفة التي يقوم بها هذا الملف ولها عدة قيم:

- ***subtitles*** : قيمة تعني أن الملف هو ملف ترجمة، إذا كانت لغة ملف الفيديو غير مفهومة للمستخدم، وربما يحتوي تعليقات أخرى، عن مكان التصوير أو الوقت أو التاريخ.
- ***description*** : قيمة تعني وصف نصي لمحتوى الفيديو، وهو مناسب لمكفوفي البصر.

wailmonir0@protonmail.com

- ***chapters*** : قيمة تعني أن الملف عبارة عن عناوين، تستخدم في حالة تعدد ملفات الفيديو، ويقوم المستخدم بالتنقل بينها.

- ***captions*** : قيمة تعني أن الملف هو نص، ينقل الحوار وكل التفاصيل الصوتية التي لا يمكن سماعها، مثل صوت الموسيقى أو أصوات الحركة والمؤثرات، وهذا النوع من الملفات ملائم للصم.

- ***metadata*** : قيمة تعني أن الملف مستخدم في جزء الإعدادات من الصفحة، الغير مرئي للمستخدم.

label

خاصية تضع عنواناً لملف الترجمة، ويستخدم المتصفح هذا العنوان لجدولة ملفات الترجمة المتاحة .

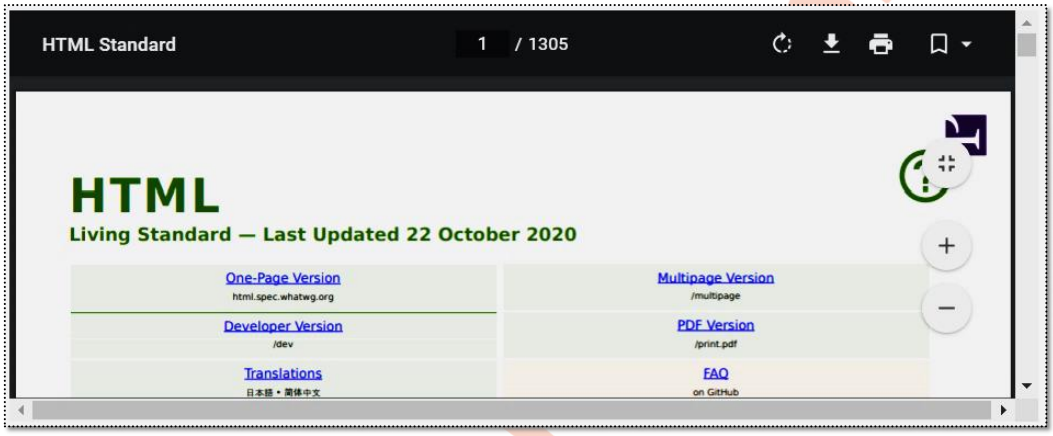
ويوجد الكثير من القواعد والإعدادات التي تتحكم في بناء ملف الترجمة، أو الملف المصاحب لملف الفيديو، ولدراسة كيفية إنشاء ملف ترجمة يراعي هذه القواعد والإعدادات والشروط، انظر ملحق 3 .

wailmonir0@protonmail.com

<embed>

عنصر empty element يقوم بإدخال محتوى خارجي إلى الصفحة، وهذا المحتوى قد يكون ملفاً لتطبيق خارجي، أو محتوى تفاعلي من الإضافات (plug-in) المستخدمة في المتصفح.

```
<embed src="html-standard.pdf" width="500" height="200"
type="application/pdf">
```



ويمكن تعيين عدة خصائص لهذا العنصر:

src

خاصية تحدد مسار الملف المطلوب إدخاله إلى الصفحة.

type

خاصية تحدد نوع الملف المطلوب إدخاله إلى الصفحة.

width

خاصية تحدد عرض العنصر <embed> .

height

خاصية تحدد ارتفاع العنصر <embed> .

wailmonir0@protonmail.com

<object>...</object>

عنصر مخصص لإدخال مصدر خارجي للصفحة، والذي يمكن أن يعامل كصورة أو يتم التعامل معه على أنه plug-in (والـ plug-in هو تطبيق أو برنامج يزيد من قدرات المتصفح في مجال معين مثل anti-virus).

```
<object  name="view"  data="grafitti.jpg"  width="220" height="150"
type="image/jpg"  usemap="#mnt">
</object>
<map  name="mnt">
<area  shape="circle"  coords="150,75,100"  href="mount.jpg">
</map>
```

لاحظ أن العنصر <object> في هذا المثال يعامل كصورة، وتم تحديد الخاصية usemap له لتحديد منطقة معينة على العنصر تكون حساسة للنقر ، وعند النقر عليها تقوم بنقلك إلى الرابط الموجود بالخاصية href الموجودة بالعنصر <area> ، ولكن هذه الخاصية مدعومة في متصفح Firefox فقط.

ويمكن تعيين عدة خصائص للعنصر <object> :

data

خاصية تحدد مسار العنصر الذي سيتم إدخاله إلى الصفحة.

name

خاصية تحدد اسم العنصر <object> .

width

خاصية تحدد عرض العنصر <object> .

height

خاصية تحدد ارتفاع العنصر <obeject> .

wailmonir0@protonmail.com

type

خاصية تحدد نوع الملف الذي سيتم إدخاله عن طريق العنصر <object> .

form

خاصية تربط العنصر <object> بنموذج معين، إذا وجد رابط بينه وبين أحد عناصر النموذج، حيث يسند لهذه الخاصية قيمة المعرف الفريد id للنموذج.

<param>

عنصر empty element داخلي للعنصر <object> ، يقوم باختيار خاصية معينة للعنصر <object> وإسناد قيمة لها، ويمكن تعيين خاصيتين له:

name

خاصية تحدد اسم الخاصية التي سيتم تعيينها للعنصر.

value

خاصية تحدد قيمة الخاصية التي تم تعيينها في الخاصية name .

```
<object name="movie" data="video.mp4" width="220" height="150"
type="video/mp4">
<param name="controls" value="false">
</object>
```

ورغم البحث في طبيعة سلوك هذا العنصر، وتجربة العديد من الأمثلة على موقع w3schools وغيرها من المواقع، ، وبرغم أن العنصر <param> مدعوم في كل المتصفحات، إلا أنني لم أجد له أي تأثير ملحوظ ولم أجد تفسيراً لذلك برغم الاستفسار في موقع MDN ويمكنك الاطلاع على الموضوع ومناقشته من خلال هذا الرابط:

https://github.com/mdn/content/issues/9223

<iframe>...</iframe>

عنصر مخصص لإنشاء إطار، ليتم دمج مصادر خارجية داخله، لدمجها في الصفحة، وهذه المصادر قد تكون صفحات ويب html أو صور jpg و png و gif أو ملفات صوت mp3 و wav أو ملفات فيديو mp4 و avi أو ملف تطبيق doc و pdf .

```
<iframe  width="560"  height="315"
src="https://www.youtube.com/embed/C5QFHp1oAws"
allow="accelerometer; autoplay; clipboard-write; encrypted-media;
gyroscope; picture-in-picture"  allowfullscreen>
</iframe>
```

ويتم التحكم في سلوك المصدر الذي سيتم إدخاله عن طريق عدة خصائص، يتم تعيينها للعنصر <iframe> وهي كالتالي:

src

خاصية تحدد مسار المصدر الخارجي الذي سيتم دمجه في الإطار .

width

خاصية تحدد عرض الإطار مقدراً بالـ px والقيمة الافتراضية هي 300 .

height

خاصية تحدد ارتفاع الإطار مقدراً بالـ px والقيمة الافتراضية هي 150 .

name

خاصية تحدد اسم الإطار.

allow

خاصية تحدد سياسة الميزات المتاحة لهذا الإطار، مثل إمكانية الوصول إلى الميكروفون أو الكاميرا أو البطارية أو تغيير السلوك الافتراضي للتشغيل التلقائي للفيديوهات، أو التأكد من أن الصور تتخذ حجماً مناسباً للعرض داخل الإطار.

wailmonir0@protonmail.com

allowfullscreen

خاصية تحدد إمكانية أن يعرض الإطار ومحتواه بملء شاشة العرض، ويسند لها قيمتان true أو false .

allowpaymentrequest

خاصية تحدد إمكانية إدخال الأكواد البرمجية الخاصة بالدفع الإلكتروني إلى الإطار، ليتضمن عمليات البيع والشراء والسداد الإلكتروني، ويسند لها قيمتان true أو false .

referrerpolicy

خاصية تحدد نوع معلومات صفحة المحيل referrer التي تحتوي عنصر <iframe> والتي سيتم نقلها إلى الصفحة destination ، عند جلب المصدر الخارجي منها، وقيم الخاصية ومدلولها ملحق 1 .

loading

خاصية تحدد طريقة تحميل الإطار، ولها قيمتان:

- *eager* : القيمة الافتراضية، وتعني أن تحميل الإطار يتم فوراً، عند بداية تحميل الصفحة، وإن كان الإطار خارج شاشة العرض (أي لم ينتقل إليه المستخدم، أو لم يشاهده بعد).

- *lazy* : تعني أن تحميل الإطار لا يتم إلا عندما يقترب الإطار من شاشة العرض بمسافة معينة، ليكون جاهزاً أمام المستخدم.

sandbox

خاصية تطبق قيوداً معينة على الإطار، وإذا كانت قيمة هذه الخاصية فارغة فهذا يعني تطبيق جميع القيود على الإطار، ويمكن إسناد قيم متعددة لها حسب القيود المطلوب تطبيقها، على أن يكون بين القيمة والقيمة التي تليها مسافة فارغة، والقيم هي:

- *allow-downloads* : قيمة تعني السماح بتحميل الملفات من رابط داخل الإطار بطلب من المستخدم.

- *allow-forms* : قيمة تعني السماح باعتماد أو submit للنموذج <form> من داخل

الإطار ، وعدم إسنادها يعني عدم إمكانية اعتماد النموذج من داخل الإطار.

- allow-modals : قيمة تعني إمكانية استخدام الأكواد التي تتعامل مع نافذه المتصفح من داخل الإطار، مثل window.alert() أو window.prompt() وغيرها.

- allow-orientation-lock : قيمة تعني عدم السماح بتغيير اتجاه الشاشة .

- allow-pointer-lock : قيمة تعني السماح بإدخال بيانات، أو تفاعل من المستخدم مع بيانات معروضة داخل الإطار، عن طريق تحريك الفأرة أو النقر بها أو استخدام أحداثها Mouse Events .

- allow-popups : قيمة تعني السماح بفتح نوافذ جديدة من داخل الإطار، في تبويب جديد بفعل الخاصية target="_blank" أو الدالة window.open() أو الدالة showModalDialog() ، وعدم إسناد هذه القيمة يعني عدم فتح أي نافذة جديدة من داخل الإطار.

- allow-popups-to-escape-sandbox : قيمة تعني السماح بفتح نافذة جديدة من داخل الإطار دون قيود، حيث لا ترث هذه النافذة القيود التي يفرضها الإطار الذي فتحت من خلاله.

- allow-presentation : قيمة تعني إمكانية فتح عروض تقديمية من داخل الإطار.

- allow-same-origin : قيمة تعني معاملة محتوى الإطار كما لو كان من نفس العنوان الرئيسي origin .

- allow-scripts : قيمة تعني إمكانية تشغيل الأكواد البرمجية من داخل الإطار، ولكن لا تسمح بتشغيل النوافذ المنبثقة من داخله.

- allow-top-navigation : قيمة تعني إمكانية التنقل من داخل الإطار، إلى الصفحات التي تفتح فوق الصفحة التي تحتوي الإطار أي target="_top" .

- allow-top-navigation-by-user-activation : قيمة تعني السماح بالتنقل إلى الصفحات التي تفتح فوق الصفحة الحالية من داخل الإطار، إذا بدأ المستخدم عملية التنقل للمستوى الأعلى أولاً.

srcdoc

خاصية تكون قيمتها عبارة عن كود html ، فتجعل قيمتها تعمل كصفحة html ، فتكون

wailmonir0@protonmail.com

عبارة عن مصدر وهمي، أو قيمة وهمية للخاصية src ، وعندها يمكننا ألا نقوم بتعيين الخاصية src .

```
<iframe  width="250"  height="100"  srcdoc="
<head>
<style>
body{background: lightblue;}
</style>
</head>
<body>
<h2>Hello World !!</h2>
<p><i>Let's start a web design course</i></p>
</body>" >
</iframe>
```

لاحظ أنه على الرغم من عدم تعيين الخاصية src ، إلا أن الخاصية srcdoc جعلت الكود في قيمتها يعمل كصفحة html واعتبرت هي قيمة الخاصية src .

رغم أن الخاصية srcdoc تلغي عمل الخاصية src ، إلا أنه ينصح دائماً بتعيين الخاصية src ، للاستعانة بها إذا كان المتصفح لا يدعم الخاصية srcdoc .

العناصر الوصفية (الغير مقروءة في جسم الصفحة)

Metadata Elements

كما سبق وأن ذكرنا، فهي العناصر التي تتحكم في إعدادات وتنسقات الصفحة والملفات المرتبطة بها وبيئات العمل المناسبة لها، وكل هذه العناصر توضع داخل العنصر <head> ، خارج جسم الصفحة المرئي للمستخدم، وهو العنصر <body> .

```
<head>....</head>
```

وكما ذكرنا، فإن العنصر <head> يحتوي كل عناصر metadata ، أو العناصر الغير مقروءة في جسم الصفحة وهذه العناصر:

- عنصر عنوان الصفحة <title> .
- عنصر الرابط الرئيسي للصفحة <base> .
- عنصر الصفحات أو الملفات المرتبطة بالصفحة <link> .
- عنصر تنسيق الصفحة باستخدام css العنصر <style> .
- عنصر اختيار وإعداد بيئات العمل للصفحة <meta> .
- عنصر إدخال أكواد برمجية أو سكريبتات، للتحكم في سلوك العناصر <script> .

wailmonir0@protonmail.com

<title>....</title>

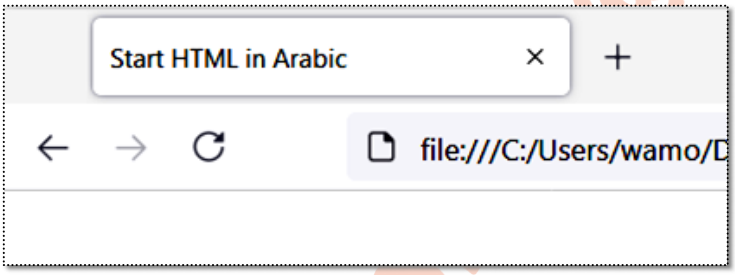
هو عنصر يحتوي داخله عنوان الصفحة، وهو العنوان الذي يظهر في شريط عنوان الصفحة title bar ، أو في تبويب الصفحة page's tab ، ومحتوى العنوان يجب أن يكون نصاً، دون أي عناصر أخرى.

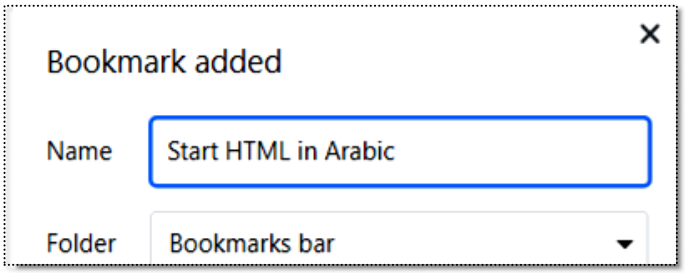
والصفحة لا يكون لها أكثر من عنصر <title> ، وتحفظ الصفحة تحت هذا العنوان عند عمل bookmark للصفحة، أو عند حفظها في الصفحات المفضلة favorite .

العنصر <title> هام لعملية SEO ، حيث تستخدم محركات البحث search engines خوارزميات لترتيب العناوين، للعثور على الصفحات بطريقة أسرع.

تجنب أن يكون العنوان من كلمة أو كلمتين، ويفضل دائما أن يكون وصفياً، وحيث إن محركات البحث تقوم بالبحث في عدد معين من الحروف من 55 إلى 60 تقريباً، فيجب ألا يزيد عدد حروف محتوى العنوان عن هذا العدد، حيث سيتم تجاهل ما يزيد عن هذا العدد من الحروف والكلمات.

```
<head>
        <title>Start HTML in Arabic</title>
</head>
```





التنسيق الافتراضي:

```
title {
        display: none;
        }
```

وهو التنسيق الافتراضي لكل عناصر metadata .

<base>

عنصر empty element يستخدم ليكون الرابط أو المسار الرئيسي، لكل الملفات والصور والامتدادات داخل الصفحة، والذي تتفرع منه باقي الروابط والمسارات الفرعية، ويجب ألا يكون في الصفحة أكثر من رابط رئيسي واحد، ويمكن تعيين خاصيتين لهذا العنصر:

href

خاصية تحدد رابط أو مسار العنوان الرئيسي للصفحة.

target

خاصية تحدد الموضع الذي سيفتح فيه رابط العنوان الرئيسي عند الضغط عليه، والقيم التي تسند إلى هذه الخاصية هي: _blank, _parent, _self, _top, framname ، وشرح مدلول كل قيمة ملحق 1 .

```
<!DOCTYPE html>
<html>
<head>
<base href="https://www.finestwallpaper.com/" target="_blank">
</head>
<body>
<p><a href="">Search for wallpaper</a></p>
<p><a href="store/c35/cool_kids.html">Kidswallpaper</a></p>
<p><img
src="uploads/5/7/7/9/5779447/s774058804322417752_p6571_i70_w1266.pn
g">
</p>
</body>
</html>
```

Search for wallpaper

Kidswallpaper

- لاحظ أن الضغط على الرابط الأول Search for wallpaper يفتح رابط صفحة جديدة، بالرغم من أن قيمة الخاصية href للعنصر <a> فارغة ولا يوجد مسار للرابط المطلوب فتحه، إلا أن ذلك يعني أن قيمة هذه الخاصية تساوي قيمة نفس الخاصية للعنصر <base> ، أي أن رابط الصفحة الجديدة هو نفسه رابط العنصر الرئيسي.

- لاحظ أن الضغط على الرابط الثاني Kidswallpaper يفتح رابط صفحة جديدة، بالرغم من أن قيمة الخاصية href للعنصر <a> في حد ذاتها لا تعتبر مساراً ولا تفتح أي صفحات، ولكن عند فتح الرابط، يتم وضع رابط هذه الصفحة كرابط فرعي بعد الرابط الرئيسي، وهو قيمة الخاصية href للعنصر <base> ، فيكون الرابط كاملاً للصفحة الجديدة.

- لاحظ أن الصورة <img> تظهر في الصفحة، بالرغم من أن قيمة خاصية مصدر الصورة، وهو الخاصية src كانت قيمتها غير مكتملة، لأنها لا تتضمن أي موقع للتحميل، ولكن بعد وضع الرابط الرئيسي قبل رابط الصورة ، يكون رابط الصورة كاملاً.

wailmonir0@protonmail.com

<link>

عنصر empty element يستخدم في ربط الملف أو الصفحة الحالية بمصادر خارجية أخرى، وفي الغالب يستخدم لربط ملف تنسيقات css ، وقد يستخدم لوضع أيقونة للصفحة في شريط العنوان، أو وضعها في تبويب الصفحة، بجانب عدة استخدامات أخرى.

ويمكن تعيين عدة خصائص لهذا العنصر :

href

خاصية تحدد رابط أو مسار الملف الخارجي المطلوب ربطه بالصفحة الحالية.

rel

خاصية تحدد نوع العلاقة بين الصفحة الحالية والملف الخارجي، انظر ملحق 1 .

type

خاصية تحدد نوع الملف الخارجي المرتبط بالصفحة، ومن أمثلة أنواع الملفات المستخدمة مع هذا العنصر : text/css أو image/x-ico أو font/woff2.

```
<link rel="preload" href="mFont.woff2" as="font" type="font/woff2" >
```

sizes

خاصية تحدد حجم الملف الخارجي المرتبط بالصفحة إذا كانت قيمة الخاصية rel هي icon ، حيث يتم إسناد قيمة العرض وبعدها "x" أو "X" ثم قيمة الارتفاع، وقد يسند لهذه الخاصية القيمة any ، وتعني استخدام أي حجم للصورة لاستخدامها كأيقونة.

```
<link rel="icon" sizes="114x114" href="logo.png" as="image"
type="image/png">
```

disabled

خاصية تقوم بتعطيل عمل الملف الخارجي المرتبط بالصفحة.

```
<link rel="stylesheet" href="styles.css" disabled>
```

wailmonir0@protonmail.com

hreflang

خاصية تحدد لغة الملف الخارجي المرتبط بالصفحة الحالية.

title

خاصية تضع عنواناً للملف الخارجي المرتبط بالصفحة.

```
<link rel="stylesheet" href="styles.css"  title="mystyle"  disabled>
```

media

خاصية تحدد نوع البيئة التي سيكون الملف الخارجي فعالاً ومؤثراً فيها، ومن الأمثلة على أنواع هذه البيئات: النوع screen ويعني عرض الصفحة على شاشة عرض، سواء حاسب أو هاتف جوال أو تابلت، والنوع print ويعني طباعة الصفحة، والنوع all ويعني أن الملف يكون نشطاً في كل حالات عرض الصفحة.

```
<link   rel="stylesheet"  href="styles2.css"  media="print, screen">
```

crossorigin

خاصية تحدد كيفية الحصول على الملف الخارجي، وهل يجب إرسال بيانات عن الصفحة مثل cookies أو authentication إلى المصدر أم لا، وشرح الخاصية وقيمها ملحق 1 .

as

تستخدم هذه الخاصية إذا كانت قيمة الخاصية rel هي preload أو prefetch ، حيث تحدد نوع الملف الخارجي، الذي سيتم تحميله مسبقاً للصفحة الحالية.

```
<link   rel="preload"   href="styles.css"   as="style">
```

imagesrcset

إذا كانت قيمة الخاصية rel="preload" وقيمة الخاصية as="image" فإن هذه الخاصية تقوم بالتحميل المسبق، لمجموعة صور أو نسخ لصورة معينة، كل نسخة ذات جودة معينة أو عرض معين، وهذه النسخ سوف تستخدم لاحقاً كقيمة للخاصية srcset في العنصر <img> .

imagesizes

خاصية تقوم بتحديد حجم وبيئة عرض، نسخ الصور المحملة في الخاصية imagesrcset ، والتي سيتم إسنادها لاحقاً للخاصية sizes في العنصر <img> .

```
<head>
<link rel="preload"  as="image"  type="image/jpg" imagesrcset="river-1X.jpg  400w, river-2X.jpg  590w,river-3X.jpg 800w" imagsizes="(max-width:400px) 200px, (max-width:590px) 400px, (max-width: 800px) 600px"
</head>
<body>
<img src="river-1X.jpg"
srcset="river-1X.jpg  400w, river-2X.jpg  590w,river-3X.jpg 800w"
sizes="(max-width:400px) 200px, (max-width:590px) 400px, (max-width: 800px) 600px">
</body>
```

wailmonir0@protonmail.com

<style>...</style>

عنصر مخصص لاحتواء تنسيقات العناصر الأخرى داخل الصفحة، مثل لون النص وتنسيق الخلفية والحدود والهوامش والفلاتر والانتقال والحركة وغيرها.

ويمكن تعيين عدة خصائص لهذا العنصر:

title

خاصية تقوم بإسناد عنوان لهذا العنصر.

type

خاصية تحدد نوع الكود الذي سيتم كتابة العنصر به، وقيمتها هي text/css ، وهي خاصية اختيارية ليس من الضروري أن يتم تعيينها، لأن القيمة الافتراضية دائماً text/css.

media

خاصية تحدد البيئات التي يعمل بها التنسيق داخل العنصر.

```
<head>
<style  title="my-styles" media="screen" type="text/css">
  div {
        width: 170px;
        color : red;
        background-color : yellow;
        border : 3px solid black;
        padding: 3px;
        font-size : 30px;
       }
</style>
</head>
<body>
<div> Style Element </div>
</body>
```

لاحظ أن العنصر <style> قد منح العنصر تنسيقاً عبارة عن:

Style Element

- عرض أو width قدره 170px .
- لون أمامي أو color أحمر red .
- لون خلفية أو background أصفر yellow .
- حدود أو border سمكها 3px ولونها أسود black وخط مصمت solid .
- هامش داخلي أو padding بين محتوى العنصر والحدود قدره 3px .
- حجم خط أو font-size قدره 30px .

wailmonir0@protonmail.com

<meta>

عنصر empty element ، يتحكم في مجموعة بيانات غير مرئية metadata ، تؤثر في طريقة عرض البيانات المرئية، ويكون العنصر <meta> دائماً داخل محتوى العنصر <head> ويختص بمجموعة وظائف منها، تحديد طريقة ترميز النصوص charset ، ووصف الصفحة، والكلمات المفتاحية التي تستخدم في البحث، ومعلومات عن مطور الصفحة، وإعدادات مشاهدة الصفحة، وإعادة تحميل الصفحة، ومعرفة مدى استجابة http headers وغيرها من الوظائف الهامة.

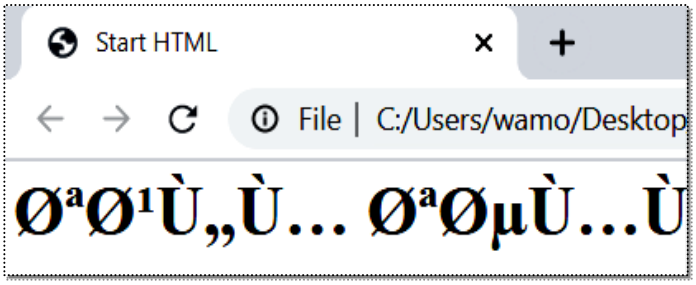
ويمكن تعيين عدة خصائص لهذا العنصر:

charset

خاصية تحدد طريقة ترميز حروف النصوص في الصفحة، والقيمة utf-8 هي القيمة الوحيدة الصحيحة لصفحات html ، وهي غير حساسة لحالة الأحرف case-insenstive فالقيمة utf-8 هي نفسها القيمة UTF-8 ، ويجب أن يتم تعيينها في أول 1024 بايت من الصفحة، ولتوضيح مفهوم هذه القيمة، انظر الخاصية accept-charset ، ص 104 .

```
<head>
<title>Start HTML</title>
<meta charset="utf-8">
</head>
<body>
        تعلم تصميم مواقع الإنترنت
</body>
```

utf-8



ISO-8859-1

content

خاصية تفسر القيم التي يتم إسنادها للخاصية name والخاصية http-equiv .

name

خاصية تحدد اسم البيانات الغير مرئية metadata والتي سيوضع وصف لها في قيمة الخاصية content ، والخاصية name تقبل عدة قيم:

- ***description*** : قيمة تعني وصف للصفحة.

```
<meta name="description" content="Learning Web Design">
```

- ***keywords*** : قيمة تعني كلمات مفتاحية تستخدم في البحث عن الصفحة.

```
<meta name="keywords" content="learn, web, html, css">
```

- ***author*** : قيمة تعني اسم مؤلف موضوع معين في الصفحة أو اسم مطور الصفحة.

```
<meta name="author" content="Wail Monir">
```

- ***generator*** : قيمة تعني البرنامج أو الأداة التي تقوم بإنشاء الصفحة.

```
<meta name="generator" content="FrontPage 4.0">
```

- ***referrer*** : قيمة تعني طريقة إرسال referrer http header أو محتويات رأس صفحة المحيل.

```
<meta name="referrer" content="no-referrer">
```

- ***viewport*** : قيمة تحدد للمتصفح الأبعاد ومقياس العرض للصفحة، وتستخدم هذه الخاصية في حالة الأجهزة المحمولة فقط.

```
<meta name="viewport" content="width=device-width,
initial-scale=1.0">
```

وتقبل الخاصية content عدة قيم تصف القيمة viewport وهي:

- **width** : قيمة تعني تحديد عرض إطار الرؤية، ويسند لها قيمة عبارة عن عدد صحيح موجب، أو القيمة device-width .
- **height** : قيمة تعني تحديد ارتفاع إطار الرؤية، ويسند لها قيمة عبارة عن عدد صحيح موجب، أو القيمة device-height .
- **initial-scale** : قيمة تعني النسبة بين عرض الجهاز في الوضع الأفقي portrait أو ارتفاع

الجهاز في الوضع الرأسي landscape ، وحجم إطار العرض، وتكون النسبة عدد موجب بين 0.0 و 10.0.

- maximum-scale : قيمة تعني أقصى قيمة لتكبير إطار العرض، ويجب أن تكون أكبر من أو تساوي القيمة minimum-scale ، وتكون قيمتها عدداً موجباً بين 0.0 و 10.0 ، ويمكن إعداد المتصفح لتجاهل هذه القيمة.

- minimum-scale : قيمة تعني أقل قيمة لتصغير إطار العرض، ويجب أن تكون أقل من أو تساوي القيمة maximum-scale ، وتكون قيمتها عدداً موجباً بين 0.0 و 10.0 ، ويمكن إعداد المتصفح لتجاهل هذه القيمة.

- user-scalable : قيمة تقبل قيمتين فرعيتين هما:

القيمة yes وتعني أن المستخدم يستطيع تكبير وتصغير الصفحة.

القيمة no وتعني أن المستخدم لا يستطيع تكبير أو تصغير الصفحة.

ويمكن للمتصفحات تجاهل هذه القيمة.

- viewport-fit : قيمة تحدد كيف يتم تهيئة الإطار viewport ، ليلائم جهاز العرض، وهي قيمة تقبل ثلاث قيم فرعية وهي:

auto : قيمة فرعية لا تؤثر في الشكل الأساسي للإطار، والصفحة سيتم عرضها بالكامل على شاشة عرض الجهاز.

contain : قيمة فرعية تعني أن الإطار سيتم تحجيمه داخل الجهاز، ليتم إحتواء الإطار داخل شاشة الجهاز.

cover : قيمة فرعية تعني أن الإطار سوف يتم تحجيمه بحيث يملأ شاشة الجهاز مع المحافظة على النسبة بين عرضه وارتفاعه، وقص ما يزيد من العرض أو الارتفاع.

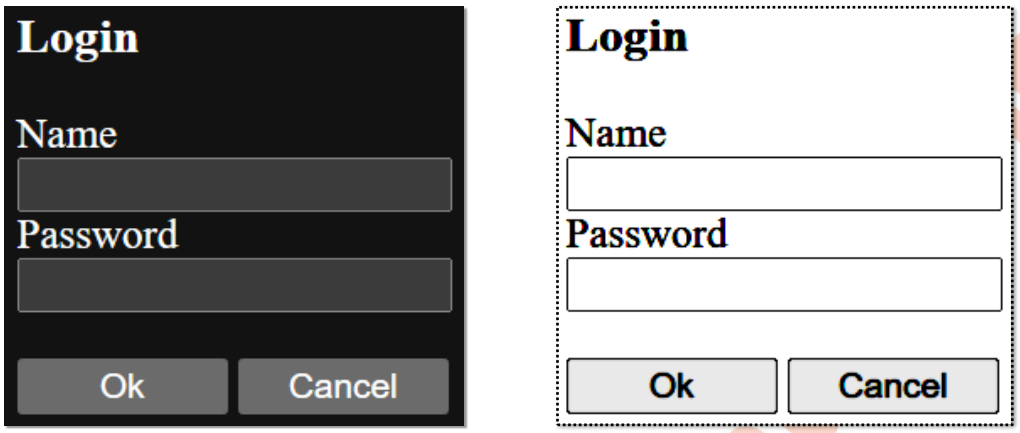
- *color-scheme* : قيمة تعني النمط اللوني الذي ستعرض فيه الصفحة، والذي يؤثر على خلفية الصفحة ولون النص وغيرها، وتقبل الخاصية content عدة قيم:

- *normal* : لا تستطيع الصفحة تحديد النمط اللوني، وتعرض الصفحة باستخدام الألوان الافتراضية.

- [*dark|light*] : إما النمط اللوني الداكن أو النمط اللوني الساطع أو الاثنين معاً.

- *only light* : استخدام النمط اللوني الساطع فقط.

```
<meta name="color-schem" content="dark">
```



dark light

والقيمة color-scheme كانت مدعومة في متصفح chrome فقط وأصبحت الآن مدعومة في متصفح Firefox أيضاً.

http-equiv

خاصية كل قيمها عبارة عن أسماء http headers ، فتقوم بتحديد المعلومات والقيم الموجودة التي سيتم تفسيرها في الخاصية content ، كما يمكن استخدامها لمحاكاة استجابة http header .

وهذه الخاصية تقبل عدة أنواع من القيم:

- content-type : قيمة تحدد نوع الصفحة الحالية، ونوع الترميز للنصوص.

وتقبل الخاصية content قيمتها على هيئة نوع الصفحة وهو text/html يليه ";" ثم خاصية نوع الترميز charset ، ويسند لها نوع الترميز وهو utf-8 كالتالي:

```
<meta http-equiv="content-type" content="text/html;charset=utf-8">
```

- refresh : قيمة تجعل الصفحة تقوم بعمل refresh تلقائياً، وتحدد الفترة بين refresh كقيمة للخاصية content ، فهي تساوي - مثلاً - 30 ثانية، كما في الكود التالي:

```
<meta http-equiv="refresh" content="30">
```

ويمكن أن يتم توجيه الصفحة لصفحة أخرى بعد عملية refresh ، عن طريق إضافة url للصفحة المطلوب توجيه الصفحة الحالية إليها، بعد عملية refresh ، على أن يفصل بين الفترة الانتقالية وعنوان الصفحة الجديدة ";".

```
<meta http-equiv="refresh" content="30; url=https://google.com">
```

لاحظ أنه سيتم توجيه الصفحة الحالية إلى موقع جوجل بعد 30 ثانية.

- ***default-style*** : قيمة يقصد منها:

إما تحديد العنصر <link> الذي يربط الصفحة بملف التنسيق الخارجي، إذا كانت التنسيقات external ، أو تحديد العنصر <style> الذي يحتوي تنسيقات الصفحة داخله، إذا كانت التنسيقات internal أي داخل نفس صفحة html .

وبناء على ذلك فقد تكون قيمة الخاصية content إحدى القيمتين التاليتين:

- قيمة الخاصية title للعنصر <link> الذي يقوم بربط ملف التنسيق الخارجي بالصفحة الحالية.

- قيمة الخاصية title للعنصر <style> الذي يحتوي تنسيقات الصفحة داخله.

```
<head>
<link rel="stylesheet" href="styles.css">
<meta   http-equiv="default-style"  content="my-style">
<style title="my-style">
div {
      width: max-content;
      padding: 5px;
      font-size: 2em;
      font-weight: bold;
      color: darkred;
      background: pink;
      }
</style>
</head>
<body>
<div>Default Style</div>
</body>
```

Default Style

لاحظ ظهور العنصر بالتنسيقات الموجودة داخل العنصر <style>، ولاحظ أن قيمة الخاصية title لهذا العنصر تساوي قيمة الخاصية content للعنصر <meta> .

قم بتغيير قيمة الخاصية title للعنصر <style> واترك قيمة الخاصية content دون تغيير، ولاحظ اختفاء تنسيقات العنصر وظهوره بتنسيقاته الافتراضية.

Default Style

- X-UA-compatible : قيمة تساعد متصفح IE (internet explorer) قبل النسخة 11 على عرض صفحات الويب الحديثه، بنوع من التوافق مع المواصفات الحديثة، التي لا تدعمها هذه المتصفحات القديمة، وذلك بجعلها متوافقة مع أحد المتصفحات الحديثه، وعند اختيار القيمة ie=edge فهذا يعني أن المتصفحات القديمة، ستكون متوافقة بأكبر قدر ممكن مع مواصفات متصفح edge .

```
<meta   http-equiv="X-UA-comaptible"    content="IE-edge">
```

- content-security-policy : سياسة أمن المحتوى لـ http header تجعل مديري الصفحات يتحكمون في المصادر التي يتم تحميلها إلى الصفحات، حيث تقوم بتحديد نوع الخادم أو server الذي سيتم التحميل منه، وكذلك نقطة نهاية الكود البرمجي، لمنع ما يعرف بهجمات الاختراق المسماة XSS أو cross-site scripting .

وهذه القيمة (الخاصية) تقبل عدة قيم فرعية كقيمة للخاصية content :

- child-src : قيمة تحدد المصادر الصالحة، لأي برامج أو أكواد تعمل في الخلفية، أو أي طلبات للتحميل من داخل الإطارات <iframe> .
- connect-src : قيمة تعني تقييد عناوين url التي يمكن تحميلها بواسطة الأكواد البرمجية (scripts).
- default-src : قيمة تعمل كقيمة احتياطية لكل القيم الأخرى.
- font-src : قيمة تعني توفير مصادر صالحة لتحميل أنواع الخطوط.
- frame-src : قيمة تعني توفير مصادر صالحة للتحميل، عند طلبها من داخل الإطار أو عنصر <iframe> .
- img-src : قيمة تعني توفير مصادر صالحة لتحميل الصور والأيقونات.
- manifest-src : قيمة تعني توفير مصادر صالحة لتحميل ملفات التطبيقات الأساسية.
- media-src : قيمة تعني توفير مصادر صالحة للتحميل بواسطة العناصر <audio> و<video> و <track> .

- *object-src* : قيمة تعني توفير مصادر صالحة للتحميل، بواسطة العناصر <object> و<embed> .
- *script-src* : قيمة تعني توفير مصادر صالحة لأكواد الجافاسكريبت.
- *style-src* : قيمة تعني توفير مصادر صالحة لتنسيقات css .

وقيمة الخاصية content تكون إحدى هذه القيم الفرعية السابقة، ثم يتبعها كلمة مفتاحية تحدد مصدر التحميل، وهذه الكلمة المفتاحية لها أنواع:

- *none* : قيمة تعني عدم السماح بتحميل أي مصادر.
- *self* : قيمة تعني السماح بتحميل المصادر من العنوان الرئيسي "origin" للصفحة.
- *https* : قيمة تعني السماح بتحميل المصادر من المواقع الآمنة فقط.
- *http* : قيمة تعني السماح بتحميل المصادر من المواقع الغير آمنة.
- *http://example.com* : قيمة تعني السماح بتحميل المصادر من موقع معين.

```
<meta  http-equiv="content-security-policy"
content="default-src  https  'self'; font-src  http://fonts.com">
```

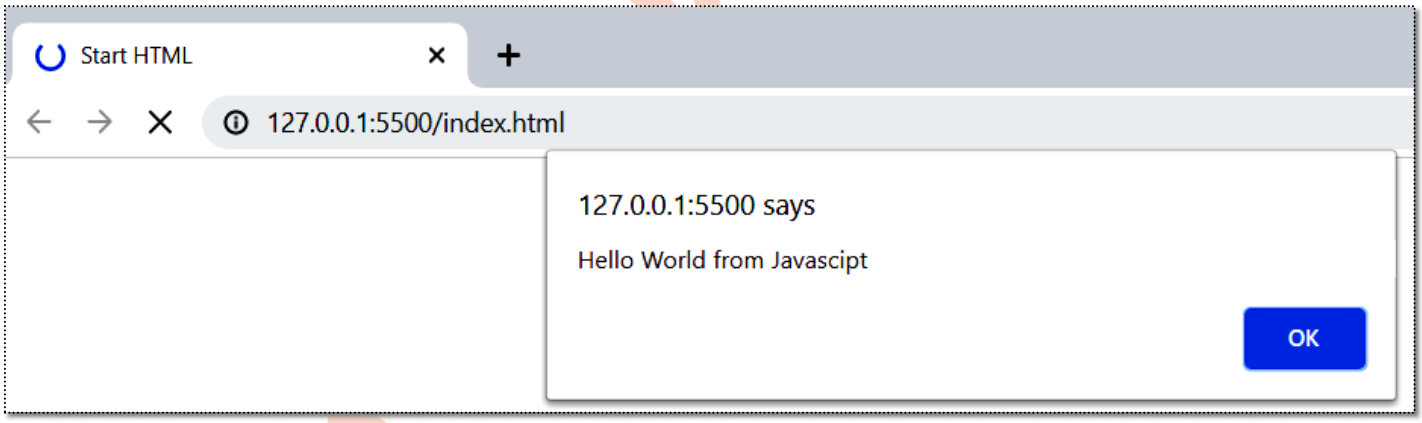
وهذا شرح بشكل مبسط، حتى يستوعب كل مبتدئ سلوك وتأثير هذه القيمة أو الخاصية، لأنها تحتوي على كثير من التفاصيل التي لا يتسع المجال لذكرها.

wailmonir0@protonmail.com

<script>...</script>

عنصر مخصص لاحتواء الأكواد البرمجية للجافاسكريبت Javascript أو JSON ، والجافاسكريبت هي لغة برمجة تختلف عن باقي لغات البرمجة، التي تحتاج إلى ترجمة أكوادها باستخدام مترجم (compiler) خاص، إلا أن جافاسكريبت تتم ترجمة أكوادها من خلال المتصفح على جهاز المستخدم مباشرة، وهو ما يسمى client side scripting ، والجافاسكريبت تستخدم لإضفاء نوع من الديناميكية على الصفحة، أو التفاعل مع المستخدم والتجاوب معه، فيمكنها إنشاء أو حذف عناصر html ، وتعيين خصائص جديدة للعنصر أو إسناد قيم جديدة للخصائص، يمكنها التلاعب بالعناصر والتحكم فيها، مثل إخفاء عنصر أو إظهاره أو تحريكه ورسم الأشكال عن طريق العنصر canvas ، وغيرها الكثير والكثير من الوظائف، التي تضفي على الموقع روح الحركة والتفاعل.

```
<head>
<title>Start HTML</title>
<script> alert ("Hello World from Javascipt")</script>
</head>
```



لاحظ أن مخرجات كود العنصر <script> أدت لظهور رسالة، يجب التعامل معها أولاً، قبل تصفح الموقع، ويمكن تعيين عدة خصائص لهذا العنصر:

src

خاصية تحدد مسار الملف الخارجي الذي يحتوي الكود البرمجي، في حالة عدم كتابة الكود بين وسمي البداية والنهاية للعنصر<head> أو عدم كتابته في جسم الصفحة، وإنما تمت كتابته في ملف خارجي (وهو الاستخدام الأكثر شيوعاً).

type

خاصية تحدد نوع الملف الخارجي أو نوع الكود البرمجي لهذا العنصر، وأنواع الملفات الخارجية هي application/javascript أو application/ecmascript ويوجد نوع آخر هو module (وهو ملف يحتوي أكواد برمجية معدة مسبقاً يتم استيراده داخل أي ملف)، وأي نوع آخر من أنواع الملفات، سيسبب تجاهل لقيمة الخاصية src .

async

خاصية boolean attribute تعني أن النص البرمجي سيتم تحميله بشكل غير متزامن، مع تحليل الصفحة (أي في نفس الوقت)، وتحليل الصفحة هو قراءة كود html وبناء هيكل العناصر dom وإظهار مخطط الصفحة basic layout والنصوص text ، ويتم تنفيذ الكود فور جلبه (بعد اكتمال تحميله)، ويتوقف تحليل كود الصفحة خلال عملية تنفيذ الكود، وهذه الخاصية تطبق على الكود إذا كان في ملف خارجي فقط، أي أن العنصر <script> تم تعيين الخاصية src له.

defer

خاصية boolean attribute مشابهة للخاصية السابقة، والفرق بينهما أن الكود لن يتم تنفيذه فور جلبه، ولكن سيتم تأجيل التنفيذ، حتى الانتهاء من تحليل الصفحة.

في حالة عدم تعيين أي من الخاصيتين السابقتين، سيتم تحميل الكود وتنفيذه مباشرة بطريقة متزامنة، أي سيتم تحميل وتنفيذ الكود فوراً، ويتوقف تحليل الصفحة لحين الانتهاء من تنفيذ الكود ، ويتم استئنافه بعد التنفيذ.

integrity

خاصية تسمح للمتصفح بالتحقق، من أن ملف الكود الذي تم جلبه، لم يتم التلاعب به من جانب طرف ثالث، وأنه هو نفسه الملف المرسل من السيرفر، عن طريق شفرة معينة لهذا الملف، حيث يحتفظ السيرفر بالملف وتحتفظ الصفحة التي تقوم بجلب الملف بهذه الشفرة، و بعد تحميل الملف، يقوم المتصفح بمقارنة الشفرة الموجودة بالصفحة (قيمة الخاصية integrity) وبين شفرة الملف الذي تم تحميله، فإذا تطابقا يتم استخدام الملف، وإن

اختلفا يتم حجب الملف.

nomodule

خاصية boolean attribute تعني أن الأكواد البرمجية، لن يتم تنفيذها في المتصفحات التي تدعم نسخة الجافاسكريبت es6 لأنها تدعم modules ، ويجب تجربة الأكواد على متصفحات أقدم لا تدعم modules .

nonce

خاصية تسمح للخادم بتحديد رقم فريد عشوائي، من النوع base64-encoded ليكوّن سلسلة نصية، يصل حجمها على الأقل إلى 128 bits ، ويستخدم هذا الرقم لمرة واحدة فقط، لأي مصدر خارجي يمكن جلب الكود منه بأمان، لتكوين قائمة بيضاء للمصادر الآمنة، حيث يستخدم هذا الرقم في سياسة أمن المحتوى Content Security Policy .

crossorigin

خاصية تحدد كيفية الحصول على ملف الكود، وهل يجب إرسال معلومات عن الصفحة الحالية للصفحة التي تحتوي الكود أم لا، وشرح الخاصية وقيمها ملحق 1 .

referrerpolicy

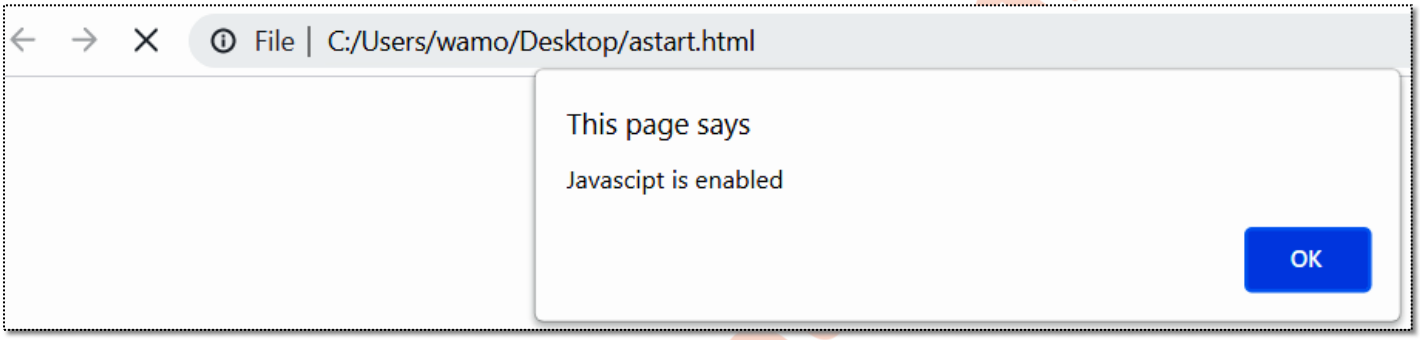
خاصية تحدد نوع المعلومات التي سيتم إرسالها في http header ، من صفحة المحيل referrer إلى صفحة destination ، وللتعرف على الخاصية وقيمها انظر ملحق 1 .

wailmonir0@protonmail.com

<noscript>..</noscript>

عنصر لا يظهر محتواه إلا في حالة عدم تفعيل أكواد جافاسكريبت في الصفحة.

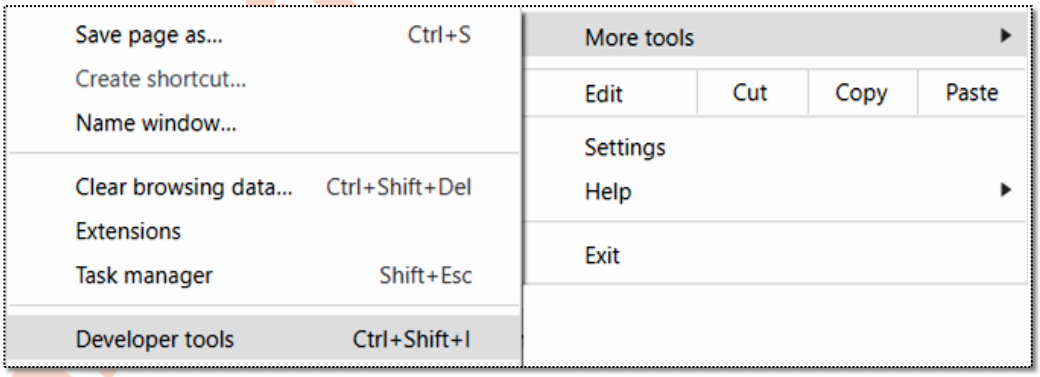
```
<head>
<title>Start HTML</title>
<script> alert ("Javascipt is enabled")</script>
<noscript> Javascript is disable </noscript>
</head>
```



لاحظ أن كود الجافاسكريبت داخل العنصر <script> تم تنفيذه، ولم يظهر النص داخل محتوى العنصر <noscript> ، لأن أكواد الجافاسكريبت مفعلة في هذه الصفحة.

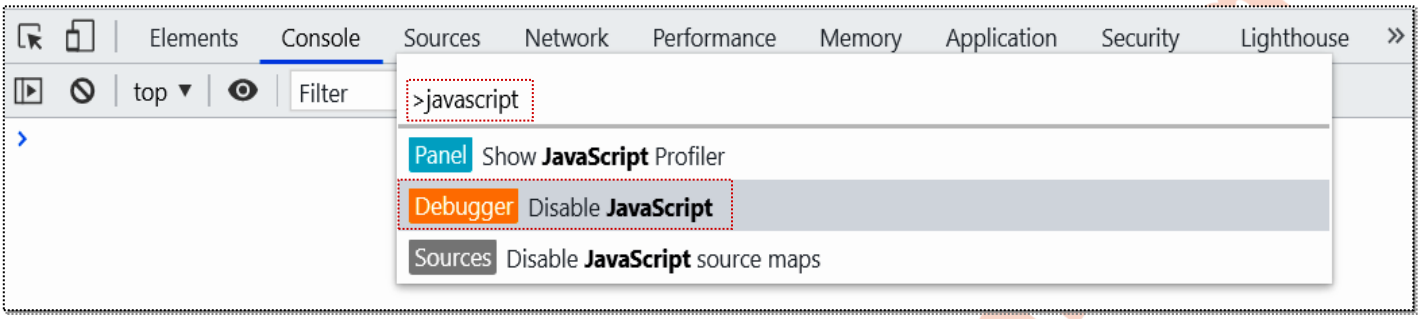
لنقم الآن بتعطيل أكواد الجافاسكريبت في متصفح Chrome (على سبيل المثال):

- قم بفتح Develeper tools إما بالضغط على مفتاح F12 ، أو الضغط على مفاتيح Ctl+Shift+i ، أو من القائمة الجانبية، كما بالشكل التالي:



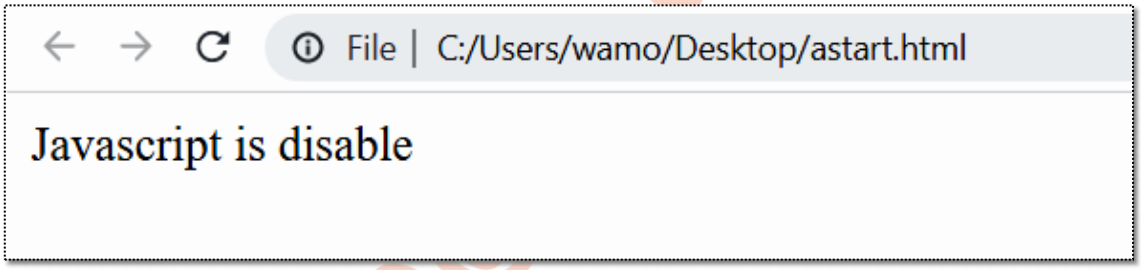
لتفتح نافذة Developer tools بالشكل التالي:

- قم بفتح Command menu ، بالضغط على المفاتيح Ctl+Shift+p واكتب في مربع البحث javascript كما بالشكل التالي:



- ستفتح لك قائمة منسدلة، قم بالضغط على الاختيار Disable Javascript .
- انتهت عملية تعطيل أكواد الجافاسكريبت وستغلق القائمة المنسدلة.

قم الآن بعمل refresh للصفحة أو التبويب الذي قمنا بتعطيل أكواد Javascript داخله :



لاحظ أن الكود داخل العنصر <script> لم يتم تنفيذه، وظهر بدلاً منه النص الموجود داخل محتوى العنصر <noscript> ، لتنبيه المستخدم أن أكواد الجافاسكريبت معطلة في هذه الصفحة أو هذا التبويب.

الخصائص العامة

Global Attributes

wailmonir0@protonmail.com

wailmonir0@protonmail.com

الخصائص العامة Global Attributes

هي خصائص مشتركة بين جميع عناصر html وليست قاصرة على عناصر معينة.

accesskey

خاصية تحدد مفتاح واحد على لوحة المفاتيح، قد يكون حرفاً أو رمزاً أو رقماً، عند الضغط عليه مع مفتاح Alt في متصفح Chrome ، أو مع مفتاحي Alt+Shift في متصفح Firefox ، يتم تنشيط وتفعيل العنصر، وكأنه قد تم الضغط عليه.

```
<button accesskey="b" onclick="style='background:red'"> Press Me
</button><br><br>
<a href="https://cdn.pixabay.com/photo/2018/03/30/15/11/deer-
3275594_960_720.jpg" accesskey="r" target="_blank">Go to photo</a>
```

قم بالضغط على المفتاحين b+Alt في متصفح Chrome ، أو المفاتيح b+Alt+Shift في متصفح Firefox ، ولاحظ تغيير خلفية الزر إلى اللون الأحمر، حسب الكود المطلوب تنفيذه في الخاصية onclick ، ثم قم بالضغط على الأزرار r+Alt أو r+Alt+Shift (حسب المتصفح) ولاحظ أن الرابط Go to photo قد قام بفتح رابط الصورة الموجودة في العنصر <a> وقد تغير لون الرابط، كدليل على أنه تم الضغط عليه وفتحه .

بعض النقاط التي تجعلنا نتوخى الحذر عند استخدام هذه الخاصية:

- قد يتعارض اختصار معين لهذه الخاصية، مع اختصارات نظام التشغيل، أو مع اختصارات المتصفح، أو مع اختصارات البرامج المساعدة.
- استخدام الأرقام في هذه الخاصية، قد يسبب ارتباكاً للمستخدم، خاصة إذا كان المفتاح المستخدم هو رقم ليس له صلة بالوظيفة التي يقوم بها.
- عدم إعلام المستخدم بتعيين هذه الخاصية لأحد العناصر، قد يتسبب في تنشيط هذا العنصر بطريق الخطأ.

wailmonir0@protonmail.com

class

خاصية يتم تعيينها للعنصر، ليتم اختيار العنصر عن طريق قيمتها لتنسيقه بـ css ، أو استدعاؤه واستخدامه برمجياً في javascript أو غيرها من لغات البرمجة.

قيمة هذه الخاصية قد تكون حرف واحد أو أكثر، ويمكن أن تبدأ بحرف أو مسافة فارغة أو "_" أو "-" ولا تبدأ برقم أو رمز، ولا يتخللها فاصلة، أو رمز ، أو مسافة فارغة (المسافة الفارغة ستعتبر ما قبلها قيمة وما بعدها قيمة أخرى) وقد يتخللها أي أرقام أو "_" أو "-" .

وعليه، فهذه القيم غير صحيحة: @cls و c ls و .cls و 9cls و cl&s .

وهذه القيم صحيحة : cls و _cls و -cls و c9ls و cl_s و cl-s و صف1 .

```
<head>
<style>
.1صف{
        width: max-content;
        background: lightblue;
        padding: 5px;
        color: blue;
        font-size: 3em;
        font-weight: bold;
       }
</style>
</head>
<body>
<div class="1صف">Using Class</div>
</body>
```

Using Class

وقد يكون للعنصر أكثر من قيمة داخل قيمة الخاصية class ، بينها مسافة واحدة فارغة على الأقل، تمثل كل منها العنصر، ويستخدم كل منها بشكل منفرد ليعبر عن هذا العنصر، وقد يشترك أكثر من عنصر في قيمة الخاصية class .

```
<div class="dv    dv_cls">First Class</div>
<div class="cls   dv_cls">Second Class</div>
```

contenteditable

خاصية تسمح بالتعديل على محتوى العنصر داخل المتصفح وتقبل قيمتين:

- *true* : قيمة تعني أن العنصر يقبل التعديل.
- *false* : قيمة تعني أن العنصر لا يقبل التعديل.

```
<div contenteditable="true">Click to edit it in the browser</div>
```

Click to edit it in the browser

قم بالنقر على العنصر، ولاحظ ظهور مؤشر الكتابة، الذي يسمح لك بالتعديل على محتوى العنصر، سواء بالحذف أو الإضافة.

يمكن تعيين هذه الخاصية بدون إضافة أي قيمة لها وستعامل على أنها true .

```
<div contenteditable>Click to edit it in the browser</div>
```

إذا لم يتم تعيين الخاصية للعنصر ، أو تم تعيينها بطريقة خاطئة وأصبحت غير مؤثرة على العنصر، فإن العنصر يرث قيمتها من العنصر الأب، سواء كانت true أو false .

hidden

خاصية تقوم بإخفاء محتوى العنصر الذي يتم تعيينها له، لأنه قد يكون حقلاً للتعديل في قاعدة البيانات، أو رمز مرور أو معلومة مهمة للصفحة، لا يجب أن يشاهدها المستخدم، وهي خاصية boolean attribute ، أي لا يسند إليها أي قيمة، ومجرد تعيينها يعني أن قيمتها true ، وعدم تعيينها يعني أن قيمتها false .

```
<div>This element will be shown in the browser</div>
<div hidden>This element won't be shown in the browser</div>
```

This element will be shown in the browser

لاحظ أن العنصر الأول ظهر في المتصفح، بينما لم يظهر العنصر الثاني نتيجة لتعيين الخاصية hidden له.

data-*

خاصية تقوم بإنشاء خاصية جديدة غير معرفة في html ، حيث يقوم المطور بتعريفها واختيار اسم لها على أن يبدأ بالمقطع data- ، ويقوم بمنحها القيمة التي يرغبها، ثم يقوم باستخدامها في التنسيق، أو الأكواد البرمجية مثل أي خاصية معرفة في html .

```
<head><style>
        div[data-name]{
            background: lightgreen;
            font-weight: bold;
            padding: 5px;
            width: max-content;
            margin-bottom: 10px;
            }
          </style>
</head>
<body>
<div data-name>A div with data attribute</div>
<div>A div without data attribute</div>
</body>
```

A div with data attribute

A div without data attribute

لاحظ أنه تم تنسيق العنصر الذي تم تعيين الخاصية data-name له، وتم استخدامها لتمييز هذا العنصر عن أي عنصر آخر.

الاسم بعد المقطع data- لا يجب أن يحتوي أي حروف كبيرة، وإذا احتوى حروفاً كبيرة بطريق الخطأ، فإن المتصفح سوف يحولها لحروف صغيرة، فلا تندهش إن استخدمت حروفاً كبيرة، ووجدت أن الكود يعمل بشكل جيد، ويمكن للاسم أن يبدأ بحرف صغير أو رقم، ويمكن أن يتخلله أي حروف صغيرة أو أرقام، ولا يمكن أن يبدأ بمسافة فارغة أو رمز ما عدا "-" و"_" ، ولا يتخلله أي مسافات فارغة ولا أي رموز ما عدا "-" و "_".

dir

خاصية تحدد اتجاه النص، أو اتجاه محتوى عنصر معين، ويسند لها عدة قيم:

- ***rtl*** : قيمة تعني أن اتجاه المحتوى داخل العنصر يبدأ من اليمين إلى اليسار.
- ***ltr*** : قيمة تعني أن اتجاه المحتوى داخل العنصر يبدأ من اليسار إلى اليمين.
- ***auto*** : قيمة تعني أن المتصفح يحدد اتجاه المحتوى، حيث يقوم بتحليل الحروف حتى يجد حرفاً في النص له اتجاه معين، فيكون هذا هو اتجاه النص أو المحتوى.

```
<div style="width:400px" dir="rtl">Right To Left Direction</div>
<div style="width:400px" dir="ltr">Left To Right Direction</div>
<div style="width:400px" dir="auto">هذا النص باللغة العربية</div>
<div style="width:400px" dir="auto">This is English Text</div>
```

Right To Left Direction

Left To Right Direction

هذا النص باللغة العربية

This is English Text

لاحظ أن اتجاه محتوى العنصر الأول من اليمين إلى اليسار لأن قيمة الخاصية rtl .

لاحظ أن اتجاه محتوى العنصر الثاني من اليسار إلى اليمين لأن قيمة الخاصية ltr .

لاحظ أن اتجاه محتوى العنصر الثالث من اليمين إلى اليسار، بالرغم من أن قيمة الخاصية dir هي auto ، حيث أدرك المتصفح أن اتجاه حروف النص من اليمين إلى اليسار (اللغة العربية).

وكذلك اتجاه محتوى العنصر الرابع من اليسار إلى اليمين، بالرغم من أن قيمة الخاصية هي auto ، حيث أدرك المتصفح أن اتجاه حروف النص من اليسار إلى اليمين.

تأكد من تعيين الخاصية charset للعنصر <meta> ، واسند لها القيمة utf-8 حتى تظهر حروف اللغة العربية بالشكل الصحيح في متصفحك.

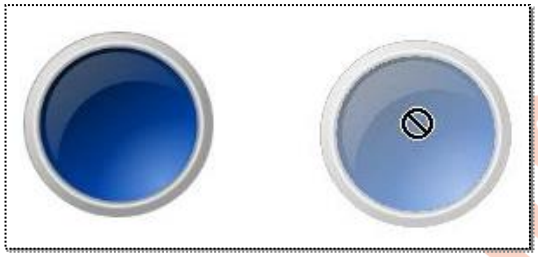
draggable

خاصية تسمح بسحب وإفلات عنصر معين، ويسند لها عدة قيم:

- ***true*** : قيمة تعني إمكانية سحب وإفلات العنصر.
- ***fasle*** : قيمة تعني عدم إمكانية سحب وإفلات العنصر.
- ***auto*** : قيمة تعني أن المتصفح يتبع القيمة الافتراضية للعنصر، فمثلاً العنصر **<img>** قابل للسحب والإفلات تلقائياً، بينما العنصر **<p>** أو العنصر ** ** عناصر غير قابلة للسحب والإفلات افتراضياً.

```
<div  draggable="true">Drag Me</div><br>
<img  src="button.png" width="50" height="50">
```

لاحظ أن العنصر ** ** أصبح قابلاً للسحب والإفلات بعد إسناد القيمة **true** للخاصية **draggable** .



لاحظ أن العنصر **<img>** قابل للسحب والإفلات بدون تعيين الخاصية **darggable** ، حيث إنه قابل للسحب والإفلات افتراضياً.

رغم أن الخاصية **draggable** يمكن أن تعمل بشكل صحيح بدون إضافة القيمة **true** ، إلا أن استخدام القيمة **true** يعتبر ضرورياً، لأن استخدام اسم الخاصية دون إسناد القيمة **true** وإن كان يعمل بشكل صحيح، إلا أنه يوحي بأن عدم تعيين الخاصية سيجعل قيمتها **false** ، وهذا غير صحيح في بعض العناصر، لأنه كما ذكرنا من قبل، يوجد عناصر قابلة للسحب والإفلات افتراضياً، ولا تحتاج لتعيين هذه الخاصية، وعليه فإن الإيحاء بأن عدم تعيينها سيجعل قيمتها **false** لكل العناصر، هو إيحاء خاطئ.

id

خاصية تمنح العنصر معرفاً فريداً ، يجب ألا يتكرر مع أي عنصر آخر في نفس الصفحة، ويستخدم في تنسيق العنصر، أو في الأكواد البرمجية، أو في التنقل بين أجزاء الصفحة fragments ، والخاصية id لا تقبل إلا قيمة واحدة فقط، وفي حالة إسناد أكثر من قيمة بينها مسافة فارغة (مثل الخاصية class) ستعتبر القيمة غير صحيحة.

وتكون قيمة الخاصية نصاً، ولا تبدأ برقم أو رمز أو مسافة فارغة، ورغم أنه يمكن أن تبدأ بحرف أو "_" أو "-" ، إلا أنه يفضل أن تبدأ دائماً بحرف، وقد يتخللها أي عدد من الأرقام أو "_" أو " -" ولا يتخللها أي رموز أو أي مسافات فارغة.

وعليه فإن هذه القيم صحيحة: pid , p2id , _pid , -pid , p_id , p-id .

وهذه القيم غير صحيحة: .pid , @pid , p$id , p id , 3p.id .

```
<head>
    <style>
      #a1{
          background: orange ;
          width: max-content;
          font-weight: bold;
          padding: 5px;
          }
    </style>
</head>
<body>
<p  id="a1">This paraghrap is styled by using id</p>
</body>
```

This paraghrap is styled by using id

لاحظ أن العنصر <p> قد تم تنسيقه بال css ، حيث تم ربط التنسيق داخل العنصر <style> بالعنصر <p> باستخدام الخاصية id التي تم تعيينها للعنصر <p> .

lang

خاصية يسند لها قيمة عبارة عن اللغة المستخدمة في محتوى العنصر، الغير قابل للتعديل، أو في قيمة العنصر القابلة للتعديل، والقيمة الافتراضية لهذه الخاصية تكون غير معرفة، لذلك يجب عليك دائماً أن تعين هذه الخاصية، وتسند لها قيمة تعبر عن لغة الصفحة، وتكون قيمة الخاصية في الغالب، عبارة عن حرفين يعبران عن هذه اللغة مثل: en , fr , ar .

وقد تلي هذه الحروف بعض الرموز ، أو اللاحقات لتدل على المنطقة التي تأتي منها هذه اللغة، وقد تتكون هذه اللاحقات من حرفين كبيرين مثل: en-GB ، وتعني الإنجليزية في بريطانيا، و es-ES وتعني الأسبانية في أسبانيا، وقد تتكون هذه اللاحقات من ثلاثة أرقام تدل على هذه المنطقة، مثل es-013 وتعني الأسبانية في أمريكا الوسطي.

```
<p lang="ar">هذا العنصر يستخدم اللغة العربية</p>
<p lang="en">This paragraph uses English Language</p>
<p lang="fr">Ce paragraphe est défini en français.</p>
```

هذا العنصر يستخدم اللغة العربية

This paragraph uses English Language

Ce paragraphe est défini en français.

style

خاصية تقوم بتنسيق العنصر بتنسيقات css ، داخل وسم الافتتاح للعنصر opening tag وهو ما يعرف بال inline style ، وذلك بدلاً من استخدام العنصر <style> داخل العنصر <head> ، أو استخدام ملف css يضم كل التنسيقات وربطه بملف html عن طريق العنصر <link> داخل العنصر <head> .

يعتبر تنسيق العنصر باستخدام الخاصية style أقوى من النوعين الآخرين , internal external ، حيث إنه أسرع لوجوده في وسم بداية العنصر، بالإضافة إلى أنه شديد التخصص حيث يتم تعيينه للعنصر المراد تنسيقه مباشرة.

```
<head>
<style>
   div{
        width: 215px;
        background: darkred;
        color: pink;
        padding: 5px;
        font-weight: normal;
        }
</style>
<body>
<div style="width: 215px; background: blue; color: white;
      padding: 5px;font-weight: bold;">
      This div is styled by inline style
</div>
</body>
```

This div is styled by inline style

لاحظ أن العنصر تم تنسيقه بتنسيقات الخاصية style في وسم الافتتاح، وتم تجاهل تنسيقات العنصر<style> الموجود داخل العنصر <head> .

title

خاصية تقدم معلومات إضافية عن العنصر، تكون على هيئة تلميح tooltip ، يظهر عند المرور بالفأرة فوق العنصر، مثل العنصر <abbr> ، والعنصر <dfn> ، كما تستخدم قيمة هذه الخاصية مع العنصر <meta> ، لتحديد التنسيق الافتراضي للصفحة، انظر ص 215 .

وإذا لم يتم تعيين الخاصية title للعنصر، فإنه سيرث قيمة الخاصية من العنصر الأب، الذي تم تعين الخاصية له كالتالي:

```
<abbr title="North Atalntic Treaty Organization"> NATO
<div>Political Organization</div>
</abbr>
```

لاحظ أن مرور الفأرة على العنصر الابن أدى لظهور تلميح الخاصية title ، بالرغم من عدم تعيين الخاصية له، وإنما تم تعيينها للعنصر الأب وورثها العنصر الابن.

ولكن إذا لم نرد للعنصر الابن أن يرث العنصر الأب، ولا يظهر التلميح عند مرور الفأرة فوقه، فنقوم بتعيين الخاصية title له، ونسند لها قيمة عبارة عن سلسلة نصية فارغة:

```
<abbr title="North Atalntic Treaty Organization">NATO
<div title="">Political Organization</div>
</abbr>
```

وعموماً، فإن أي عنصر يتم تعيين الخاصية title له، سوف يظهر تلميح tooltip ، يعرض قيمة الخاصية، عند مرور الفأرة فوقه.

```
<img src="mount.jpg" width="200" title="Amazing Nature">
```

tabindex

خاصية تختص بنقل تركيز لوحة المفاتيح focus بين العناصر، حيث تكون قيمتها عبارة عن رقم معين، يمثل ترتيب هذا العنصر في دورة انتقال التركيز بين عناصر الصفحة، ويكون الانتقال بالترتيب بين العناصر بواسطة الضغط على زر tab .

وتسند لهذه الخاصية عدة أنواع من القيم:

- قيمة موجبة : تعني ترتيب معين لانتقال التركيز إلى العنصر بين عناصر الصفحة.
- قيمة سالبة : تعني أن العنصر ليس له ترتيب لانتقال التركيز إليه، وأنه خارج ترتيب العناصر ولن ينتقل إليه التركيز أبداً.
- القيمة 0 : تعني أن ترتيب انتقال التركيز إلى هذا العنصر، يكون بعد كل العناصر ذات الترتيب الموجب.

```
<head>
<style>
   div{
        display: inline-block;
        width: max-content;
        background: darkorange;
        color: navy;
        padding: 5px;
        font-weight:  normal;
        }
</style>
</head>
<body>
      <div tabindex="3">Tab order 3</div>
      <div tabindex="-1">Tab order -1</div>
      <div tabindex="2">Tab order 2</div>
      <div tabindex="0">Tab order 0</div>
      <div tabindex="1">Tab order 1</div>
</body>
```

Tab order 3 | Tab order -1 | Tab order 2 | Tab order 0 | Tab order 1

قم بالضغط المتكرر على الزر Tab ولاحظ أن:

العنصر Tab order -1 لم ينتقل إليه التركيز focus مطلقاً.

العنصر Tab order 0 انتقل إليه التركيز بعد العنصر Tab order 3 وهو آخر ترتيب موجب بين العناصر.

لا ينتقل التركيز من العنصر الأب إلى العنصر الابن، طالما لم نقم بتحديد ترتيب له في دورة tabindex، بمعنى أن التركيز لا يتم وراثته، والعنصر الأ ب والعنصر الابن يعاملان كعنصرين منفصلين.

يفضل دائماً أن تضع العناصر في ترتيبها المنطقي والعملي، بدلاً من أن تلجأ إلى تعديل قيمة الخاصة tabindex ، لأن برامج قراءة الشاشة قد تواجه صعوبة في قراءة ترتيب العناصر المعدل.

اعلم أن أعلى قيمة موجبة يمكن أن يصل إليها ترتيب انتقال التركيز كقيمة للخاصية tabindex هي 32767 .

إذا تساوى عنصران في قيمة الخاصية tabindex ، فإن العنصر الذي سينتقل إليه التركيز سيكون حسب اتجاه العنصر الأب، الذي يحتوي هذه العناصر، فإذا كان اتجاهه من اليسار إلى اليمين ltr ، فإن التركيز سينتقل بين العناصر الداخلية من اليسار إلى اليمين، وإذا كان اتجاه العنصر الأب من اليمين إلى اليسار rtl ، فإن التركيز سينتقل بين العناصر الداخلية من اليمين إلى اليسار.

inputmode

خاصية تحدد نوع البيانات التي سيقوم المستخدم بإدخالها في حقل معين، وذلك عن طريق تحديد نوع لوحة المفاتيح التي ستظهر للمستخدم، وبالطبع ليس لهذه الخاصية تأثير عند عرض الصفحة على أجهزة الحاسب، لأن لوحة المفاتيح المادية تضم كل أنواع المدخلات، وإنما تأثير هذه الخاصية يظهر عند عرض صفحة الويب على الهاتف الجوال أو التابلت، حيث تُعرض لوحة مفاتيح افتراضية، تناسب نوع المدخلات المطلوبة، والتي سيتم تحديدها عن طريق هذه الخاصية.

ويمكن إسناد عدة قيم لهذه الخاصية:

- ***none*** : قيمة تعني عدم وجود لوحة مفاتيح افتراضية.
- ***text*** : قيمة تعني عرض لوحة المفاتيح القياسية للغة المستخدمة.
- ***decimal*** : قيمة تعني عرض لوحة مفاتيح لإدخال كسور الأرقام، أي تحتوي على فاصل عشري (سواء كان نقطة أو فاصلة).
- ***numeric*** : قيمة تعني عرض لوحة مفاتيح لإدخال الأرقام من 0 إلى 9 .
- ***tel*** : قيمة تعني عرض لوحة مفاتيح لإدخال أرقام الهاتف، تحتوي الأرقام من 0 إلى 9 بالإضافة إلى "#" و "*" .
- ***search*** : قيمة تعني عرض لوحة مفاتيح محسنة، قد تحتوي على زر رجوع أو زر اعتماد submit .
- ***email*** : قيمة تعني عرض لوحة مفاتيح لإدخال البريد الإلكتروني أي تحتوي "@" .
- ***url*** : قيمة تعني عرض لوحة مفاتيح لإدخال عنوان موقع إلكتروني تحتوي "/ " .

```
<input type="text" name="user" inputmode="text">
<input type="number" name="age" inputmode="numeric">
<input type="email" name="mail" inputmode="email">
<input type="tel" name="phone" inputmode="tel">
<input type="search" name="find" inputmode="search">
```

wailmonir0@protonmail.com

Microdata

مجموعة خصائص يمكن إضافتها للعنصر، للمساعدة في برامج قراءة الشاشة، أو المساعدة في استخدام هذا العنصر في نص برمجي معين، أو استخدامه في تنسيق معين.

وهذه الخصائص تتكون من اسم لهذه الخاصية وقيمة لها.

itemscope

خاصية تحدد نطاق عمل البيانات الوصفية metadata الخاصة بهذه الخاصية، حيث تفيد هذه الخاصية، ببدء إضافة خاصية جديدة لعنصر معين.

itemid

خاصية تحدد معرف فريد لهذه الخاصية التي يتم إضافتها، ولا يتم تعيين هذه الخاصية إلا للعنصر الذي تم تعيين الخاصيتين itemscope و itemtype له، ويكون عبارة عن عنوان إلكتروني صحيح url ، أو عنوان urn دون تحديد موقعه أو صحته.

itemtype

خاصية توضح العنوان url الذي يحتوي على المفردات، التي يتم استخدامها في تعريف الخاصية الجديدة، ومن أمثلة هذا العنوان schema.org .

itemprop

خاصية تقوم بإضافة خاصية جديدة للعنصر، وكذلك إضافة قيمة لهذه الخاصية، ويتم إضافة القيمة بعدة طرق كما في الكود التالي:

- قد تكون بين وسمي البداية والنهاية (سطر 3) .
- قد تكون قيمة للخاصية src إذا كان اسم الخاصية image (سطر 4).
- قد يتم تعريف عنصر <data> ووضع الخاصية وقيمتها داخله (سطر 5).
- قد يكون للخاصية أكثر من قيمة (سطر 8).
- قد يكون للخاصية أكثر من اسم (سطر 7 و 8).
- قد تكون الخاصية متداخلة nested أي يتم تعيينها لعنصر ابن له itemscope داخل عنصر أب له itemscope آخر (سطر 6) .

```
<div itemscope  itemtype="schema.org/Person"
itemid="urn:name:55:12:5923">
<p itemprop="name"> Wail Monir </p>
<img itemprop="image" src="wailmonir.jpg ">
<data itemprop="age" value="48">The author age</data>
<div itemscope>
<span itemprop="address">Egypt, Cairo</span>
<span itemprop="address  new_address">Egypt, Alexandria</span>
</div>
</div>
```

itemref

خاصية يتم تعيينها لعنصر، تم تعيين الخاصية itemscope له ولم يتم تعيين الخاصية itemtype له، وذلك لربطه بعناصر أخرى خارجية، تم تعيين الخاصية itemprop لها، ويسند لهذه الخاصية قيمة عبارة عن المعرف الفريد id للعناصر الخارجية المرتبطة بهذا العنصر.

```
<div itemscope  itemref="dv1  im"></div>
<p itemprop="name" id="dv1"> Wail Monir </p>
<img itemprop="image"  id="im" src="wailmonir.jpg ">
```

هذه الخاصية لا تدخل ضمن خصائص microdata ، ولكنها تستخدم كدليل للمطورين في التعرف على العناصر الخارجية المرتبطة بالعنصر الذي تم تحديد itemscope له.

wailmonir0@protonmail.com

خصائص الحدث

Event Attributes

wailmonir0@protonmail.com

wailmonir0@protonmail.com

خصائص الحدث Event Attributes

هي خصائص تعمل كشرط، عند تحققه، يتم تنفيذ قيمة هذه الخاصية، والتي تكون عبارة عن script ، أي كود برمجي، ينفذ وظيفة معينة، وفي الغالب يكون هذا الكود عبارة عن كود javascript ، ويمكن أن تكون هذه القيمة تنسيق css ، يمكن تطبيقه عند تحقق الشرط الذي تتطلبه هذه الخاصية.

ويوجد العديد من أنواع هذه الخصائص، ويوجد عدد كبير من الخصائص تحت كل نوع، وحيث إن المجال لا يتسع للشرح المفصل، والتطبيق على كل هذه الخصائص، إذ أنها كثيرة العدد كما ذكرنا، وربما تحتاج مرجعاً منفصلاً ليغطيها بشكل كامل، كما أنها تحتاج معرفة جيدة بلغة الجافاسكريبت، إلا أننا سنقوم بعرض كل الخصائص، وشرحها شرحاً مبسطاً لنتعرف على وظيفتها، وطبيعة عملها، وسنقوم بتطبيق بعض الأكواد البسيطة على كثير من هذه الخصائص، لتوضيح سلوكها وتأثيرها على الصفحة أو عناصرها .

قد يصعب على أي مبتدئ أن يستوعب تفاصيل كود Javascript الذي يتم تنفيذه، بالرغم من أنني قد استخدمت أكواداً بسيطة جداً وشبه مكررة، ولكن الاستيعاب الجيد والفهم الأكيد لهذا الكود، غير مطلوبين في الوقت الحالي، المطلوب فقط أن تقرأ الكود جيداً، وتفهم ما تسطيع منه قدر إمكانك، ثم تقوم بكتابته في محرر النصوص، وشاهد التغيير المتوقع عند تحقق الشرط المتعلق بوقوع الحدث، ليكون ذلك تمهيداً طيباً لك، عندما تبدأ في دراسة Javascript قريباً.

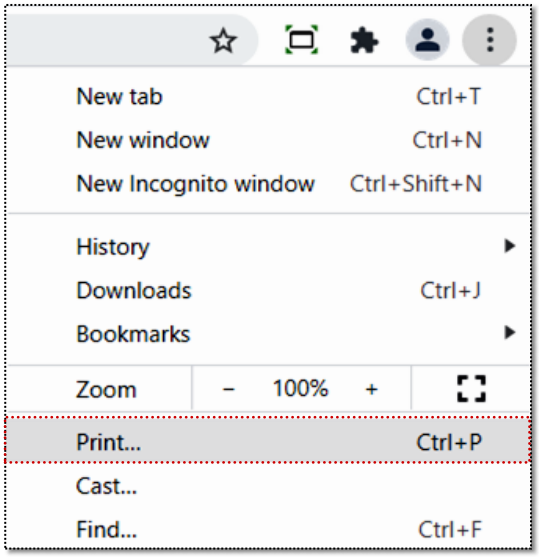
wailmonir@protonmail.com

خصائص حدث الصفحة Window Event Attributes

خصائص متعلقة بالصفحة ككل، ويتم تعيينها في الغالب للعنصر <body> وهي:

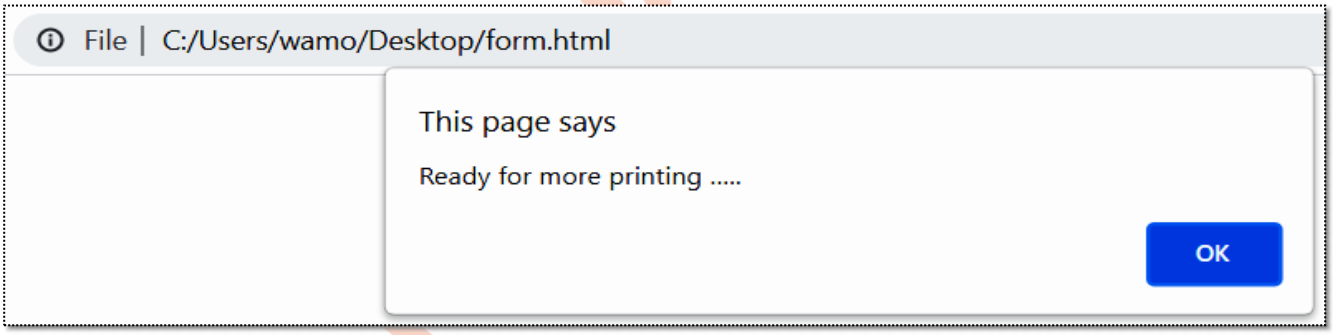
- *onafterprint* : يتم تنفيذ الكود البرمجي بعد الموافقة على طباعة الصفحة.

```
<body onafterprint="alert('Ready for more printing .....')">
</body>
```



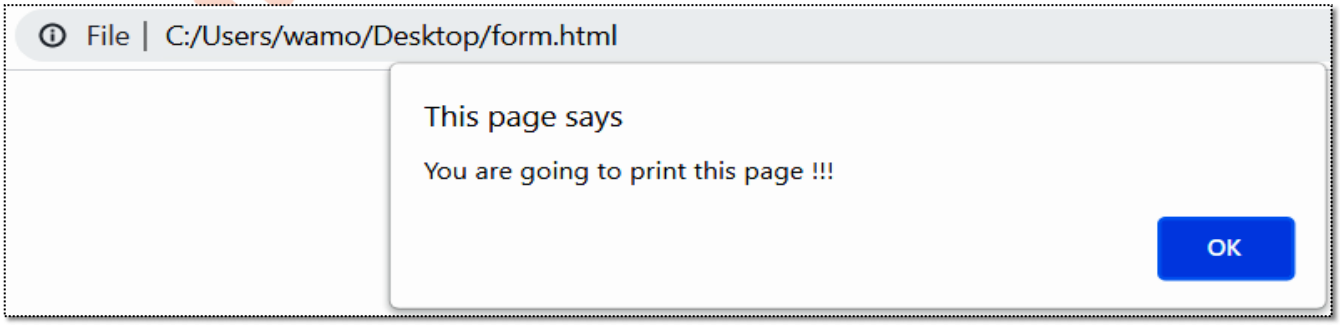
لاحظ أنك بعد أن تضغط على Ctl+p من لوحة المفاتيح، أو تضغط على الأمر Print... ، من القائمة الجانبية الخاصة للمتصفح، ستفتح لك صفحة إعدادات الطباعة، وبعد أن توافق على الإعدادات ثم تضغط Print لتنفيذ أمر الطباعة:

تظهر لك رسالة، تعرض النص الموجود في الكود:



- *onbeforeprint* : يتم تنفيذ الكود البرمجي قبل طباعة الصفحة.

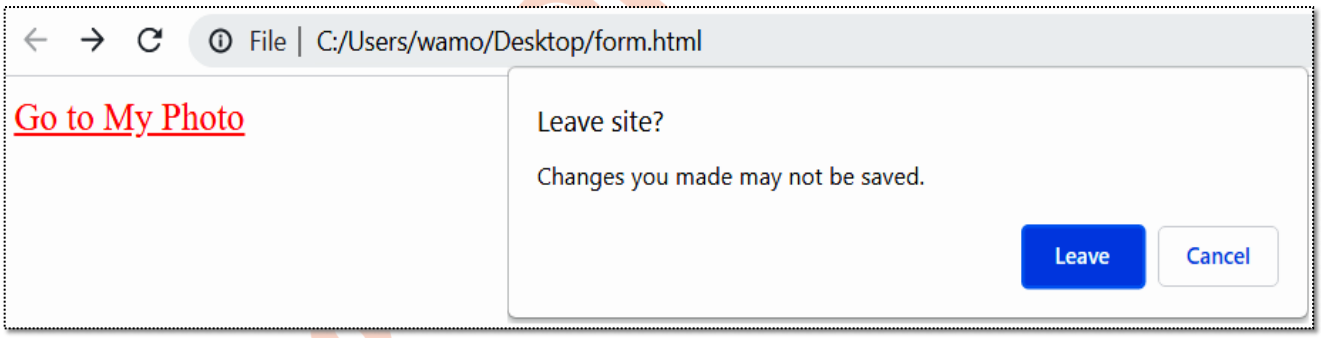
```
<body onbeforeprint="alert('You are going to print this page')">
</body>
```



لاحظ أنك بعد أن تضغط على Ctl+p من لوحة المفاتيح، أو الأمر Print... من القائمة الجانبية الخاصة للمتصفح، وقبل أن تفتح لك صفحة إعدادات الطباعة، لتجهيز الصفحة بالإعدادات المطلوبة للطباعة، تظهر لك رسالة، تعرض النص الموجود في الكود البرمجي.

- *onbeforeunload* : يتم تنفيذ الكود البرمجي قبل الانتقال إلى صفحة أخرى، ورغم أنك يجب أن تعد نصاً في الكود البرمجي الذي سيتم تنفيذه، إلا أن هذا النص لن يظهر فعلياً، ولكن تظهر بدلاً منه رسالة افتراضية، تخبرك بأنك تغادر الصفحة الحالية.

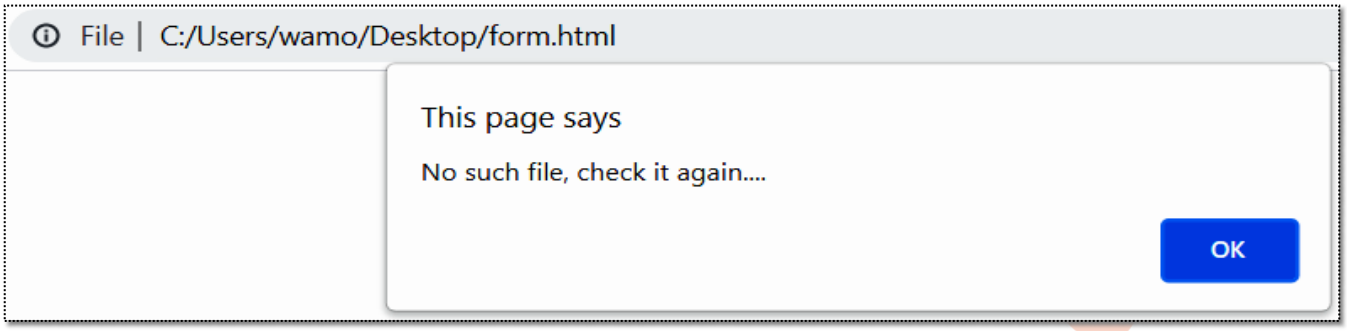
```
<body onbeforeunload="return myfunc()">
<a href="https://cdn.pixabay.com/photo/2018/05/17/09/18/away-3408119_960_720.jpg" target="_top">Go to My Photo
</a>
<script>
function myfunc(){
return ("Good Bye")
}
</script>
</body>
```



لاحظ أنه بعد الضغط على الرابط، تظهر الرسالة قبل الانتقال إلى الصفحة الجديدة، ويتم تجاهل الرسالة التي كتبتها.

- *onerror* : يتم تنفيذ الكود البرمجي عند حدوث خطأ معين، ويتم تعيين هذه الخاصية للعناصر <img> و <object> و <link> و <script> والنوع image من العنصر <input> .

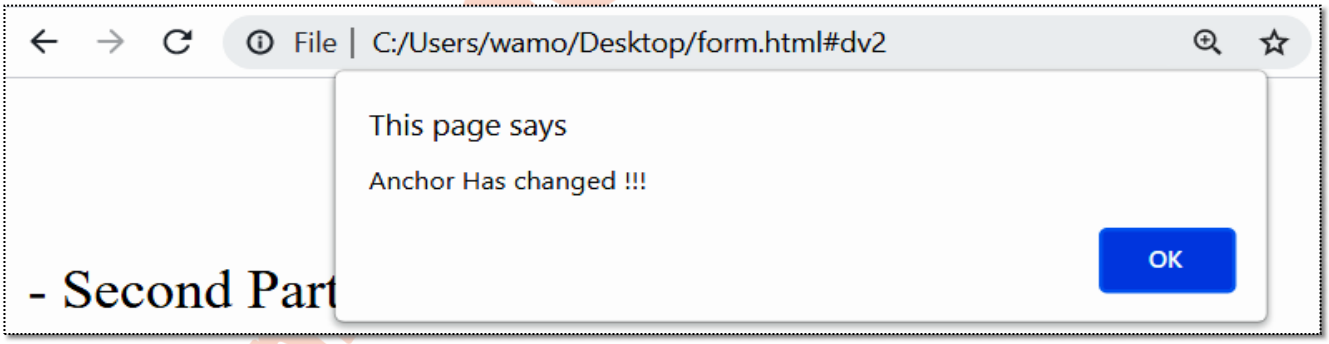
```
<link rel="stylesheet" href="file.css" onerror="alert('No such file, check it again....')">
```



لاحظ أن عدم وجود الملف الموجود في الرابط، أدي لتفعيل الكود البرمجي، وبالتالي ظهرت الرسالة الخاصة بعدم وجود الملف المطلوب داخل العنصر <link> .

- *onhashchange* : يتم تنفيذ الكود البرمجي عند حدوث تغيير في عنوان رابط تشعبي داخل الصفحة، ويكون التغيير في جزء الصفحة anchor part ، الذي يتم الانتقال إليه والذي يبدأ بالرمز "#".

```
<body onhashchange="alert('Anchor Has changed !!!')">
<a href="#dv1">One</a> |
<a href="#dv2">Two</a><br>
<div id="dv1">- First Part</div><br><br><br>
<div id="dv2">- Second Part</div>
</body>
```

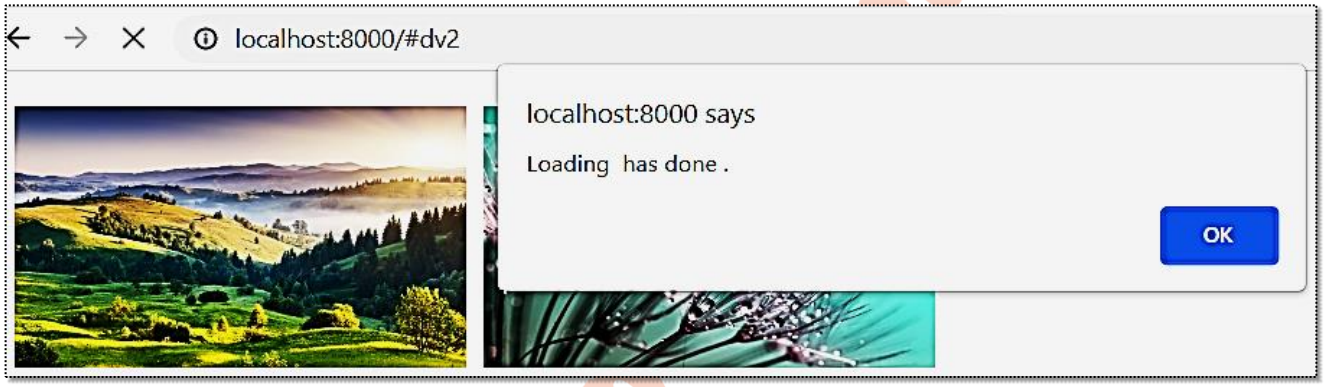


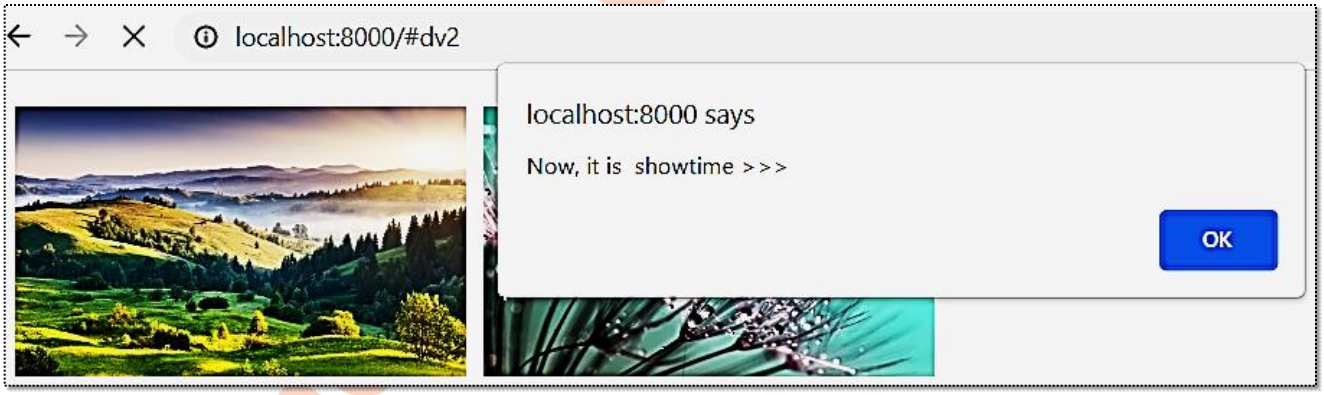
لاحظ وجود رابطي anchor ، عند الضغط على أحدهما بصورة متتابعة، لا تظهر الرسالة، ولكن عند تبادل الضغط بينهما، تظهر الرسالة نتيجة تغير anchor part .

- *onload* : يتم تنفيذ الكود البرمجي بعد انتهاء تحميل الصفحة وعناصرها، وهذا الحدث مدعوم في العناصر <body> و <iframe> و <img> و<style> و <link> و <script> والنوع image من العنصر <input> .

- *onpageshow* : لها نفس تأثير الخاصية onload ، والاختلاف أنه يتم تنفيذ الكود البرمجي بعد انتهاء تحميل الصفحة، أي بعد تنفيذ onload ، ليتم التحميل أولاً، ثم العرض.

```
<body onload="alert('Loading  has done .')"  onpageshow="alert('Now, it is  showtime >>>')">
<img src="https://cdn.pixabay.com/photo/2017/10/10/07/48/hills-2836301_960_720.jpg" width="100" height="60">
<img src="https://cdn.pixabay.com/photo/2014/09/14/18/04/dandelion-445228_960_720.jpg" width="100" height="60">
</body>
```



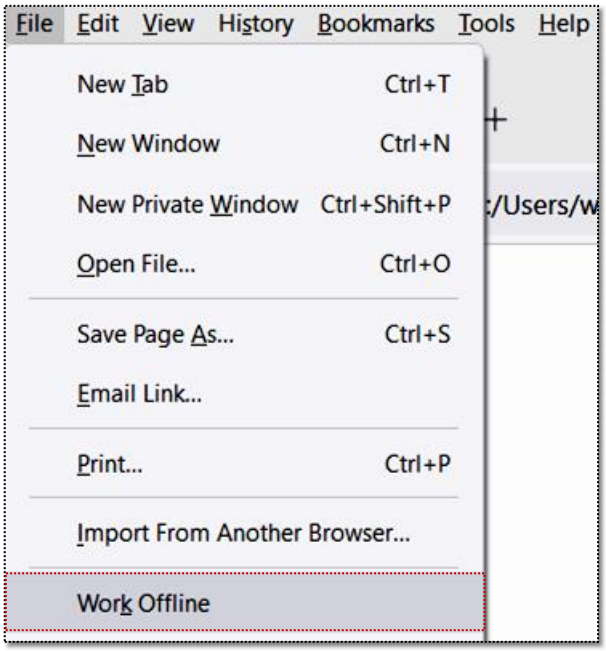


لاحظ أنه تم تنفيذ الحدث onload أولاً، ثم الحدث onpageshow ثانياً.

- *onoffline* : يتم تنفيذ الكود البرمجي عندما يكون عمل المتصفح offline، وهذه الخاصية والتي تليها غير مدعومتين في متصفح Chrome .

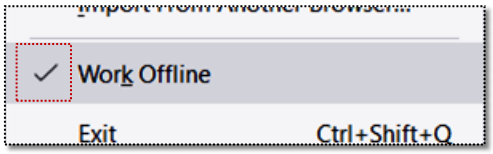
```
<body onoffline="alert('We are going to offline')">
</body>
```

ولجعل متصفح Firefox يعمل offline ، قم بالضغط على مفتاح alt ، ليظهر لك شريط القوائم العلوي، ومن القائمة file اختر Work Offline ، ولاحظ ظهور الرسالة الموجودة في الكود البرمجي.



- *ononline* : يتم تنفيذ الكود البرمجي عندما يعمل المتصفح online .

قم بالضغط على Work Offline لإلغاء العلامة أمامها، للعودة مرة أخرى للعودة للعمل Online ، ولاحظ ظهور الرسالة الموجودة في الكود البرمجي.




```
<body ononline="alert('We are back online')">
</body>
```

- *onresize* : يتم تنفيذ الكود البرمجي عند تغيير أبعاد الصفحة.

```
<body onresize="alert('Changing Window Dimensions >>>>')"> </body>
```

قم بسحب الصفحة من أحد أطرافها للداخل أو الخارج، لأعلى أو لأسفل ولاحظ ظهور الرسالة الموجودة بالكود البرمجي بمجرد تغيير أبعاد الصفحة.

- ***onunload*** : يتم تنفيذ الكود البرمجي عند إغلاق المتصفح، أو عند الانتقال إلى صفحة أخرى غير الصفحة الحالية.

```
<body onunload="alert('We are leaving, Good Bye, Guys.')">
<a href="https://cdn.pixabay.com/photo/2017/12/23/22/12/orange-3036097_960_720.jpg">Leave this page to my photo</a>
</body>
```

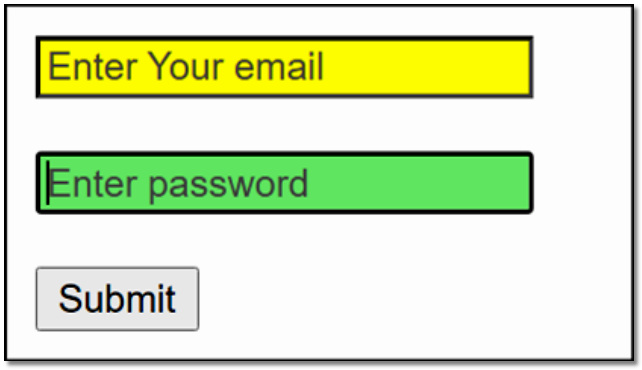
قد لا تعمل هذه الخاصية بالشكل المتوقع، نظراً لاختلاف إعدادات المتصفحات.

الأحداث التي لم أذكر العناصر التي تدعمها، هي أحداث مدعومة في العنصر <body> فقط، أي يتم تعيينها في وسم البداية للعنصر <body> .

wailmonir0@protonmail.com

خصائص حدث النموذج Form Event Attributes

خصائص تنشط أحداثها بفعل التعامل مع عنصر النموذج <form> ، أو التعامل مع عناصره الداخلية، حيث يتم تعيين هذه الخصائص للعنصر <form> وعناصره الداخلية.



- *onblur* : يتم تنفيذ الكود البرمجي أو التنسيق عند خروج العنصر من التركيز (نقل التركيز لعنصر آخر عن طريق مفتاح tab أو الضغط بالفأرة على عنصر آخر).

- *onfocus* : يتم تنفيذ الكود البرمجي أو التنسيق، عند انتقال التركيز لهذا العنصر (عن طريق مفتاح tab أو الضغط بالفأرة على هذا العنصر).

```
<form style="border:solid;width:200px;padding:10px" >
<input type="text" placeholder="Enter Your email"
onblur="style='background:yellow'"
onfocus="style='background:lightgreen'"><br><br>
<input type="pasword" placeholder="Enter password"
onblur="style='background:yellow'"
onfocus="style='background:lightgreen'"><br><br>
<input type="Submit">
</form>
```

لاحظ تغير لون خلفية العنصر للون الأصفر ، عند فقدان التركيز، وتلونها باللون الأخضر عند انتقال التركيز إليه مرة أخرى.

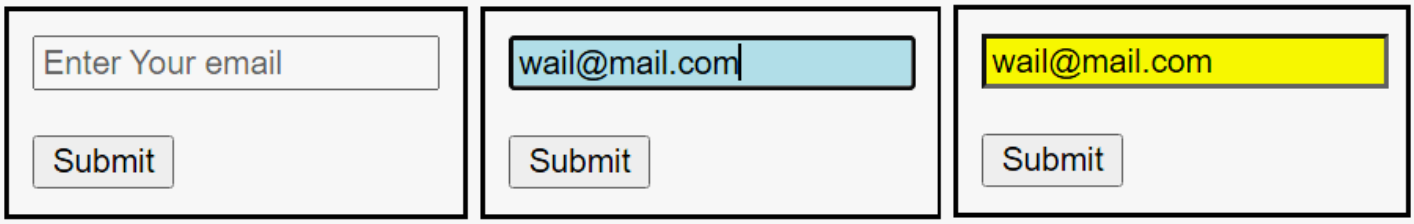
الخاصيتان السابقتان تدعمان كل عناصر html ما عدا <html> و <bdo> و
 و<param> و<iframe> و<head> وكل عناصر metadata ما عدا العنصر <link> .

- *oninput* : يتم تنفيذ التنسيق أو الكود البرمجي عند بدء إدخال القيمة إلى العنصر، والخاصية مدعومة في عنصر <textarea> و الأنواع text وsearch وpassword من العنصر <input> .

- *onchange* : يتم تنفيذ التنسيق أو الكود البرمجي بعد الخروج من العنصر، أو بعد انتقال

تركيز لوحة المفاتيح خارجه، والخاصية مدعومة في العنصر <textarea> والعنصر <select> والأنواع text و password وsearch وcheckbox وradio وfile وrange من العنصر <input> .

```
<form style="border:solid;width:200px;padding:10px" >
<input type="text" placeholder="Enter Your email" name="user"  value=""
onchange="style='background:yellow'"><br><br>
<input type="text" placeholder="Enter Your email" name="user"  value=""
oninput="style='background:lightblue'">
<input type="Submit">
</form>
```

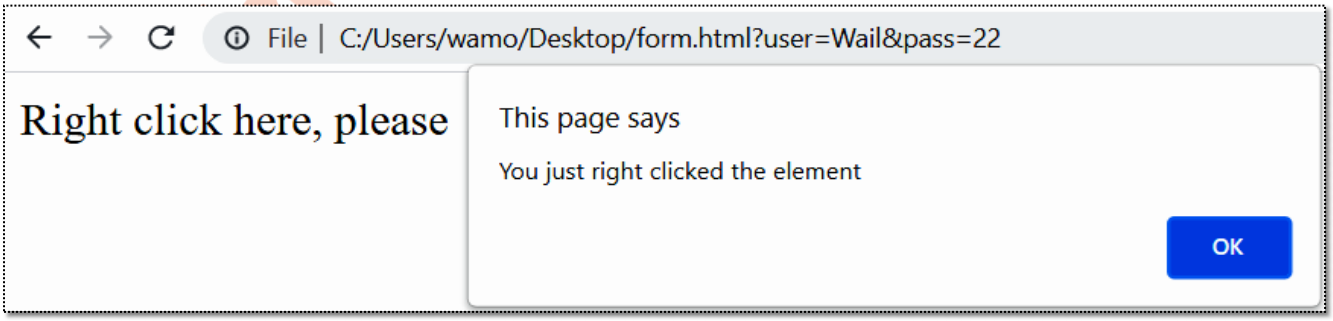


before input oninput onchange

لاحظ تغير لون خلفية حقل الإدخال، عند بدء إدخال البيانات بفعل الخاصية oninput ، وبعد إدخال البيانات والخروج من الحقل، ونقل التركيز خارجه، تغير لونه مرة أخرى بفعل الخاصية onchange .

- *oncontextmenu* : يتم تنفيذ الكود أو التنسيق عند الضغط على العنصر بمفتاح الفأرة الأيمن، وقبل ظهور القائمة، والخاصية مدعومة في كل عناصر html .

```
<div oncontextmenu="alert('You just right clicked the element')">
Right click here, please <div>
```



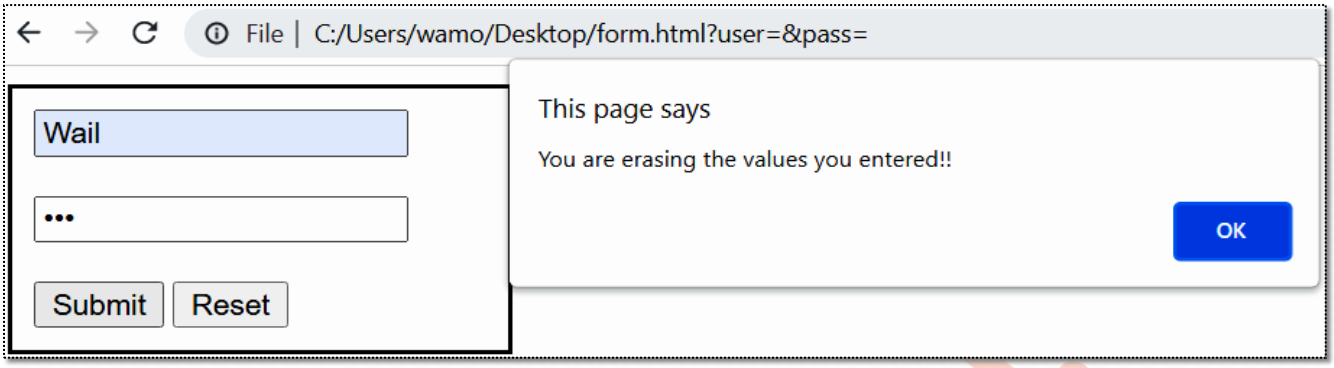
لاحظ أن الضغط بمفتاح الفأرة الأيمن على العنصر، أدى لتنفيذ الكود وظهور الرسالة.

- ***oninvalid*** : يتم تنفيذ الكود البرمجي إذا كانت القيم المدخلة، غير متوافقة مع نوع بيانات العنصر، والخاصية مدعومة في العنصر <input> type .

```
<form style="border:solid;width:200px;padding:10px" >
<input type="email" placeholder="Enter Your email" name="user"
value=""
oninvalid="alert('Invalid entries, try again!!!')"><br><br>
<input type="password" placeholder="Enter password"  name="pass"
value=""><br><br>
<input type="Submit">
</form>
```

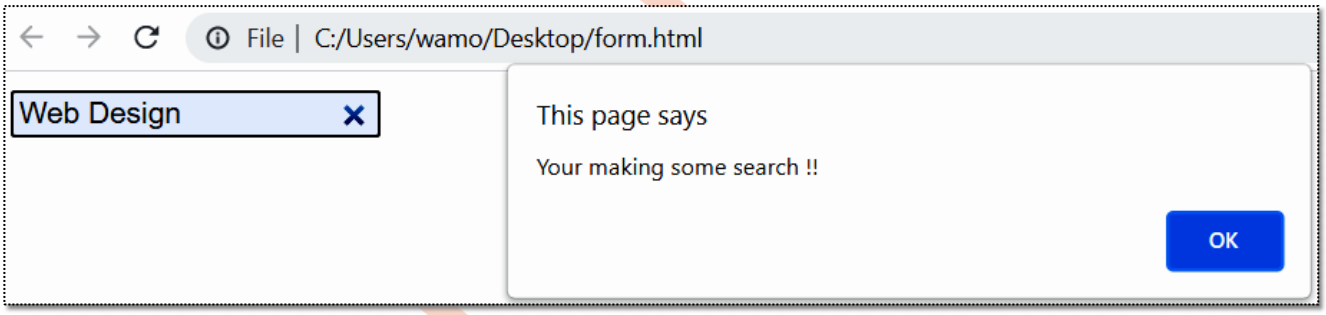
- ***onreset*** : يتم تنفيذ الكود البرمجي بعد الضغط على زر reset ، وقبل مسح قيم العناصر التي تم إدخالها سابقاً ، والخاصية مدعومة في العنصر <form> .

```
<form style="border:solid;width:200px;padding:10px" onreset="alert('You
are erasing the values you entered!!')">
<input type="email" placeholder="Enter Your email" name="user"
value=""><br><br>
<input type="password" placeholder="Enter password"  name="pass"
value="" ><br><br>
<input type="Submit" > <input type="Reset" >
</form>
```



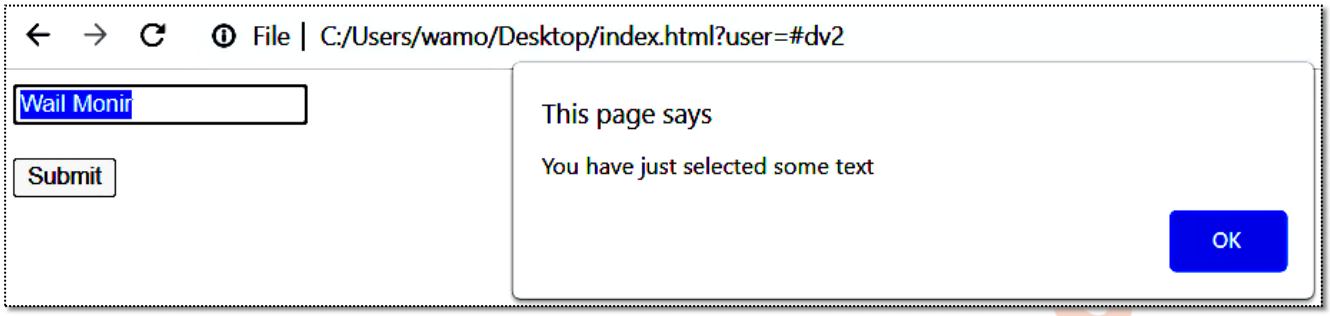
- *onsearch* : يتم تنفيذ الكود البرمجي بعد كتابة نص داخل العنصر من النوع search ، ثم الضغط enter ، والخاصية غير مدعومة في متصفح Firefox و Edge ومدعومة في النوع search من العنصر <input> .

```
<form>
<input type="search" placeholder="Find something or someone"
name="find"
onsearch="alert('Your making some search !!')"><br><br>
</form>
```



- *onselect* : يتم تنفيذ الكود البرمجي عند اختيار نص داخل العنصر، والخاصية مدعومة في العنصر <textarea> والأنواع text وpassword و file من العنصر <input> .

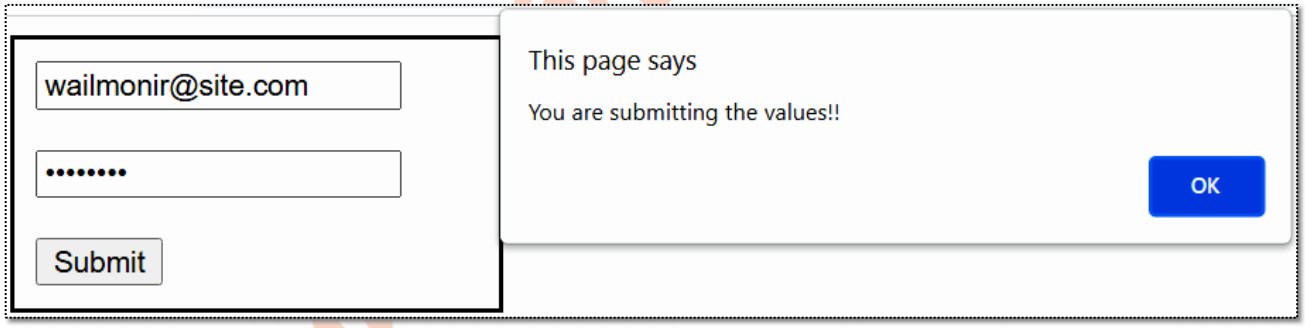
```
<form>
<input type="text" placeholder="Enter your name "
onselect="alert('You have just selected some text')"><br><br>
<input type="Submit" >
</form>
```



لاحظ ظهور الرسالة، عند اختيار النص أو جزء منه.

- *onsubmit* : يتم تنفيذ الكود البرمجي بعد الضغط على زر submit ، والخاصية مدعومة في العنصر <form> .

```
<form style="border:solid;width:200px;padding:10px"
onsubmit="alert('You are submitting the values!!')">
<input type="email" placeholder="Enter Your email" name="user"
value=""><br><br>
<input type="password" placeholder="Enter password"   name="pass"
value="" ><br><br>
<input type="Submit" >
</form>
```



لاحظ ظهور الرسالة عند الضغط على زر submit .

خصائص حدث لوحة المفاتيح Keyboard Event Attributes

خصائص تنشط أحداثها بفعل التعامل مع لوحة المفاتيح، عند إدخال قيم في بعض العناصر.

- *onkeydown* : يتم تنفيذ الكود البرمجي عند الضغط على أي مفتاح.
- *onkeypress* : يتم تنفيذ الكود البرمجي عند إدخال حرف بعد الضغط على المفتاح.

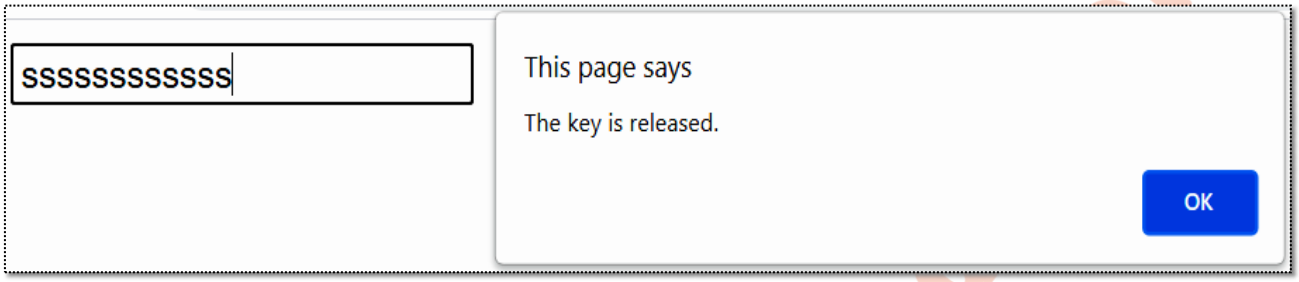
```
<input type="text" placeholder="Enter your name"
onkeypress="alert('The key is pressed.')"
onkeydown="alert('The key is down.')">
```

لاحظ أننا قد قمنا بتعيين خاصيتي حدث للعنصر <input> ، ورغم أن الخاصية onkeypress تسبق الخاصية onkeydown ، إلا أن تنفيذ حدث الخاصية onkeydown يحدث أولاً، لأنه يبدأ عند الضغط على مفتاح معين، ثم يتبعه تنفيذ حدث الخاصية onkeypress ، والذي يبدأ عند إدخال الحرف، ولكن قبل ظهوره في حقل الإدخال:

اعلم أن الخاصية onkeydown تنشط عند الضغط على أي مفتاح في لوحة المفاتيح، بينما الخاصية onkeypress تنشط فقط، عند الضغط على الحروف، بمعنى أنها لا تنشط عند الضغط على بعض المفاتيح مثل Ctl و Shift و Alt و Esc و Delete و Home و Insert وCaps-lock وغيرها.

- *onkeyup* : يتم تنفيذ الكود البرمجي عند رفع الضغط عن المفتاح.

```
<input type="text" placeholder="Enter your address"
onkeyup="alert('The key is released.')">
```



لاحظ أن استمرار الضغط على مفتاح معين، أدخل الحرف عدة مرات في العنصر <input> ولم تبدأ الخاصية في تنفيذ الكود، إلا بعد رفع الضغط عن المفتاح، وبالتالي ظهرت الرسالة.

الخصائص الثلاث السابقة مدعومة في كل عناصر html ما عدا <html> و <bdo> و
 و <param> و <iframe> و <head> وكل عناصر metadata ما عدا العنصر <link>

wailmonir0@protonmail.com

خصائص حدث الفأرة Mouse Event Attributes

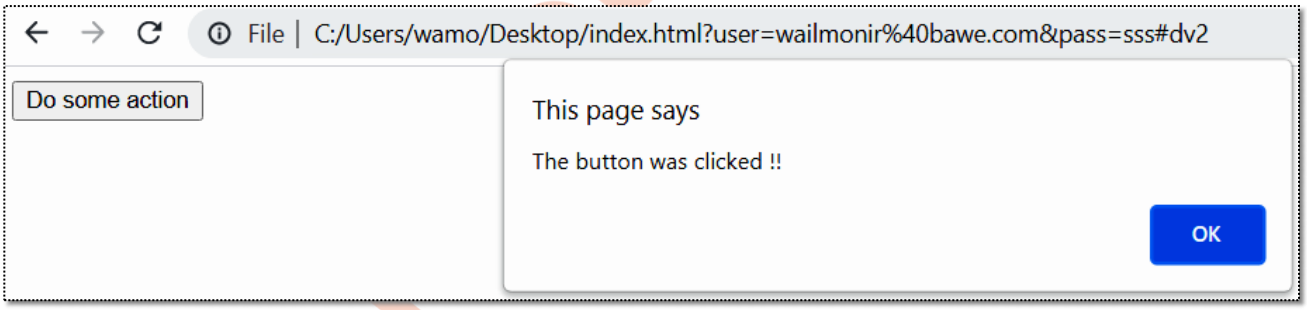
خصائص تنشط أحداثها بفعل تعامل الفأرة مع العناصر.

- *onmousedown* : يتم تنفيذ الكود البرمجي بعد النقر بزر الفأرة الأيسر ، وقبل رفع الضغط عن الزر.

```
<button onmousedown="alert('The button is down !!')">
Do some action</button>
```

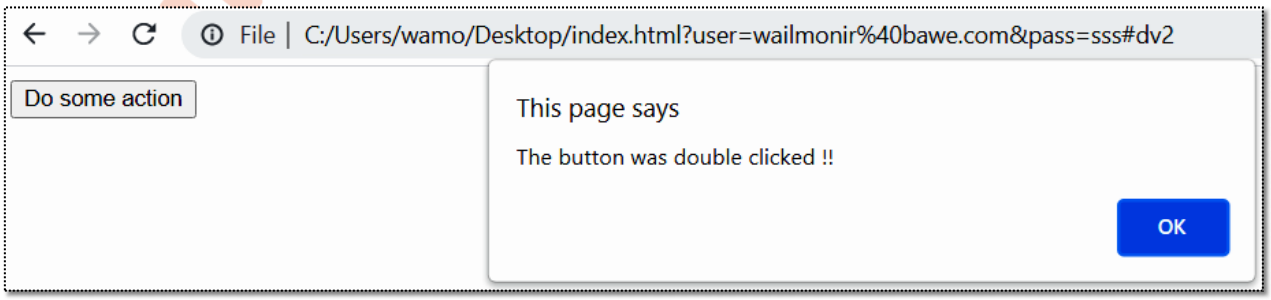
- *onclick* : يتم تنفيذ الكود البرمجي بعد النقر على العنصر، وبعد رفع الضغط عن زر الفأرة الأيسر.

```
<button onclick="alert('The button was clicked !!')">
Do some action</button>
```



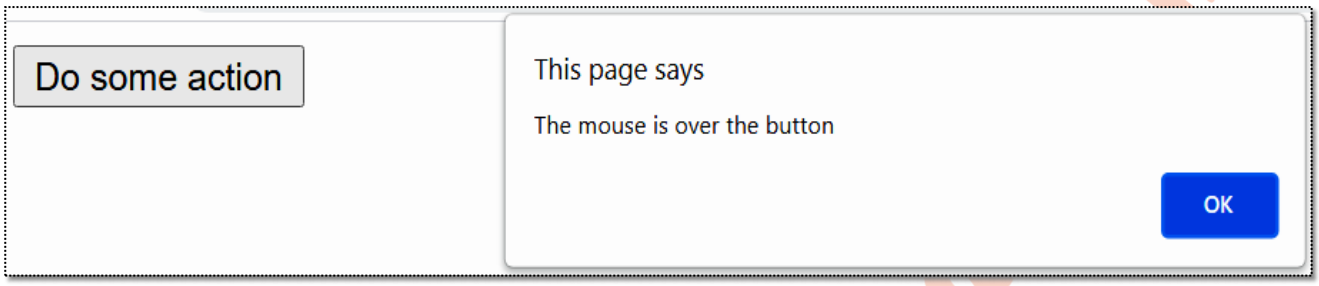
- *ondblclick* : يتم تنفيذ الكود البرمجي بعد النقر المزدوج على العنصر.

```
<button ondblclick="alert('The button was double clicked !!')">
Do some action</button>
```



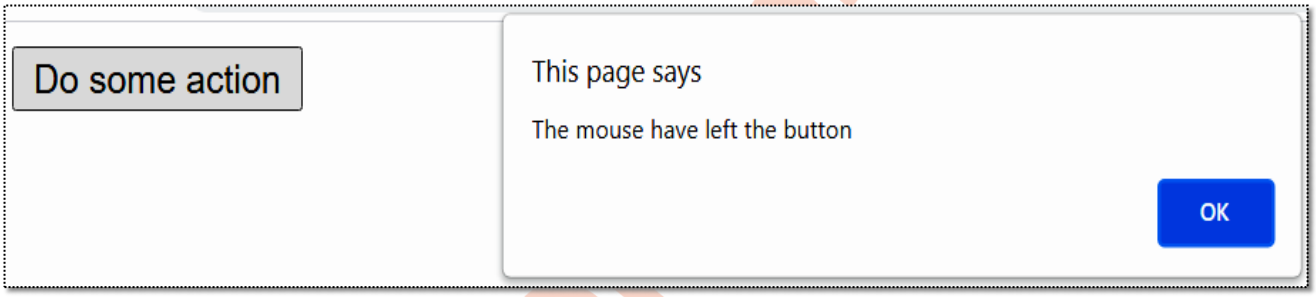
- *onmousmove* : يتم تنفيذ الكود البرمجي عند مرور الفأرة فوق العنصر.

```
<button onmousemove="alert('The mouse is over the button')">
Do some action</button>
```



- *onmouseout* : يتم تنفيذ الكود البرمجي بعد خروج الفأرة من فوق العنصر.

```
<button onmouseout="alert('The mouse have left the button')">
Do some action</button>
```



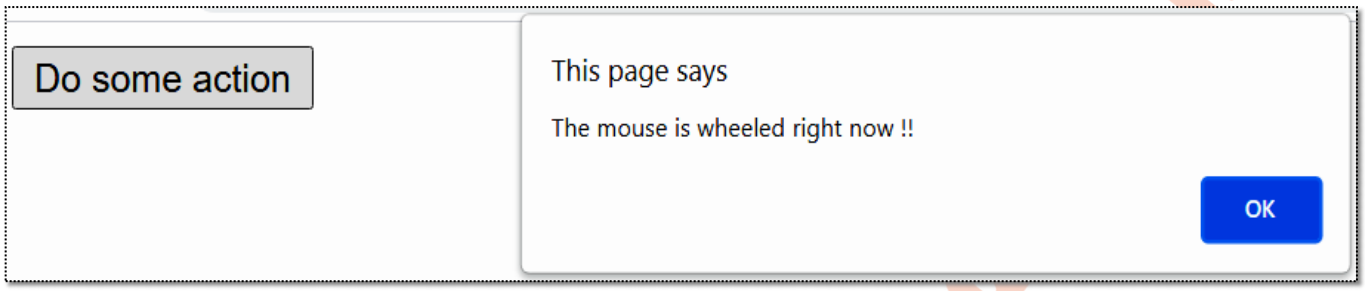
- *onmouseup* : يتم تنفيذ الكود البرمجي عند رفع الضغط عن زر الفأرة، سواء كان الضغط بمفتاح الفأرة الأيمن أو الأيسر.

```
<button onmouseup="alert('The button is released after left or right
clicks')"> Do some action</button>
```

الخصائص السابقة مدعومة في كل عناصر html ما عدا <html> و <bdo> و
 و <param> و <iframe> و <head> وكل عناصر metadata ما عدا العنصر <link> .

- *onwheel* : يتم تنفيذ الكود البرمجي عند دوران عجلة الفأرة فوق العنصر.

```
<button onwheel="alert('The mouse is wheeled right now !!')"> Do some action</button>
```



هذه الخاصية مدعومة في كل عناصر html .

wailmonir0@protonmail

خصائص حدث السحب والإفلات

Drag and Drop Event Attributes

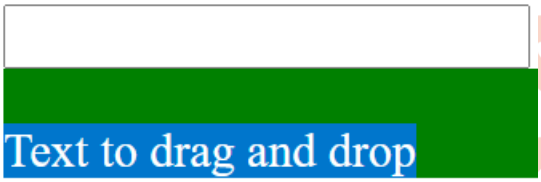
خصائص تنشط أحداثها بفعل عمليتي السحب والإفلات للعناصر، وهذه الخصائص تنقسم لقسمين، قسم يتم تعيينه للعنصر الذي يتم سحبه، وقسم آخر يتم تعيينه لمنطقة السحب أو الإفلات.

وسوف نبدأ أولاً بالخصائص، التي يتم تعينها للعنصر ، الذي يتم سحبه وإفلاته:

- ***ondragstart*** : يتم تنفيذ الكود البرمجي أو التنسيق، عند التقاط العنصر، أو النص، وبدء عملية السحب.

- ***ondrag*** : يتم تنفيذ الكود البرمجي أثناء سحب العنصر إلى منطقة الإفلات.

```
<input type="text">
<div ondragstart="style='background:green'"
        ondrag="style='background:red'"><br>
Text to drag and drop</div>
```



ondragstart

ondrag

لاحظ أنه عند بدء سحب العنصر يتحول لون الخلفية للون الأخضر، وأثناء عملية السحب يتحول لون الخلفية إلى اللون الأحمر (حسب كود التنسيق في قيمة الخاصيتين).

- ***ondragend*** : يتم تنفيذ الكود البرمجي عند إفلات العنصر.

```
<img src="mount.jpg" width="100" ondragend="alert('We have just finish
dragging !!')">
```

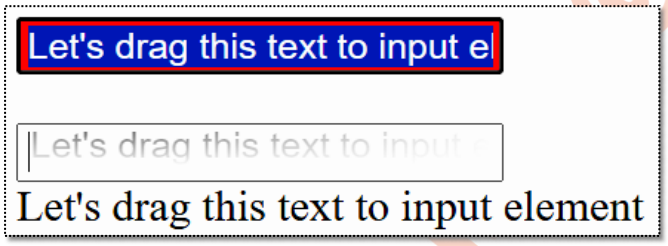
والآن ننتقل للقسم الثاني، والخصائص التي يتم تعيينها لمنطقة الإفلات:

- *ondragenter* : يتم تنفيذ الكود البرمجي بعد التقاط العنصر أو النص، وبدء سحبه إلى أي منطقة صالحة للإفلات.

```
<input type="text"
ondragenter="alert('We entered the dropping area now!!')">
<div> Let's drag this text to input element</div>
```

- *ondragleave* : يتم تنفيذ الكود البرمجي أو التنسيق عند التقاط عنصر أو نص ، وسحبه ليترك منطقة الإفلات التي يوجد بها، لإفلاته في منطقة أخرى.

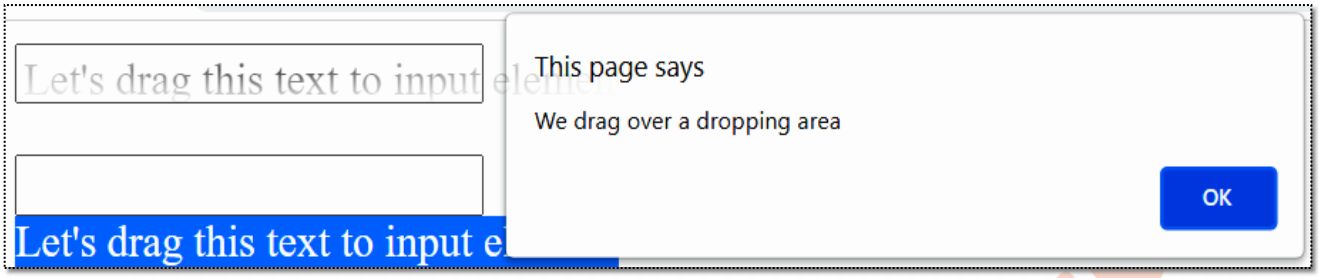
```
<input type="text" ondragleave="style='background:red;'"><br><br>
<input type="text">
<div> Let's drag this text to input element</div>
```



قم بسحب النص من محتوى العنصر ، وقم بإفلاته في العنصر <input> الأول، ثم قم بسحب نفس النص، وإفلاته في العنصر <input> الثاني، ولاحظ تلون خلفية العنصر الأول (طبقاً للكود في الخاصية ondragleave) ، باللون الأحمر عند مغادرة النص إلى العنصر الثاني.

- *ondragover* : يتم تنفيذ الكود البرمجي بعد سحب العنصر أو النص، ومروره فوق المنطقة المطلوب إفلاته فيها، من بين المناطق الصالحة للإفلات.

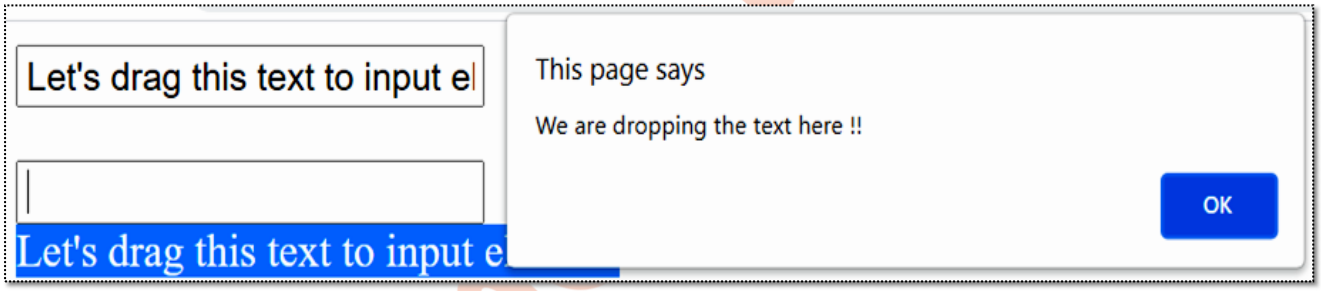
```
<input type="text"><br><br>
<input type="text" ondragover="alert('We drag over a dropping area')">
<div > Let's drag this text to input element</div>
```



لاحظ أن الكود لا ينفذ، والرسالة لا تظهر عند المرور فوق عنصر <input> الثاني ، وعند المرور فوق العنصر الأول، ظهرت الرسالة حسب الكود في قيمة الخاصية ondragover .

- *ondrop* : يتم تنفيذ الكود البرمجي الخاص بمنطقة الإفلات، عند إفلات العنصر أو النص في هذه المنطقة، دون غيرها من المناطق الأخرى الصالحة للإفلات.

```
<input type="text"><br><br>
<input type="text" ondrop="alert('We are dropping the text here !!')">
<div > Let's drag this text to input element</div>
```



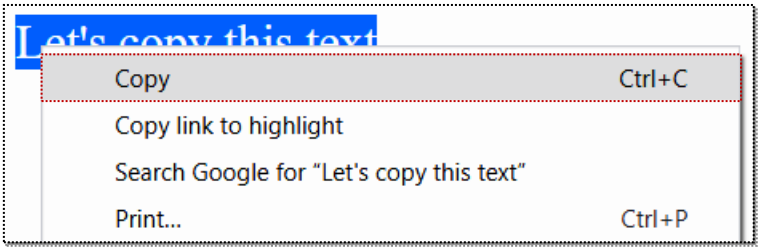
لاحظ أن الرسالة لم تظهر إلا عند إفلات النص في العنصر <input> الثاني.

وكل خصائص حدث السحب والإفلات، مدعومة في كل عناصر html .

خصائص حدث النسخ والقص واللصق

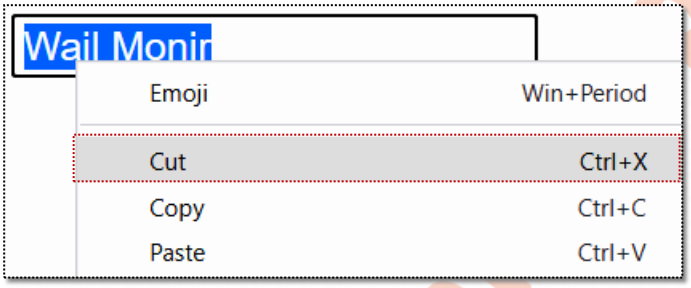
Clipboard Event Attributes

خصائص تنشط أحداثها بفعل القيام بعمليات النسخ والقص واللصق.



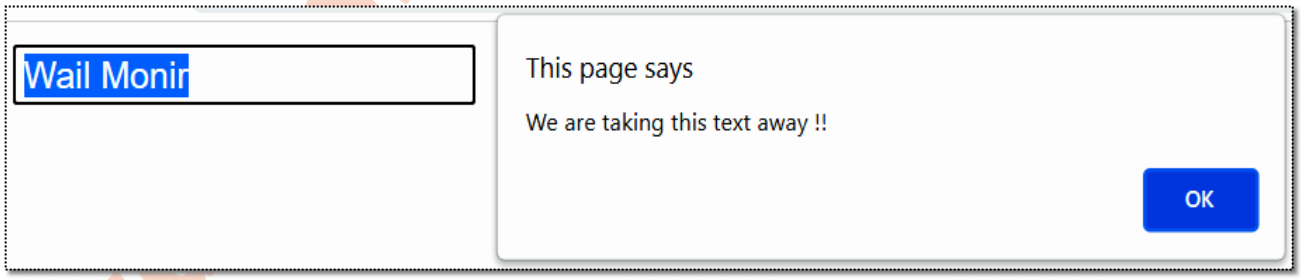
- ***oncopy*** : يتم تنفيذ الكود البرمجي عند محاولة نسخ نص معين، موجود في محتوى أو في حقل إدخال.

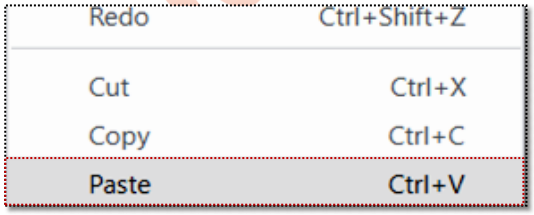
```
<div oncopy="alert('We copy the element content')">
Let's copy this text</div>
```



- ***oncut*** : يتم تنفيذ الكود البرمجي عند محاولة قص نص معين.

```
<input type="text" oncut="alert('We are taking this text away!!')">
```





- ***onpaste*** : يتم تنفيذ الكود البرمجي عند محاولة لصق نص داخل عنصر معين.

```
<input type="text" onpaste="alert('We got you a new text !!')">
```

وكل خصائص حدث النسخ والقص واللصق مدعومة في كل عناصر html .

wailmonir0@protonmail.

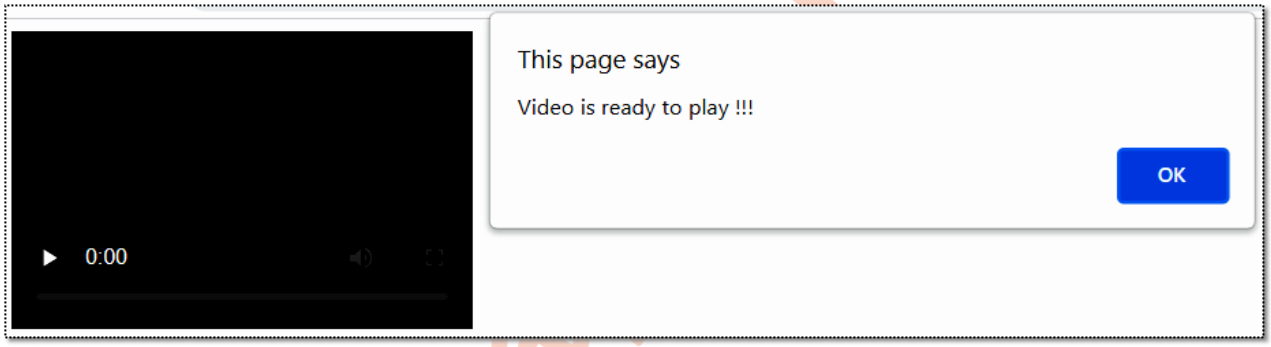
خصائص حدث الوسائط Media Event Attributes

خصائص تنشط أحداثها بفعل التعامل مع عناصر الوسائط، مثل <img> و<video> و<audio> .

- *onabort* : يتم تنفيذ الكود البرمجي عند توقف تحميل مصدر العنصر.

- *oncanplay* : يتم تنفيذ الكود البرمجي عندما يكون مصدر العنصر جاهزاً للتشغيل، مثل فيديو لا يوجد أخطاء في تحميله، أو في تحديد مساره أو نوعه.

```
<video controls width="300" height="200"
oncanplay="alert('Video is ready to play !!!')">
<source src="kids.mp4"></source>
</video>
```

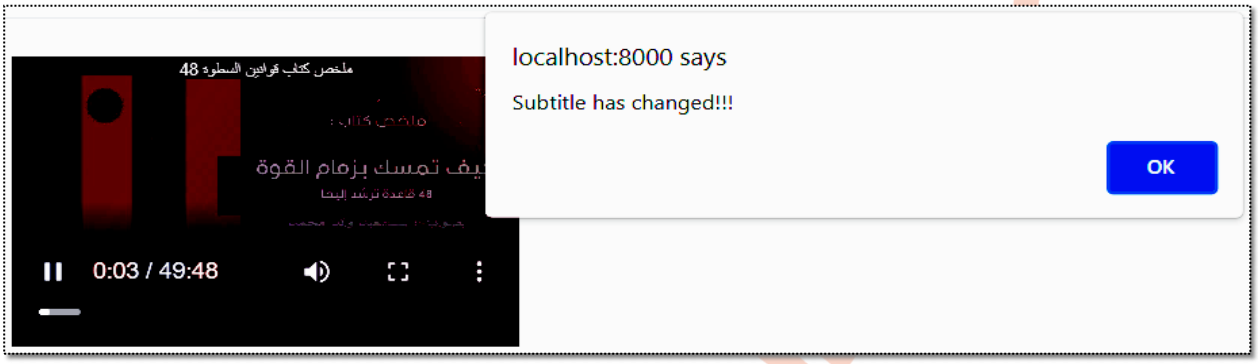


- *oncanplaythrough* : يتم تنفيذ الكود البرمجي إذا كان مصدر العنصر مكتمل التحميل، وسيعمل للنهاية دون أن يتوقف لتحميل أي أجزاء منه.

```
<video controls  width="300" height="200"
oncanplaythrough="alert('Video will play with no stop !!!')">
<source src="kids.mp4"></source>
</video>
```

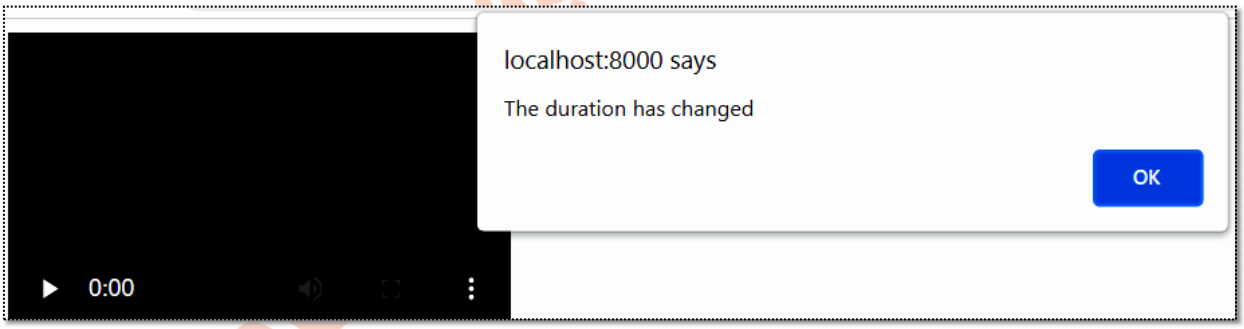
- *oncuechange* : يتم تنفيذ الكود البرمجي عند كل تحديث في الترجمة.

```
<video muted width="300" height="200" >
<source src="kids.mp4"></source>
<track src="kids.vtt" kind="subtitles" default oncuechange="alert('Subtitle has changed')"></track>
</video>
```



- ***ondurationchange*** : يتم تنفيذ الكود البرمجي عند تغيير توقيت تشغيل الفيديو، ويكون أول تغيير عند بداية عرض الفيديو حيث يزيد زمن التشغيل عن 0:00 .

```
<video controls width="300" height="200" ondurationchange="alert('The duration has changed')">
<source src="kids.mp4" ></source>
</video>
```



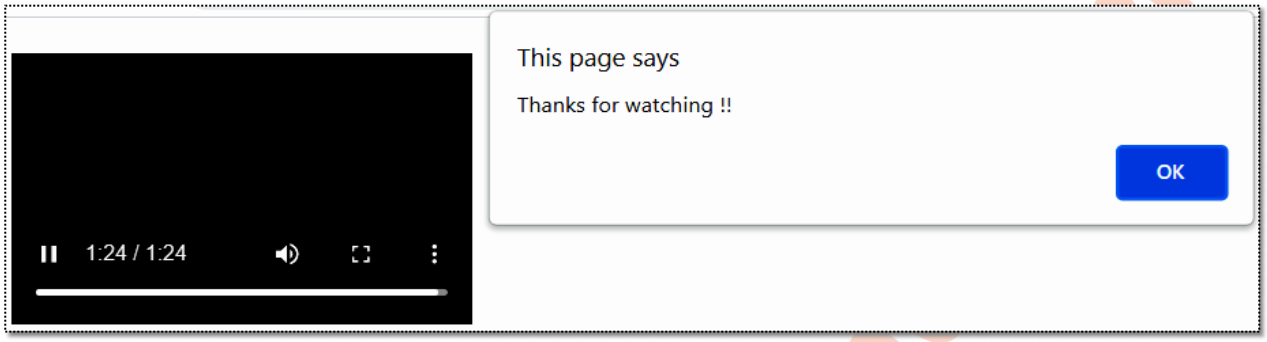
- ***onemptied*** : يتم تنفيذ الكود البرمجي عند حدوث خطأ مفاجئ في ملف المصدر.

```
<video controls  width="300" height="200" onemptied="alert(' Something wrong happened !!')">
<source src="kids.mp4" ></source>
</video>
```

- ***onended*** : يتم تنفيذ الكود البرمجي عند نهاية عرض الفيديو أو نهاية ملف الصوت.

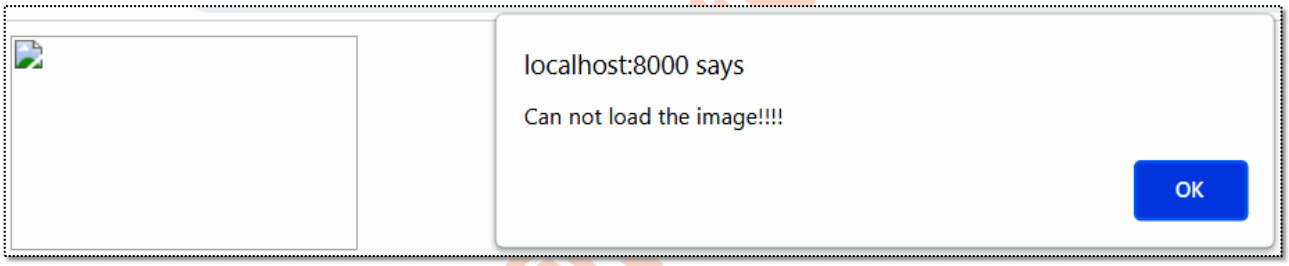
```
<video controls width="300" height="200" onended="alert('Thanks for
```

```
watching !!')">
<source src="kids.mp4" ></source>
</video>
```



- ***onerror*** : يتم تنفيذ الكود البرمجي عند حدوث خطأ في تحميل الملف.

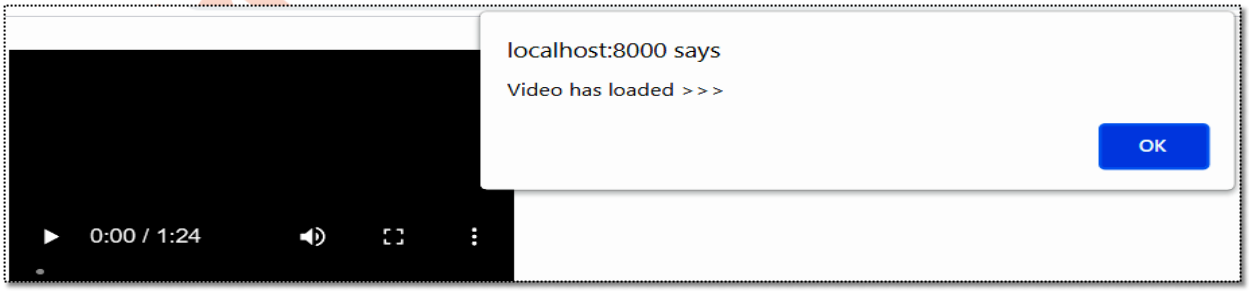
```
<img src="some_image.jpg" width="200" height="120"
onerror="alert('Can not load the image!!!!')">
```



قد يكون الخطأ في اسم مصدر العنصر ، أو مساره أو عدم وجود الملف.

- ***onloadeddata*** : يتم تنفيذ الكود البرمجي عند اكتمال تحميل ملف المصدر.

```
<video controls   width="300"   height="200"
onloadeddata="alert('Video has loaded >>>')">
<source src="kids.mp4"></source>
</video>
```

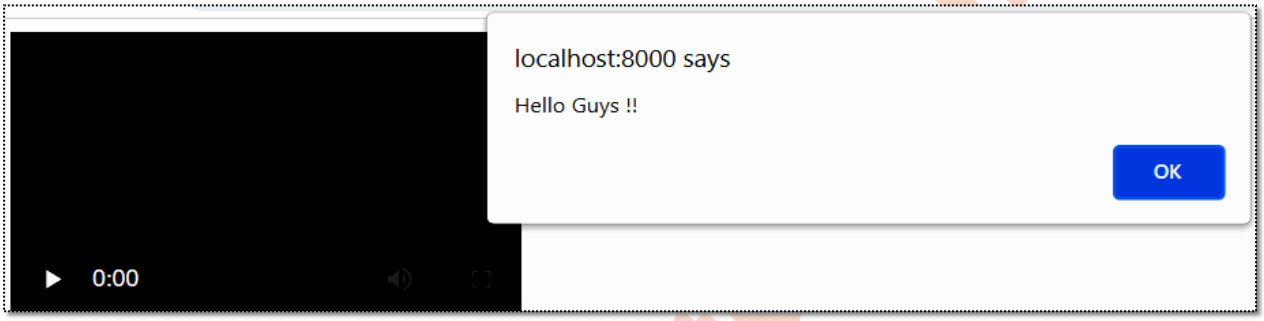


- ***onloadedmetadata*** : يتم تنفيذ الكود البرمجي عند تحميل البيانات الخاصة

بإعدادات ملف المصدر ، مثل الطول والعرض.

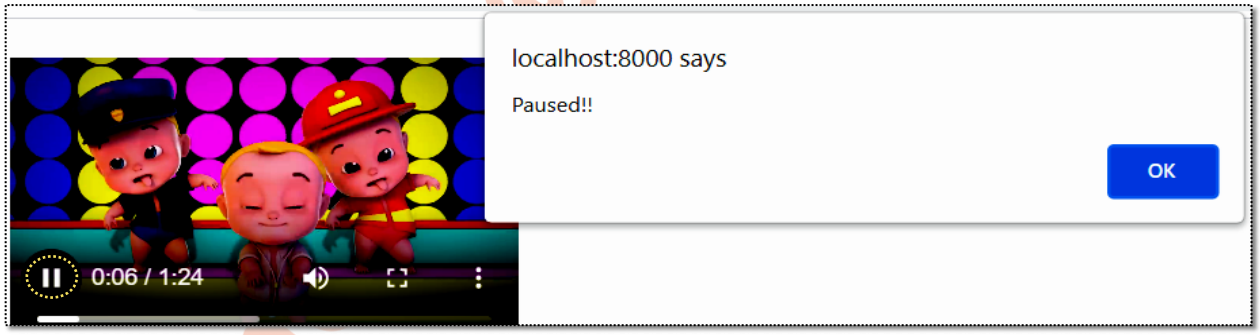
- ***onloadstart*** : يتم تنفيذ الكود البرمجي عند بداية تحميل وعرض ملف المصدر.

```
<video controls  width="300"  height="200"
onloadstart="alert('Hello Guys !!')">
<source src="kids.mp4" ></source>
</video>
```



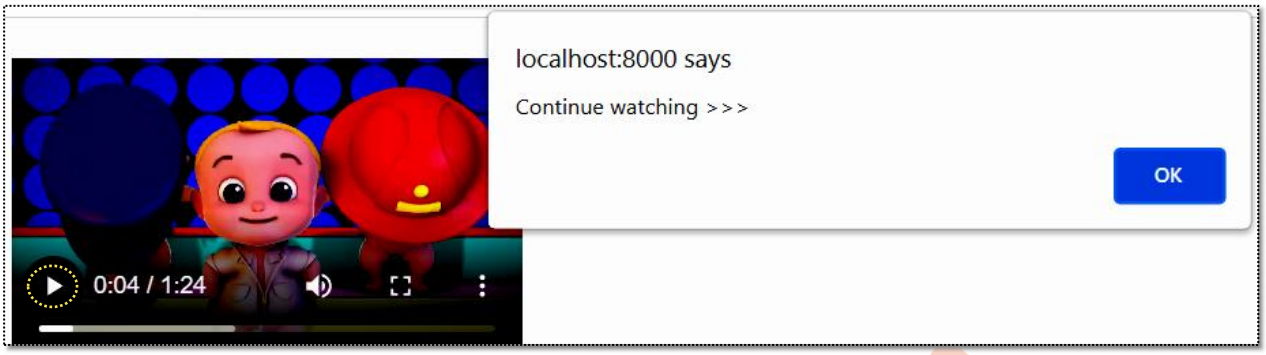
- ***onpause*** : يتم تنفيذ الكود البرمجي عند توقيف عرض ملف المصدر.

```
<video controls width="300" height="200" onpause="alert('Paused!!')">
<source src="kids.mp4"></source>
</video>
```



- ***onplay*** : يتم تنفيذ الكود البرمجي عند تشغيل ملف المصدر بعد توقيفه.

```
<video controls  width="300" height="200" onplay="alert('Continue
watching')">
<source src="kids.mp4"></source>
</video>
```

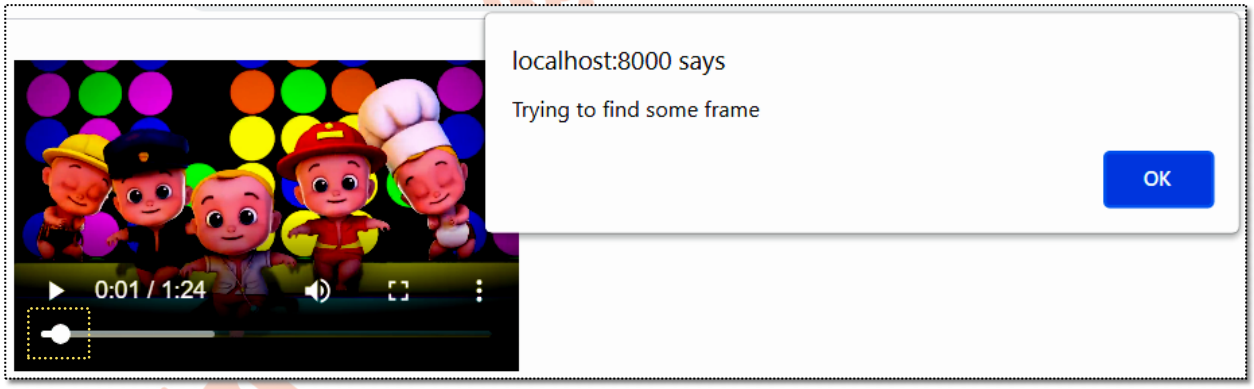


- *onprogress* : يتم تنفيذ الكود البرمجي طالما كان هناك تقدم في تحميل وتشغيل ملف المصدر.

- *onratechange* : يتم تنفيذ الكود البرمجي عند زيادة أو تقليل سرعة عرض ملف المصدر.

- *onseeking* : يتم تنفيذ الكود البرمجي عند محاولة الانتقال إلى نقطة معينة في ملف المصدر ، عن طريق استخدام شريط التنقل في الفيديو، عند أول التقاط للمنزلق.

```
<video controls   width="300"   height="200"
onseeking="alert('Trying to find some frame')">
<source src="kids.mp4"></source>
</video>
```

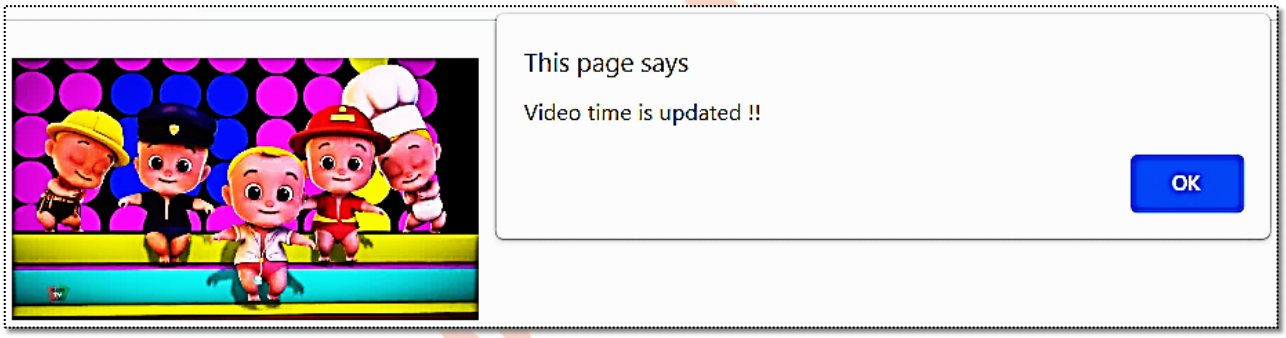


- *onseeked* : يتم تنفيذ الكود البرمجي عند وصول المنزلق على شريط الانتقال إلى النقطة التي يرغب المستخدم في الوصول إليها لتشغيل الفيديو عند هذه النقطة.

```
<video controls   width="300"   height="200"
onseeked="alert('Already find frame !!')">
<source src="kids.mp4" ></source>
</video>
```

- ***ontimeupdate*** : يتم تنفيذ الكود البرمجي كلما حدث تغير في زمن تشغيل ملف المصدر (فيديو أو صوت)، أي كلما تم تشغيل الفيديو يتغير زمنه، وتظهر الرسالة بشكل دائم.

```
<video controls width="300" height="200"
ontimeupdate="alert('Video time is updated !!')">
<source src="kids.mp4"></source>
</video>
```



- ***onwaiting*** : يتم تنفيذ الكود البرمجي عند توقف عرض ملف المصدر تلقائياً نتيجة انتظار تحميل الجزء التالي من الملف.

- ***onvolumechange*** : يتم تنفيذ الكود البرمجي عند تغيير حدة الصوت.

```
<video controls   width="300" height="200"
onvolumechange="alert('Trying to change the volume >>>')">
<source src="kids.mp4"></source>
</video>
```

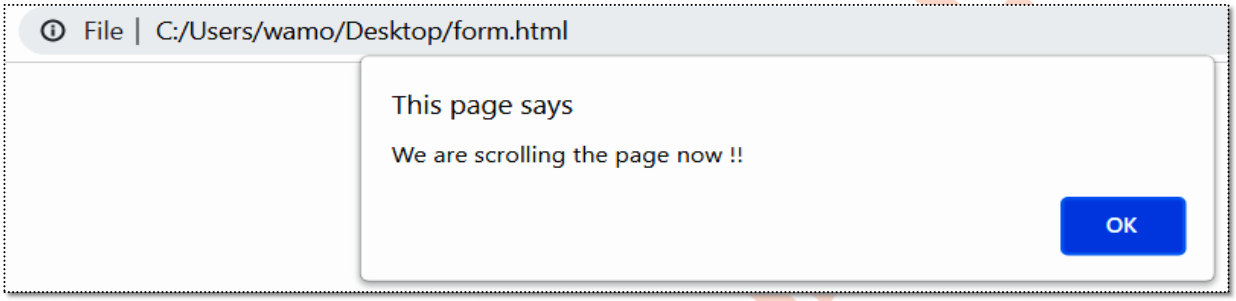
كل خصائص حدث الوسائط مدعومة في العنصرين <audio> و<video> ماعدا الخاصية onerror مدعومة في العناصر <img> و<object> و<link> و<script> والنوع image من العنصر <input> .

wailmonir0@protonmail.com

خصائص حدث أخرى Misc Event Attributes

- *onscroll* : يتم تنفيذ الكود البرمجي عند عمل scroll ، لشريط الانتقال الأفقي أو الرأسي للعنصر.

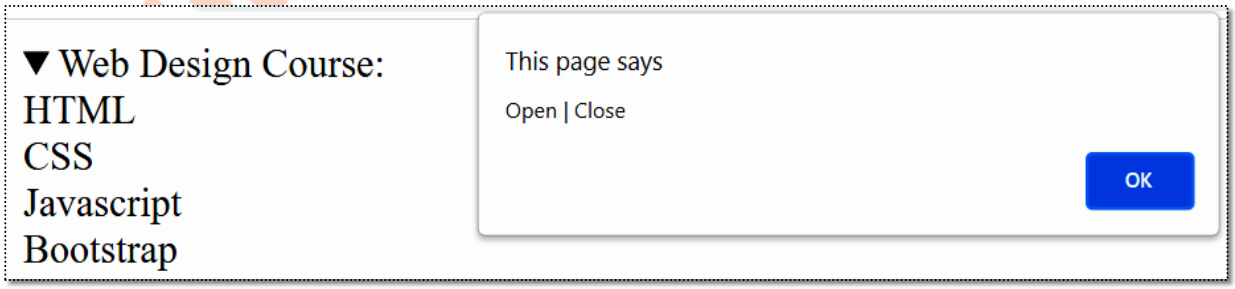
```
<body  style="height:2000px"
onscroll="alert('We are scrolling the page now !!')">
```



هذه الخاصية مدعومة في العناصر <html> و<body> و<blockquote> و<address> و <caption> و<dd> و<dl> و <dt> و و<form> و<fieldset> و<h1> إلى <h6> و<tbody> و<select> و<pre> و<p> و و <ul> و <li> و<object> و <thead> و <tfoot> و<textarea> .

- *ontoggle* : يتم تنفيذ الكود البرمجي عند فتح أو غلق العنصر <details> .

```
<details   ontoggle="alert('Open | Close')">
   <summary>Web Design Course:</summary>
                HTML<br>
                CSS<br>
                Javascript<br>
                Bootstrap
</details>
```



الخاصية مدعومة في العنصر <details> فقط.

ملاحق إضافية

Appendices

wailmonir0@protonmail.com

wailmonir0@protonmail.com

ملحق 1

خصائص مشتركة

يتناول هذا الملحق شرح بعض الخصائص المشتركة بين بعض العناصر، وحتى لا نكرر شرح نفس الخاصية في أكثر من موضع، فقد وضعت هذا الملحق للرجوع إليه، عند الرغبة في دراسة أي من هذه الخصائص، وهي:

rel

خاصية تحدد نوع الرابط أو الملف المرتبط بالصفحة أو النموذج <form> ، ويسند لهذه الخاصية العديد من أنواع القيم وهي:

- ***alternate*** : الرابط هو ملف بديل للصفحة، يعمل في media أو بيئات أخرى يتوافق معها، ويعرض الصفحة بشكل ملائم لهذه البيئة، ومسموح بهذه القيمة مع العناصر <a> و<area> و <link> ، وغير مسموح بها مع العنصر <form> .

- ***author*** : الرابط هو معلومات عن مطور الصفحة، أو كاتب المقال، ومسموح بهذه القيمة مع العناصر <a> و <area> و <link> ، وغير مسموح بها مع العنصر <form> .

- ***bookmark*** : الرابط هو إشارة مرجعية لأقرب عنصر <article> سابق، وفي حالة عدم وجود عنصر <article> ، يكون الارتباط لأقرب عنصر تجزئة في الصفحة لهذا الرابط، ومسموح بهذه القيمة مع العناصر <a> و<area> ، وغير مسموح بها مع العناصر <link> و<form> .

- ***canonical*** : الرابط هو عنصر html يساعد مديري الصفحات على منع تكرار المحتوى، عن طريق تحديد الصفحة الأساسية أو الصفحة المفضلة، مما يؤدي لتحسين أداء محركات البحث، ومسموح بهذه القيمة مع العنصر <link> ، وغير مسموح بها مع العناصر <a> و <area> و<form> .

- ***dns-prefetch*** : الرابط يجعل المتصفح يحلل مسبقاً dns server ، الذي يحتوي عنوان ip الخاص بأصل هذا الرابط، ومسموح بهذه القيمة مع العنصر <link> ، وغير مسموح بها مع العناصر <a> و <area> و <form> .

- ***external*** : الرابط يؤدي إلى مصدر خارج الصفحة الحالية، وهو ما يعني أن المستخدم

سوف يغادر الموقع الحالي، ومسموح بهذه القيمة مع العناصر <a> و <area> و <form> ، وغير مسموح بها مع العنصر <link> .

- ***help*** : الرابط هو ملف لتقديم مساعدة قد تحتاجها أثناء تصفحك، ومسموح بهذه القيمة مع العناصر <a> و <area> و <form> و<link> .

- ***icon*** : الرابط هو أيقونة تظهر في شريط العنوان أو تبويب الصفحة، ومسموح بهذه القيمة مع العنصر <link> ، وغير مسموح بها مع العناصر <a> و <area> و<form> .

- ***license*** : الرابط هو مستند ترخيص منتج معين، ومسموح بهذه القيمة مع العناصر <a> و <area> و <form> و<link> .

- ***manifest*** : الرابط هو تطبيق PWA ، وهي تطبيقات ويب يتم تحميلها مسبقاً ثم تثبيتها لتعمل offline ، ومسموح بهذه القيمة مع العنصر <link> ، وغير مسموح بها مع العناصر <a> و <area> و<form> .

- ***modulepreload*** : الرابط يقوم بتحميل سكريبت معين قبل تحميل الصفحة، ومسموح بهذه القيمة مع العنصر <link> ، وغير مسموح بها مع العناصر <a> و <area> و<form> .

- ***next*** : الرابط هو الصفحة التالية التي سيتم الانتقال إليها، ومسموح بهذه القيمة مع العناصر <a> و <area> و <form> و<link> .

- ***nofollow*** : الرابط يشير لمنتج غير معتمد لدى مطور الصفحة، ولا يوصي به، ومسموح بهذه القيمة مع العناصر <a> و <area> و <form> ، وغير مسموح بها مع العنصر <link> .

- ***noopener*** : قيمة تجعل الرابط الذي يفتح في صفحة فارغة أي target="_blank" ، عندما يعود للصفحة الحالية، لا يتم توجيهه لأي صفحات أخرى غيرها، حيث تقوم هذه القيمة بإعطاء الخاصية window.opener القيمة null ، ومسموح بهذه القيمة مع العناصر <a> و<area> و <form> وغير مسموح بها مع العنصر <link> .

- ***noreferrer*** : عدم إرسال عنوان الصفحة أو أي معلومات أخرى إلى الرابط من خلال http header ، ومسموح بهذه القيمة مع العناصر <a> و <area> و <form> ، وغير مسموح بها مع العنصر <link> .

- opener : الرابط ينشئ مسار تصفح إضافي (مسار يمكن الرجوع منه إلى نفس الصفحة)، إذا كان الرابط سيؤدي إلى مسار تصفح عالي المستوى (أي ليس له صفحة يعود إليها أو ليس له صفحة رجوع أو أب يسبقه في التصفح)، ومسموح بهذه القيمة مع العناصر <a> و <area> و<form> ، وغير مسموح بها مع العنصر <link> .

- pingback : الرابط يحدد عنوان سيرفر يتتبع الصفحة الحالية، للإدلاء بتعليق أو اقتباس حول الصفحة، ومسموح بهذه القيمة مع العنصر <link> ، وغير مسموح بها مع العناصر <a> و <area> و<form> .

- preconnect : الرابط يقدم تلميح للمتصفح، ليفتح اتصالاً مسبقاً مع الرابط، دون إرسال أي بيانات أو تحميل أي محتوى، بحيث يتم جلب المطلوب منه بسرعة أكبر، ومسموح بهذه القيمة مع العنصر <link> ، وغير مسموح بها مع العناصر <a> و <area> و<form> .

- prefetch : يقوم المتصفح بجلب الملف المرتبط ويخزنه مؤقتاً، لأنه قد يكون مطلوباً لاستمرار التصفح، ومسموح بهذه القيمة مع العنصر <link> ، وغير مسموح بها مع العناصر <a> و <area> و<form> .

- preload : يقوم المتصفح بجلب الملف المرتبط وتخزينه مؤقتاً، حتى يتبين كيفية توجيهه حسب قيمة الخاصية as، ومسموح بهذه القيمة مع العنصر <link> ، وغير مسموح بها مع العناصر <a> و <area> و<form> .

- prerender : يجب على المتصفح أن يجلب الصفحة المطلوبة، وأن يستعرضها أو يحملها في الخلفية قبل أن يتصفحها المستخدم، وبالتالي يسرع من عملية تصفح المستخدم لها، ومسموح بهذه القيمة مع العنصر <link> ، وغير مسموح بها مع العناصر <a> و <area> و<form> .

- prev : الرابط هو الصفحة السابقة التي انتقلت منها الصفحة الحالية، ومسموح بهذه القيمة مع العناصر <a> و <area> و <form> و<link> .

- search : الرابط هو مصدر للبحث، وإذا كانت قيمة الخاصية type هي application/opensearchdescription+xml فهذا يعني أن الرابط هو محرك بحث، يمكن أن يضاف في واجهة الصفحة ليستخدم في البحث، ومسموح بهذه القيمة مع العناصر <a> و <area> و <form> و<link> .

- ***stylesheet*** : الملف المرتبط بالصفحة هو ملف تنسيق للصفحة، ومسموح بهذه القيمة مع العنصر <link> ، وغير مسموح بها مع العناصر <a> و <area> و<form> .

- ***tag*** : الرابط يشير إلى صفحة تحمل إشعاراً عن الصفحة الحالية، ومسموح بهذه القيمة مع العناصر <a> و<area> ، وغير مسموح بها مع العناصر <link> و<form> .

wailmonir0@protonmail.com

target

خاصية تحدد الموضع الذي سيفتح فيها الرابط التشعبي عند الضغط عليه، أو الموضع الذي سيتجه إليه النموذج <form> عند اعتماده، أي عند الضغط على زر submit ، والخاصية يسند إليها خمس قيم:

_blank

قيمة تعني أن الرابط أو النموذج سيفتح في تبويب جديد new tab .

_self

قيمة تعني أن الرابط أو النموذج سيفتح في نفس الإطار iframe ، أو نفس الصفحة الموجود بها.

_parent

قيمة تعني أن الرابط أو النموذج سيفتح في الإطار الأب، الذي يحوي الإطار الموجود فيه الرابط أو النموذج.

_top

قيمة تعني أن الرابط أو النموذج سيفتح في كامل الصفحة الموجود فيها.

framename

قيمة تعني تحديد اسم الإطار <iframe> ، الذي سيفتح فيه الرابط.

الخاصية formtarget لها نفس السلوك ، ولكن يتم تعيينها إما إلى النوع submit من العنصر <button> أو إلى النوعين submit و image من العنصر <input> .

wailmonir0@protonmail.com

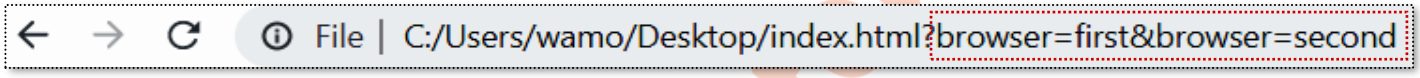
method

خاصية تحدد طريقة إرسال بيانات النموذج <form> ، عند الضغط على زر submit أو طريقة إرسال بيانات الصفحة إلى السيرفر ، أو إلى الصفحة الموجودة في قيمة الخاصية action ، أو قيمة الخاصية formaction ، عند الضغط على الرابط التشعبي أو زر submit ، وهذه الخاصية تقبل قيمتين:

get

وتتميز هذه الطريقة بما يأتي:

- إلحاق بيانات الـ form أو الرابط التشعبي بعنوان URL الموجود في address bar ، فتكون مرئية للمستخدم على هيئة الاسم والقيمة name/value pairs .



- طول عنوان URL محدود حوالي 2048 حرف.
- لا تستخدم هذه الطريقة في إرسال المعلومات الحساسة.
- تفيد عند عمل submit لملاحظة النتائج في شريط العنوان والاستفادة منها.
- أفضل طريقة في حالة إرسال البيانات الغير مهمة.

post

وتتميز هذه الطريقة بما يأتي:

- إلحاق بيانات الـ form أو بيانات الرابط التشعبي بـ http header وبذلك تكون غير مرئية للمستخدم.
- غير محدودة الحجم وغير محدودة لعدد الحروف.
- لا تظهر نتائج عملية اعتماد النموذج submit ، أو الضغط على الرابط التشعبي للمستخدم، مما يوفر أماناً في نقل البيانات.
- أفضل طريقة في نقل البيانات الحساسة، والتي يفضل ألا يطلع عليها المستخدم.

📌 الخاصية formmethod لها نفس السلوك ، ولكن يتم تعيينها إما إلى النوع submit من العنصر <button> أو إلى النوعين submit و image من العنصر <input> .

wailmonir0@protonmail.com

enctype

طريقة تشفير بيانات النموذج عند دخولها إلى السيرفر بعد اعتمادها، وتستخدم هذه الخاصية إذا كانت قيمة الخاصيتين method أو formmethod هي post ، ويسند لهذه الخاصية ثلاث قيم:

application/x-www-form-urlencoded

القيمة الافتراضية للخاصية، وتعني تشفير كل الحروف قبل دخولها إلى السيرفر ويتم تحويل المسافة الفارغة إلى "+" والرموز الخاصة تتحول إلى ما يقابلها في جدول asci .

multipart/form-data

قيمة تستخدم عند رفع upload ملف إلى النموذج، أي إذا كان النموذج داخله عنصر <input> وكان النوع أو type لهذا العنصر هو file .

text/plain

قيمة تعني عدم تشفير أي حروف أو رموز، وإرسال النص على هيئته الأصلية، ولا يوصى باستخدام هذه القيمة.

الخاصية formenctype لها نفس السلوك ، ولكنها تسند إما إلى النوع submit من العنصر <button> أو إلى النوعين submit و image من العنصر <input> .

crossorigin

خاصية تحدد هل يتم إرسال بيانات خاصة بالصفحة مثل cookies و basic authentication وغيرها، إلى المصدر أثناء الحصول على الملف المرتبط، أو أثناء التعامل معه، أم لا يتم إرسال أي بيانات، وهذه الخاصية يسند لها قيمتين:

anonymous

الحصول على الملف دون إرسال أي بيانات للصفحة المُرسِلة (المصدر).

use-credentials

الحصول على الملف مع إرسال البيانات المطلوبة للصفحة المُرسِلة (المصدر).

referrerpolicy

خاصية تحدد إذا كان الانتقال إلى الصفحة الحالية أو destination قد تم من خلال صفحة سابقة فتسمى الصفحة السابقة بـ referrer أو المحيل، لأنها أحالت المستخدم إلى هذه الصفحة الحالية، ويمكن للصفحة الحالية الحصول على معلومات المحيل referrer عن طريق إرسال http header الخاص بها إلى الصفحة الحالية destination ، وتنظيم هذه العملية هو وظيفة الخاصية referrerpolicy ، والتي يسند إليها عدة قيم، ولكن علينا أولاً أن نتعرف على مصطلح هام، سوف يقابلنا كثيراً عند شرح هذه القيم وهو origin .

origin

هو العنوان الرئيسي للموقع أو الصفحة ويبدأ من برتوكول http ويسمى schema ثم host أو domain ثم رقم المنفذ port number ، كما بالشكل التالي:

```
https://www.somedom.com:80/pages/file.html?name=wail#info
└──────────────────────────┘
            origin
```

ويسند للخاصية عدة أنواع من القيم:

no-referrer

لن يتم إرسال أي معلومات عن المحيل أو referrer في http header إلى الصفحة الحالية.

no-referrer-when-downgrade

يجب أن تكون للصفحتين نفس مستوى برتوكول http أو تكون صفحة المحيل أقل من مستوى الصفحة الحالية، ليتم الإرسال كما بالجدول المقابل:

referrer	sent	destination
https	√	https
http	√	http
http	√	https
https	X	http

لاحظ أنه في الحالة الرابعة لم يتم الإرسال أو إحالة أي معلومات في http header عن المحيل referrer لأن الصفحة الحالية destination أقل في برتوكول http عن صفحة المحيل

referrer، وبالتالي لن تنقل الصفحة الأعلى في مستوى البروتوكول، معلوماتها إلى الصفحة الأقل منها في المستوى، وهذا يسمى downgrade .

origin

إرسال الرابط أو العنوان الرئيسي فقط، أي أنه إذا كان عنوان المحيل هو http://www.somesite.com:80/somelink/page.html فسيتم إرسال العنوان الرئيسي وهو http://www.somsite.com:80 .

origin-when-cross-origin

إذا كانت صفحة المحيل referrer والصفحة الحالية destination على نفس origin ، فيتم الإرسال، أما إذا كانت الصفحة الحالية على origin آخر، فيتم فقط إرسال عنوان المحيل الرئيسي origin.

same-origin

إذا كانت صفحة المحيل referrer والصفحة الحالية destination على نفس العنوان الرئيسي أو origin فيتم إرسال http header ، أما إذا اختلف العنوان الرئيسي origin فلا يتم الإرسال.

strict-origin

إذا كانت صفحة المحيل referrer والصفحة الحالية destination لهما نفس المستوى لبروتكول الأمان https فيتم إرسال العنوان الرئيسي origin ، أما إذا كانت الصفحة الحالية لها بروتكول http فلا يتم الإرسال.

strict-origin-when-cross-origin

إذا اختلف origin الصفحة الحالية destination عن origin المحيل referrer فيتم إرسال origin صفحة المحيل referrer إذا اتفقا في مستوى بروتكول الأمان، أو كان بروتوكول صفحة المحيل (http) أقل من برتوكول الصفحة الحالية (https) ، إما إذا كان أعلى، فلا يتم الإرسال.

unsafe-url

يتم إرسال العنوان الرئيسي والفرعي، ولكن لا يتم إرسال باقي العنوان، وهي أجزاء الصفحة أو fragments ولا الاسم أو كلمة السر.

wailmonir0@protonmail.com

ملحق 2

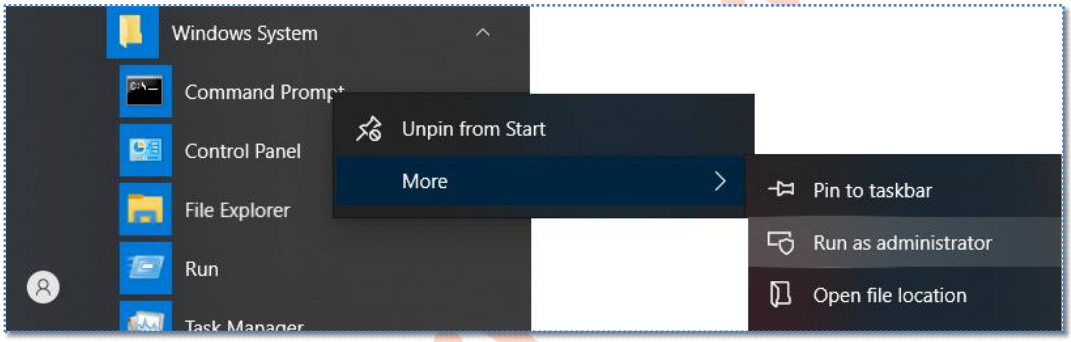
تثبيت خادم محلي Installing a localhost

تحتاج بعض الملفات لأن تعمل بواسطة خادم محلي، لأن بعض المتصفحات لا تقوم بتنفيذ الطلبات الغير متزامنة، أي تنفيذ أكثر من طلب في نفس الوقت، وهذا يسمى بعدم التزامن، حيث يجب تنفيذ جميع الطلبات في وقت واحد، وللتغلب على هذه المشكلة نقوم بتثبيت خادم محلي localhost بأحد الطرق التالية:

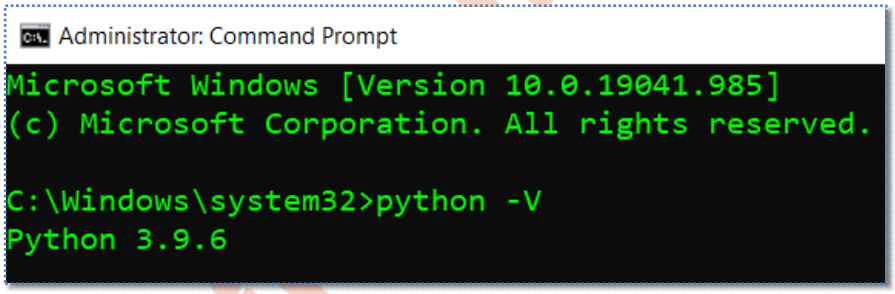
1- إنشاء خادم محلي عن طريق تثبيت برنامج Python

قم بزيارة موقع python.org وقم بتحميل أحد نسخ برنامج python المناسبة لجهازك، ثم قم بتثبيته ببساطة مثل أي برنامج عادي، وبعد الانتهاء من تثبيت البرنامج، تأكد من تعريف مساره في متغيرات النظام، وللتأكد من ذلك قم بفتح موجة الأوامر cmd عن طريق الأمر run as adminstrator كالتالي:

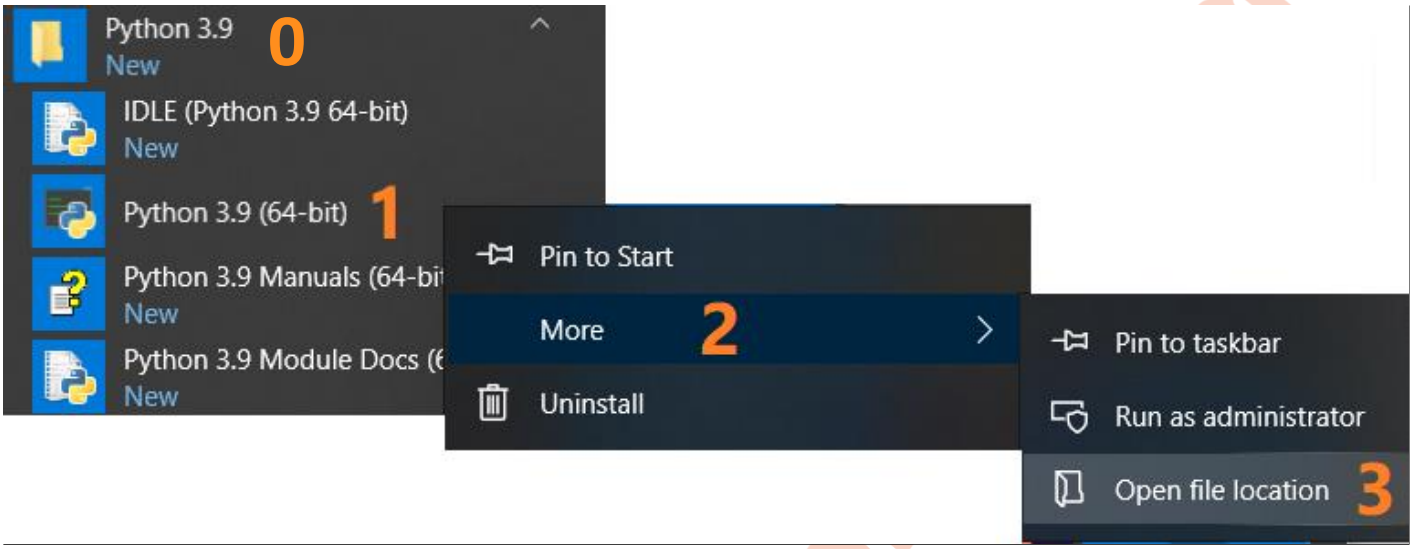
سيتم فتح موجه الأوامر كما بالشكل التالي:



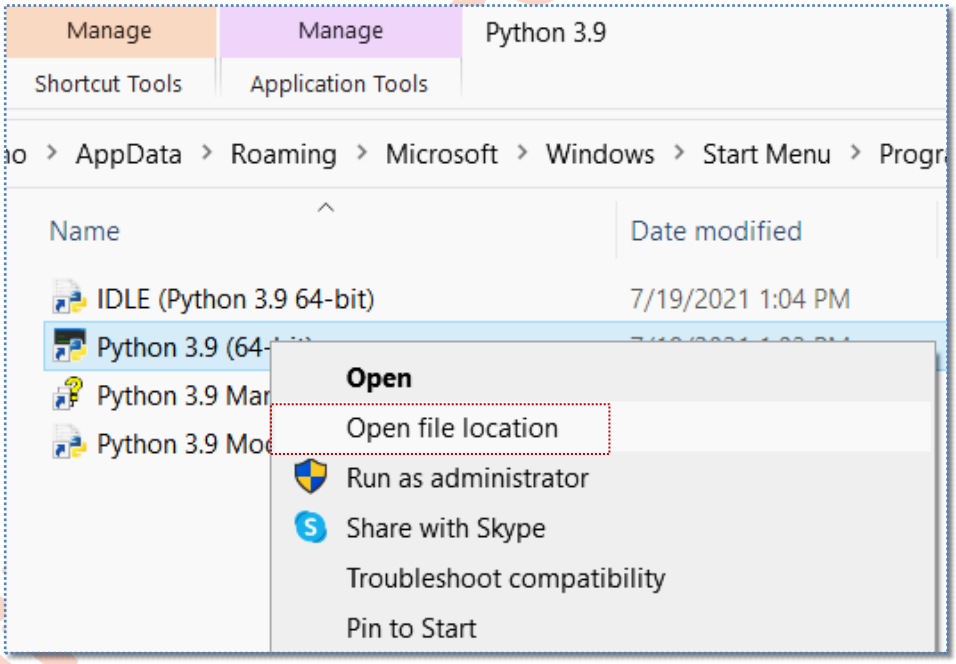
اكتب في سطر الأوامر python --version أو اكتب python -V ثم اضغط enter ولاحظ الجملة التي سيتم طباعتها، فإذا كانت عبارة عن نسخة البايثون التي قمت بتثبيتها، فهذا يعني أن مسار البرنامج معرف في متغيرات النظام، أما إذا ظهرت لك الجملة python is not recognized as internal or external command فهذا يعني أن مسار البرنامج غير معرف في متغيرات النظام، وسوف نقوم بتعريفه في الخطوات التالية:



من قائمة start اختر Python 3.9 ثم اختر Python 3.9 (64-bit) ثم اضغط بمفتاح الفأرة الأيمن واختر More ثم اختر Open file location .



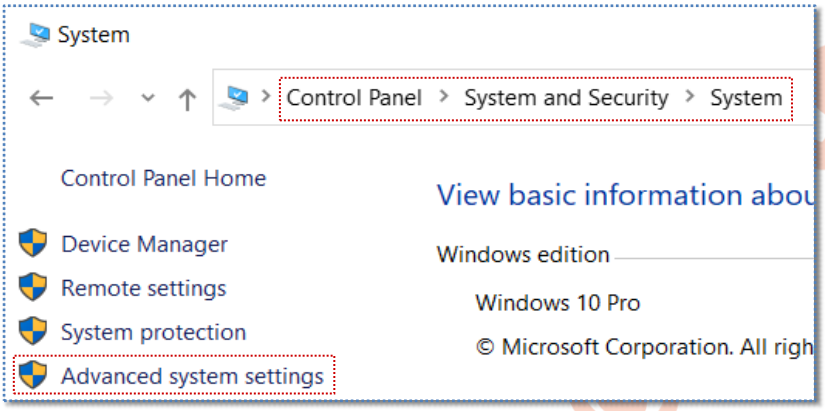
سيتم فتح folder أيقونات اختصارات البرنامج، اختر Python 3.9 (64-bit) ثم اضغط بزر الفأرة الأيمن، واختر من القائمة Open file location .



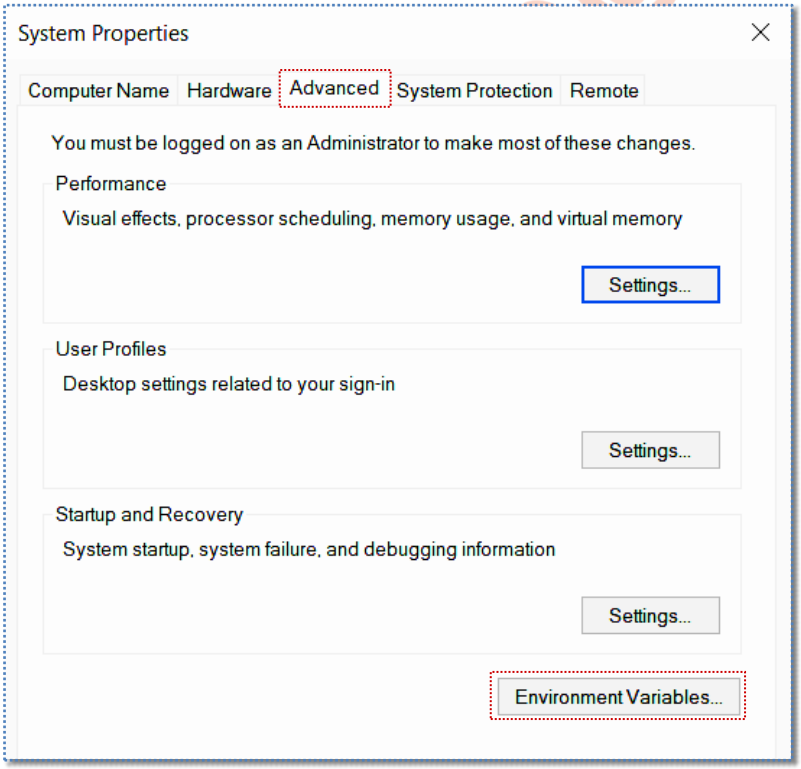
سيفتح لك المسار المحفوظ به البرنامج، كما بالشكل التالي، قم بنسخ مسار البرنامج المظلل والموجود في شريط العنوان:

C:\Users\wamo\AppData\Local\Programs\Python\Python39

Name	Date modified	Type
DLLs	7/19/2021 1:04 PM	File folder
Doc	7/19/2021 1:04 PM	File folder
include	7/19/2021 1:03 PM	File folder
Lib	7/19/2021 1:04 PM	File folder
libs	7/19/2021 1:04 PM	File folder
Scripts	7/19/2021 1:04 PM	File folder
tcl	7/19/2021 1:04 PM	File folder
Tools	7/19/2021 1:04 PM	File folder
LICENSE.txt	6/28/2021 4:09 PM	Text Document
NEWS.txt	6/28/2021 4:10 PM	Text Document
python.exe	6/28/2021 4:08 PM	Application

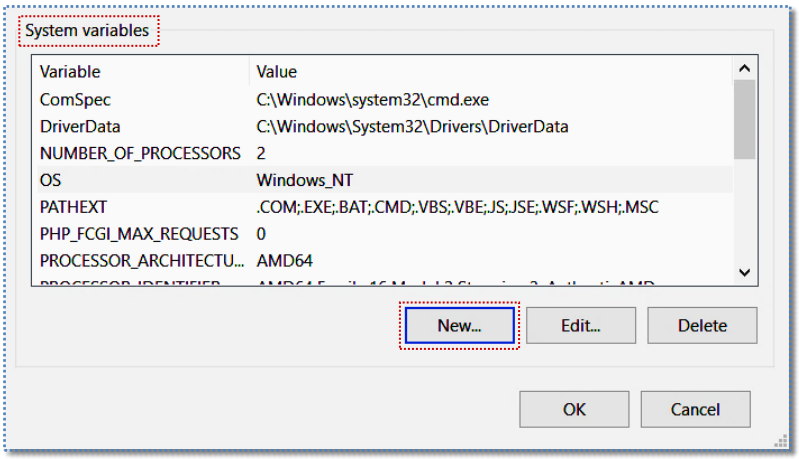


الآن قم بفتح Control Panel ثم System and Security ثم System ثم اختر من القائمة Advanced system settings .

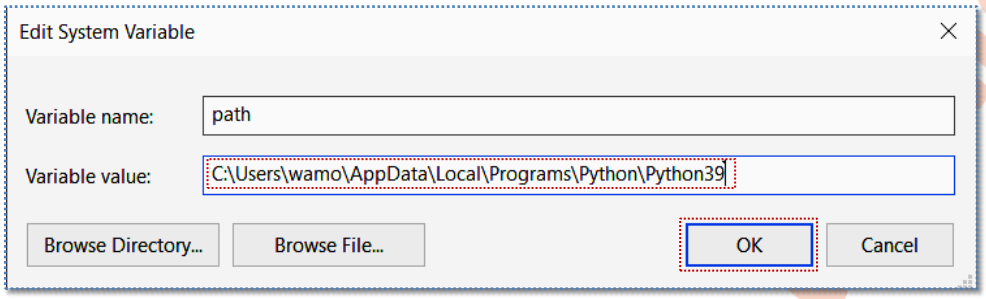


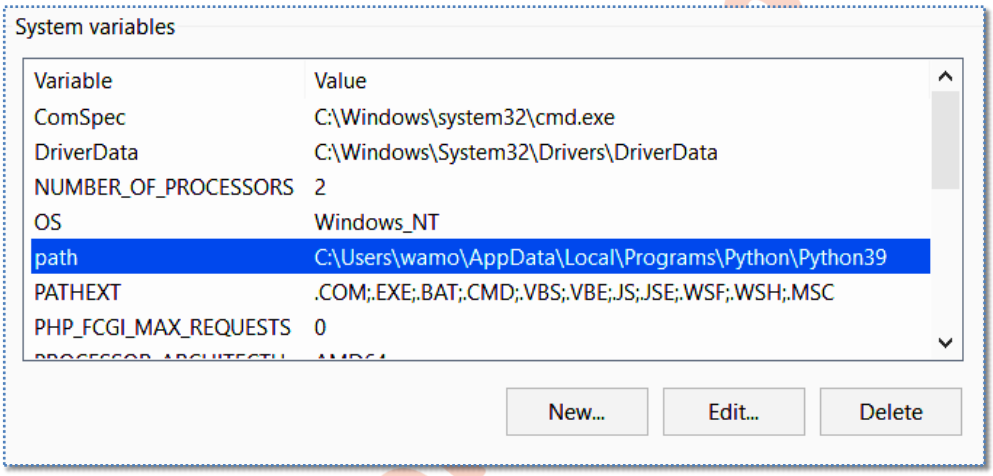
اختر التبويب Advanced ثم اضغط Environment Variables .

من الجزء المسمى System vaiables إذا لم يكن المتغير path معرفاً تحت القائمة Variable ، اضغط New .

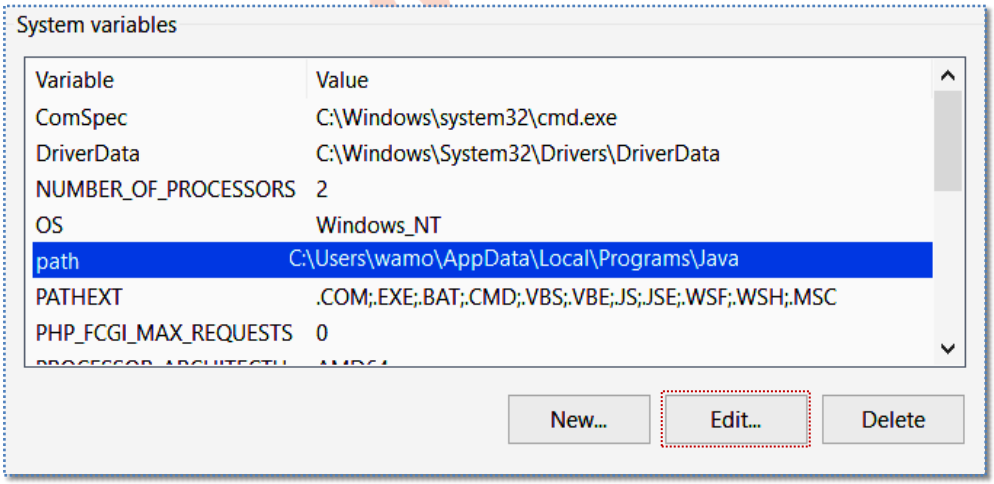


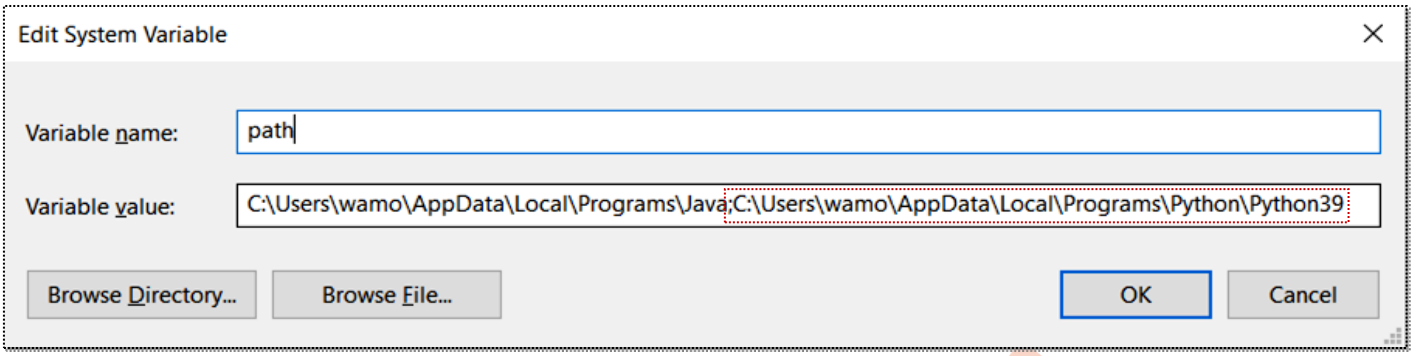
اكتب path في Variable name ثم قم بلصق المسار الذي قمت بنسخه من شريط العنوان في الصفحة السابقة، ثم اضغط OK، ولاحظ ظهور اسم المتغير والعنوان أمامه.

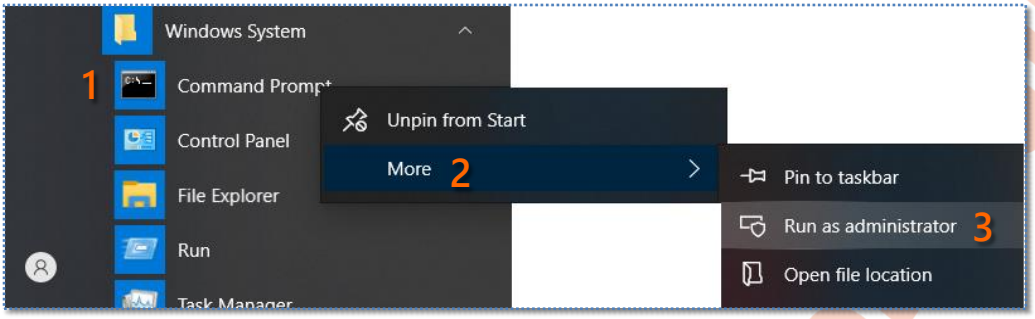




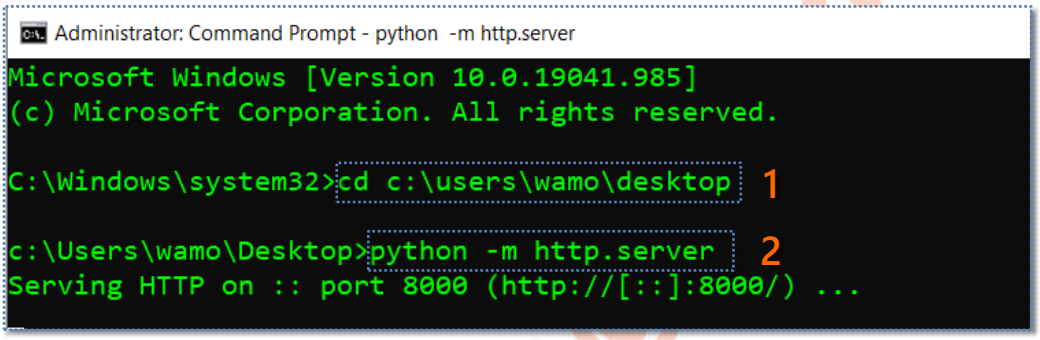
أما إذا كان المتغير path معرفاً وله قيمة، فنختار edit ويتم إضافة فاصلة منقوطة ثم إضافة عنوان أو مسار python له.



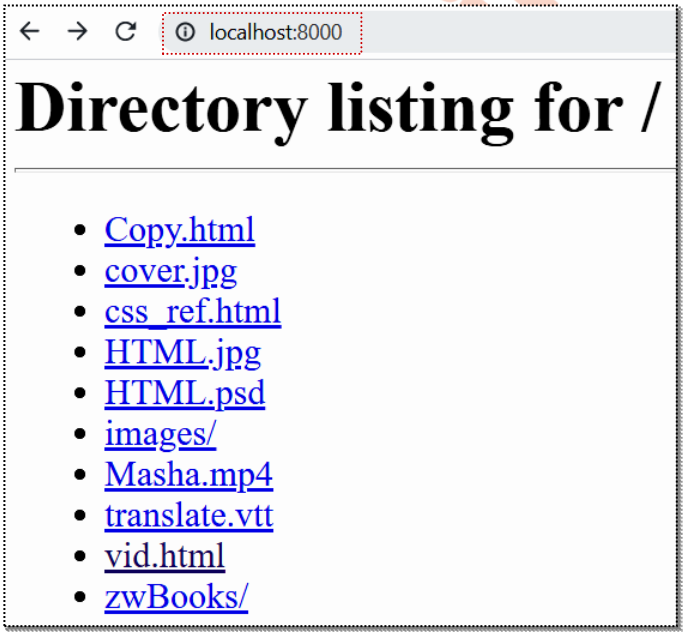




قم بفتح موجه الأوامر cmd كما يظهر بالشكل المقابل.



وجه cmd أو سطر الأوامر إلى مسار الملف الذي تريد فتحه باستخدام الخادم المحلي واكتب الأمر التالي لتشغيل الخادم المحلي http.server python -m ولاحظ أن السيرفر بدأ العمل بالفعل على منفذ 8000 .



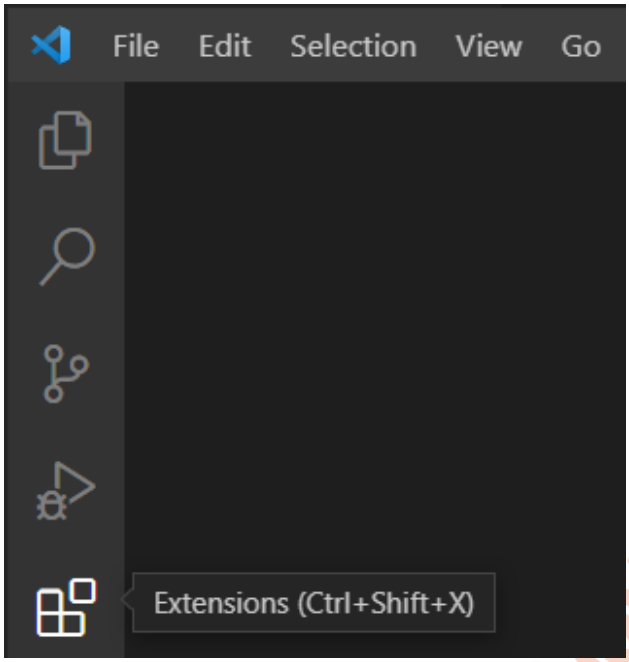
قم بفتح متصفحك واكتب في العنوان localhost:8000 ثم اضغط enter ، ولاحظ ظهور قائمة بجميع ملفاتك في المتصفح، اختر الملف المطلوب تشغيله واضغط عليه.

إذا كان لديك صفحة html ولها الاسم index ، فستفتح مباشرة دون المرور على هذه القائمة، أي أنها ستكون الاختيار الافتراضي للتشغيل من خلال الخادم.

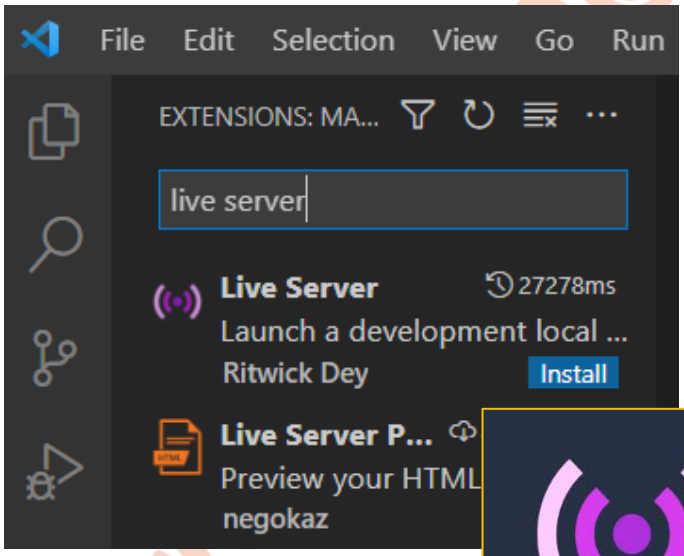
2 – إنشاء خادم محلي عن طريق vs code

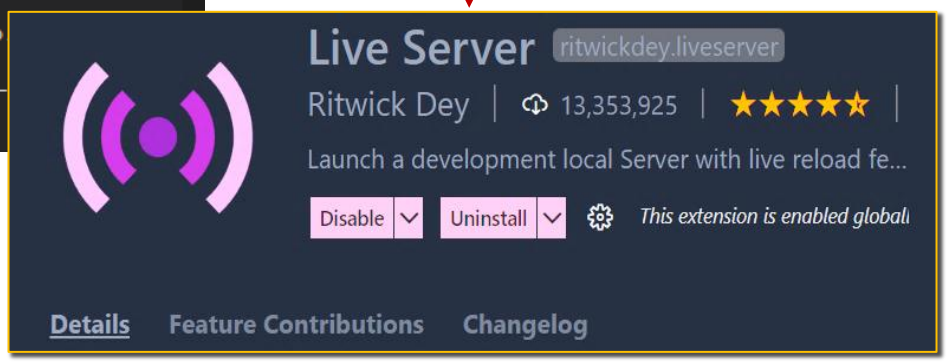
يقوم برنامج visual studio code بتثبيت امتداد live server ، حيث إن للسيرفر المحلي وظيفة أخرى، فهو يمكنك من مشاهدة تحديثات الكود الذي تكتبه بمجرد حفظه، دون عمل refresh للصفحة .

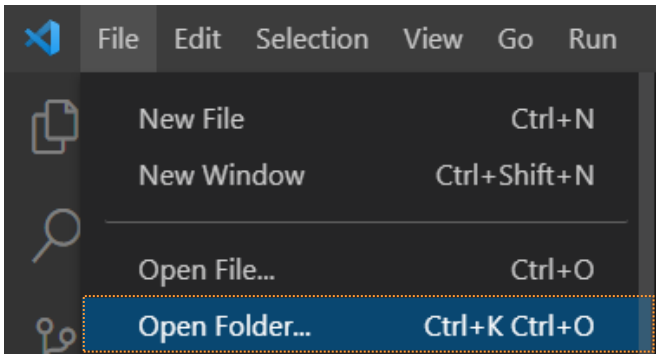
قم بتحميل برنامج vscode من الرابط code.visualstudio.com/Download ثم قم بتثبيته كالمعتاد، ثم قم بفتحه، ومن الشريط الجانبي قم باختيار الخاصية extensions كما بالشكل المقابل.



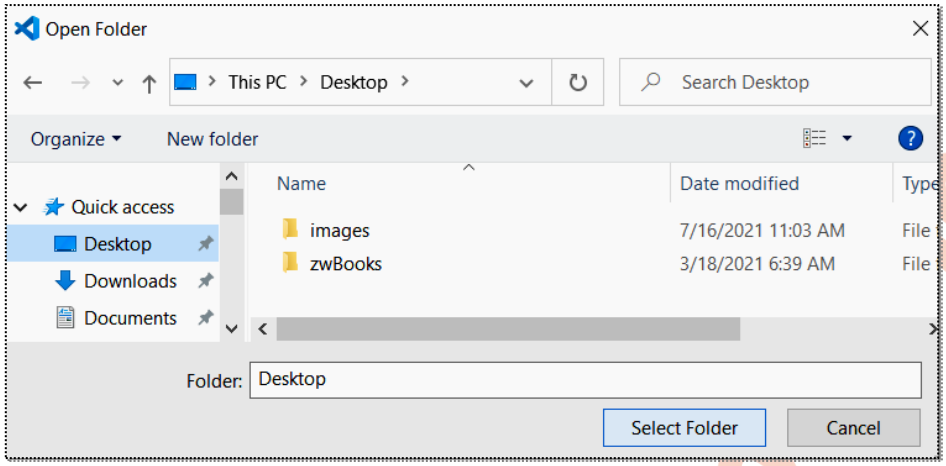
اكتب في خانة البحث live server ثم اضغط install ، ولاحظ قيام البرنامج بتثبيت الامتداد، ليظهر لك بالشكل التالي بعد انتهاء التثبيت:



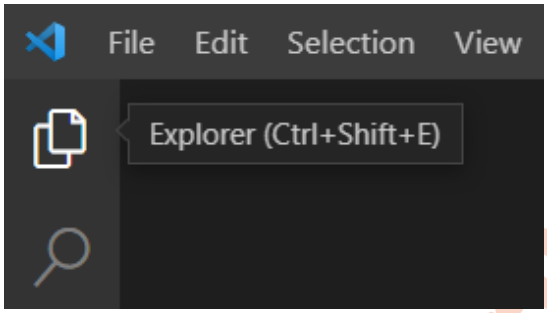




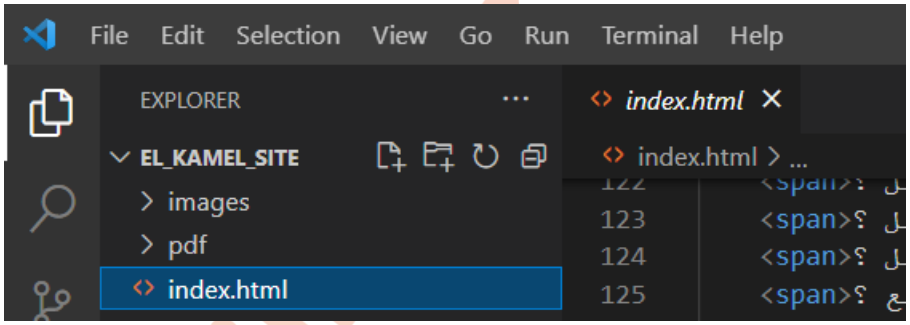
من قائمة file اختر open folder ثم اختر الفولدر الذي يحتوي الملفات التي تريد استعراضها من خلال الخادم المحلي.



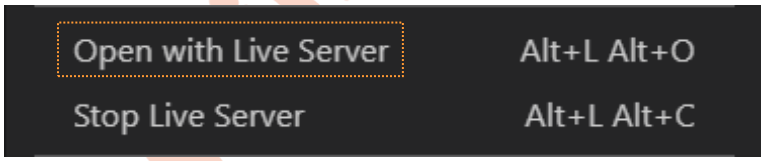
اختر الفولدر ثم اضغط select folder .



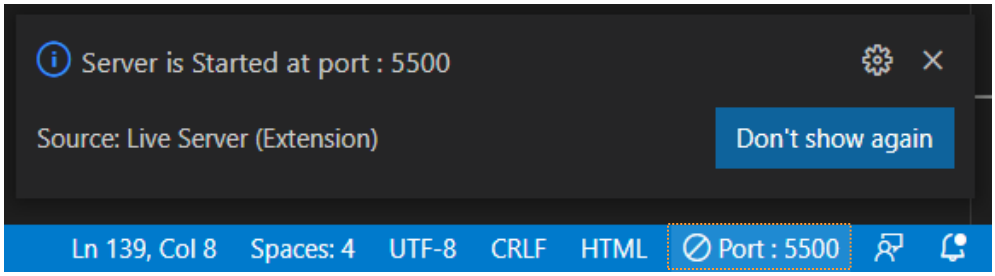
من الشريط الجانبي اختر Explorer ليفتح الفولدر الذي اخترته وبداخله الملفات المطلوبة.



اختر الملفات التي تريد فتحها ثم اضغط مرة واحدة على كل ملف.

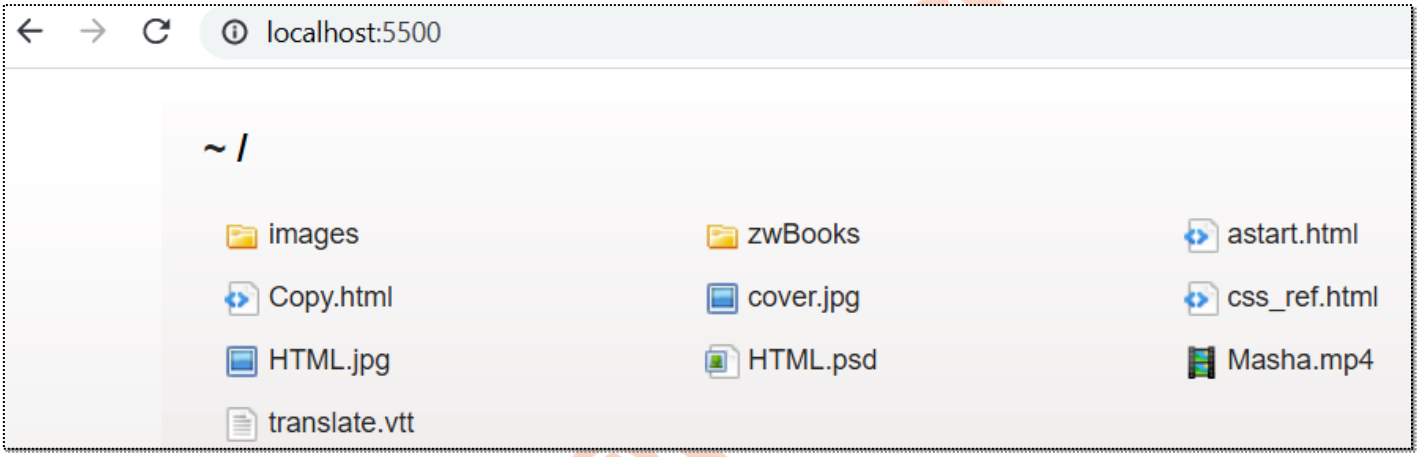


والآن لتشغيل الخادم المحلي، انقر بزر الفأرة الأيمن داخل صفحة html التي تريد استعراضها ومن القائمة اختر open with live server أو اختر من الشريط السفلي Go live



لتتحول Go live إلى Starting ثم Port:5500 وهو المنفذ الذي يعمل عليه الخادم المحلي.

إذا لم تفتح صفحة html مباشرة في متصفحك الافتراضي، قم بفتح متصفحك واكتب localhost:5500 في شريط العنوان ولاحظ ظهور الملفات في الفولدر الذي تم اختياره، اختر الملف المطلوب استعراضه واضغط عليه ليفتح لك في المتصفح.



wailmonir0@protonmail.com

ملحق 3

إنشاء ملف ترجمة أو ملف مصاحب لملف الفيديو

يمكنك إنشاء ملف الترجمة في خطوات بسيطة، كل ما تحتاج إليه هو محرر نصوص لتكتب فيه الترجمة، وكذلك تحتاج إلى تثبيت خادم محلي localhost لتشاهد نتيجة عملك على الطبيعة.

```
WEBVTT - Simple VTT file

00:00:01.000 --> 00:00:03.000
هيا بنا نلعب الكرة، هل تستطيع أن تلعب مثلي

NOTE  Next cue will not be shown

أنا ماهرة جداً
00:00:04.000 --> 00:00:06.000
طبعا أنا  أتمتع بمهارة عالية في كرة القدم
```

1 - يجب أن تتأكد من أنك متصل بخادم محلي localhost قبل تشغيل ملفات الترجمة وشرح كيفية تثبيت خادم محلي ملحق 2 .

2 - افتح ملف جديد في محرر الأكواد الخاص بك، واكتب في السطر الأول WEBVTT .

ويوجد بعض الشروط من الواجب مراعاتها عند إعداد ملف الترجمة وهي:

- الكلمة WEBVTT يجب أن تكون أول ما يكتب في ملف الترجمة، بمعنى أنه يجب ألا يوجد قبلها أي سطر سواء كان فارغاً أو به نص، وألا يوجد قبلها أي مسافات سواء كانت فارغة أو بها نص، حتى وإن كانت مسافة واحدة، وخلاف ذلك سيؤدي إلى عدم عمل ملف الترجمة.

- يجب أن تكتب هذه الكلمة بحروف كبيرة، بمعنى أن كتابتها بحروف صغيرة أو مزيج من

الحروف الصغيرة والكبيرة، سيؤدي إلى عدم عمل ملف الترجمة.

3- يمكنك إضافة header أو رأس للملف، وهو نص يشرح نقطة معينة، ويتم ذلك بإضافة مسافة فارغة بعد كلمة WEBVTT ثم إضافة النص، والذي لن يظهر في سطر الترجمة.

4 - يفضل أن تترك سطراً فارغاً، ثم تبدأ في سطر جديد كتابة التوقيت الذي ستعرض فيه الترجمة على ملف الفيديو.

تنسيق توقيت الترجمة:

- يكون تنسيق توقيت الترجمة بهذا الشكل hh:mm:ss.ttt أو 00:00:00.000 حيث تمثل الأصفار من اليسار إلى اليمين، أولاً الساعات، ثم ثانياً الدقائق، ثم ثالثاً الثواني، ثم رابعاً وأخيراً الجزء من الألف من الثانية.

- يمكن الاكتفاء بالثلاثة أجزاء الأولى، والاستغناء عن الجزء المخصص للجزء من الثانية، وإذا لم تكن في حاجة لجزء الساعات فيمكن أن تستغني عنه، ولكن لا يمكنك الاستغناء عن جزء الدقائق أو الثواني.

- يجب أن نبدأ بتوقيت بداية عرض الترجمة يتبعها توقيت نهاية العرض، على أن يفصل بينهما الرمز "<--"، وبالطبع يجب أن يكون توقيت نهاية العرض أكبر من توقيت بدايته.

- يفضل أن تترك مسافة فارغة بين توقيت البداية والرمز "<--" بعده، وأن تترك مسافة فارغة بين توقيت النهاية والرمز "<--" قبله، ولكن في حالة عدم ترك أي مسافة فالملف سيعمل بطريقة صحيحة.

- يمكنك إضافة توقيت على هيئة timestamp tag في سطر الترجمة، وهو وسم يتم وضعه بين الجمل في سطر ترجمة معين، ويحتوي هذا الوسم على التوقيت بالتنسيق السابق، ويفيد هذا الوسم في تحديد توقيت النص النشط active text ، ويكون هو النص بين هذا الوسم وتوقيت النهاية، ويعرف النص قبل هذا الوسم وتوقيت البداية بالنص السابق أو previous text ، ويطلق على طريقة التوقيت هذه karaoke style text ويكون هذا الوسم بهذا الشكل:

```
00:00:01.000 --> 00:00:04.000
هيا بنا نلعب الكرة، <00:00:03.000> هل تستطيع أن تلعب مثلي
```

5 - في السطر التالي مباشرة اكتب أسطر الترجمة أو cues التي تريد أن تظهر على الفيديو ولا مشكلة في إدخال مسافة فارغة أو أكثر قبل سطر الترجمة.

6 - السطر الفارغ بين سطر التوقيت وسطور الترجمة يعني عدم عرض سطور الترجمة، طالما لم يسبقها سطر التوقيت مباشرة، وتعرض سطور الترجمة المتصلة بسطر التوقيت مباشرة أياً كان عددها، طالما لم يفصل بينها سطر فارغ.

7- يمكنك إضافة تعليق عن طريق إضافة سطر فارغ، ثم في السطر الذي يليه نضيف الكلمة NOTE وإضافة النص المطلوب بعدها والذي لن يظهر في الترجمة.

8 - لا يجب إدخال الرمز "<--" أو الرموز الخاصة (&, <, >) في التعليق والاستعاضة عنها بما يكافئها من رموز html .

9 - قم بحفظ الملف بامتداد file_name.vtt ، وقم بإسناده كقيمة للخاصية src في العنصر <track> ، داخل العنصر <video> .

وبتطبيق ما تناولناه على الكود السابق، لاحظ أن سطر الترجمة الأول يظهر في التوقيت من الثانية 1 إلى الثانية 3، بينما سطر الترجمة الثالث يظهر في التوقيت من الثانية 4 إلى الثانية 6 ، في حين أن السطر الثاني "أنا ماهرة جداً" تم تجاوزه ولم يظهر، لوجود سطر فارغ قبله، وعدم وجود توقيت لعرضه قبله مباشرة.

إعدادات عرض سطر الترجمة cue

يوجد عدة خصائص تتحكم في اتجاه وموضع وحجم سطر الترجمة، وتضاف الخاصية بعد توقيت النهاية، على أن يفصل بينهما مسافة فارغة على الأقل، ويفصل بين الخاصية وقيمتها ":" ويفصل بين الخاصية والتي تليها، مسافة فارغة على الأقل، وهذه الخصائص هي:

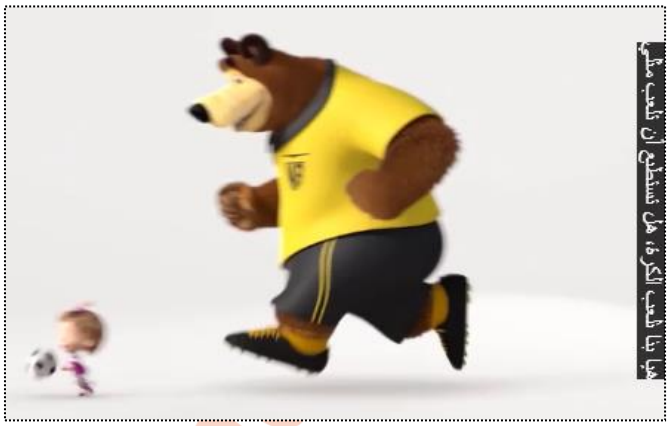
vertical

خاصية تعرض سطر الترجمة رأسياً، ولها قيمتان:

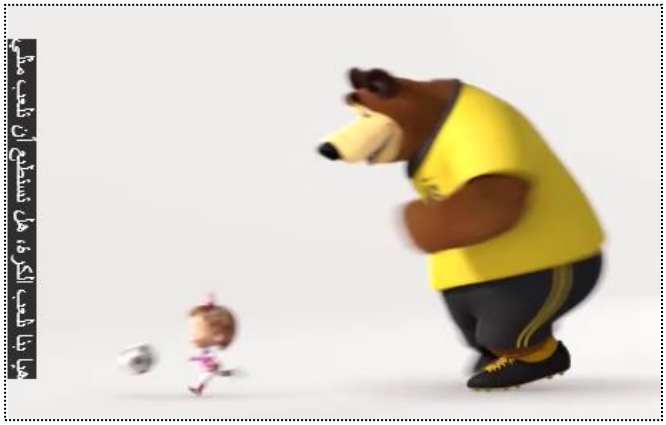
- ***lr*** : قيمة تعني أن يعرض سطر الترجمة من اليسار.
- ***rl*** : قيمة تعني أن يعرض سطر الترجمة من اليمين.

```
WEBVTT – cue settings(vertical)
00:00:01.000 --> 00:00:04.000  vertical:lr
```

هيا بنا نلعب الكرة، هل تستطيع أن تلعب مثلي



rl



lr

وبما أن الاتجاه الرأسي يبدو غير مألوف أو غير مستخدم، ما عدا في اللغات الصينية واليابانية، فسوف نستكمل شرح باقي الخصائص على الاتجاه الأفقي horizontal .

align

خاصية تحدد محازاة النص، ويسند لها ثلاث قيم، فإذا كان النص أفقياً تكون:

- ***start*** : موضع النص الحد الأيمن لملف الفيديو.
- ***center*** : موضع النص في منتصف ملف الفيديو أفقياً.
- ***end*** : موضع النص الحد الأيسر لملف الفيديو.

```
WEBVTT – cue settings(align)

00:00:01.000 --> 00:00:04.000  align:center
هيا بنا نلعب الكرة، هل تستطيع أن تلعب مثلي
```

end

center

start

position

خاصية تحدد موضع سطر الترجمة أفقياً من منتصفه بالنسبة لعرض width ملف الفيديو، وتقبل قيمتها على هيئة نسبة مئوية من بداية عرض أو width ملف الفيديو (من الحد الأيسر) فتكون مثلاً: 25% أو 50% أو 75% .

```
WEBVTT – cue settings(position)

00:00:01.000 --> 00:00:04.000  position:25%
هيا بنا نلعب الكرة، هل تستطيع أن تلعب مثلي
```

75%

25%

line

خاصية تحدد موضع سطر الترجمة من منتصفه رأسياً بالنسبة لارتفاع ملف الفيديو ، وتقبل قيمتها على هيئة نسبة مئوية من بداية ارتفاع height ملف الفيديو (من الحد العلوي) فتكون مثلاً: 10% أو 60% أو 80% .

```
WEBVTT – cue settings(line)

00:00:01.000 --> 00:00:04.000  line:10%
هيا بنا نلعب الكرة، هل تستطيع أن تلعب مثلي
```

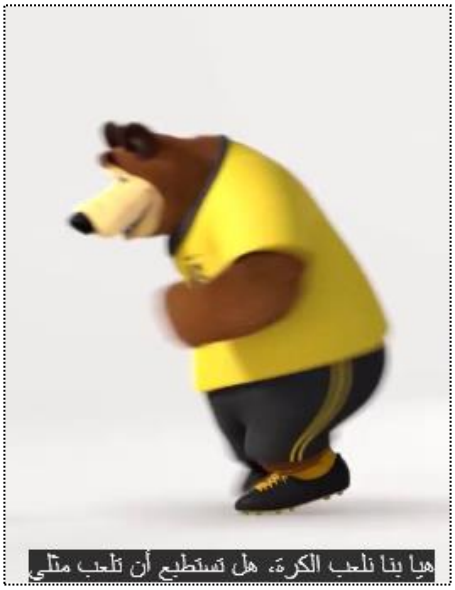
80%

10%

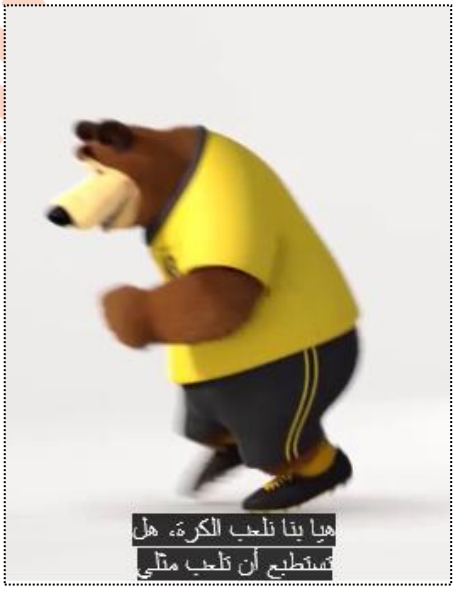
size

خاصية تحدد الحجم الذي سيشغله سطر الترجمة (الأفقي) كنسبة مئوية من عرض ملف الفيديو فتكون مثلاً: 15% أو 25% أو 50% .

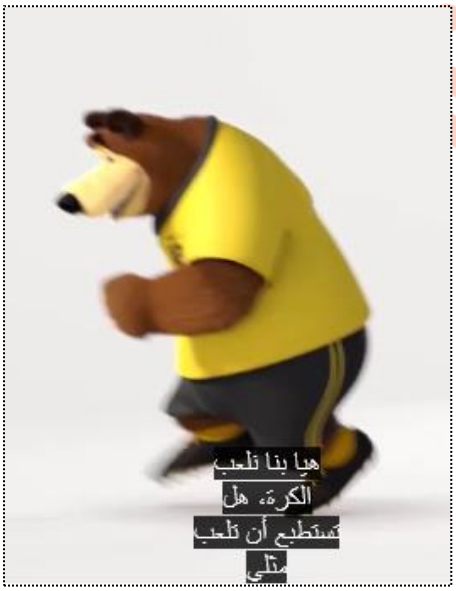
```
WEBVTT – cue settings(size)

00:00:01.000 --> 00:00:04.000  size:15%
هيا بنا نلعب الكرة، هل تستطيع أن تلعب مثلي
```



50%



25%

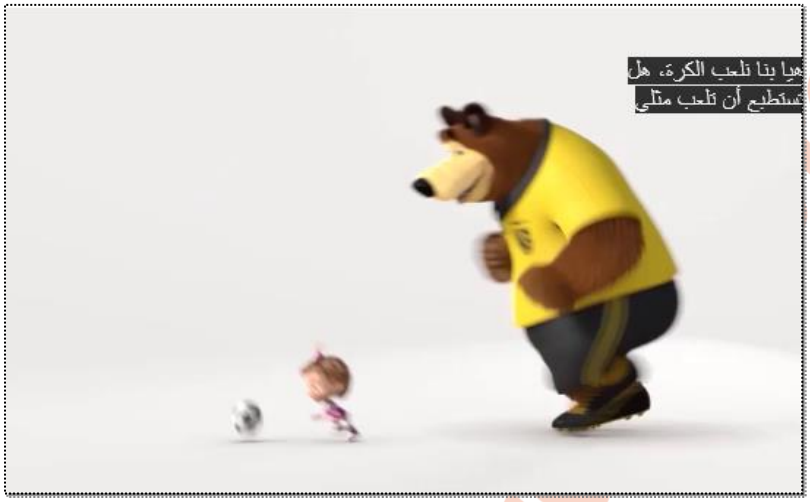


15%

لاحظ أن نسبة عرض أو size نص الترجمة في الشكل السابق، ليست متناسبة مع عرض width ملف الفيديو ، لأنني قمت بقص بعض الأجزاء الزائدة من صورة الفيديو، لوضع الصور الثلاث بجانب بعضها لسهولة المقارنة بينها.

مثال على تطبيق أكثر من خاصية على سطر الترجمة:

```
WEBVTT – cue settings

00:00:01.000 --> 00:00:04.000  size:25%  line:10%  align:start
هيا بنا نلعب الكرة، هل تستطيع أن تلعب مثلي
```



لاحظ أنه يتم الفصل بين كل خاصية والتي تليها بمسافة واحدة فارغة على الأقل، كما يجب أن تلاحظ أنه لا يمكن الفصل بين الخاصية وقيمتها بأي مسافة فارغة، وإلا فلن تعمل الخاصية بالشكل المطلوب.

استخدام عناصر html لتنسيق سطر الترجمة:

يمكننا وضع سطر الترجمة بين وسمي عنصر معين من عناصر html ، للتحكم في تنسيق هذا السطر حسب نوع العنصر، فيمكننا وضع سطر الترجمة بين وسمي العنصر <b> لنحصل على خط سميك، كما يمكننا وضعه بين وسمي العنصر <i> ليكون الخط مائلاً وغيرها من عناصر التحكم في تنسيق الخط.

```
WEBVTT – cue settings(html elements)

00:00:01.000 --> 00:00:04.000
<b><i>هيا بنا نلعب الكرة، هل تستطيع أن تلعب مثلي</i></b>
```

لاحظ أن سطر الترجمة ظهر بخط سميك لوجوده بين وسمي العنصر <b> كما أنه ظهر بخط مائل لوجوده بين وسمي العنصر <i>.

عندما تدرس css قريباً، سوف تتعلم الكثير من الخصائص التي تتحكم في سلوك سطر الترجمة، مثل لونه ولون الخلفية وغيرها من الخصائص.

تم بحمد الله وتوفيقه

اليـوم: السابع من صفر سنة 1443 هـ

الموافق: الرابع عشر من سبتمبر سنة 2021 م

wailmonir0@protonmail.com