بسم اللهالرحمن الرحيم

PeRl SImPlE TuToRIaLs

ماهي لغة البيرل ؟؟؟

هي لغة برمجة عامة قدطورت هذه اللغةلكي يصبح في مقدورهامعالجة النصوص المكتوبة يدويا
يمكن اعتبار هذه الخطوةهي التطورالتايخي لهذه اللغة
اهم الامورالتي تستخدم فيها هي
تطويرصفحات الويب سايت
ادارة الانظمة
برمجةالشبكات
gui تطوير ال
الغرض من هذه الغةهي ان تكون سهلة الاستخدام و متكاملة وتدعم كل من ال procedual
o.o.p
كيف يتم تنفيذ برنامج في لغة البيرل
اولاوقبل ان تكتب البرنامج يجب ان يكون هذاالبرنامج الذي انت بصددان تنفذه يجب ان يكون قابل للتنفيذ يعني ماتجي تكتب اشياء ليس وتريدها تتنفذ
اماعن الية التنفيذ يجب ان تكون كمايلي
command line
perl pro .pl
و كل شي انتهى
ام عن الطرق الاخرى فانا بلنسبةالي احبذهذه الطريقة و على الرغم من هذا فنا سوف اقوم بشرح الطرق الاخرى فيالدروس القادمة
وهناك ايضا ملاحظة هامة جدا هي انهالبرنامج المكتوب بلغلة البيرل سواء كامكتوب من خطوة واحدة او اكثر فانه من غير المهم ان تتم كتابة هذه الدالة main()
وهانك ايضا ملاحظة لابدمن القوقف عليهاهي انه جملة البيرل تنتهي بعلامة الفارزة المنقوطة ;
اماعن التعليقات المكتوبة بلغة البيرل فأننا نكتب التعليق كماهو لكن فقط نضيف عليه ال#

ex:-

CODE

spawn love u all

جملة الطباعة وهي جملة سهلة و بسيطة ونكتب كمايلي

CODE

print "spawn love u";

هذه الحالة تكون عندماتكون الجملة التي نريدان نطبعهاحرف يجب ان تحصر بين علامات الاقتباس
اما عم الارقام فتكون كمايلي

3x:-

CODE

print 1986;

من دون الحاجة الى علامات الاقتباس

---

perl variable type

انواع بيانات البيرل وهي تقسم الى ثلاثة انواع

scalar

arrays

hashes

الان سوف ناخذ النو ع الاول من المتغيرات الا وهو

scalar
اذا اخذنامعنى هذه الكلمة في الغة العربيةفنلاحظ ان هذه الكلمةهي تعني
الموجهات او المدرجات
الموجهات تقدم نوع واحدمن البيانات وهي تتمثل كمايلي

CODE

```
my $name = "m_spawn";
```

CODE

```
my $friend = "all geeks";
```

ان قيمالموجهات هذه من الممكن ان تكون
intgers من المكن ان تكون
قيمالموجهات من الممكن ان تكون فلوت اي اعداد عشرية
string قيمالموجهات منة الممكن ان تون سلاسل حرفية او مليعرف ب
المميزة وهي scalars هنالك عدد من ال
line of noise
punctution
هذه الانواع من المتغيرات التي ذكرناهامن الممكن ان يتم استخدامهافي كافة
انواعالاهداف في لغة البيرل
لكن الى حدالان نحن نتقدم في لغة البيرل و لانواجهصعوبات الان قدحان
الوقت الذي يجب فيه ان نركزعلى هذه المتغير المهم الاوهو
$_;
وهو المتغيرالذي يطلق عليه برمجيافي لغة البيرل
the default variable
والذي يستخدم عادة في ال
the default argument

الان مبدئيا اصبحنلعرف ماهمية هذاالمتغير و اين يتم استخدامه
وهذاالمتغير يجهز ضمنيا
by the certain looping constructors

---

arrays

النوعه الثاني من المتغيرات الذي يختلف عن سابقه في مايلي
انه يقدم مجموعة من المتغيرات
صيغته العامة هي

CODE

```
my @xxx = ("xxx", "xxx");
```

الان ناخذمثال على هذاالنوع من انواع البيانات

CODE

```
my @friend = ("geek 1", "geek2", "all geeks");
```

الان اخذنامثال عليه الان راح نحلل هذاالكود
1- z3r0 ان المصفوفة دائماتكون معنونة بالعنوان
2- اذا اردناان نطبع مقدار نعمل مايلي

CODE

```
print  my $friend[0];
```

وسوف تلاحظ النتيجة الاتية

geek1

الان ناتي و ناخنالمتغيرات المميزةفي المصفوفات وهي المتغير الهام جداو السهل فينفس الوقت

وهذاالمتغير هو

$#array

وتكون صيغته العامةهي

CODE

print $array[$#array]

هذاالمتغيرتكون هذه صيغته العامة و هو يستخدمفي الحالات التي تكون فيها اعدادالمتغيرات في المصفوفة كبيرة جدا لنايتم استخدام هذاالمتغيرلكي يقوم باعطائنا المتغير الاخير !!!في المصفوفة اذن وظيفة هذاالمتغير هو اعطاء مقدار المتغير الاخيرفي المصفوفة

ملاحظة هامة ان المتغير

$#array

هو يعطيك مقدار المتغير الاخيروليس يطبعلك موقعه

لذايجب الانتباه

هنالك انواع اخرى من المتغيرات المميزةالاوهي

1-@argv

(the command line argument 2 ur script)

2-@_

(the argument passed to sub routine)

HASEHES

النوع الثالث من انواعالمتغيرات في لغة البيرل

ماهو الهاش؟؟!!؟

يمكن القول ان الهاش يقدملك كل شئ وقرينه ايان المتغيرات في تكون على شكل ازواج

يعني كل شي وقرينه مايمكن ان نقول مثلا ان
لون البطيخ احمر
لون العنب اسود وهكذا
يتم تم تمثيلها برمجيا كمايلي
ناخذمثال على هذه الحالة

CODE

```
my %colur = ("apple", "red", "blackberry", "black");
```

وحالات ال
=>
مسموحة يعني من الممكن تمثيل العبارة اعلاه كمايلي

CODE

```
my %fruit_color = (
apple  => "red",
banana => "yellow",
);
```

وهكذا
لكن اذا اردنانطبع مقدار موجودفيالهاش كيف تتم العملية ؟؟؟
ان العملية تتم كمايلي

CODE

```
print $colur{"apple"};
```

عند تنفيذ هذه العبارة كمايلي

red

الان نجي ناخذالدوال المميزة في هذاالنوع من المتغيرات
الاوهو
%env
وهي مختصر لكلمة
environment
وهي تعني محيط او بيئة

---

perl data structures

الان فيالدرس الثاني سوف نتناول موصوع سهل ومهم لكل من يريد ان يبرمج برنامج في لغة البيرل
ول شي راح ناخذالمتغيرات
اول شي ان المتغيرات منالمعروف انهامن الثوابت المهمة في كل لغةوفي كل لغة تكون لهاصيغة عامة اماالصيغة العامة لهافي لغتناهي

CODE

```
my $var="value";
```

; هذه هي الصيغة العامة لهاو لكن هالك نقطة مهمة هي انه ال
مهمة جداوضرورية لانه اذاتم تنفيذالجملة من دون هذه العلامة فان الحاسبة لن تعلم بانتهاء الجملة و لذلسوف تمر بمرحلة من الارتباك وسوف تظهر رسالة خطأ للمستخدم
وهناك صيغةاخرى لتعريف المتغيرهي
CODE

```
$var="value"
```

اذن عرفنا انه كلمة
my
هي كلمة من الممكن ان يتم الاستغناء عنهايعني اذاتم تنفيذالجملة البرمجية المكتوبة بلغة البيرل من دون هذه الكلمة فان النتيجة هينفس النتيجة الناتجة منم الجملة المحتوية على كلمة ال my
---
الان بعد اخذنا نظرة عامة على المتغيرات العامة في لغة البيرل الان راح ناخذ conditional and looping constrctures
طبعا لفة البيرل مثل باقي اللغات تحتوي على جمل شرطية بكل حالاتها و انواعها
لكن هانلئبعض النسخ منزوعة منها جملة ال case/switch
و لكنها متوافرة في النسخ الاحدث مثل الاصدار ال 5.8 و الاصدارات الاعلى منه
صيغة جملةالشرط في لغة البيرل هي كمايلي
CODE
```
if (conditions){
some commands
}
else if (other commands){
some commands
else(condition)
```

ولكن طبعا كماهو الحال مع التطورالبرمجي المعهودفي الانظمة المفتوحة المصدرنلاحظ هنالك حالة معاكسة لهذه الحالة وهيحالة النقيض لهاو هي كمايلي

CODE

```
unless (conditions){
some commands
}
```

طبعاهنالك موضوع مهم يجب ان انبه اليه هو انهالشرط مهماكان صغيرااو قليل الحجم او حتى مهماكان غير مهم لكن يجب ان يتم وضعه بينالاقواس والا فان البرنامج سوف لن ينفذ

الان راح ناخذمثال بسيط على برنامج يحتوي على جملة ال if

CODE

```
$a=55;
if($a<85){
print "spawn love u"
}
```

---

while statement

تكون الصيغة العامة لهذه العبارة هي كمايلي

CODE

```
while(expression){
some commands
}
```

وكذلك هنالك الحالة المعاكسة لهذه الحالة هي حالة ال

CODE

```
unless(expression)
{
some commands
}
```

---

for statement

الان سوف ناخذ عبارة برمجية تعتبر من اهم العبارت البرمجية الموجودة التركيب البرمجي و الصيغة العامة لهذه العبارة هي كمايلي

CODE

```
for($i=0;$i<$max;$i++){
some commands
}
```

الان سوف ناخذ مثال برمجي على هذه الحالة

CODE

```
$a=33;
for($a=0;$a<32;$a++){
print $a;
}
```

---

build in function

عندما تنزل نظام الى حاسبتك نظام مفتوح المصدر تنزل معه عدة من التطبيقات و ينزل معه هذه اللغة و من الطبيعي فانه كل لغة برمجية يكون لها نوعان من الداول

النوع الاول هي النوع النيي يعرفهالمبرمج اثناء كتابة برنامجه
الدوال المضمنة و المبنية داخل اللغة
ونحن الان قد اخذنابالفعل عدد من هذه الداوال منها جملة الطباعة وهي
print

العمليات الرياضية في لغة البيرل

Mathematics
الجمع addition +
الطرح subtraction -
الضرب multiplication *
القسمة division %

المقارنات العددية numeric compartment

يساوي equal to ==
لايساوي not equal to =!
اقل من less than <
اقل من او يساوي less than or equal to =<
اكبر من greater than >
اكبر من او يساوي greater than or equal to >=

المقارنة بين السلاسل الحرفية sting copmeration

المساواة eq equalty
عدم المساواة NE INEQUALITY
اقل من LT LESS THAN
اكبر من GT GREATER THAN
اصغر من او يساوي LE LES THAN OR EQUAL TO
اكبر من او يساوي GE GREATER THAN OR EQUAL TO

وطبعانحن الان نورد هذهالعمليات ونمر عليها مرور الكرام لكن بعدان نتوغل قليلافي لغة البيرلسوف نورد كل عملية من هذهالعمليات بشكل مفصل

BOOLEAN LOGIC

و && AND
او || OR
نفي ! not

منوعات miscellaneous
مساواة == assignment
ربط السلاسل الحرفية string contact .
x string multiplication
ان شاء اللهفهمتم الدرس

Perl reference tutorial

ان لغة البيرل واحدة من اهماللغات الموجدوة حاليا نظرلمدى سهولتهاولمدى مرونتهافي التعامل مع ال
complex data structures
وبماان لغة البيرل لهاالقدرة على التعامل
complex data structures
مثل
المصفوفةذات البعدالثنائي
الهاشات المتعددة او المتشعبة
وان لغة البيرل كان لابد لهامن ان تقدمبعض الخائص لدعم كل هذهالامكانيات
او الذي يعرف في اللغة العربية reference لذاقدمت لغة البيرل شئ يدعى ب
بالمصدر ان استخدام هذاالمصدر هو اشبه باستخدام المفتاح للتعامل مع
complex data structures
الان قديطرح احدسؤال و يقول لماذا قد يحاتج احد لا وهو لماذا قد يحتاج
complex data structures شخص الى هذه
والجواب على هكذاسوال هو انه المبرمجين السابقين في لغة البيرلو بالاخص
لغة البيرل ذات الاصدار الرابع
ان من اصعب الاشياء لديهم كانت هي الهاشات التي كانت بشكل قائمة وكيفية
تمثيلها وتقديمهاولكن الان ايضا قديسأل احد لما قد يجتاجشخص الى هذه الى
استعمال هذهالهاشات وكما ذكرنا سابقاان فكرة عملالهاش هي انه هو كل
ونظيرة يعني
تفاح احمر
موز اصفر
وهذه اشارة الى انه لون التفاح احمرولون الموز اخضر
والان ناخذمثال عمليلنفرض كان لديك هذاالهاش كيف منتسوف تحله
Baghdad-> Iraq
Mosul -> Iraq

Cairo -> Egypt

Gaza -> Palastine

Jerusalem->Palastine

الان الغاية من هذاالمقطع البرمجي لو كان لديك هكذا فهل كنتسوف تحله بهذه الطريقة الطويلة الجواب لاطبعاو الحل الطبيعي على هكذا مقطع هو انه نعمل على ترتيب المقاطع كمايلي

Iraq: Mosul,Baghdad

Egypt: Cairo

Palastine : Gaza ,jerusalem

هنلسوف يكون الحل على هكذاسوال الان يجب انعرف اذاكانت قيمالهاش من غير الممكن ان تكون قوائم يمكن القول في هذه الحال ان الكود سوف لن ينفذ في لغة البيرل ذات الاصدار الرابع الهاشات كانت من غير المككن ان تكون سوى على شكل سترينك وفي هذه الحالة لو كنت من مشنتخدمين لغة البيرل ذات الاصدار الرابع كان عليه ان يدمج كل هذه المدن في سترينك واحدةفقط بطريقة ما وعندمايحين الوقت لكي الناتج او تطبعه عليك ان تكسر هذه ال سترينك الى قائمة ثم تقوم بترتيب هذه العناصر ثم اعادتهاالى سترينك واحدة وهنايمكن اعتباره الفوضى العارمة بذاتها

THE SOLUTION

مع تطور الزمن تقدمت لغة البيرل وقدمت لنا الاصدارالخامس والذي فيه hashes value must be a scalar value

Reference الحل لهذه الحالة هو ال

قديسال احدماهو هذاال Reference الان بعدكل هذاالحديث عن هذا ال Reference

Reference اذن يمكن تعريف ال

التي من الممكن ان تشيرالى المصفوفة بكاملهااو scalar على انه قيمة من نوع من الممكن ان تشيرالىالهاش بكامله

في لغةالبيرل هي اشبة بالاسمللمصفوفات والهاشات Reference ال يشيرالى شي واحد فقط Reference ال

الى مصفوفة يمكن ان تشيرالى المصفوفة بكاملهامن Reference واذاكان لديك الىهاش من الممكن ان تشيرالىالهاش كله Reference خلاله واذا كان لديك Reference من خلال ال
array و انت لايمكنك ان تملكهاش تكون القيم التي بداخله من نوع لكن كام ذكرنا لكم ان ال scalar قيمالهاش يجب فقط ان تكون من نوع من الممكن ان يشيرالى مصفوفة كاملة وكما ذكرنا ان قيم ال Reference لذا من الممكن الحصول على لكن من الممكن الحصول scalar هي Reference على
a hash of
references to arrays,

Syntax

references الان سوف نتكلم عن طرق الحصول على
الطريقةالاولى هي
اذا علامة
\
في بداية المتغير او امام المتغيرفانك في مثل هذه الحالةسوف تحصل على لذلك المتغير references
EX:-
$aref=\@array 1:- $aref now holds a reference to @array
$href=\%hash 2:-$href now holds a reference to %hash
$sref=\$scalar 3:-$sref now holds a reference to $scalar
الى مصفوفة reference الان تمسك -:1
الىهاش reference الان تمسك $href -:2
scalar الى reference الان تمسك $sref -:3
من الطريقةالاولى اصبحت سهلة ان reference و الان اتوقع طيريقة عمل ال شاء الله

$aref , $href في المتغير مثل reference ولكن مايحدث عندمايخزن ال في هذه الحالة من الممكن ان تقوم بعملية نسخه تماما كماهو الحال مع اي قيمة من نوع scalar

Ex:-

$xy = $aref; 1:- $xy now holds a reference to @array

$p[3] = $href; 2:-$p[3] now holds a reference to %hash

$z = $p[3]; 3:- $z now holds a reference to %hash

1:- $xy الان تحمل reference الى مصفوفة

2:- $p[3] الان تحمل reference الىهاش

3:- $z الان تحمل reference الىهاش

هذه الامثلة تعلمالشخص كيف يعمل reference

الى متغير يحمل اسم

MaKe RulE TwO

الان بعدان تناولنا الطريقة الاولى و اصبحت مفهومة الى الجميع باذن الله

الان شوف تناول الطريقة الثانية

و هي ال

جديدة و عيالناتج الى تلك anonymous array وهي تصنع [ITEMS]

الى تلك المصفوفة references

الى ذلك references وتعيدال anonymous hash تصنع [ITEMS] و ال الهاش

EX:-

$href = { spawn => 5 , perl =>4};*

$href now holds a references to a hash

$aref = [“spawn”, “soqor” ,”securitygurus”;];

$aref now holds a references to an array

*N0T3 :-1

بس احب الى ان انبه انه في مثال الهاش انه عدا الاحرف هو الذي قمت متكونة من 5 احرف وكنا الحال مع كلمة بيرل spawn باستخدامه يعني انه كلمة فهي مكونة فقط من الربع احرف لذا حبيت ان انبه

*N0T3:-2

عندي ملاحظة ثانية لكنها يمكن القول انه مجرد تنبيه مو اكثر هو انه فارغ anonymous hash عندما تكتب بلا عناصر اي فارغة فانها تصنع {} -:1
anonymous array عندما تكتب فارغة بدون اي عناصر فانها تصنع [] -:2
فارغة
وهذه ملاحظة هامة نوعا ما

UsIng ReFeReNcEs

الان قد يسأل شخص ماهي الفائدة المرجوة من او ما 1 ايمكن ان نعمل حالما نحصل عليه ولقد علمنا مسبقا انه عبارة REFERENCES باستخدام ال وانه من الممكن ان يتم استعادته في اي وقت مثل اي scalar عن قيمة من نوع و الان سوف نستعمل طريقتان من اجل استخدامه scalar قيمة من نوع

UsE RuLe 1

في الاقواس المتعرجة {} هذه الاقواس array references دائما يمكنك باستخدام اعني كمثال يعني

This

"@{aref}"

instead of

$aref

SomE ExampLes

1:-@a	@{$aref}	an array

2:-reverse @a	reverse @{aref}	reverse an array
3:-$a[3]	${$aref}[3]	an element of the array
4:-$a[3]=[144]	${$aref}[3]=[144]	Assigning an element

مصفوفة عادية -:1

تعني عمس او قلب في اللغة العربية reverse عكس المصوفة حيث كلمة -:2

عنصر في المصوفة-:3

تحديد عنصر في المصوفة -:4

*N0T3:-3

الان على الجانيبين توجد تعابير و ان االتعابير التي على الطرف الايمن هي ذات نفسها التعابير التي على الطرف الايسر هي ذات نفسها الموجدة على الطرف الايمن الان كيف لو كان نريدان نطبق بعض العبارت البرمجية على و بشكل في الوقت الذي انا كل ماملك هو عبارة عن ال (Loop) خاص عبارات ال اذا كان لديك هكذا سوال فان الجواب عليه يكون كمايلي reference في حالة ال array

```
for  my $element (@array){
xxxxxx;
}
```

نقوم بعمل تعديل وهو كما يلي reference هذه في الحالات العادية لكن مع ال

```
for my $element (@{$aref}){
xxxxxxx;
}
```

ونفس السؤال مع حالة الهاش عندما نقول كيف نطبع هاش بينما نحن كل ما نملك هو reference

الحل في هذه الحالة

```
for my $keys (keys %hash){
xxxxxxx;
}
```

هذه في الحالة العادية و الان سوف نرى ماذا سوف يحدث في حالة ال

reference

```
for my $keys (%{$herf}){
print "$key => ${$href}{$key}\n";
}
```

reference الان بعد ان اخذنا الطريقة الاولى من طرق صنع ال
الان نأتي الى الطريقة الثانية

Use Rule 2

ان استخدام الطريقة الاولى هو حقيقة كل ما تريد لانها كل ماتحتاج تقريبا لانها
reference تخبرك كيف تماما ماذا تفعل او تريد ان تفعل مع ال
وان استخدام الطريقة الاولى والتدوين فيها يمكن القول انه مزعج لذا فانه في
هذه الطريقة يوجد اختصار
في الطريقة الاولى كنا نقوم بعمل هذه

```
"${$ref}[3]"
```

الان وفي هذه الطريقة الجديدة سوف نقوم بعمل هذا التغير

```
$aref->[3]
```

هي العنصر $aref->[3] للمصفوفة فأن reference تحمل ل $aref اذا كانت
الرابع من المصفوفة وقلنا العنصر الرابع لانه ذكرنا في دروس ماضية انه تعداد
العنصر في المصفوفة يبدأ من الرقم 0
هذا التغيير بدلا من الطريقة المعقدة الموجودة في الاعلى وان كلا المقدارين هما
نفس الشئ يعني الان بعد انا قمنا بعملية التغيير فأن كل من المقدارين الذين
في الاعلى والذي في الاسفل كلاهما نفس المقدار هذه الحالة كانت مع
المصفوفة الان سوف ننتقل الى الهاش

```
${$href}->{red}
```

هذه كانت الطريقة المتبعة في الطريقة الاولى الان وبعد التغيير سوف تتحول
الى

```
$href->{red}
```

وايضا الان كل من المقدراين هما انفس الشي و ان اختلفت طرق الكاتبة الان سوف نأخذ مثال سريع لترى كل هناكم هو مفيد

```
@spawn  = ( [1, 2, 3],
            [4, 5, 6],
            [7, 8, 9]
          );
```

هي عبارة عن مصفوفة مكونة من ثلاث عناصر وكل واحد هو @spawn الى مصفوفة reference

التي تتكون من 3 عناصر وهي references هو واحد من هذه ال $spawn[1]

تشير الى مصفوفة وهذه المصفوفة مكونة من 3 عناصر وطبقا للطريقة الثانية فانه من الممكن ان نحصل على عنصر من المصفوفة عند طباعة مايلي

```
$spawn[1]->1;
```

عند طباعة هذه الجملة سوف نحصل على المقدار 5 لانه ثاني عنصر من المصفوفة 1 لذا نحن الان لدينا شئ يشبة المصوفة ذات البعد الثنائي

ان عملية التدوين لاتزال صعبة ومزعجة الى حد الان لايمكن القول انه هناك اختصار واحد بعد وهو

اذا كانت لديك هذه المصوفة

```
@spawn  = ( [1, 2, 3],
            [4, 5, 6],
            [7, 8, 9]
          );
```

بدلا من ان تقوم بطباعة هذه الجملة من اجل الحصول على رقم 5 مثلا

```
print $spawn[1]->[1];
```

من الممكن ان نقوم بعمل مايلي من اجل الحصول على الناتج ولكن بخطة اقصر وهي

```
print $spawn[1][1];
```

وعند تنفيذهذه الجملةسوف نحصل علىنفس الناتج في كلتاالحالتين ويمكنك ان تتاكد بنفسك

الان وبعد ان انتهى هنالدرس البسيط في ال
اتمنى ان اكون قد وفقت في تقديم كل ماهو مفيد وجيد REFERENCES

GR33Tz To :

HACKERS PAL (Th3 Und3r W0rld M45t3r)

GaCkEr (H0Pl355 C453)

And SpEcial Gr33tz to

www.soqor.net

www.securitygurus.net

ThIs SiMpLe TuTorIal is wRot3 By

M_sPaWn

Mail Me:- mahmoud_najafy@hotmail.com

THX A LOT 4 READING MY SIMPLE TUTORIAL

تم بحمد الله