

بسم الله الرحمن الرحيم  
دورة في الباتش سكريبت .. إعداد معزز محمد حامد حسين الجعل

### أولاً: مقدمة حول الباتش سكريبت :

هي لغة بسيطة ، تعتمد في تركيبها على أوامر موجودة أصلاً في نظام التشغيل ، تستدعي هذه الاوامر على شكل سلاسل متتابعة ، مما يختصر الوقت و الجهد ، فعلاً ، لو أننا في شبكة ، و على كل حاسب مجلدات موجودة في المستندات(Mydocs) و موجودة في C: و نريد نقل كل هذه الملفات إلى D: لاننا سنمسخ الأجهزة جميعاً ، سيكون من الصعب علينا التوجه الى كل حاسب و القيام بهذه الخطوات ، مرة تلو الأخرى ، فبدل ذلك ، نصنع ملف باتش سكريبت ، يقوم بذلك بمجرد تشغيله ، و نشاركه على جميع الحواسيب ، و بدل أن نذهب الى الحواسيب واحداً واحداً ، نطلب من العمال تشغيل هذا الباتش من النت وورك ، و بذلك سنوفر الكثير من الوقت.

أي أن الهدف الرئيسي منها ، الاختصار في تنفيذ المهام ، و توفير الجهد و الوقت

و تسهيلات لذلك ، ارفقت لكم ملف shell\_batch ، و هو أصلاً ملف باتش سكريبت يقوم باضافة الخيار New MS-DOS batch script الى قائمة new في الوندوز، و هذا دليل على جدوى الباتش سكريبت

### الدرس الأول : برنامج بسيط (Hello Paint):

في هذا الدرس ، سنعمل ملف باتش ، يقوم بتشغيل برنامج الرسام ، و يكتب عبارة (Hello paint) ، و عند انتهاء برنامج الرسام ، سيكتب Good bye paint. Enjoy real\_mtz turtoials: و خلال الدرس سنتعارف على الاوامر الخاصة بالتعامل مع الواجهة البسيطة للمستخدم ، من:

1-البداية

2- كيفية اظهار مخرجات على الشاشة .

3- و تغيير لون موجه الاوامر ،

4- استدعاء برامج مدمجة في الوندوز (اي موجودة في مجلد الوندوز windows او system32)

1-كيف تبدأ ،

سنستخدم محرر النصوص الموجود مع الوندوز (notepad) ، لكتابة السكريبتات ، و عند الحفظ نلصق الملف باحتداد bat. او .

cmd اي نضيفهما بعد الاسم الذي نريده

الآن لنبدأ مع الباتش فايل الصغير الذي نريد ان نصنعه كاول مثال ، شغل ال notepad (لو لا نعلم كيف تشغيلها ، اعتقد ان الموضوع

لا يناسبك) و اكتب الآتي

echo off@

color 1f

title hello paint batch by real\_mtz

echo helloooo Mr.paint

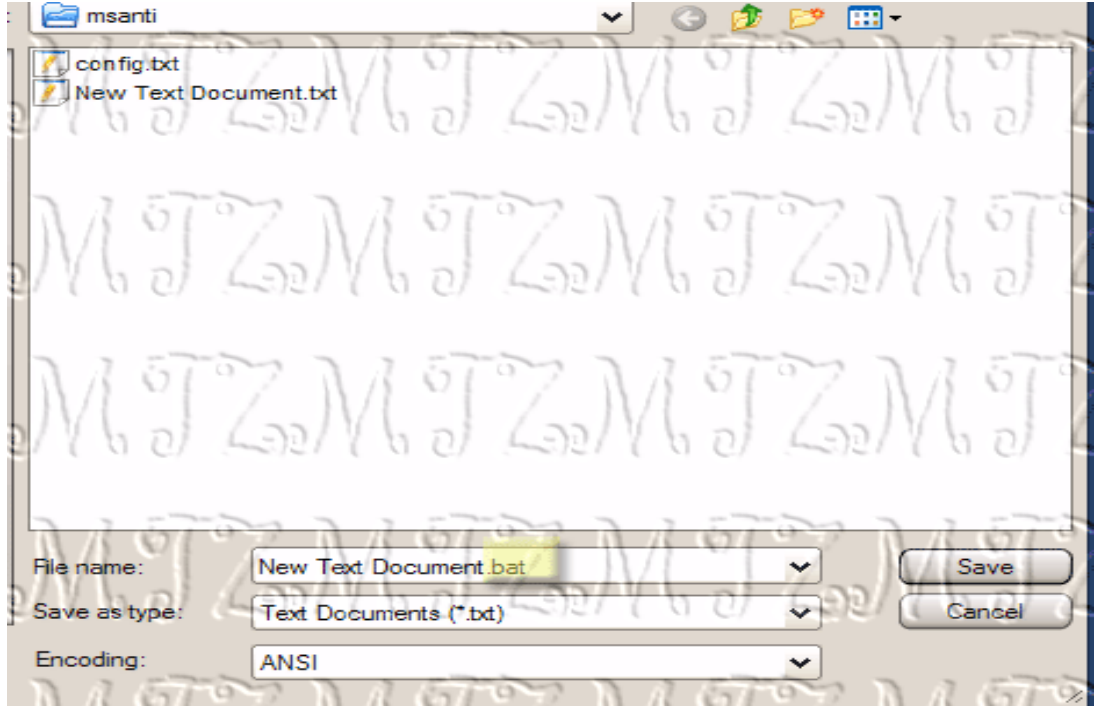
c:\windows\system32\mspaint.exe

echo goodbye paint

pause

Exit

، ثم احفظه باحتداد bat. كما في الصورة



ثم شغله ، و لاحظ ما يقوم به البرنامج ، و الآن لنحلل ما يقوم به ، خطوة بخطوة:  
في السطر الاول نجد الأمر @echo off

شرح @echo off:

وظيفة هذا الامر هو ان يجبر موجه الاوامر بان يتجاهل الكلام المكتوب (الاوامر) أي لو اننا كتبنا في الملف اسم برنامج او ملف او اي شيء غير النصوص الظاهرة التي نحدد نحن ظهورها ، لن تظهر .

**title**

هذا الامر يحدد عنوان النافذة التي سيعمل فيها الباتش فايل

**echo**

echo ALoooooooooooo هذا الامر وظيفته اظهار الكلام الذي نختاره على الشاشة ن فعندما نكتب  
aloooooooooooooooo سيطلع كلمة

و هو الجزء الوحيد الذي يظهر ، من الكلام الذي نكتبه في الباتش فايل ما دمنا استخدمنا @echo off  
**c:\windows\system32\mspaint.exe**

هذا اسم البرنامج الذي شغلناه ، فكما قلنا ان اساس الباتش سكريبت ، الاعتماد على مكونات النظام ،

فنحن شغلنا الملف من النظام نفسه ، ولو انا كتبنا اسم اي ملف آخر فان هذا الملف سيعمل ، باختصار ، بإمكاننا تبديل انفسنا اثناء كتابة

الباتش فايل ، اننا نتعامل مع الدوس ، فائناء تعاملنا مع الدوس نكتب اسم البرنامج ليشتغل ، و كذلك في الباتش فايل ، لكن نذكر ان

الأوامر تعالج سطرًا سطرًا فلو اننا اردنا تشغيل اكثر من برنامج واحدًا تلو الآخر ، سنكتب الاسماء كل واحد في سطر ، و بالتالي ، تعمل

حسب الترتيب اللتي هي مرتبة في دايل الباتش فايل، قد يتساءل أحدكم ، لو انني أشغل الباتش من مصدر متغير المسار (قرص ليزري)

كيف ساجدد مسار البرنامج ، الجواب ببساطة هو انا نكتب اسم الملف فقط دون المسار ، و تراعي ان يكون الملف الذي استدعيته من

ملف الباتش في نفس مسار الباتش فايل ، أي بجانبه ،

و كذلك لو اردت الباتش فايل أن يتعامل مع ملفات النظام او يشغل اجهدها بإمكانك استعمال متغيرات النظام التي سأعطيهها لاحقاً ، او ان

نكتب اسم البرنامج ليعمل فوراً ، كما في dxdiag.

**echo**

مرة اخرى استخدمنا echo لاطهار كلمات على الشاشة

**pause**

هذا الامر يوقف تنفيذ الباتش فايل ، الى ان يقوم المستخدم بضغط اي زر على الكيبورد ، ثم يتابع النظام تنفيذ محتويات هذا الملف بعد

ان يضغط المستخدم اي زر .

**exit**

هذا الامر كما هو واضح ، ينهي الباتش فايل و يغلقه

لاحظ ان الملف يتم تنفيذه سطرًا بسطر ، كما رأيتم ، هذه لا تستحق ان تسمى لغة برمجة لكن بإمكانها تنفيذ مهمات جميلة و جذبة و بديل كثيرة ، خصوصاً بتعاونها مع مكونات النظام و البرامج الأخرى (هي اصلاً ليست الا اعتماد على البرامج و مكونات النظام العامة) ، و سنغطي ذلك في الدروس القادمة بذن الله

الواجب :

- 1- حاول تنفيذ ملف hello Microsoft Word يجعل برنامج الورد يعمل بدل الرسام .
- 2- اجعل البرنامج بعد تشغيل الرسام ، يشغل الورد ثم الباوربوينت
- 3- للمحترفين : اصنع باتش فايل لمجموعة من الهزات الثيبية ، تعمل واحداً تلو الآخر بحيث تتسجل جميع البرامج دون تدخل المستخدم.

في انتظار ردودكم و تفاعلاتكم ، و اريدو تجديد رغبتكم في استمرار الموضوع..

بسم الله الرحمن الرحيم ، الدرس الثاني ،

أولاً :

الاخ ابو فرج هو الوحيد الذي حل الواجب ، فله كل الشكر على المتابعة و التفاعل ، و جله كان يأخذ عليه 10/9

### البداية:

بنود درس اليوم :

- 1- فكرة اكبر عن عمل ملفات الباتش
- 2- الاوامر و سويتشاتنا
- 3- تشفير ملف الباتش ، حتى لا يستطيع أحد العبث في الملف بعدك
- 4- مقدمة في متغيرات النظام

### 1-فكرة عمل الباتشات :

الباتشات ، هي استخدام لبرامج صغيرة مدمجة في النظام او موجودة في نفس مجلد الباتش. منها ما يكون بارز ، و منها ما هو مدمج بملفات النظام و لا نستطيع ايجاده مباشرة ، فكر أمر نستخدمه ، هو برنامج صغير ، فمثلاً ، عند استخدام الامر `chkdsk` و هو متفحص الاقراص ، في موجه اوامر الدوس فإن الوندوز يستدعي البرنامج `chkdsk.exe` الموجود في ملفات النظام ، فكل امر نستخدمه في بيئة عملنا (الوندوز) ، يجب ان يكون برنامج إما مدمج و المدمجة واضحة و تختص بالنسخ و النقل و المسح و غيرها من العمليات البسيطة ، او برنامج `exe` موجود في مجلد الوندوز او `system32` او نفس مجلد الباتش ، و هذه البرامج لا يحصر لها ، حيث يمكن ان تضع انت برنامج هناك ، و يصبح بإمكانك استدعاؤه من ملفات الباتش التي تعمل على الجهاز ، و للتوضيح اليكم مثال على ذلك ، هو الامر `sleep` (الامر يستخدم لايقاف تنفيذ الباتش فترة من الزمن قبل الانتقال الى السطر التالي) فهذا الأمر كان موجود في الوندوز 95 و 98 و لكنه غير موجود في الاكس بي ، لذلك لا بد لنا من وضع هذا البرنامج في مجلد النظام `system32` و اليكم الآن هذا المثال الذي يوضح ذلك بالصور :

(a)لنعمل ملف باتش يستخدم الامر `sleep` ، و نرى ما الذي سيحدث

echo off@

echo I wil sleep for 10 seconds

sleep 10

? echo I have just got up now , have ten seconds passed

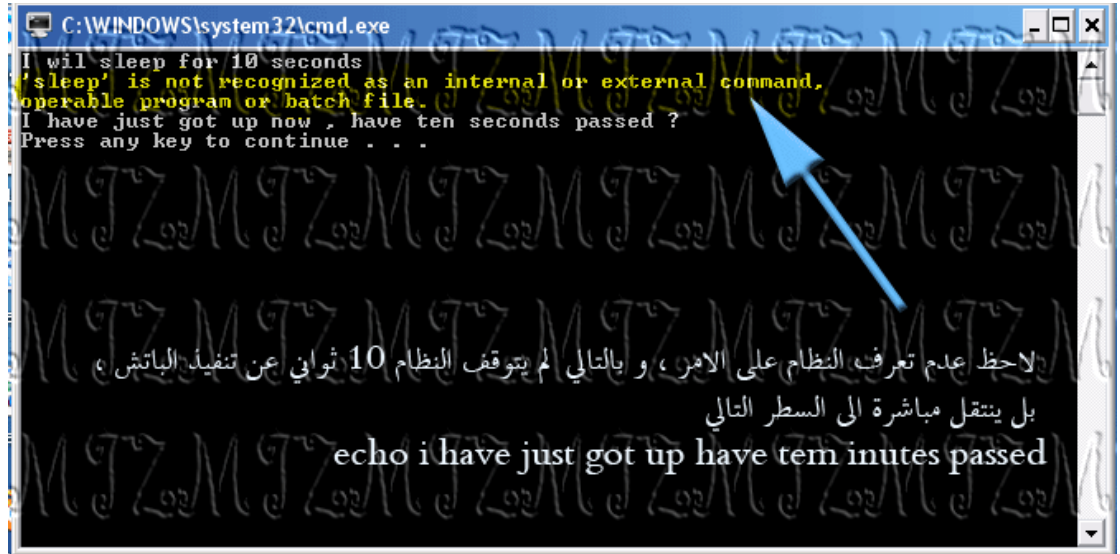
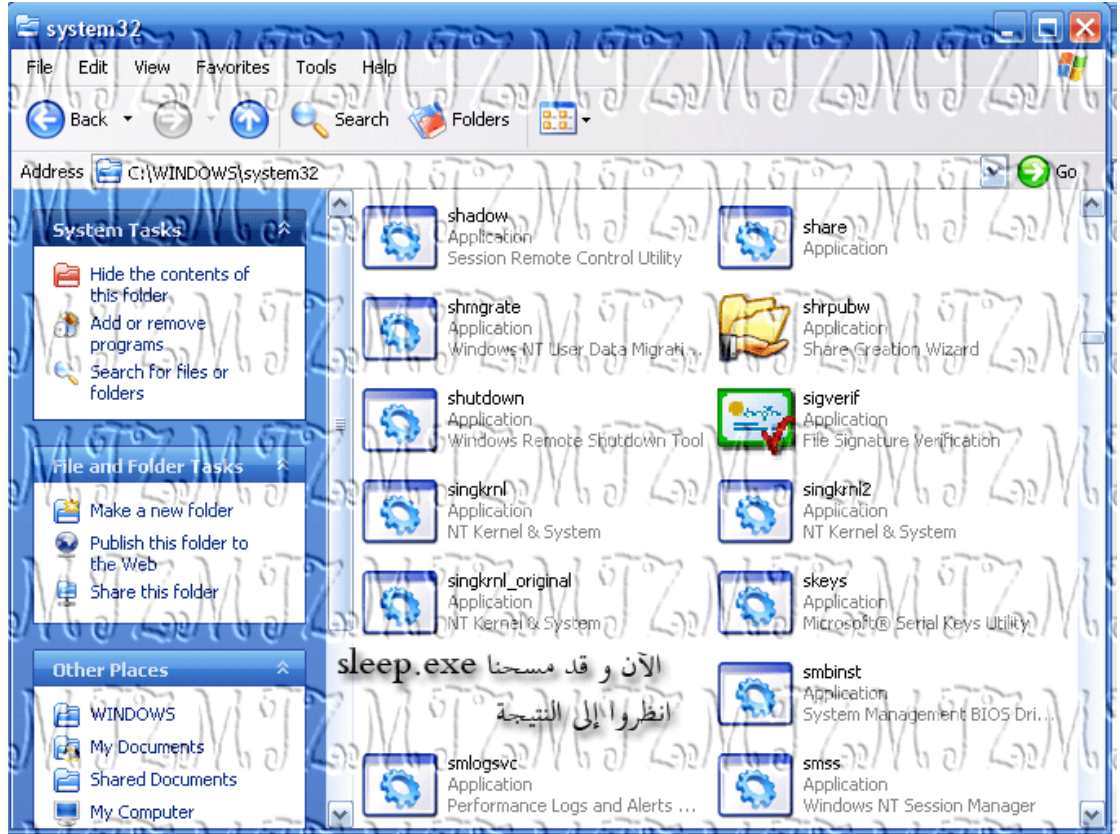
pause

2-سترون الآن ، ان هذا الكود نفسه سيعمل بشكلين مختلفين ، حسب وضع ملفات النظام ووجود sleep.exe هناك  
الذالة الأولى : (الكود مع موجود sleep.exe في ال system32 كذلك وجوده في windows او مجلد الباتش نفسه ، يؤدي  
نفس المهمة)

عند تشغيل ملف الباتش ، ستكون هذه هي النتيجة :



الحالة الثانية : (نفس الكود مع غياب sleep.exe عن system32 و مجلد windows و مجلد الباش نفسه)



**\*ملاحظة :** تجنب تسمية الملف **sleep** فذلك يؤدي الى ان يكرر الباتش تشغيل نفسه

، يعني انت اسميته **sleep** ، فالنظام سيبحث في المجلدات الثلاث التي اشرنا اليها (**system32** و **windows** و مكان وجود الباتش) فسيجد **sleep.bat** و يشغله ، اي ان الباتش سيستدعي نفسه ، ثم يكرر ذلك و يكرر ذلك فنكون النتيجة الاتي:



```

C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.
C:\Documents and Settings\Moataz>cd ..
C:\Documents and Settings>cd ..
C:\>cd windows
C:\WINDOWS>cd system32
C:\WINDOWS\system32>chkdsk.exe /?
Checks a disk and displays a status report.
CHKDSK [volume[[path]filename]] [/F] [/U] [/R] [/X] [/I] [/C] [/L[:size]]

volume          Specifies the drive letter (followed by a colon),
                 mount point, or volume name.
filename        FAT/FAT32 only: Specifies the files to check for fragmentation
/F             Fixes errors on the disk.
/U            On FAT/FAT32: Displays the full path and name of every file
              on the disk.
              On NTFS: Displays cleanup messages if any.
/R            Locates bad sectors and recovers readable information
              (implies /F).
/L:size       NTFS only: Changes the log file size to the specified number
              of kilobytes. If size is not specified, displays current
              size.
/X            Forces the volume to dismount first if necessary.
              All opened handles to the volume would then be invalid
              (implies /F).
/I            NTFS only: Performs a less vigorous check of index entries.
/C            NTFS only: Skips checking of cycles within the folder
              structure.

The /I or /C switch reduces the amount of time required to run Chkdsk by
skipping certain checks of the volume.
C:\WINDOWS\system32>

```

لاحظ التنقل للخلف باستخدام الأمر

cd..

cd

الانتقال الى المجلد الفرعي بالأمر متبوعاً باسم المجلد

اسم البرنامج

السويتش الذي يحدد وضع البرنامج

لاحظ في الصورة أعلاه ، عندما حددنا السويتش للبرنامج ، اخترنا سويتش علامة الاستفهام ، و هو سويتش متكرر في كثير من البرامج يوضح سويشات البرنامج و عمل كل منها ، بمعنى آخر عندما استخدمنا هذا السويتش طلبنا تشغيل البرنامج متبوعاً بسويتش علامة الاستفهام شغلنا البرنامج في وضع التعريف به ، و هذا الوضع يدلنا على كل الاوضاع التي يمكن أن يعمل بها البرنامج و سويشاتها فالآن لو عدنا و شغلنا البرنامج في وضع /F يقوم البرنامج عند تفحص القرص بتصحيح الأخطاء تلقائياً ، بينما لو لم نستخدمه لن يقوم بذلك. و من الجدير بالذكر انه من الممكن استخدام اكثر من سويتش في نفس الوقت لنجعل البرنامج يعمل في الوضعين في نفس الوقت فمثلاً بالامكان تشغيل `chkdsk.exe /F /L` (لاحظ المسافة بين كل سويشين) ذلك سيجعل البرنامج يصحح الأخطاء تلقائياً و يحاول استرجاع البيانات من القطاعات التالفة . و هكذا فإن كل سويتش يفعل وضع معين .. و لذلك علاقة ببرامج التثبيت ، حيث ان فيها وضعاً للتنصيب الهادئ يفعل بسويتش معين

ملاحظة: ليش شرطاً ان يكون علامة "/" هي المستخدمة لتفعيل السويتش فقد يستخدم "-" و قد يكون دون اي رمز ، فقط بمسافة متبوعة بالسويتش

الآن بإمكاننا استخدام اي أمر ، مادام أننا علمنا كيفية ، التعرف على سويشاته بال:"/?" الآن اترك لكم انتم الابداع و التعرف على كل امر و صيغته . و الفت نظركم الى انني سأعطي بعض الاوامر و استخدامها بالتفصيل في درس لاحق

تذكر : ان كل سطر في الباتش ، امر يتوجه الى موجه الاوامر `command prompt` \*\*\*

### 3- كيف نمنع إعادة التعديل على الباتش الخاص بك :

ببساطة ، كل ما عليك ، هو أن تحول ملف الباتش الى برنامج `exe` باستخدام الاداة المرفقة `bat2exe` .

طريقة الاستخدام:

ضع ملف الباتش بجانب `bat2exe` ، و شغل البرنامج من موجه الاوامر ، مع الانتقال الى موضع الملف بموجه الاوامر ، ثم كتابة `bat2exe` متبوعاً بمسافة ، ثم متبوعاً باسم ملف الباتش المراد تحويله الى `exe` كما في الصورة .

\*ملاحظة: طريقة التنقل بين المجلدات كما في الصورة السابقة باستخدام الاوامر `cd` .

### 3- متغيرات النظام `system variables`:



عندما نعمل ملف باتش ، فإننا عادة ما نصنعه ليعمل على أي حاسوب ، ليس فقط على الحاسب الذي نستخدمه ، و عندما كان للمستخدم خيار التغيير في مجلد الوندوز و اسم الحاسب و غيره ، من المتغيرات التي تختلف من حاسب الى آخر ، كان لا بد من حل لمشكلة التنوع هذه ، فكان الحل هو متغيرات النظام ، حيث ان كتابة رمز اجد المتغيرات كمتغير (مجلد الوندوز) ، يوجهنا فوراً الى مسار الوندوز المستخدم على الجهاز الحالي ، فهذه المتغيرات تلائم اختلاف البيئات التي يعمل عليها الملف و اعداداتها ، و هذا القسم شيق و يجعل و ترقبوا في الدرس القادم ، كيفية عمل ساعة بسيطة عن طريق باتش فايل و بالاعتماد على متغيرات النظام.....

الآن الى التعميرين في الدرس الماضي ، :

**الوحيد و هو أبو فرج بدرية 9 من 10** ، كان لديه شبه خطأ، انه لم يضع المسارات مع الاسم ، بل وضع الاسم منفرداً ، و دفع نفسه الى نقل الباتش الى مكان تلك البرامج.

PROFILE	%ALLUSERSPROFILE%	مجلد الملفات المشتركة All users
	%APPDATA%	المجلد الذي تسجل فيه البرامج بياناتها عادة C:\Documents and Settings\Moataz\Application
	%CD%	المجلد الحالي الذي يعمل منه الباتش
COMMANDLINE	%CMDCMDLINE%	الملف التنفيذي للأوامر مع مساره عادة "C:\WINDOWS\system32\cmd.exe"
VERSION	%CMDEXTVERSION%	اصدار موجه الاوامر
COMPUTERNAME	%COMPUTERNAME%	اسم الكمبيوتر
COMSPEC	%COMSPEC%	مسار موجه الاوامر
	%DATE%	تاريخ اليوم
ERRORLEVEL	%ERRORLEVEL%	خطأ في آخر امر تم تنفيذه ، كذلك بإمكاننا التحكم به ، و سنرى ذلك في الدرس القادم
HOMEDRIVE	%HOMEDRIVE%	حرف يبين القرص الرئيس للمستخدم ، حسب ما هو محدد في اعدادات النظام
HOMEPATH	%HOMEPATH%	المجلد الرئيس للمستخدم ، حسب ما هو موضوع في اعدادات النظام في اعدادات المستخدمين و المجموعات
HOMESHARE	%HOMESHARE%	مجلد الشبكة الرئيس حسب ما هو موضوع في اعدادات النظام في اعدادات المستخدمين و المجموعات
NUMBER_OF_PROCESSORS	%NUMBER_OF_PROCESSORS%	عدد المعالجات التي تعرف عليها الوندوز ، حيث ان الحواسيب التي تستخدم تقنية Hyper Threading يتم التعرف عليها كمعالجان
	%OS%	اسم نظام التشغيل الحالي
	%PATH%	يحدد المجلد الذي يقرأ منه منفذ الاوامر ، الاوامر
	%PATHEXT%	يعطي قائمة بالملفات التي يرى النظام انها تنفيذية
PROCESSOR_IDENTIFIER	%PROCESSOR_IDENTIFIER%	وصف للمعالج
PROGRAMFILES	%PROGRAMFILES%	المسار الاساسي للبرامج
	%RANDOM%	يجعل النظام يولد رقم عشوائي بين 0 الى 32727
SYSTEMDRIVE	%SYSTEMDRIVE%	المجلد الذي يجوي النظام
SYSTEMROOT	%SYSTEMROOT%	القرص لاذي يجوي النظام
	%TEMP%	المجلد المؤقت

	%TMP%	المجلد المؤقت
	%TIME%	الوقت الحالي
MAIN	%USERDOMAIN%	اسم مضيف الشبكة الذي يحدّث اليه المستخدم الحالي
ME	%USERNAME%	اسم المستخدم
PROFILE	%USERPROFILE%	المجلد الذي يحتوي ملفات المستخدم الحالي
	%WINDIR%	مسار ملفات التشغيل

بسم الله الرحمن الرحيم ، و الصلاة و السلام على نبي المرسلين و آله و صحبه أجمعين  
الدرس الثالث:

سنناول اليوم في درسنا بضعة أمور ،

- 1- متابعة متغيرات النظام
- 2- استخدام التعليقات ، و تغيير لون تنفيذ الاوامر. و انشاء مجلد جيد و اوامر الجرى
- 3- استدعاء باتش من باتش
- 4- عمل العلفات النصية من الباتشات و الاوامر ، و طباعتها

## 1- تابع ، متغيرات النظام ...

كما قلنا ، متغيرات النظام هي متغيرات تختلف موجودة في النظام ، و هذه المتغيرات بإمكاننا التحكم بها و تغييرها و عرضها ، وذلك له فوائد عديدة أهمها ، تسهيل الحصول على معلومات النظام للمبرمجين.  
و اليكم هذا الجدول لمتغيرات النظام

%ALLUSERSPROFILE%	مجلد الملفات المشتركة All users
%APPDATA%	المجلد الذي تسجل فيه البرامج بياناتها عادة C:\Documents and Settings\Moataz\Application Data
%CD%	المجلد الحالي الذي يعمل منه الباتش
%CMDCMDLINE%	الملف التنفيذي للواحد مع مساره عادة "C:\WINDOWS\system32\cmd.exe"
%CMDEXTVERSION%	اصتار موجه الاوامر
%COMPUTERNAME%	اسم الكمبيوتر
%COMSPEC%	مسار موجه الاوامر
%DATE%	تاريخ اليوم
%ERRORLEVEL%	متغير يبين وجود خطأ في آخر امر تم تنفيذه ، كذلك يمكننا التحكم به ، وسنرى ذلك في الدرس القادم
%HOMEDRIVE%	متغير يبين القرص الرئيس للمستخدم ، حسب ما هو محدد في اعدادات النظام
%HOMEPATH%	المجلد الرئيس للمستخدم ، حسب ما هو موضوع في اعدادات النظام ، في اعدادات المستخدمين و المجموعات
%HOMESHARE%	مجلد الشبكة الرئيس حسب ما هو موضوع في اعدادات النظام ، في اعدادات المستخدمين و المجموعات
%NUMBER_OF_PROCESSORS%	عدد المعالجات التي نعرف عليها الوندوز ، حيث ان الحواسيب التي تستخدم تقنية Hyper Threading يتم التعرف عليها كمعالجان
%OS%	اسم نظام التشغيل الحالي ،
%PATH%	يحدد المجلد الذي يقرأ منه منفذ الاوامر ، الاوامر
%PATHEXT%	يعطي قائمة بالملفات التي يرى النظام انها تنفيذية
%PROCESSOR_IDENTIFIER%	وصف للمعالج
%PPROGRAMFILES%	المسار الاساسي للبرامج
%RANDOM%	يجعل النظام يولد رقم عشوائي بين 0 الى 32727
%SYSTEMDRIVE%	المجلد الذي يحوي النظام
%SYSTEMROOT%	القرص لذي يحوي النظام
%TEMP%	المجلد المؤقت
%TMP%	المجلد المؤقت
%TIME%	الوقت الحالي
%USERDOMAIN%	اسم مضيف الشبكة الذي يدخل اليه المستخدم الحالي
%USERNAME%	اسم المستخدم
%USERPROFILE%	المجلد الذي يحوي ملفات المستخدم الحالي
%WINDIR%	مسار ملفات التشغيل

**الاستخدام:** كل ما علينا هو باختصار شديد جداً ، ان تضع المتغير كما هو ، في الباتش ، و سيعطي

تلقائياً قيمته ، اليكم المثال الآتي ، :

1- نريد ان ننشئ ملف باتش يعطينا تقرير عن الحاسب الذي عمل عليه، ثم يطبع هذا التقرير :

**و نريد التقرير يكون كالآتي:**

1- تاريخ انشاء التقرير و وقته

2- عدد المعالجات التي نعرف عليها النظام ، و نوع المعالج

3- مسار الوندوز

4- مسار اعدادات المستخدم

5- اسم المستخدم الذي قدم التقرير

6- الشبكة التي ينتمي اليها ، و اسم الجهاز

7- المعجل الذي تم تشغيل الباتش منه

## الأوامر المستخدمة:

- echo1-
- title2-
- rem3-
- color4-
- pause5-
- MKDIR6-
- start7-
- sleep8-
- cls9-
- exit10-

شرح الأوامر الجديدة:

1- **rem** : يستخدم هذا الأمر للتعليقات على الكود فقط ، أي أن منفذ الأوامر عند تنفيذه

للباتش لن يقرأ ما بعد هذا الأمر إطلاقاً ، بل سيتخطاه كأنه غير موجود فإينما وجدت

سطر بدايته rem ، اعلم أنه كأنه غير موجود

2- **color** : يستخدم هذا الأمر لتغيير الوان موجه الأوامر ، و له مرافقان عبارة عن رمزان

الاول يتك تطبيقه على الخلفية ، و الثاني يتم تطبيقه على النص ، و اليكم هذا الجدول

بوضوح كل رمز و لونه:

فمثلاً لو أردنا نجعل الخلفية بيضاء باللون الابيض الساطع و النص باللون الأحمر العادي يكون شكل الأمر كالآتي:

color F4

و عندها سيتغير

شكل موجه

الأوامر ،

3- MKDIR

او RD.

يستخدم هذا

الأمر لإنشاء مجلد

جديد ، و ذلك

اللون	الرمز للعادي	الرمز للفاتح
اسود	0	8
ازرق	1	9
اخضر	2	A
اخضرمزرق	3	B
احمر	4	C
بنفسجي	5	D
اصفر	6	E
ابيض	7	F

بكل بساطة بكتابة المسار الذي سيكون فيه المجلد ، مثلاً

[\MKDIR c:\document\mtz\damas\etc](#)

ذلك سينشئ المجلد etc في القرص C: ، في المجلد document في المجلد mtz في

المجلد damas .

و اذا لم يكن اي من هذه المجلدات الثلاث موجوداً ، فسيقوم الامر بانشاءه ، حتى يصبح المجلد etc  
 كما طبته انت  
 اما اذا كان الامر :

## MKDIR etc

سينشئ المجلد etc في المجلد الحالي ، اي المجلد الذي تم تفعيل الباتش منه ،  
**\*\*لا حظ :** لا يوجد فرق اذا وضعنا ال "\" او لم نضعها بعد اسم المجلد المراد انشاؤه  
 و اذا أدركنا أن ننشئ مجلد داخل مجلد تشغيل الباتش نكتب

## MKDIR 1<sup>st</sup>\2<sup>nd</sup>\etc

ذلك سينشئ المجلد etc في المجلد الحالي ( الذي تم تشغيل الباتش منه ) ، في مجلد 2<sup>nd</sup>  
 الموجود في 1<sup>st</sup> و هكذا

**4-Start:** هذا الامر نستخدمه لتشغيل امر ما ، او برنامج من مسار معين، السؤال الذي يطرح نفسه ، أليس مجرد كتابة الامر من دون شيء ك: sleep يكفي لتشغيل الامر ، الجواب نعم ، لكن start يعطينا الفرصة لاستخدام خيارات متقدمة ، كانتظار انتهاء البرنامج ، و المتابعة بعد ذلك ، و كتحديد أهمية الامر (البرنامج) المطلوب تشغيله بالنسبة للمعالج (process priority) ، إضافة الى ذلك ففيها تجنب لكثير من المشاكل الممكنة اثناء استدعاء الاوامر و البرامج ، فمن الآن و صاعداً سنستخدم Start لاستدعاء البرامج و الاوامر و لن نستخدم الطريقة القديمة إلا في الاوامر البسيطة و الواضحة ، المدعمة مع الوندوز

الاستخدام:

Start "(switchone) /(switchtwo) /(switchthree) .... / (" عنوان النافذة " (switches\_of\_program) / ("مسار الملف او البرنامج المراد تشغيله

يتضح في صيغة الامر ، انه بالامكان استخدام كثير من السويتشات ، ل الامر start اليكم بعض السويتشات و استخدامها ،

wait/ الايقاف التام لعنفذ الاوامر الى ان يتم البرنامج الذي قمت بطلبه في start	
تحدد اهمية البرنامج اثناء عمله بالنسبة للمعالج ، هل يعطيه اهتمام اكثر من العمليات الاخرى (يجوز استخدام واحد فقط من هذه السويتشات)	LOW/ BELOWNORMAL/ NORMAL/ ABOVENORMAL/ HIGH/ REALTIME/
تشغيل ملف الباتش او البرنامج في نفس نافذة موجه الاوامر الحالية	B/

تشغيل برامج 16 بت في ذاكرة منفصلة  
، ذاكرة مشتركة مع برامج 16 بت

SEPARATE/  
Shared/

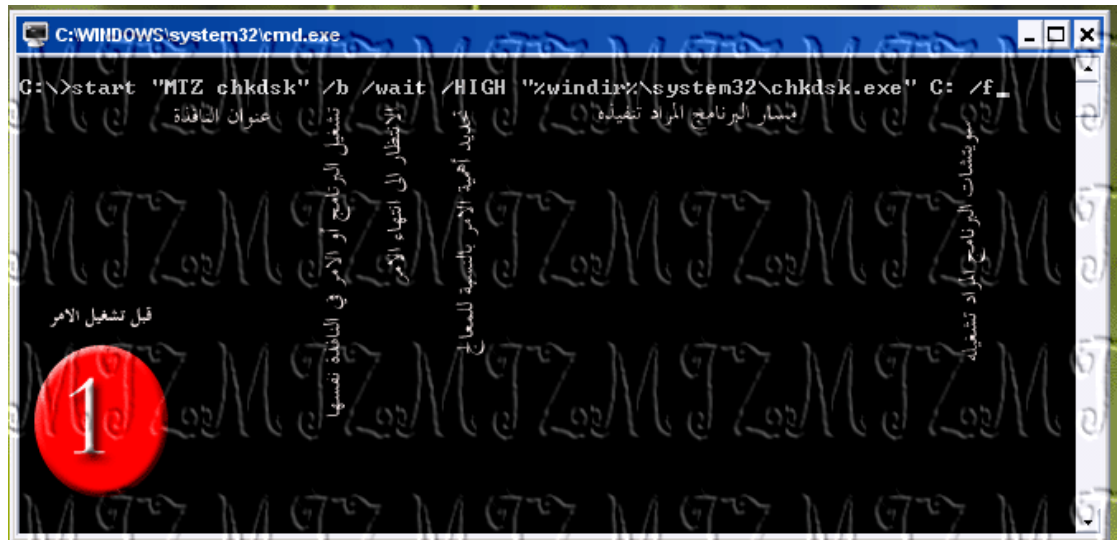
مثال: تشغيل برنامج نفحص الاقراص في وضع التصحيح التلقائي ، مع اعطاؤه نصيب اكبر من طاقة المعالج ،

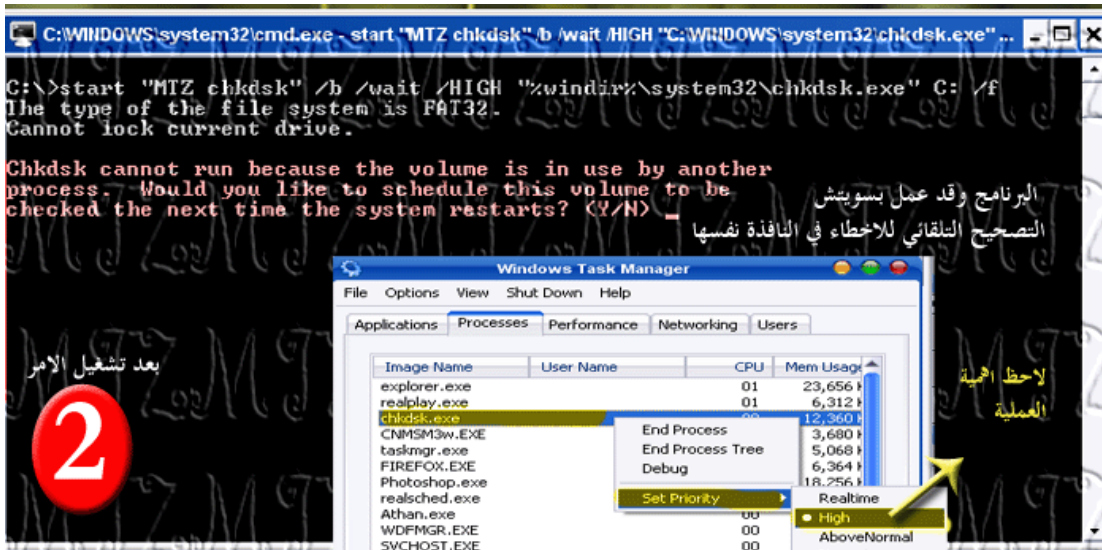
\*ملاحظة:

- 1- من الآن فصاعداً سنستخدم %windir% و باقي المتغيرات ، لن نستخدم قيمة ثابتة حتى يعمل الباتش على كل الحواسيب.
- 2- لم يكن هناك داعي لكتابة مسار الملف لان الملف موجود في مجلد النظام
- 3- نستخدم علامتي التنصيص ، اذا كانت اسماء الملفات او العجلدات تهوي مسافة ، كذلك لو لم تكن تهوي مسافات فلا بأس في ذلك ، والسويشبات المرافقة للبرنامج المراد تشغيله باستخدام **start** تكون خارج علامتي التنصيص.
- 4- يجب وضع عنوان بين علامتي التنصيص الاولتين، و إلا فلن يعمل الامر بشكل صحيح .

أ- في نفس نافذة موجه الاوامر الحالي

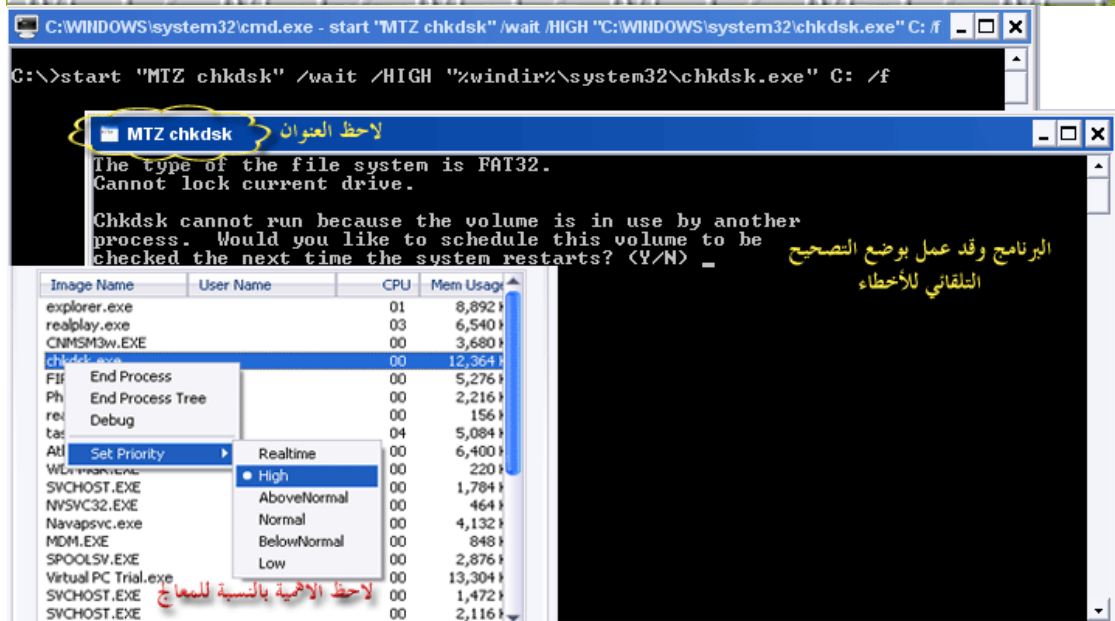
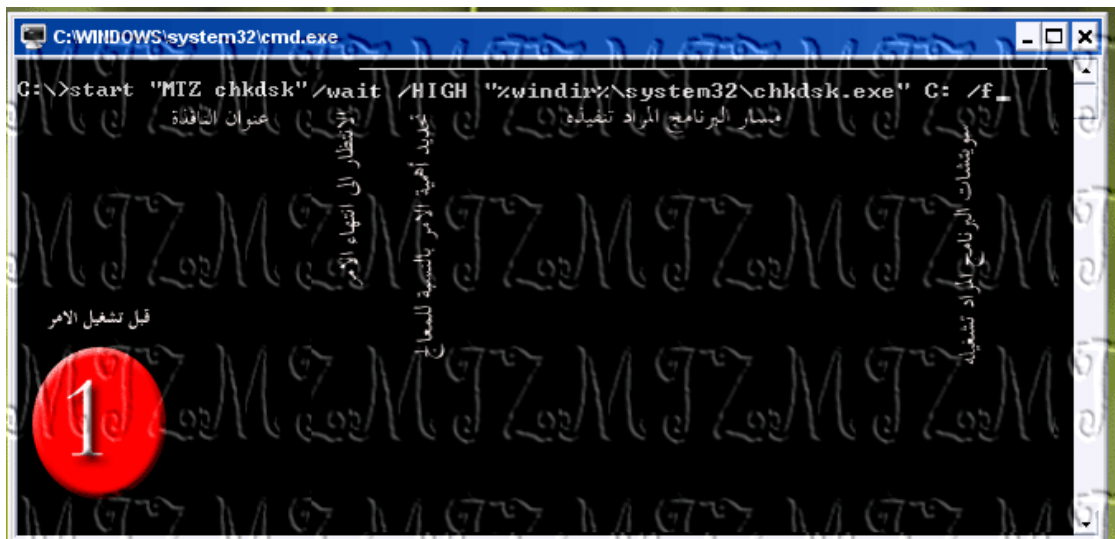
```
start "MTZ chkdisk" /b /wait /HIGH  
"%windir%\system32\chkdisk.exe" C: /f
```





ب- في نافذة منفصلة:

start "MTZ chkdsk" /wait /HIGH  
"%windir%\system32\chkdsk.exe" C: /f



يستخدم لمسح كل ما في موجه الاوامر ، فلا يبقى شيء في موجه الاوامر  
كما في الشكل:

```
C:\WINDOWS\system32\cmd.exe
C:\>start "MTZ chkdsk" /wait /HIGH "%windir%\system32\chkdsk.exe" C: /f
^C
C:\>start "MTZ chkdsk" /wait /HIGH "%windir%\system32\chkdsk.exe" C: /f
^C
C:\>kpnca[klmbg
'kpnca[klmbg' is not recognized as an internal or external command,
operable program or batch file.

C:\>kblmksr' lbnkol' lzd
'kblmksr' lbnkol' lzd' is not recognized as an internal or external command,
operable program or batch file.

C:\>lkbm
'lkbm' is not recognized as an internal or external command,
operable program or batch file.

C:\>sl
'sl' is not recognized as an internal or external command,
operable program or batch file.

C:\>cls
CLS قبل تشغيل امر
```

```
C:\WINDOWS\system32\cmd.exe
C:\>
نافذة موجه الاوامر وقد تم تفريغها تماماً
```

استدعاء ملفات باتش أخرى صنعناها، من خلال ملف الباتش الآخر  
يتم ذلك باستخدام الامر Start بشكل طبيعي كالآتي :

```
start "External batch" /wait /b "batch.bat"
```

طبعاً اذا لم يكن ملف الباتش ، جينياً بجانب الى الباتش الرئيسي ، يجب وضع العسار بدل الاسم فقط ،  
مع مراعاة وضع علامتي التنصيص اذا كان اي من الاسماء يهوي مسافات ، سواء كانت المسافات في  
اسماء المجلدات او اسم الباتش نفسه  
\*\* عند استدعاء ملف باتش من ملف باتش آخر كاننا بالختصار نكتب كل ما في ذلك الباتش ، في الباتش  
الذي قام بعميلة الاستدعاء

**ملاحظة: لا تحاول اطلاقاً ان تشغل باتش من باتش آخر دون start**



المخرج النص من امر الى ملف نصي ، :

مثلاً لو أردنا استخدام الامر dir لرؤية قائمة ملفات في مجلد ، فاننا نكتب dir مباشرة ، ماذا لو اردنا ان نحفظ هذه المخرجات الناتجة من الامر dir ، للقيام بذلك نقوم بكتابة الامر متبوعاً بعلامة "<" ثم اسم الملف النصي المراد بحفظه ، و التوضيح كالاتي:

dir>directory.txt

```
C:\WINDOWS\system32\cmd.exe
D:\Web Downloads>cd mobile
D:\Web Downloads\Mobile>dir
Volume in drive D is MTZ CENTER
Volume Serial Number is 2B52-0D02

Directory of D:\Web Downloads\Mobile
06/03/2005  10:20 AM    <DIR>      .
06/03/2005  10:20 AM    <DIR>      ..
06/01/2005  06:35 PM           59,159 Toom.zip
06/01/2005  06:39 PM        527,498 need.zip
06/01/2005  06:39 PM        28,371 Darts_7210[1].zip
06/01/2005  06:40 PM        62,788 Superhase.zip
06/01/2005  06:40 PM        61,942 Spyrescue.zip
06/01/2005  06:40 PM        32,352 tower.zip
06/01/2005  06:40 PM        63,249 yogihear.zip
06/01/2005  06:40 PM        28,949 musicpp.zip
06/01/2005  06:40 PM        32,897 nineball.zip
06/01/2005  06:40 PM        60,973 DwangoRacing.zip
06/01/2005  06:40 PM        57,564 evil_dead_pinball.zip
06/01/2005  06:40 PM        49,736 fireblaster.zip
06/01/2005  06:40 PM        38,035 dragonfire7210.zip
06/01/2005  06:40 PM        34,801 quicxo.zip
06/01/2005  06:42 PM        364,138 Jungl eking.zip
06/01/2005  06:42 PM       2,080,364 metal.zip
06/01/2005  06:42 PM        613,320 Agile Fighter 1.zip
06/03/2005  10:20 AM    <DIR>      Toom
07/02/2005  05:46 AM    <DIR>      themes
07/03/2005  06:57 PM    <DIR>      Darts_7210[1]
07/18/2005  08:42 PM       709,806 photoshop.zip
07/18/2005  08:43 PM       475,574 frames.zip
          19 File(s)          5,381,516 bytes
           5 Dir(s)      5,435,981,824 bytes free

D:\Web Downloads\Mobile>
```

\*\* من الممكن تحديد مسار الملف الناتج ، كأن نكتب:

`dir>%systemdrive%\mobilelist.txt`

طباعة ملف نصي `txt`: (طبعا يجب ان يكون لديك طابعة )

يتم ذلك ببساطة باستخدام الامر :

`notepad /p filename.txt`

طبعا نضع اسم الملف المراد طباعته بدل `filename.txt` ، و كما في اي امر ، يمكن تحديد المسار للملف و يجب مراعاة وضع علامتي تنصيص اذا اجنوى اي من الاسماء مسافة ..

الآن بعد هذه التوضيحات ، هيا إلى هدفنا ، باتش يعطي تقرير و يقوم بطباعته:

\* سنقوم بعمل ملفين ، اجهدهما اسمه `work.bat` و الآخر `work2.bat`

[Work.bat](#)

@echo off

```

REM كما نرون هذا الجزء لا يؤثر في الملف ، فنفذ الاوامر لن يقرأ ما هو بعد هذه الكلمة
echo Welcome to another MTZ turtoial
title MTZ turtoials,system variables
rem لاحظوا، انه عند تغيير لون الواجهة نستخدم رمزين ، الاول يتم تنفيذة على الخلفية ، و الثاني
على النص
color 1F
echo To Begin system report
pause
echo Reporting....
REM لانشاء ملف نصي بجوي مخرجات معينة ، نكتب الامر متبوعاً بإشارة اكبر من ثم اسم الملف
REM اذا حددنا مسار تكوين الملف النصي ، سيوضع في ذلك المسار ، و ان لم نحدد سيوضع في
نفس مسار الباتش
MKDIR %systemdrive%\ourlog\
REM يجب ان يكون المجلد موجوداً قبل الجراج الملف اليه
start /b work2.bat>%systemdrive%\ourlog\mtz.txt

echo If u want to print the report ,
pause
title the report will be printed in: 5
sleep 1
title the report will be printed in: 4
sleep 1
title the report will be printed in: 3
sleep 1
title the report will be printed in: 2
sleep 1
title the report will be printed in: 1
sleep 1
title Printing.....
notepad /p %systemroot%\ourlog\mtz.txt
cls
title ALL DONE !!!
echo Allright ,thx alot 4 using mtz turtoials
REM لا نسوننا من الدعاء الصالح

```

```
sleep 2
exit
```

## Work2.bat

```
@echo off
echo  \\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\Here come ur report\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
echo  \\ Processor: %processor_idenfifier%
echo  \\ No.of processors:%Number_of_processors%
echo  \\ Ur Windows folder is in: %windir%
echo  \\ ur profile is in: %USERPROFILE%
echo  \\ ur domain:%USERDOMAIN%
echo  \\ Ur pc name is: %COMPUTERNAME%
echo  \\ Directory this batch was started from is: %CD%
echo  \\ This report was taken by %USERNAME% on
%DATE% at %time%
```

### خيرات من الباتش work.bat السابق :

- 1- الغاء اظهار الاوامر في واجهة منفذ الاوامر
- 2- تغيير العنوان
- 3- تغيير لون الواجهة
- 4- اظهار النصوص فقط ، دون معالجة منفذ الاوامر لها كاوامر
- 5- استخدام التعليقات
- 6- انشاء مجلد جديد
- 7- تشغيل برنامج او باتش من خلال باتش
- 8- وضع مخارج الامر في ملف نصي
- 9- عمل عد تنازلي في العنوان
- 10- طباعة الملف باستخدام الامر notepad /p
- 11- مسح كل ما شاشة تنفيذ الاوامر

### خيرات من الباتش work2.bat

- 1- استخدام متغيرات النظام فقط
- 2- اسلوب التفكير في ربط ملفات الباتش ببعضها البعض

الواجب : لدينا من هذ الدرس فصاعداً ، سيكون هناك قسم مخصوص ، عليكم فيه القيام بعكس الواجبات السابقة

أنا أعطيك الملف ، و عليكم وضع تصور للكود الخاص به ، و عليه درجة الواجب كله ، و القسم الأخر قسم bonus للعابرة منكم ، فمن يجله يأخذ زيادة درجتين:

- 1 - ضع تصور لملف الباتش في mtz.zip الموجود في المرفقات ، 10 درجات
  - 2- ان تنشئ ساعة ، باستخدام الباتش سكريبت ... ( للفاهمين 100% ) + 3 درجات
- مدة الواجب : 4 أيام ...

## أمثلة

امثلة تطبيقية: المثال الاول :

**الهدف :** عمل قرص خاص ، عليه مجموعة من البرامج ، يتم تثبيتها بمجرد ادخال القرص .

المخطط :

- 1- ترتيب الملفات و المجلدات
- 2- عمل الباتش :
- 1- استخدام echo@ لمنع اظهار الاوامر عند تشغيل الباتش
- 2- العد التنازلي باستخدام الامر Sleep
- 3- تشغيل الهزم التثبيتية باستخدام الامر Start او الطريقة التقليدية
- 4- الخروج عند الانتهاء .
- 3- عمل الملف : Autorun.inf المسؤول عن تحديد ما سيعمل عند ادخال القرص .
- 4- حرق السدي باستخدام أي برنامج .

التنفيذ:

- 1- فلنبدأ بوضع تصور لشكل الملفات على القرص ، فلنفرض اننا نريد وضع البرامج الآتية :

Adaware  
Yahoo Messenger  
Azerues

سنضع كل البرامج في نفس مسار الباتش كي نتجنب تحديد مسار الملفات التي سيثقلها الباتش ، و بذلك نكون قد اتبعنا العرجلة الاولى بشكل تمثيلي للملفات في وجه القرص ..

2- عمل الباتش:

أ-

لنفكر الآن بالباتش ، الذي سنعمله ، فكرته الاساسية بسيطة جداً ، و هي تشغيل مجموعة من البرامج بصورة متتابعة ... الاول ثم الثاني ثم الثالث ، اذا طبقنا الفكرة مباشرة سنكتب الآتي في الباتش (بفرض ان الملفات كلها بجانب الباتش ، اي في نفس مساره) :

"yahoo setup.exe"  
aawsepersonal.exe  
Azureus.exe

: ملاحظات على الكود السابق

في الكود السابق افترضنا أن الملفات توجد بجانب ملف الباتش أي في نفس مساره ، و لذلك تعمل مباشرة دونما تحديد لمسارها ، مثلها -1  
كمثل أي أمر موجود. فقد ذكرنا في الدروس السابقة ، ان الباتش عند تنفيذه لامر معين (ذكرنا ان الامر هو برنامج بالنسبة للباتش) ، يبحث  
: عن هذا الامر (البرنامج) في المسارات الثلاث الآتية

1- مسار ملفات النظام -2 system32 مسار الوندوز 3- مسار الباتش نفسه ..

2- اذا اجتوى اسم لملف على مسافة يجب وضع علامتي تنصيص ..كما هو الحال في الياهو مسنجر

ذلك سيقوم لنا بالفكرة ، فهذا سيعمل أولاً ، تثبيت الياهو ، ثم ال ad-aware ثم ال msn messenger

لو فكرنا قليل في الكود السابق .. سنجد اننا لم نقم بشيء .. ما فائدة تشغيل هذه البرامج واحداً تلو الآخر ... ما دامت لا تزال تحتاج  
الى متابعتي الشخصية لاتمام خطوات التثبيت .. سنجد الجواب في الوضع الصامت للجزعات التثبيتية ، والسويتشات .. الشرح في هذا الرابط :

<http://www.damascgate.com/vb/showpost.php?p=340740&postcount=329>

الآن وقد اصبح لدينا فكرة حول السويتشات و التثبيت الصامت .. فلنعد النظر في ملف الباتش ..

لو اننا أضفنا سويتشات هذه البرامج (الوامر) لها ، سنجد الفائدة .. فسيكون الموضوع مهم نوعاً ما و ذو جدوى فيصبح الباتش كالآتي :

```
"Yahoo setup.exe" /S  
aawsepersonal.exe /S  
Azureus.exe /S
```

و عند تشغيله .. سيقوم بنفس ما قام به الباتش السابق (من حيث تشغيل البرامج) ، لكن هذه المرة ستعمل البرامج في الوضع الصامت

.....

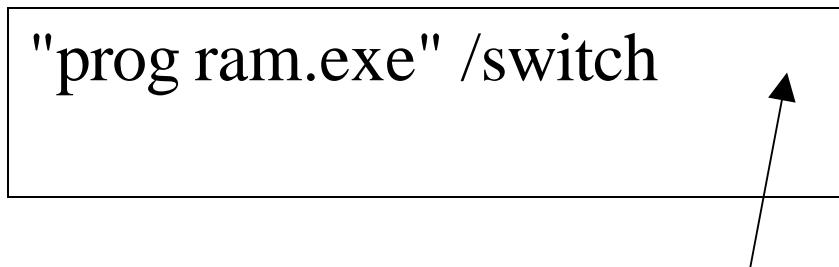
**ملاحظات على الكود السابق :**

1-السويتش S/ لا يصلح لكل البرامج ، راجع درس التثبيت الصامت ....

2-عند استخدام ملفات اسماءها تهوي مسافات نكتب الامر بحيث يكون اسم الملف بين علامتي التنصيص ، متبوع

بمسافة ، ثم السويتش ... كالآتي :

"prog ram.exe" /switch



انتبه لوضع مسافة

3- ايضاً الاسماء العادية يجب ترك مسافة بين الاسم و السويتش ، لو لم نفعل ذلك ، سيفهم منفذ الاوامر ان السويتش جزء من اسم الملف ،  
مثلا : aawsepersonal.exe/S ، سيفهم منفذ الاوامر ان هذه كله اسم ..

الآن أجمعنا هيكل الباتش ، لكن لا يزال شكل الملف أثناء تنفيذه بدائياً ، لذلك سنقوم بواحد من الخيارين الآتيين :

1- اما ان نستخدم : cmdow @ /HID ، و ذلك لإلغاء نافذة الباتش تماماً أثناء تنفيذ الملف ، و يتطلب ذلك وجود cmdow.exe  
في واحد من المسارات الثلاث التي أكثرنا الحديث عنها ( system32 ، الوندوز ، مسار الباتش نفسه ) .. بإمكانك تحميل cmdow  
العرفقات ..

الاستخدام : فقط نكتب : cmdow @ /HID و سنختفي النافذة بمجرد ظهورها ..

و بالتالي يكون شكل الباتش كالآتي :

cmdow @ /HID

"Yahoo setup.exe" /S"

aawsepersonal.exe /S  
Azureus.exe /S

2- أن نستغل نافذة الاوامر لعرض الخطوات الحالية مع الترتيب . و ذلك بمجموعة من الاوامر :

أ- الامر : echo لإظهار الاوامر الموجهة لتنفيذ الامر

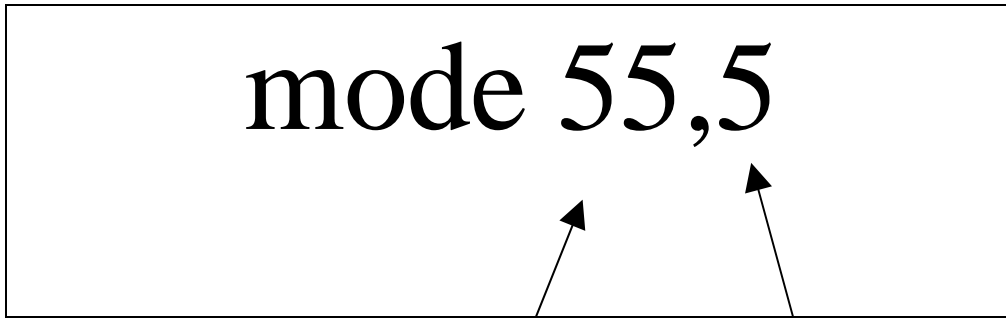
ب- الامر: color لتغيير لون النص و الخلفية

ج- الامر: mode : لتغيير حجم النافذة ..

د- الامر : title لتغيير العنوان

**ملاحظة:** طبعاً ، كما هو الحال في لغتنا البسيطة هذه ، الامر ينفذ في سطره ، أي لو كتبنا مثلاً color 1F في السطر الاول ، ثم كتبنا في السطر التالي " /S" yahoo setup.exe ثم عدنا في السطر التالي لنكتب color F1 سنجد ان منفذ الاوامر سيبدأ بلون ، ثم ينفذ امر تشغيل برنامج تثبيت الباهو . ثم يعود ليغير لون موجه الاوامر مرة أخرى و هكذا ... هذه هي ملاحظة فقط ...  
الامر : mode :

يمكن استخدام هذا الامر لتغيير حجم نافذة العرض .. كالآتي :



عدد الاعمدة التي تتسع لها النافذة ،  
حيث كل عمود يساوي حرف

عدد الصفوف التي  
تتسع لها النافذة  
حيث كل صف يمثل  
سطر



اما عن الاوامر الأخرى ، فقد درسناها في الدروس السابقة من حيث الصياغة و الاستخدام ، الآن أعتقد ان لدينا فكرة الباتش .. و صرته مكتملة و هي كالآتي :

1- سنجعل العنوان MTZ Apps installer

2- سيكون هناك اعلام للمستخدم بالبرنامج الجاري تثبيته عن طريق الامر echo

3- بعد ما يتم تثبيت البرنامج سيكتب الملف للمستخدم رسالة : !!DONE

الآن الى التنفيذ:

لا تنسو ان rem للتعليقات ، اي ان ما بعدها يتجاهله منفذ الاوامر تماماً

الامر التالي لمنع منفذ الاوامر من اظهار اسماء الاوامر اثناء تنفيذها REM

```

@echo off
REM الآن نضع العنوان
title MTZ Apps Installer
REM الآن نغير حجم النافذة
mode 55,5
REM الآن نغير الألوان
Color 1F
REM الآن نبدأ بالبرنامج الاول
REM نكتب قبل بداية أمر تثبيتياهو أمر باظهار رسالة
echo Installing Yahoo ...
"Yahoo Setup.exe" /S
REM بعد ان انتهى امر تثبيتياهو نريد رسالة اعلام بان التثبيت تم
echo DONE !!!
REM البرنامج الثاني
echo Installing Ad-Aware
aawsepersonal.exe /S
echo DONE !!!
REM الثالث
echo Installing Azerues
Azureus.exe/S
echo DONE !!
echo All DONE !!!
exit

```

و هكذا نكون أتممنا النقطة 2 . الآن الى النقطة 3. وهي عمل ذلك في قرص له auto run

**فكرة ال AutoRun في الوندوز**

هناك ملف يوضع في وجه السيدي . اسمه Autorun.inf هو الذي يحدد البرنامج الذي سيعمل عند ادخال القرص .. بإمكانك للتأكد من ذلك ان تتصفح اي قرص لديك فيه خاصية التشغيل التلقائي . ستجد ملف autorun.inf في وجه السيدي . و فيه الملف الذي سيعمل عند ادخال القرص و ايقونة القرص ..

```

[AutoRun]
open=setup.exe
icon=WPI.ico

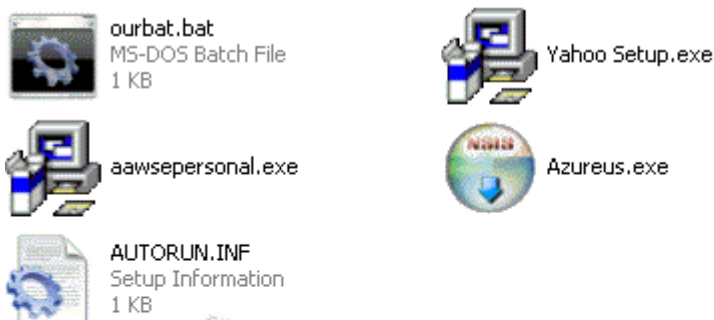
```

بإمكاننا وضع ملفنا الذي صنعناه بدل من Setup.exe

و الايقونة المناسبة بدل WPI.ICO

و ينتهي معنا شكل العلفات في السيدي كالآتي:





مثال 2

### المثال الثاني: باتش التنظيف للنظام

مثال اليوم ، مثال بسيط جداً ، عمله كالآتي :

- 1- يقوم بمسح الملفات المؤقتة .
- 2- يقوم بتفحص قرص النظام
- 3- يقوم بإلغاء تجزئة القرص
- 4- بعد انتهاء هذه العمليات ، سيقوم بإطفاء الجهاز ..

### فائدة مثل هذا الباتش ..

بعد انتهاءك من العمل على الجهاز ، في ساعة متأخرة من الليل ، بإمكانك عمل هذه المهمة ، حتى لا تضيق ساعات عملك في انتظار تعام هذه المهمات ..

### أميرات سابقة ننتاها من الدروس السابقة

- 1- الامر echo للتحكم في عرض الاوامر .
- 2- الامر pause لإيقاف تنفيذ الملف ، حتى يضغط المستخدم زر على لوحة المفاتيح
- 3- الامر rem للتعليقات
- 4- استدعاء باتش من باتش
- 5- الامر start لتشغيل ملف من خلال الباتش
- 6- الامر chkdsk لفحص القرص
- 7- الامر defrag لإلغاء التجزئة
- 8- الامر cmdow لإلغاء نافذة الباتش
- 9- الامر MKDIR او MD لإنشاء مجلد جديد

## خيارات جديدة:

- 1- الامر RD لالغاء مجلد ..
- 2- الامر shutdown لايقاف الجهاز

## شرح الاوامر الجديدة :

### RD1-

تناولنا في الدرس الثالث كيفية انشاء مجلد جديد و لم نعلم كيف نهذف مجلد ما ، هذا الامر يقوم بهذف المجلد .  
عندما نهذفنا عن mkdir قلنا انه اذا كتبنا ،

```
pic1  
mkdir "c:\1\2\3\4\5"
```

سيقوم الامر بانشاء جميع المجلدات ، 1 و 2 و 3 و 4 وصولاً الى 5 ، حتى اذا لم تكن موجودة.  
السؤال الآن ، اذا استخدمنا RD بنفس الطريقة .. اي كتبنا

```
pic2  
RD "c:\1\2\3\4\5"
```

هل سيقوم الامر بهذف ، جميع المجلدات ... 5 ثم 4 ثم 3 وصولاً الى واحد  
الجواب : لا ، ذلك سيلغى فقط المجلد المسمى 5 بكل ما يدويه من مجلدات و ملفات ، اي اذا فرضنا ان المجلد 5  
يدوي ، المجلدات 6 و 7 و 8 ، جميعها بداخل بعضها البعض ، ستهذف هذه المجلدات

## الصيغة :

عادة ما نستخدمه بالصيغة :

```
pic3  
RD /S /Q "path"
```

بيث path تمثل المسار ، بحيث يكون آخر مجلد فيه ، هو المجلد الذي سيتم حذفه ، اي كما ذكرنا في شرح الامر ،  
السويتشات (خيارات الامر) :

```
pic4  
/S لـحذف كل ما بالداخل تماماً  
/Q لمنع سؤال التأكد من الحذف
```

\*\* ملاحظة ، أبرئ نفسي من سوء استخدام هذا الامر فيما يضر ،  
( كاستخدامه لهذف المستندات (... my documents )

## shutdown2-

يستخدم هذا الامر لايقاف تشغيل الحاسب ..

## الصيغة العامة :

```
pic 5  
shutdown -s -t 00
```

السويتشات (الخيارات):

```
pic6  
-S ايقاف التشغيل  
-T إعادة التشغيل
```

- t تحديد الزمن قبل ايقاف التشغيل
- (الرسالة المرفقة كسبب اغلاق الجهاز (طبعاً يجب ان يكون هناك وقت مهدد للايقاف لكي تظهر الرسالة C-
- (ايقاف عملية ايقاف التشغيل الحالية (طبعاً يجب ان يكون هناك وقت مهدد للايقاف لكي تتمكن من الالغاء a-

طبعاً كما قلنا ... بالامكان ان نستخدم اكثر من سويتش في الوقت نفسه .. لكن لا يجوز استخدام سويتشان متضاربان كاستخدام S- مع I- لانك بذلك تطلب من الامر ان يعيد التشغيل و يوقفه في وقت واحد ... بالله عليكم كيف ببسويتها.

### الفكرة لعمل الباتش

سنقدم الآن فكرة عمل الباتش .. فلنبدأ بالتحليل نقطة نقطة .

#### 1- حذف الملفات المؤقتة :

سنقوم بذلك عن طريق ازالة المجلد تماماً ثم اعادة تكوينه عن طريق الامر `mkdir` ، فازالة المجلد تعني ازالة المجلد و كل ما به من الداخل ..

#### 2- تفحص قرص النظام ..

هذه المهمة بسيطة عن طريق .. `CHKDSK` ، و لتحديد قرص النظام نستخدم المتغير %  
systemdrive

#### 3- الغاء تجزئة قرص النظام

هذه المهمة كالمهمة السابقة ، باستخدام الامر `defrag` بدلاً من `chkdsk`

#### 4- اطفاء الجهاز

بسيط عن طريق الامر `shutdown` الذي ذكرناه ..

الآن الى العمل و طريقته ،

ترتيباً للعمل سنقسم الباتش على ملفين ، الاول ، يقوم بالمسح و اعادة التكوين للمجلدات ، و الثاني عليه الاكتمال الى نهاية المهمات ..

قبل ان نبدأ بكتابة الكود .. يجب ان نفكر بالمجلدات التي تهوي ملفات مؤقتة لنمسحها ، حسب معرفتي هذه المجلدات هي ..

1- %userprofile%\Local Settings\Temporary Internet Files\Content.IE5

2- %userprofile%\Local Settings\Temporary Internet Files\Content.IE5

3- %userprofile%\Local Settings\history

4- %windir%\prefetch

5- %userprofile%\local settings\Temp\

طبعاً اذا كنت تريد مسح ملفات اخرى من خلال الباتش ، يمكن وضعها ،

الآن سنمسح هذه المجلدات ( المجلد الاخير في المسار) ، باستخدام RD ثم نعيد تكوينها بـ `mkdir`

الآن الى العمل ، فلنكتب الكود لملف التنظيف ( المسح )

## clean.bat

---

### pic 7

@echo off

@cmdow @ /HID

REM هذا هو الملف الذي سيقوم بمسح كل الملفات المؤقتة و الغير ضرورية

REM اسنستخدم امر الحفاء النافذة

REM سنعمد فكرة بسيطة هي : ان نهدف المجلد بكل تفرعاته ، ثم نعيد انشاؤه

RD /S /Q "%userprofile%\Local Settings\Temporary Internet Files\Content.IE5"

mkdir "%userprofile%\Local Settings\Temporary Internet Files\Content.IE5"

RD /s /Q "%userprofile%\cookies"

MD "%userprofile%\cookies"

RD /s /Q "%userprofile%\Local Settings\history"

MD "%userprofile%\Local Settings\history"

RD /s /Q "%windir%\prefetch"

MD "%windir%\prefetch"

RD /S /Q "%userprofile%\local settings\Temp\"

MD "%userprofile%\local settings\Temp\"

rem انتهى دور باتش التنظيف

الباتش الرئيسي ( الثاني ) سنسميه main هذا هو ..

### pic 8

## main.bat

---

```

@echo off
echo Welcome TO MTZ system cleaner
echo the cleaner Will clean temprory files , then check your
system disk , defrag it ,then shutdown ur PC to proceed ,
pause
rem نستخدم الامر start حتى نشغل الباتش
start "" "clean.bat"
rem الامران الآتيين ، تم مناقشتهما من قبل
rem كما ترون ، استخدمنا متغير قرص النظام
chkdsk %systemdrive%
defrag %systemdrive%
rem هذا هو الامر الجديد المختص بإيقاف تشغيل الجهاز
shutdown -s -t 0

```

طبعاً التشغيل يكون من الباتش. main.bat و يجب مراعاة وجود clean.bat في واحد من المسارات الثلاث التي طالما تحدثنا عنها ، و للتأكيد هي

- 1- windows
- 2- system32
- 3- مسار الباتش

انا جاهز لاي استفسار او سؤال ( مع انني شبه متأكد من تفاعل اي عضو .. غير ابو فرج.. اعزج )  
هكذا نكون اتعمنا العنال الثاني .

بسم الله الرحمن الرحيم

العنال الثالث : حول عمليات العلفات ، النقل ، النسخ

## تجربات سابقة :

معمم .. اعتقد اننا مش بنستخدم حاجة قديمة بس بنحتاج فهمنا للباتش سكريبت ...

## تجربات جديدة :

- 1- الامر move
- 2- الامر Xcopy و الامر copy
- 3- التعامل مع الملفات بحسب النوع و الخواص .. ما يسمى wildcards

## شرح الاوامر :

### الامر move :

يستخدم لنقل مجلد او ملف من مكان الى آخر ، صيغته كالاتي :

"الملف او المجلد المراد نقله . مع المسار بدقة " "المسار الذي سيتم النقل اليه " move

اليكم المثال الاتي للتوضيح ..

لدينا الملف file1.doc ، موجود على سطح المكتب للمستخدم الحالي ، اي :

```
%userprofile%\desktop\file1.doc
```

هذا هو مسار الملف الاول بدقة .. لو اردنا نقله الى القرص d:\ في مجلد موجود مسبقاً اسمه folder ، اي ان مسار هذا المجلد هو d:\folder ، فاننا نكتب الاتي ،

```
move "%userprofile%\desktop\file1.doc" "d:\folder"
```

هكذا ينتقل الملف file1.doc الى المسار d:\folder ، اي سنجد الملف في المجلد folder الموجود في القرص d: . و لن نجده بعد الآن في المسار السابق ...

السويتشات :

يستخدم لمنع ظهور الرسالة التأكيدية عند استبدال الملف او المجلد اذا كان موجود في المسار الجديد /Y

## ملاحظات :

- 1- يجب تحديد اسم الملف المراد نقله مع الامتداد ، اي انه لا يكفي ان نكتب file1
- 2- اذا كان المراد نقله مجلد ، طبعا لن نكتب الامتداد (ما عمريش شفت مجلد لو امتداد)
- 3- احياناً تكون النقطة جزء من اسم الملف و ليست امتداده ، مثلاً : Moataz.part11.rar نلاحظ ان الامتداد هو rar و ليس part11 ، فالامتداد الحقيقي للملف هو آخر نقطة متبوعة بمجموعة احرف .
- 4- طبعا لن ننسى ذكر اهمية وضع علامتي التنصيص ، عند التعامل مع اسماء نهوي مسافات
- 5- لو اردنا ان ينتقل الملف الى المسار الجديد ، و لكن باسم آخر ، نكتب :

```
move "move "%userprofile%\desktop\file1.doc" "d:\folder\file1renamed.doc"
```

بذلك بعد ان ينتقل الملف الى المجلد folder سيتم تغيير اسمه الى file1renamed.doc ، و اعود لكرر و اؤكد ضرورة وضع الامتداد ، فلو انا لم نضع في المسار الجديد الامتداد .doc سنحصل على ملف عديم النوع

## الامر xcopy و copy ،

يستخدم لنسخ مجلد او ملف ، صيغته العامة :

للملفات:

```
"الملف المراد نسخه مع المسار بدقة " "الملف الذي سيتم نسخ بيانات الملف المنسوخ اليه " xcopy
```

مثال :

```
xcopy "%userprofile%\desktop\file1.doc" "d:\folder\file1copied.doc"
```

هذا الامر سيفقوم بنسخ الملف file1 من سطح مكتب المستخدم الحالي ، الى المجلد folder ، باسم file1copied.doc ، ليس file1.doc

نسخ الملفات التي تحمل خاصية "archive" فقط بين ملفات المجلد، و هذا السويتش يمنع تغيير خصائص الملف لعد النسخ  
 نسخ الملفات التي تحمل خاصية "archive" فقط بين ملفات المجلد، و هذا السويتش يلغي خاصية "archive" للملف المنقول  
 نسخ الملفات التي تم التعديل عليها بعد التاريخ المعطى ، لو لم تدخل تاريخاً ، سيتم نسخ الملفات الأحدث في المصدر من D:m-d-y  
 تلك التي في الهدف ..

/EXCLUDE:file1[+file2][+file3]...

استثناء ملفات معينة من عملية النسخ ...

/P سؤال المستخدم قبل نسخ أي ملف  
 /S نسخ المجلد و المجلدات الفرعية فيه بكل ما بها من ملفات .. عدا المجلدات الفارغة  
 /E نسخ المجلد و المجلدات الفرعية فيه بكل ما بها من ملفات .. مع المجلدات الفارغة  
 /U التأكد من أن كل ملف تم نسخه بنجاح ... اي ان ملف المصدر يطابق ملف الهدف 100%  
 /W مطابقة المستخدم بضغط أي زر قبل بدأ النسخ  
 /C متابعة النسخ حتى إذا صادف عملية النسخ خطأ  
 /I إذا كان الهدف المكتوب ، غير موجود ، و يتم نسخ أكثر من ملف .. فإن هذا السويتش يدل على أن الهدف المحدد. مجلد و ليس ملف  
 /Q عدم إظهار أسماء الملفات أثناء النسخ  
 /F إظهار المسارات كاملة للملفات أثناء النسخ  
 /L عرض أسماء الملفات التي سيتم نسخها  
 /G يسمح بالنسخ ، حتى إذا كان المصدر يدعم خاصية تشفير الملفات و الهدف لا يدعمها  
 /H نسخ الملفات المخفية و ملفات النظام ، حيث أن عدم وضع هذا السويتش يمنع ذلك  
 /R استبدال الملفات التي تحمل خاصية "للقراءة فقط" في الهدف  
 /T بدلاً من نسخ الملفات التي داخل المجلد .. فإن هذا الخيار يعمل على نسخ شجرة المجلد الرئيسي ، دون نسخ أي ملف  
 /U نسخ و استبدال الملفات الموجودة في المصدر و الهدف ، فإذا كان هناك ملف موجود في المصدر و غير موجود في الهدف ، فلن يتم نسخه  
 /R نسخ الملفات ، مع المحافظة على خواصها (الامر دون هذا الخيار.. يلغي الخواص عند النسخ)

/Y عدم السؤال عن استبدال الملفات

/-Y

/Z نسخ الملفات من الشبكة في النظام القابل لإعادة

### نوضيح عملية النسخ ،

ذكرت انا عبارة " العلف الذي سيتم نسخ بيانات العلف المنسوخ اليه " . الجملة تبدو مبهمه نوعاً ما لعن لا يعلم خطوات النسخ للعلف . ما يحدث عند تحديد "اسم العلف الذي سيتم نسخ بيانات العلف المنسوخ اليه" . عند بداية عملية النسخ . فإن اول ما يحدث هو انشاء ملف بحجمه 0 . بالاسم الذي حددناه في الامر (file1copied.doc) . ثم يبدأ نقل البيانات من العلف الاصلي الى العلف الجديد ...

\*\* سنبالظ عند استخدام الامر Xcopy لنسخ ملف بالطريقة السابقة . سيقوم موجه الاوامر بسؤال المستخدم . إذا ما كان الاسم file1copied.doc اسم ملف او مجلد . كما في الصورة :

```
C:\Documents and Settings\Moataz>xcopy "%userprofile%\desktop\file1.doc" "d:\folder\file1copied.doc"
Does D:\folder\file1copied.doc specify a file name
or directory name on the target
(P = file, D = directory)?
```

إذا كنت لا تحبذ ، ذلك ، بإمكانك استخدام الامر Copy . فهو مخصص لنقل ملف واحد فقط ... و يتميز الامر Copy بخيارات متقدمة . كتحديد نوع عملية النقل للبيانات .. هل هو نظام text او binary . او دمج ملفين في ملف . و هذه الخيارات . عادة لا نحتاجها . الا في حالات نادرة جداً ..

===== المثال العملي =====

### الجملة:

لديك في حاسبك قرصين . C و D

تريد ان تنسخ ملفات المستندات على القرص D . كذلك تريد إعادة تنظيم مجلد المستندات المكتظ بشئى أنواع الملفات ( EXE,PPT,DOC,MDB,ZIP,RAR,MP3 ,.... ) بحيث يكون كل نوع في مجلد خاص ...

نبدأ الآن بتقسيم المهام :

1- نسخ الملفات

2- إعادة التنظيم للمجلد الجديد بعد النسخ و تنضيم :

a. إنشاء مجلدات بأسماء مختلفة . باستخدام MKDIR

1b. استخدام الامر move لنقل الملفات مع الاعتماد على الامتدادات (wildcards)

-1 نسخ الملفات:

سنقوم بذلك باستخدام الامر Xcopy لنسخ مجلد المستندات ( my documents ) بأكمله إلى القرص D: . و بذلك نكتب ...  
xcopy "%USERPROFILE%\my documents" "d:\my documents" /E /I

2a . إنشاء مجلدات حسب النوع باستخدام MKDIR

```
cd "d:\my documents"  
mkdir "Applications"  
mkdir "Access"  
mkdir "powerpoint"  
mkdir "zipped files"  
mkdir "Word documents"  
mkdir "RAR files"  
mkdir "Songs"
```

2b. استخدام الامر move لنقل الملفات مع الاعتماد على الامتدادات (wildcards)

بإمكاننا الاعتماد على نوع الملف . او وصفه . او امتداده لفصل الملفات حسب نوعها . كالآتي:

```
move *.exe "Applications"  
move *.mdb "Acces"  
move *.ppt "PowerPoint"  
move *.zip "Zipped files"  
move *.doc "word documents"  
move *.rar "RAR files"  
move *.mp3 "Songs"
```

هذه الاوامر ستنقل كل نوع الى مجلد . فستجد كل ملفات powerpoint في مجلد PowerPoint . طبعاً فرضنا أنه منشأ أصلاً

هكذا نكون اتعمنا المثال الثالث .... و يؤسفني .. أنني مضطر للتوقف هنا و عدم المتابعة . حيث أنني لم أجد متابعة كافية من الأعضاء . صديح أن الموضوع وصل إلى 11 صفحة .. لكن لو نظرنا إلى الإحدى عشر صفحة . لن نجد أي مشاركة حقيقية ... كلها مشاركات شكر . طبعاً أنا أقدر كل من شكرني . و شجع الموضوع . لكنه شيء يفتأ يجزن . أن يكون ليس هناك سوى شخص أو شخصين استفادوا من كل هذا المجهود ... في الواقع كنت أريد التوسع أكثر بكثير من الموضوع الذي توقفت به ...كنت أريد ان أصل إلى :

- العمل مع المتغيرات ، و متغيرات النظام

تنفيذ أوامر وفق حسب تحقق شرط معين باستخدام if و for

التعامل مع مدخلات المستخدم ،

استخدام أوامر متنوعة في مهمات مختلفة

الخدع و أساليب استخدام ملفات البائس ، كاستخدام ملفات البائس ، لإنشاء سكريبتات فيجوال بيسك و تنفيذها ..

لكن للأسف .. لم أجد التفاعل الكافي . كذلك لم أجد من يتعاون معي في الموضوع ... فأنا غير قادر على شرح جميع الأوامر في الوندوز . و أحتاج من يساعدي بذلك . و إذا كنت سأكمل لوحدتي .. فإنني لا أملك الوقت الكافي و سأخبرك بالسبب في الوقت المناسب ..

في النهاية ... أشكر كل من ساهم في الموضوع . سواءً بمشاهدة . أو رد . أو تشجيع . أو عمل . و أتقدم بشكر خاص لكل من :

1-الأخ أبو فرج ... على متابعته المتميزة للموضوع

2-الأخ Slim shady . على تشجيعه للموضوع ... و تفاعله

3-الأخ mzar . على متابعته

4-الأخ New Star

5-الأخ يوسف ..

6-الأستاذ الكبير شرارة . فهو صاحب فكرة الموضوع ....



و الآن أستودعكم الله .. على أمل أن يستفيد أحد من ما كتبتة بالموضوع ...

للمتابع (بن) بصراحة هو متابع واحد .. الأي أبو فرج ..

توقف الموضوع، أين نتابع تعليقا؟

في الواقع لا أعلم .. أنا لا أملك مرجع معين للشرح ... لكن ..

تقريباً تعلمنا كل ما يهمنا من أسلوب التعامل مع الأوامر (و ما هي هذه الاوامر) ، كل ما عليك إذا أردت التوسع .. أن تبحث عن أوامر جديدة ... (لا أعلم من أين يمكنكم إيجادها ، جوووجل أمامكم...) فهذه الاوامر هي التي تمكنكم من عمل أي شيء ... فعلاً هناك الامر RAR ، موجود مع برنامج WinRAR ، و كثير من البرامج قد تجدون معها مثل هذه البرامج الصغيرة ، التي يمكن استخدامها بملفات باتش ... و قد تجدون اوامر مختصة بالتعامل مع الدخول إلى السيرفرات كما هو الحال في telnet ، و هناك الكثير الكثير من الأوامر ، التي عن طريقها تستطيع عمل أي شيء تريده باليهاز .. مثل الأمر reg ... للتعرف على الاوامر المدعومة في وندوز.. اذهب إلى موجه الاوامر .. اكتب .. help ، و سترى الاوامر المدعومة مع الوندوز

و الآن ، بكل جزن و أسى .. ستكون هذه آخر مشاركة .. تجعل معلومة جديدة حول الباتش سكريبت في هذا الموضوع ...

**بسم الله الرحمن الرحيم ...**

**الدرس الرابع ....**

**1- المتغيرات في الباتشات ، و التعامل معها .....**

**2- الجمل الشرطية if**

**3- العلامات (labels) ....**

درس اليوم ، كبير نوعاً ما ... سنتناول فيه كثير من الأمور ... التي تعتبر نواة قوة الباتش سكريبت و التعامل معها ... فحتى الآن ، كل تعاملنا مع الباتش سكريبت ، لم يتعدى استخدام أوامر و سويتشاتها ... لكن الآن ، سنتعدى ذلك ، لنتعامل مع المتغيرات و الجمل الشرطية و العلامات ... و الآن فلنبدأ بشرح المتغيرات ...

**1- المتغيرات**

كما ذكرت سلفاً في درس "متغيرات النظام" أن كل متغير يتم ترجمته إلى قيمته عند كتابته بين علامتي "%" ، فإن المتغيرات العادية ينطبق عليها نفس الشيء .. فعلاً متغير "قرص النظام" الذي هو . systemroot ، عندما نريد الحصول على قيمته في ملف باتش و استخدامها نكتب ... %systemroot% ، و يتولى موجه الأوامر ترجمة المتغير إلى القيمة المناسبة ... سواءً كانت C أو d أو e ... الخ ... ، و بالتالي سيصبح الباتش قابل للعمل على كل الأجهزة ، أي ليس مقتصرًا على الأجهزة التي تجوي النظام (ويندوز) في القرص : ... C

\*\*\* نحتاج إلى استخدام المتغير "systemroot" إذا أردنا القيام بنسخ ملف إلى مسار النظام ... او ما شابه ذلك

**الامر الذي نستخدمه لإنشاء متغير هو الامر Set ..**

صيغته ..

set variable = value

variable : حيث اسم المتغير :

value : و قيمته هي قيمة نصية :

لاستخدام المتغير variable في أي مكان من ملف الباتش .. نضع اسمه بين علامتي "%" ، كالآتي

%variable%

فإنما كتبنا في ملف الباتش هذا المتغير بهذا الشكل ، فسيتم ترجمته إلى القيمة value

لكن .. لنفرض أننا نريد أخذ قيمة المتغير من المستخدم .. أي أردنا أن نعلم .. ما القيمة التي أدخلها المستخدم للتعامل معها ...

سنقوم عندها باستخدام سويتش خاص هو السويتش P/ و بالتالي تصبح صيغة الأمر ..

set /p variable=ur string

بالعربي

(المتغير) = (النص الذي سيظهر للمستخدم عند مطالبتك بادخال قيمة للمتغير) set /p

فمثلاً ما كتبناه أعلاه ، عند وضعه في باتش ، فإنه عند تنفيذه ، سيظهر للمستخدم عبارة "ur string" ، بعدها يكتب ما يريد .. و عند ضغطه Enter ، سيتم تحويل ما أدخل إلى قيمة المتغير variable ..

## 2-الجمل الشرطية if

عادة ما يكون استخدامها مع المتغيرات ، فمثلاً لو كان المتغير قيمته (س) ، افعل الامر الآتي ..  
و لها استخدام آخر ، حيث يتم عمل الامر المطلوب .. إذا كان ملف موجود ، كما أن لها استخدام عكسي ، هو if not ، أي إذا كانت العبارة غير صحيحة ... و بشكل عام صيغة الأمر if

if %variable% == string command

حيث :

string:القيمة التي يجب أن يعتمدها المتغير لتصبح العبارة صحيحة و يتم تنفيذ الأمر command  
command: الامر الذي سيتم تنفيذه

و هناك استخدام خاص .. حيث إذا كانت العبارة غير صحيحة ، فإن الأمر سيتم تنفيذه .. أما إذا كانت صحيحة فلن يتم تنفيذه ، هذا الاستخدام ، يتم بوضع كلمة not

if not %variable% == string command

في هذه الحالة .. سيتم عمل الأمر command طالما أن الجملة غير صحيحة ، فإذا كانت صحيحة ، لن يتم التنفيذ مئلاً إذا كانت القيمة الفعلية ل variable في كل من الحالات الآتية فإن الأمر command سيتم تنفيذه ..

variable : sdksgm  
variable : fas  
variable :damas  
variable :mtz

و هناك حالة جديدة لن يتم تنفيذ الأمر command فيها .. هي :

variable : string

أقصد إذا كانت قيمة variable هي "string"

## استخدام if exist

استخدام هذا الأمر يتم بالتحقق من وجود ملف معين ... فمثلاً إذا أردنا من الباتش أن ي حذف الملف file.doc إذا وجد .. نكتب  
if exist file.doc del file.doc

و قد نحتاج في بعض الأحيان إلى استخدام

if not exist

مثلاً نريد أن يقوم الباتش بنسخ ملف معين إلى النظام... فقط إذا لم يكن موجود فيه فيكون

if not exist %systemroot%\system32\mtz.cpl copy "%CD%\mtz.cpl"  
"%systemroot%\system32\mtz.cpl"

\*الامتداد .cpl يمثل عناصر التحكم الموجودة في لوحة التحكم ، فمثلاً ، appwiz.cpl ، هو عنصر إضافة و إزالة البرامج ، و هكذا فإن أي عنصر في لوحة التحكم ، له ملف cpl خاص في مجلد system32 و طبعاً بإمكاننا استخدام أي أمر غير الأمر copy ، بما يناسب احتياجاتنا ...

## 3-العلامات labels

تستخدم العلامات لتقسيم الكود الذي نكتبه و التحكم به ، فمثلاً الكود الآتي مقسم إلى 4 علامات ..  
يتم تحديد علامة (label) بوضع نقطتين رأسيين قبل الكلمة ..

@echo off  
:label1

```
echo this is label 1
:label 2
echo this is label 2
:label 3
echo this is label 3
:label 4
echo this is label 4
```

إذا كتبنا الكود الآتي في ملف باتش .. لن نجد أي فرق بينه و بين الآتي ، فهو بتنفيذ سطرًا سطرًا كما هو الحال مع هذا الباتش ..

```
@echo off
echo this is label 1
echo this is label 2
echo this is label 3
echo this is label 4
```

لكن لنرى الفرق ، يجب أن نتعلم استخدام الأمر goto ...  
يستخدم الأمر goto للتنقل بين العلامات ، فعلاً لو كتبنا ..

```
goto label 3
```

ذلك سيقوم بالقفز إلى السطر الذي بجوي العلامة label 3 ، فعلاً لو وضعنا السطر :goto label 3 في نهاية العلامة label 4 ، سيتم القفز عندها إلى العلامة label 3 . هـ هكذا ، فستكون النتيجة ملف باتش غير منتهي كما في الصورة

```
@echo off
:label 1
echo this is label 1
:label 2
echo this is label 2
:label 3
echo this is label 3
:label 4
echo this is label 4
goto label 3
pause
```

كلما وصل الكود إلى هذه النقطة عاد إلى النقطة أعلاه

هذا السطر يستحيل الوصول إليه حيث أنه لن يأتي دوره أبداً في التنفيذ

### الآن إلى التنفيذ :

حتى الآن نا تعلمناه ، هو كيفية استخدام هذه الاوامر .. و لم نرى استخدام حقيقي .. الآن سنحاول عمل باتش يجمع كل ما تعلمناه اليوم .. لنبدأ ..

الباتش الذي سنصنعه ، هو قائمة تمكن المستخدم من الاختيار واحد من اثنين ، إما أن يقوم بتفحص القرص الصلب ، أو إلغاء التثبيت ... و هذا هو

```
@echo off
rem علامة البداية
:begin
```

```
cls
echo Welcome To MTZ system tools
echo 1-Run DiskDefrag
echo 2-Run Chkdsk
echo 3- Exit
rem السطر الآتي ينشئ متغير و يطلب قيمته من المستخدم
set /p userchoice=Please Enter then press enter
rem الآن سنتعامل مع ما أدخله المستخدم
rem إذا كان ما أدخله المستخدم هو الرقم 1 فان المتغير ستصبح قيمته 1 و تتحقق العملية التالية و تنفذ الأمر
if '%userchoice%' == '1' defrag %systemdrive%
rem بعد أن قام موجه الأوامر بمعالجة العملية الشرطية أعلاه سواء كانت تنفذت أم لا . سيعود لمعالجة السطر الآتي (العملية الشرطية)
if %userchoice%==2 chkdsk %systemdrive%
if %userchoice%==3 exit
rem الآن يجب أن يكون ما طلبه المستخدم قد تم .. لذلك بعد سيكون هذا السطر إعادة للبانش من بدايته حتى يختار المستخدم إذا ما
أراد الخروج
goto begin
```