

Written Sp The-o-Ne

بسم الله الرحمن الرحيم

نقاط الدرس:

- تعريف الأحرف الوصفية الخاصة (metacharacters).
- استخدام بعض الأحرف الوصفية مع بعض الأوامر:
 ١. استخدام العلامة النجمية (*) مع الأمر (ls).
 ٢. استخدام العلامة الإستفهامية (?) مع الأمر (ls).
 ٣. استخدام الأقواس المربعة ([]) مع الأمر (ls).
- استخدام الفاصلة المنقوطة (;) لربط أكثر من أمر.
- استخدام الماسورة (|) (pipe).

• الأحرف الوصفية الخاصة (metacharacter):

يوجد في أنظمة ال (*nix) ككل بعض العلامات أو إن صح التعبير أحرف يكون لها معنى خاص بالنسبة للغلاف . حيث أن هذه العلامات تكون معرفة من قبل النظام أنها تقوم بعملية معينة وأذكر هنا بعض هذه الأحرف والبقية في دروس قادمة بإذن الله تعالى:

>	<	>>	>2		;	[]	*	?
---	---	----	----	--	---	----	---	---

وتستخدم عادةً هذه الحروف لكي تجعل بعض الأوامر تعمل بطريقة معينة أي أن هذه العلامات عند كتابتها مباشرة على سطر الأمر من دون ذكر أي أمر فإن النظام سوف يعطيك رسالة خطأ أن ما أدخلته لا يعتبر أمراً. وهكذا نستنتج أنه لا يمكن استخدام هذه الأحرف من دون أمر.

• العلامة النجمية (*) واستخدامها مع الأمر (ls):

تستخدم العلامة النجمية عادة لكي تقوم بسرد المحتويات الداخلية للأدلة الحالية. ويمكن لهذه العلامة أن يتم استخدامها مع أكثر من أمر وسوف نستعرض هنا كيفية عملها مع الأمر (ls) وفي الدروس القادمة سوف نأخذ تطبيقات أخرى عنها.

يمكن استخدام العلامة النجمية مع الأمر (ls) وذلك لكي نجعل الأمر يعمل بطريقة مختلفة عن طريقة عمله عندما يكون لوحده أي يمكن أن نشبه استخدام مثل هذه العلامات مع الأوامر على أنها خيارات كما تعرفنا عن ماهية الخيارات في دروس سابقة. وعند استخدام العلامة النجمية (*) مع الأمر (ls) مباشرة فإنه سوف يتم عرض محتويات الدليل الحالي وأيضاً عرض محتويات الأدلة المندرجة تحته إن

كان هناك أي أدلة أي أن هذه العلامة عند وجودها مع الأمر (ls) فإنها تقوم بنفس عمل الخيار (R) الذي قمنا بمناقشته سابقاً. لاحظ المثال التالي:

```
bash-2.05$ ls *  
dir2
```

في المثال السابق لاحظ أن استخدام هذه العلامة قد قام بنفس عمل الخيار (R). ولكن الفرق بينهما هو أن الخيار (R) كما تعلمنا سابقاً يقوم بعرض محتويات الدليل الحالي وإذا كان هناك أي دليل مندرج تحت هذا الدليل فإنه سيتم إستعراض محتويات وإذا كان في المحتويات المستعرضة أي دليل سوف يعرض محتوياته أيضاً أي بشكل متسلسل. أما العلامة (*) فهي تقوم فقط بعرض المحتويات الداخلية لمحتويات الدليل الحالي وإن كان هناك أي دليل في هذا الدليل الفرعي فإنه لن يتم إستعراضه.

ويمكن أن نجعل هذه العلامة عندما يتم دمجها مع الأمر (ls) أن تقوم بعمل أكثر تحديداً من الأمر مثلاً كأن يتم تنفيذ هذا الأمر على أدلة معينة فقط في الدليل الحالي كأن نحدد مثلاً أن الأدلة الذي يجب أن يطبق عليها الأمر يجب أن تكون مبدئية بحروف معينة. لاحظ المثال التالي:

```
bash-2.05$ ls d*  
dir2
```

في المثال السابق لاحظ أننا حددنا أن أمر الاستعراض سوف يطبق فقط على الأدلة التي تبدأ فقط بالحرف (d) وهنا نحن نمتلك الدليل (dir1) لذلك سوف يتم عرض محتوياته الداخلية و في هذه الحالة عندما يقوم النظام بتنفيذ الأمر فإنه يقوم بالبحث عن الأدلة التي تكون مبدئية بالحرف (d) وإذا لم يجد أي دليل فإن النظام سوف يظهر لك رسالة خطأ أنه لا يوجد دليل ملف مبتدأ بالحرف الذي طلبته وذلك لأننا لا نملك في مسارنا الحالي دليل يحتوي على هذا الحرف. لاحظ المثال التالي:

```
bash-2.05$ ls c*
```

```
ls: c*: No such file or directory
```

بما أننا في دليلنا لا يوجد أي دليل يبدأ بالحرف (c) فإن النظام سوف يرد عليك برسالة خطأ مفادها أنه لا يوجد في دليلك الحالي أي دليل يبدأ بالحرف (c). كما أنه يمكننا أن نجعل التحديد أكبر كأن يتم ذكر أكثر من حرف. لاحظ المثال التالي:

```
bash-2.05$ ls di*
```

```
dir2
```

بما أننا نملك في دليلنا الحالي هذه الدليل فإنه من الطبيعي أن يتم سرد محتوياته ومحتويات الأدلة الموجودة تحته.

• العلامة الإستفهامية (?) وإستخدامها مع الأمر (ls):

وتستخدم هذه العلامة عادة لكي تجعل الأمر يقوم بعمل أكثر تحديداً فعندما يتم دمجها أو ذكرها مع الأمر (ls) مثلاً فهي تعني أن هذا الأمر سيعمل بطريقة

مختلفة عن طريقته الطبيعية وهكذا. وتستخدم عادةً هذه العلامة لكي تحصر عدد معين من الحروف. لاحظ المثال التالي:

```
bash-2.05$ ls ?
```

```
ls: ?: No such file or directory
```

في المثال السابق يخبر الغلاف النظام أنه يجب أن يقوم الأمر (ls) بعرض فقط الأدلة المكونة حروفها من حرف واحد فقط وعندما يجد النظام في الدليل المتواجد فيه أنت دليل يكون عدد حروفه واحد فقط فإنه سيتم عرض محتوياته الداخلية. من خلال ما سبق نستنتج أنه عند وجود هذه العلامة فهي تعني عدد معين من الحروف. ولكي تتضح الصورة بشكل أكبر سنجعل الأمر محصور على أن يعرض الأدلة التي يتكون عدد حروفها من أربعة حروف فقط وحددنا أربعة حروف لأننا نمتلك في دليلنا الحالي دليل يتكون اسمه من أربعة أحرف وهو (dir1). لاحظ المثال التالي:

```
bash-2.05$ ls ????
```

```
dir2
```

في المثال السابق كان يوجد لدينا الدليل (dir1) وهو دليل مكون من أربعة حروف لذلك فإنه سوف يتم عرض محتويات هذا الدليل الداخلية وهي هنا الدليل (dir2).

كما يمكننا أن نجعل عمل امر ما أكثر تحديداً من عدد الحروف فقط كأن نذكر بعض الأحرف من الدليل المراد مثلاً لدينا في دليلنا الحالي عدة أدلة مسماها الأول فقط هو (dir) أما النهاية فهي متعددة بصورة تسلسلية مثل (dir1, dir2) وهكذا ونحن نريد أن نستعرض محتويات هذه الأدلة فإنه وباستخدام العلامة الإستفهامية مع الأمر (ls) فإن الأمر سيكون على الشكل:

```
bash-2.05$ ls dir?
```

```
dir2
```

في المثال السابق حدد أول ثلاثة أحرف أما بالنسبة لحرف الأخير فيعتبر متغير. كما أنه يمكننا أن نستخدم هذه العلامة في منتصف اسم مثل:

```
bash-2.05$ ls d?r?
```

```
dir2
```

في المثال السابق سوف يتم عرض أي دليل يتكون من أربعة أحرف وأول حرف هو حرف (d) والحرف الثالث هو (r). اعتقد أن الصورة اتضحت لك عزيزي القارئ.

• استخدام العلامات المربعة ([]) مع الأمر (ls):

كما تعلمنا سابقاً أن هذه العلامات عند إستخدامها مع أحد الأوامر فإنها تعمل وكأنها أحد الخيارات ولكنها وفي بعض الأحيان تكون أكثر فائدة من إستخدام الخيارات كما لاحظنا في العلامة الإستفهامية حيث أنه لا يوجد أي خيار يقوم بمثل عملياتها. ويمكننا إستخدام علامات الأقواس المربعة ولذلك لتحديد أيضاً عمل

الأمر على فئة معينة من محتويات الدليل الحالي. كأن نحدد مثلاً للنظام بأن يقوم بعرض كافة الأدلة الذي يكون أول حرف في اسمه هو الأحرف المحصورة بين القوسين المربعين. لاحظ المثال التالي حتى تتضح لك الصورة:

```
bash-2.05$ ls [a-z]*
dir2
```

لاحظ في المثال السابق أننا أخبرنا النظام انه يجب أن يكون أول حرف في دليل يقوم بإستعراضه لابد أن يكون أول حرف منه هو أحد الأحرف المحصورة بين (a) و (z) لو نتذكر أننا نمتلك دليل في دليلنا الحالي مسمى ب (dir1) وحيث أن هذا الحرف محصور بين الحرفين (a) و (z) فإنه سوف يتم سرد محتوياته وهي كما تلاحظ الدليل (dir2). لاحظ المثال التالي:

```
bash-2.05$ ls [e-z]*
ls: [e-z]*: No such file or directory
```

في المثال السابق نعرف أن الحرف (e) يعتبر بعد الحرف (d) أي أنه بتطبيق الأمر السابق سوف لن يظهر الدليل (dir1) وذلك لأن أول حرف من حروفه لاينتمي إلى المسار الذي قمنا بتحديدده. ولاحظ أنه لابد من إستخدام العلامة النجمية مع الأقواس المربعة والا فإن النظام سوف يعطيك رسالة خطأ لاحظ المثال التالي:

```
bash-2.05$ ls [a-z]
ls: [a-z]: No such file or directory
```

أي أنه لا يوجد دليل بالاسم ([a-z]) لذلك يجب الانتباه. كما أنه يمكننا أن نضع حرفاً واحد فقط بين القوسين المربعين وفي هذه الحالة فإنه سوف يتم تنفيذ الأمر مثل تنفيذ الأمر (ls d*) أي أن ذكر الأقواس كعدمها في هذه الحالة. لاحظ المثال التالي:

```
bash-2.05$ ls [d]*
dir2
```

حاول أن تقارن بين خرج هذا الأمر وخرج الأمر (ls d*).

• استخدام الفاصلة المنقوطة (;):

يتم استخدام الفاصلة المنقوطة عادة لكي تجعل النظام يقوم بتنفيذ أكثر من أمر واحد من خلال كتابته في سطر واحد وتأخذ الفاصلة المنقوطة الشكل العام التالي:

```
bash-2.05$ command1 ; command2 ; commandn
```

حيث أننا نعني ب (command1) أي الأمر الأول و (command2) تعني الأمر الثاني أما (commandn) فهي تعني العدد الأخير من الأوامر. فيما سبق تعرفنا عن كيفية عمل الفاصلة المنقوطة وحتى يتم فهما بشكل أكبر لاحظ المثال التالي:

```
bash-2.05$ ls -al ; cd dir1
total 27
drwxr-xr-x 4 toleg denizens 512 Sep 25 23:23 .
```

```
drwxr-xr-x 152 root wheel 5632 Sep 15 12:32 ..
-rw----- 1 toleg denizens 412 Sep 25 23:31 .bash_history
-rw-r--r-- 1 toleg denizens 778 Jul 22 00:10 .cfonce
-rw-r--r-- 1 toleg denizens 1245 Jul 22 00:10 .mailrc
-rw----- 1 toleg denizens 167 Jul 22 00:10 .plan
-rw-r--r-- 1 toleg denizens 751 Jul 22 00:10 .profile
drwxr-xr-x 2 toleg denizens 512 Sep 1 23:08 .welcome
drwxr-xr-x 3 toleg denizens 512 Sep 25 22:16 dir1
-rw-r--r-- 1 toleg denizens 11986 Sep 25 23:24 result
```

في المثال السابق نلاحظ أننا قمنا باستخدام أكثر من أمر في سطر واحد الذي سيحدث أن الأمر (ls -al) سوف يرسل للنظام وعندما ينتهي النظام من تنفيذه تقوم الفاصلة المنقوطة بجعل الغلاف أيضاً يقوم بالأمر الآخر أي يمكننا القول أن المثال السابق يكافئ تماماً المثال التالي:

```
bash-2.05$ ls -al
bash-2.05$ cd dir1
```

أعتقد أن الصورة اتضحت الآن. كما يجب التنبيه أنه يمكنك أن تقوم بذكر أكثر من أمرين إلى سطر الأوامر وباستخدام الفاصلة المنقوطة. لاحظ المثال التالي:

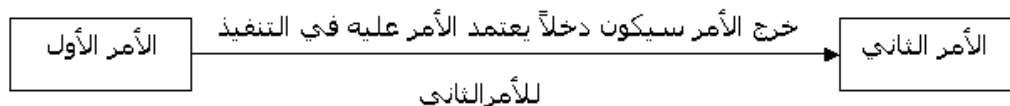
```
bash-2.05$ ls -al ; cd dir1 ; pwd
```

المثال السابق سوف يجعل النظام ينفذ ثلاثة أوامر. حاول أن تكتشفها بنفسك.

• استخدام الماسورة (pipe) (|) :

قبل أن نتعرف على وظيفة هذا الحرف يجب أن تعرف أين تقع في لوحة المفاتيح حيث أنها لا تستخدم بكثرة. يمكنك أن تصل إلى هذه العلامة من خلال المفتاح الذي يحتوي على الشرطة المائلة (|) وهو يوجد فوق زر (shift) الأيمن أي بجانب الزر (enter) وفي بعض لوحات المفاتيح تجده بجانب زر مسح حرف (backspace).

وظيفة هذه العلامة أنها تقوم أيضاً بربط أمرين في سطر أمر واحد وهي تقوم بعملية ربط أكثر من أمر في سطر واحد أعتقد أنه يتبادر إلى ذهنك الآن إذا هذه العلامة تقوم بنفس عمل الفاصلة المنقوطة ؟ والجواب هو لا وذلك لأن الفاصلة تقوم بربط أكثر من أمر في سطر واحد وذلك دون أن يكون أحد الأوامر مرتبط بالآخر أي كأنه يتم تنفيذ أمر في كل مرة. ولكن علامة الماسورة (|) فهي تقوم بأخذ الخرج من الأمر الأول وتوجهه إلى دخل الأمر الذي يليه. لاحظ الشكل التالي:



من خلال الشكل السابق يتضح لنا أن الأمرين يكونان مرتبطان ببعضهما وذلك لأن خرج الأمر الأول سوف يكون دخلاً للأمر الثاني حتى يتم تنفيذه. وحتى يتم فهم كيفية عمل هذه العلامة جرب المثال التالي:

```
bash-2.05$ ls -al /etc | more
```

```
total 35453
```

```
drwxrwxr-x 13 root    staff    3584 Sep 26 19:35 .
drwxr-xr-x 36 root    wheel     1024 Sep 19 15:24 ..
drwxr-sr-x  2 root    daemon   512 Oct 14 1994 .install.d
drwxr-s---  2 root    staff     512 Sep  4 21:29 RCS
lrwxrwxrwx  1 root    staff     10 Nov 20 1997 adm -> ../var/adm
--More--
```

(يمكنك قراءة خرج الأمر السابق كاملاً من خلال الملف (les9log) المرفق وستجد أيضاً جميع الأوامر التي تم تطبيقها في هذا الدرس في نفس الملف).
الذكرنا في دروسنا سابقة أن الدليل (etc) يتكون من ملفات النظام وإعداداته وحيث أنه كذلك فإنه من الطبيعي أن يحتوي على العديد من الملفات والأدلة. وفي هذه الحالة لو أردنا فقط أن نرى جميع الملفات الظاهرة والمخفية في هذا الدليل فإنه من الصعب رؤيتها جميعها دفعة واحدة وذلك لكثرتها. لذلك يفيد هنا استخدام علامة الماسورة (|) ففي المثال السابق طلبنا أولاً من النظام أن يقوم بإستعراض جميع محتويات الدليل (etc) وعندما يأتي النظام لتنفيذ هذا الأمر فإنه سينتبه لوجود علامة الماسورة (|) لذلك فإنه وقبل أن يرسل لك خرج الأمر فإنه يوجه خرج أمر الإستعراض إلى الأمر (more) والذي يستخدم لكي يعرض نص ما على الشاشة وسيتم مناقشته لاحقاً بإذن الله تعالى. لذلك هنا فإن الخرج القادم من أمر إستعراض الدليل (etc) سيدخل إلى الأمر (more) وعند تنفيذ الأمر على خرج الأمر الأول فإنه يرسل للمستخدم الخرج النهائي وهو عرض محتويات الدليل بشكل دفعات وتنتقل إلى كل دفعة بإستخدام زر المسافة (space) ونلاحظ في نهاية كل دفعة من هذه الدفعات تنتهي بالجملة (--More--). من خلال ما سبق يتضح لنا أن هذه العلامة تقوم بإستخلاص عمل أكثر من أمر لكي يعمل وكأنه أمر واحد. ولكي يتضح الفرق بين الماسورة والفاصلة المنقوطة استخدم الأمر السابق ولكن استبدل الماسورة بالفاصلة المنقوطة ولاحظ الفرق.
ومن الجدير بالذكر هنا أن الفاصلة المنقوطة عند حدوث أي خطأ في تنفيذ أمر ما فإنها لا تتوقف عن تكملة بقية الأوامر وذلك لأن الأوامر تكون غير مترابطة مع بعضها وإنما يكون كل أمر منفصل عن الأمر الآخر. أما بالنسبة لعلامة لماسورة (|) فإنها وعند حدوث خطأ في الأمر الأول فإنها تتوقف ولا تقوم أبداً بتنفيذ الأمر الذي يليه. لاحظ المثال التالي:

```
bash-2.05$ ls -al dir3 | more
```

```
ls: dir3: No such file or directory
```

لاحظ في المثال السابق وجد النظام خطأ في تنفيذ الأمر وذلك لأنه الدليل (dir3) لا يتعتبر أحد الأدلة الموجودة في دليلك الافتراضي وحيث أنه لا يوجد خرج من الأمر الأول فإنه لن يتم الأمر الثاني وهو الأمر (more). أما بالنسبة لو إستبدلنا علامة الماسورة بالفاصلة المنقوطة فلاحظ المثال التالي:

```
bash-2.05$ ls -al dir3 ; more
```

ls: dir3: No such file or directory

Usage: more [-dfln] [+linenum | +/-pattern] name1 name2 ...

لاحظ في المثال السابق عندما نفذ النظام الأمر الأول ووجد خطأ فإنه أظهر لك رسالة الخطأ ثم طبق الأمر الثاني وهو (more) ومن حيث أن هذا الأمر لا بد له من تحديد ملف لكي يعمل عليه فإنه أعطاك الطريقة التي يمكنك من خلالها استخدام هذا الأمر.

ونظراً لأن بقية الحروف الخاصة لا بد من تعلم بعض الأوامر الأخرى كأمر قراءة محتويات ملف ما وهكذا فإننا سوف نؤجلها لذلك الحين.

وتقبلوا خالص تحيات أخوكم المحب (The-oNe)
الرجاء إرسال مقترحاتكم وآرائكم على العناوين التالية:

The-one@pharaonics.net

OR

The_o0ne@hotmail.com

OR

The_o0one@yahoo.com