PRINCESS NORA BINT ABDULRAHMAN UNIVERSITY

College of Computer and Information Sciences
Computer Operating Systems (NET 321)
1432-1433 H
Lab 7 - Vi Text Editor , Shell Script

The topics:

- **Common shell environment variables**
- ***Rerunning commands***
- ***Adding Alias***
- **VI editor**
- **Shell script**

Common shell environment variables :

$HOME	your home directory
$PATH	list of directories used to find commands
$BASH	Contains the full path name of the *bash* command
$PWD	This is the directory that is assigned as your current directory
$OLDPWD	The directory that was the working directory before you changed to the current working directory

Rerunning commands:

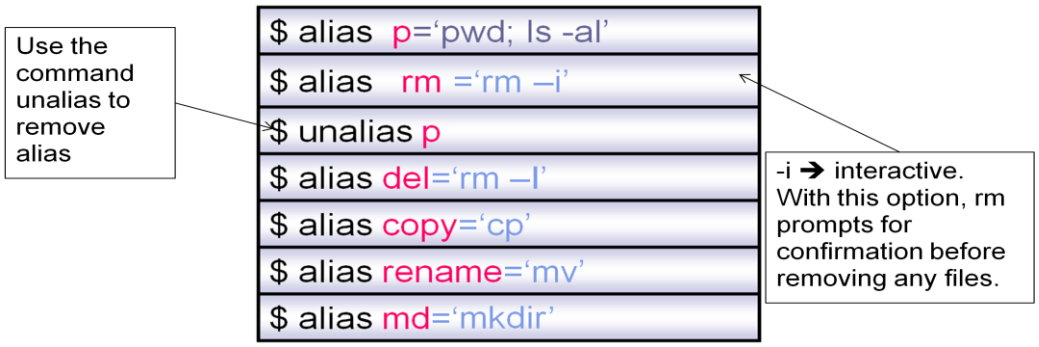
command line recall

- To view history list ⟶ **$ history**
- The last 8 commands ⟶ **$ history 8**
- Run previous command ⟶ **$!!**
- Run command number (!command number):

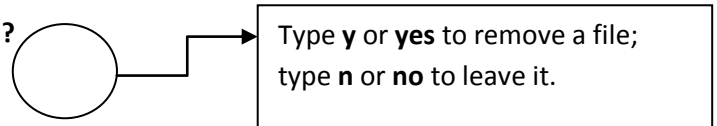
Ex. : **$!22** ⟶ If the command # 22 is pwd ➔ pwd will be run

Adding Alias:

- makes it possible to launch any command or group of commands by entering a pre-set *string* of characters.
- it allows a user to create simple names or abbreviations (even consisting of just a single character) for commands regardless of how complex the original commands are and then use them in the same way that ordinary commands are used.



Note:
rm with the "-i" option, will **prompt you before** every **removal.**
Ex . $ rm –i pnu
rm: remove regular file `pnu'?



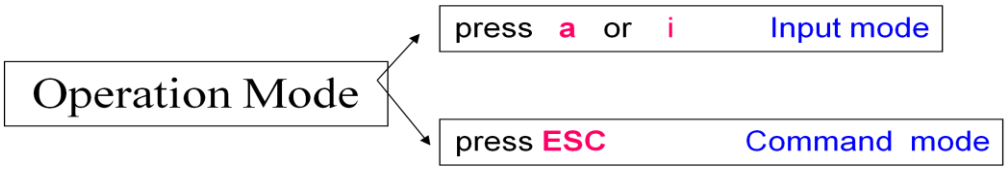
Entering VI:

- To call the VI editor and begin an editing session:
  $ vi *file-name*
- *File-name* does not exist → a new file will be created
- *File-name* exists → edit the existing file

Organization of VI :

There are two modes of operation in VI:

- ***Command mode***: where you tell the editor what you want it to do.
- ***Text input mode***: the part of the editor where you inter material (text, data, or program code) .



a : insert after the current position(cursor position).

i : insert before the current position(cursor position).

Vi Text Editor keys:

- **x** delete current character
- **X** delete previous character
- **dw** delete word forward
- **dd** delete complete line
- **d0** delete from cursor to beginning of line
- **D$** delete to the end of the line

Show file name : Ctrl+g

Save and quit:

ZZ or :wq	**Save and quit**	Then press ENTER
:w	**Save only**	
:q	**Quit (nothing added to save)** used when there are NO unsaved changes. If you need to save, vi warns you: E37: no write since last change(add ! To override)	
:q!	**Quit and don't save**	

Undo and Redo:

- **u** Undo
- **Ctrl+r** Redo

Run shell command:

:!command

For example:
:!id
:!ls
:!ps

Command mode	Input mode
You can do the following: - **Change current position(move the cursor).** - **Delete.** - **Show file name.** - **Save and quit.** - **Undo & Redo.** - **Run shell command.**	**Enter(type or write):** **- text, data and code.**

To change the current position(move the cursor):
in the command mode use the arrow in the keyboard
↑↓ → ←

Shell script:

What is Shell Script?

Normally shells are interactive. It means shell accept command from you (via keyboard)and execute them. But if you use command one by one (sequence of 'n' number of commands) , the you can store this sequence of command to text file and tell the shell to execute this text file instead of entering the commands. This is know as shell script.

Shell script defined as:
"*Shell Script is series of command written in plain text file*"

Shell script is just like batch file is MS-DOS but have more power than the MS-DOS batch file."

Why to Write Shell Script?

- Shell script can take input from user file and output them on screen.
- Useful to create our own commands.
- Save lots of time.
- To automate some task of day today life.
- System Administration part can be also automated.

How to write shell script

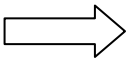
1) Use any editor like vi or mcedit to write shell script.
2) After writing shell script set execute permission for your script

syntax:
chmod *permission your-script-name*
Ex. chmod 755 test

3) Execute your script as

syntax:

bash your-script-name
sh your-script-name
./your-script-name



```
$ bash test
$ sh test
$ ./ test
```

```
$ vi first
#
# My first shell script
#
clear
echo "Knowledge is Power"
```

- After saving the above script, you can run the script as follows:

  $ chmod 755 first
  $ ./first

$ vi first	Start vi editor
# **# My first shell script** **#**	# followed by any text is considered as comment. Comment gives more information about script, logical explanation about shell script. ***Syntax:*** # comment-text
clear	clear the screen
echo "Knowledge is Power"	To print message or value of variables on screen, we use echo command, general form of echo command is as follows ***syntax:*** echo "Message"