OPS102 – Week 3 – File Systems – NDE / NDF

Activity 1: File Globing

When issuing Linux or Windows commands, it may be **more efficient** (less typing) to use **filename expansion symbols** also called **File Globing** to match files that share similar characteristics (e.g. same file extension) when issuing Linux commands.

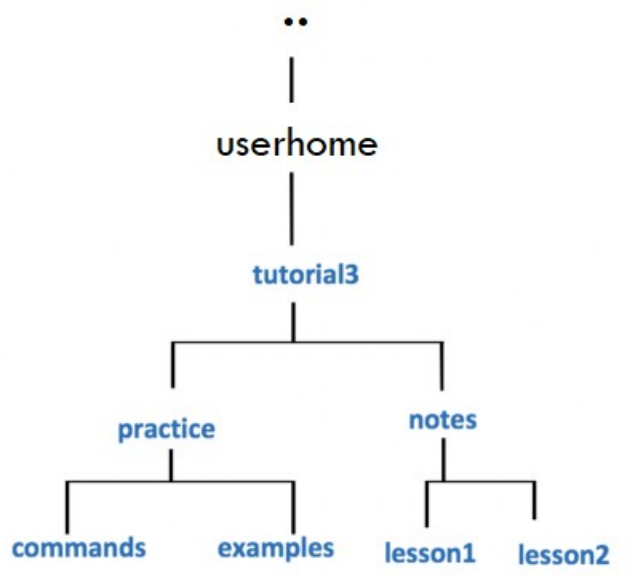
***Example**:* You can use a special character to indicate to the Bash shell to match all files that end with the extension ".txt" in your current working directory:

```
ls *.txt
a.txt b.txt c.txt 1.txt 2.txt 3.txt abc.txt work.txt
```

Below are the most common Filename Expansion symbols and how they are used for filename expansion:

Filename Expansion Symbol	Purpose
*	Asterisk (*) to represent **0 or more characters**
?	Question mark (?) to represent **exactly one character (any character)**
[ ]	Square brackets ([ ]) to represent and match for the **character enclosed within the square brackets**. It represents ONLY ONE character - it's like a **Question Mark (?)** but with **conditions or restrictions.**
[!]	Square brackets containing an exclamation mark immediately after the open square bracket ([!]) to represent and match and **OPPOSITE** character for the character enclosed within the square brackets.

Consider following file hierarchy for the activities in this section. This applies to both of Linux and Windows.



You will now get practice issuing file management commands using **filename expansion symbols**. We will be using the directory structure given above.

A great way to practice filename expansion, use the **touch** command on Linux to create a lot of empty filenames (for windows use any preferred way to create such files.), write the **ls/dir** commands that use **filename expansion**, predict the filenames that will be display, and finally run the command to check your work.

Perform the following steps for Linux and repeat them for windows using equivalent commands learnt previously:

1. Issue a Linux command to move to the **examples** directory
   (i.e. under *practice* directory as shown in diagram to the right).

2. Issue a Linux command to confirmed that you have moved to the **examples** directory.

3. Issue the **touch** command to create the following empty text files in the *examples* directory:
   (note *upper* and *lowercase* letters)

abc.txt
def.text
hij.TxT
1a4.txt
123.TXT
456.txt
6u9.txt

ab2.html
1234.txt
abcdef.txt
abcde.txt

4. If you encounter errors, then make corrections (eg. **viewing directory contents**, **check for correct filename syntax**, **case sensitivity**, **missing files**, **files in the wrong location**, etc.)

5. Issue the **ls** command to get a listing of files in your *examples* directory.

   The output should look identical to the diagram displayed below.
   You can refer to this listing to see all files so you can then predict the output from Linux commands that use filename expansion symbols.

```
ls
123.TXT  1234.txt  1a4.txt  456.txt  6u9.txt  ab2.html  abc.txt  abcde.txt  abcdef.txt  def.text  hij.TxT
```

6. What do you think the output will be from the following Linux command?
   ls ???.txt
   Write down the expected output on paper, then **issue the command** to check your answer.
7. What do you think the output will be from the following Linux command?
   ls ?????.txt
   Write down the expected output on paper, then **issue the command** to check your answer.
8. What do you think the output will be from the following Linux command?
   ls ??????.txt
   Write down the expected output on paper, then **issue the command** to check your answer.
9. What do you think the output will be from the following Linux command?
   ls [0-9].txt
   Write down the expected output on paper, then **issue the command** to check your answer.br>Did the command work?
   What does this teach you about the character class [ ] symbol?
10. What do you think the output will be from the following Linux command?
    ls [0-9][0-9][0-9].txt
    Write down the expected output on paper, then **issue the command** to check your answer.
11. What do you think the output will be from the following Linux command?
    ls [a-z][a-z][a-z].txt

Write down the expected output on paper, then **issue the command** to check your answer.

12. What do you think the output will be from the following Linux command (using character class with UPPERCASE letters)?:
    ls [A-Z][A-Z][A-Z].txt
    Write down the expected output on paper, then **issue the command** to check your answer.
13. What do you think the output will be from the following Linux command (using character class using alpha-numeric characters)?
    ls [a-zA-Z0-9][a-zA-Z0-9][a-zA-Z0-9].txt
    Write down the expected output on paper, then **issue the command** to check your answer.
14. What do you think the output will be from the following Linux command?
    ls *.txt
    Write down the expected output on paper, then **issue the command** to check your answer. Did ALL text files get listed? Why not?
15. What do you think the output will be from the following Linux command?
    ls *.[tT][xX][tT]
    Write down the expected output on paper, then **issue the command** to check your answer. Did ALL text files get listed this time? If so, why?
16. **NOTE:** We have just been using filename expansion symbols just with the ls command. Filename expansion symbols can be used for ANY Linux file management command (e.g. **cat**, **more**, **less**, **cp**, **mv**, **rm**, **ls**, etc.).

    Let's get some practice issuing these other Linux file management commands.
17. Issue the following Linux command: **file *.[tT][xX][tT]**
    What is the purpose of this command? Which files are contained in this output?

18. Change to the **commands** directory using an **absolute** pathname
    (use the diagram on right-side for reference).

19. Issue a Linux command to confirm that you are now in the **commands** directory.

20. Issue the following Linux command (lowercase "l" NOT the number "1"):
    cp /bin/l* .
    View the contents of the contents directory. What did this command do?

21. Issue the following Linux command: **rm ***

    View the contents of the contents directory. What did this command do?

22. Issue the following Linux command (lowercase "l" NOT the number "1"):
    cp /bin/l? .
    View the contents of the contents directory. What did this command do?

23. Issue the following Linux command: **rm l[!s]**
    View the contents of the contents directory. What did this command do?

24. Use a text editor (nano or vi) to create the file called **ab** in the **commands** directory that contains the line of text below,
    and then save editing changes to this file:
    This is file ab

25. Use a text editor (nano or vi) to create the file called **cd** in the **commands** directory that contains the line of text below,
    and then save editing changes to this file:
    This is file cd

26. Use a text editor (nano or vi) to create the file called **ef** in the **commands** directory that contains the line of text below,
    and then save editing changes to this file:
    This is file ef

27. Issue the following Linux command: **cat ??**

    View the contents of the contents directory. What did this command do? Why does the output look strange?

    NOTE: Press the keys **ctrl-c** to return to the shell prompt.

28. Issue the following Linux command: **cat [!l][!s]**

    View the contents of the contents directory. What did this command do? Does the output look better? If so, why?