TU Dublin, Tallaght
Department of Computing
IT Scripting and Automation
Lab3

Closing Date for submission: by 23:00, Wednesday 13th of October 2021

Instructions:

- Create a MS Word document with screen-shots of the
    1) solutions,
    2) **and** where appropriate the running scripts of each exercise.
- Please include your name and x-number.
- Mistakes in uploading cannot be rectified after the closing date.
- Attempt all exercises.

Important: In all exercises it is your responsibility to provide sufficient evidence that your script works properly. Lack of evidence will result in less marks.

Linux

Exercise 1.

Take screenshots to demonstrate you have completed all the steps:

a) Create a File "**MyFile1.txt**" then show the permissions of the file and change the permissions to allow everybody (owner, users in the same group and other users) to read, write and execute the file

b) Create a second file "**MyFile2.txt**" assign the correspondent permission to allow **only** the owner of the file to read and execute the file.

c) Create a third file "**MyFile3.txt**" assign the following permissions:
    - **Owner**: read and write.
    - **Group**: read, write and execution.
    - **Others**: execution.

Important: Provide evidence of every step.

Exercise 2.

Create the commands that will do the following, (***take screenshots to demonstrate you have completed all the steps***):

1. Remove all files that begin with a single letter, have an **a** in the middle and end a single letter, i.e., it will remove files with the names such as d**a**y, p**a**w, …

2. List all the files that begin with two letters, have an **e** in the name and end with 1 or more letter.
3. List all the files that begin with a letter, have an **a** in the name and end with a letter **s**.
4. List all the files that begin with **fil** and end with 1 or more letter.
5. Copy all files beginning in **d** to the **tmp** directory into a new directory called **tmp_copy**.
6. Remove all files ending with **w**.
7. List all files beginning with **fi** and ending in **s**, **t** or **x**.
8. List all files that begin with t or h and ends with **1,2,3,4,5,6,7,8,9** (h1,h2,..)

– You need to create empty files to test your commands. Use command **touch** to create files.
Hint: I demonstrated the use of ***touch {a..c}a{x..z}*** in the lab to create a range of empty files.

Exercise 3.

Pre-requisites - execute the following commands:

- touch file1.sh
- touch file2.py
- touch file3.txt

For each, explain the outcomes of the following instructions:

a) env | grep MAIL
b) Explain the difference between: env | grep USER and env | grep $USER
c) ls | grep 'My*'
d) ls *.{py,sh,txt}
e) ls *[ty]
f) touch nfile1243.xml
g) mv ?file*.xml mydir **Note**: "mydir" is a directory and it must exist

Exercise 4.

Create the shell scripts and provide screenshots to show the output:

Using a **variable $X=5** and the **echo** statement.
Write a script that print the following output:

```
$ A quote is “, backslash is \, backtick is `.
A few spaces are ; dollar $. $X is 5
This is \ a backslash
This is a “ a quote and this is \ a backslash
```

Exercise 5.

Create the shell scripts and provide screenshots to show the output:

a) Create a Shell script that will read the name of the
   - **module** and the
   - **week-number** (subdirectory)

   when passed as command-line **parameters** (parameter1 and parameter2) when invoking the script. Then create the following folder structure based on this information:

   `~/ModuleName/WeekNo/`

b) For each check if the directory already exists and if so, display a message with the appropriate information (*hint: use if statements. See class notes if needed*).

c) Execution sample: `./scriptDirCreation.sh itsa week1`

d) Output: the directory **itsa** and the subdirectory **week1** must be created. If they already exist, then a message indicating that must be displayed.