Stat 292 - Recitation 3

Unix Terminal - wget

Orçun Oltulu

31 / 3 / 2021

Exercise 1:

Part A:

Using echo command, print 'Hello World!'.

```
# echo 'Hello World!'
```

Part B:

Using echo command and -e option print;

'Hello World!'

'I just started using cygwin'

```
# echo -e 'Hello World!' '\nI just started using cygwin'
```

Part C:

Assign 50 * 3 + 12 to x and using echo print 'Variable x equals to: #valueofx'

Please go check how to assign variables and call them.

```
# x=$((50*3+12)); echo Variable x equals to: $x
```

Part D:

Print working directory, then set 'Desktop' as your working directory and list the files you have on your Desktop.

```
# pwd
# cd C:Users/orcun/OneDrive/Masaüstü
# ls
```

Part E:

Create a 'calculations' folder and create a txt file inside that folder.

```
# mkdir calculations
# touch calculations/calculations.txt
```

Part F:

Assign a=5, b=6, c=7, t=$a * b * c$ and k=$a + b + c + t$ using **declare** command with -i option. Then, write the values of t and k in calculations.txt file.

```
# declare -i a=5 b=6 c=7 t=a*b*c k=a+b+c+t
# echo -e "t=$t \nk=$k" > calculations/calculations.txt
```

Part G:

Using cat command, print the values inside your calculations.txt file.

```
# cat calculations/calculations.txt
```

Part H:

Copy everything inside calculations.txt file to another .txt, and then delete calculations.txt file.

```
# cp calculations/calculations.txt calculations/another_file.txt
# rm calculations/calculations.txt
```

Part I:

Use **expr** command to make logical comparisons for the following;

- $5 = 5$
- $8 < 5$ (**Hint:** use 8<5 for this type of comparisons)
- $8 > 5$
- $8 \neq 5$
- $2b = a$, where a=4, b=2

```
# skip skip skip skip skip
```

Part J:

Create myname variable for your name and using **expr** and **length** commands find how many characters your name has.

```
# myname=orcun
# expr length $myname
```

Part K:

Create a sequence of letters in reverse order.

```
# echo {Z..A}
```

Exercise 2:

Part A:

Print working directory, then set 'Desktop' as your working directory and list the files you have on your Desktop.

```
# pwd
# cd C:Users/orcun/OneDrive/Masaüstü
# ls
```

Part B:

Create a .txt file on your desktop and name it as 'Pizza'. Then, using echo command, add the favorite Pizza toppings in your Pizza.txt file, one by one.

```
# touch Pizza.txt

# echo 'Toppings:' > Pizza.txt
# echo '1. Mozzeralla' >> Pizza.txt
# echo '2. Pepperoni' >> Pizza.txt
# echo '3. Onion' >> Pizza.txt
# echo '4. Green Pepper' >> Pizza.txt
# echo '5. Corn' >> Pizza.txt

# cat Pizza.txt
```

Part C:

Create a 'Recitation3' file and move Pizza.txt file to that Recitation3 file you created. Then, set Recitation3 as your new working directory.

```
# mkdir Recitation3
# mv Pizza.txt Recitation3
# cd Recitation3
```

Part D:

Now, let's make a sandwich. Again, like the previous pizza example, you first create a Sandwich.txt file on your Desktop. Then, create a list of the items you want in your sandwich, but this time use **cat** command and create your whole list in one command.

```
# cat > Sandwich.txt
# Ingredients:
# 1. Turkey
# 2. Bacon
# 3. Ham
# 4. Cheese
# 5. Lettuce
# 6. Tomato
# 7. Mayo
```

Part E:

Create a 'Food' folder inside Recitation3 folder and move Pizza.txt and Sandwich.txt files into Food folder.

```
# mkdir Food
# mv *.txt Food
```

Part F:

Create 3 .txt files; test_1.txt ... test_3.txt in Recitation3 folder.

```
# touch test_{1..3}.txt
```

Part G:

Write 'This is the first test file' ... 'This is the third test file' for each .txt file.

```
# echo This is the first test file > test_1.txt
# echo This is the second test file > test_2.txt
# echo This is the third test file > test_3.txt
```

Part H:

Print every item inside test_1.txt ... test_3.txt files, using cat command.

```
# cat test_*
```

Exercise 3:

Part A:

Create a data folder in Recitation3 and set it as your working directory.

```
# mkdir data
```

Part B:

Use wget command to download '1987.csv.bz2' data set in Harvard Dataverse.

First, visit https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/HG7NV7

To download this data file you have to use following link: https://dataverse.harvard.edu/api/access/datafile/:persistentId?persistentId=doi:10.7910/DVN/HG7NV7/IXITH2

Use **wget –content-disposition https://dataverse.harvard.edu/api/access/datafile/:persistentId?persistentId=doi:10.7910/DVN/HG7NV7/IXITH2**

```
# wget --content-disposition https://dataverse.harvard.edu/api/access/datafile/:persis

# mv 1987.csv.bz2 data
```

Part C:

This file is zipped, unzip that file using **bzip2 -d 1987.csv.bz2** command.

```
# bzip2 -d data/1987.csv.bz2
```

Part D:

Print line, word and bytes for each file, using 'wc' command.

```
# cd data
# wc 1987.csv

### Output ###
# 1311827   1328612 127162942 1987.csv
```

Part F:

Print the first 10 rows, using head command.

```
# head 1987.csv
```

Part G:

Print last 20 rows.

```
# tail -n20 1987.csv
```

Part H:

Combine first 20 rows and last 30 rows in new_data.csv file.

```
# head -n20 1987.csv > new_data.csv
# tail -n30 1987.csv >> new_data.csv
```