

BUSA8090_Assignment_1_Rizwan

📎 Attachments	
☰ Column	
📅 Date	@Mar 18, 2021
☰ Property	
☰ Studied	

[Introduction](#)

[Question - 1](#)

[Question 2](#)

[Question -3](#)

[Question 3a](#)

[Question 3b](#)

[Question wise commands that was typed in the terminal](#)

[Conclusion](#)

Introduction

In order to solve the Assignment_1, I have created a git repository busa8090_assignment_1. And I have uploaded completed assignment files to the repository.

In the AWS unix, I have made `dataviz` directory and in that directory i have cloned the git `busa8090_assignment_1` repository.

Please find the assignment files in the below git repository.

https://github.com/mohrizwan1993/busa8090_assignment_1

Question - 1

For this question, I have followed the below steps to get the output.

```
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ touch -t 202103081800 foo
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ touch -t 202103081801 goo
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ touch -t 202103081802 hoo
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ ls
README.md  foo  goo  hoo
```

- I have created three files named `foo`, `goo` and `hoo`. I made use of `touch` command to modify timestamp for all three files. Oldest file in the directory would be `foo`
- Then I created a shell script `older.sh` using `vi` editor.

```
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ vi older.sh
```

```
#!/bin/bash
#this is our first script
ls -ltr $(find . -type f) | head -1
```

- In the first line, I have written `#!/bin/bash` shebang to invoke bash shell to interpret the script.
- Second line is a comment describing the script
- Made use of `ls` list command to list the files in the directory. The option `-l` is the long list format, `-t` sort the files by time, `-r` will list the files in reverse order. By doing this, we will get our oldest file first. Then I made use of pipeline to show only the first file. As we need just the oldest file in the list, I used `head -1`. The `find` option will find only items of type file in the current directory `.`

```
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ vi older.sh
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ chmod u+x older.sh
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ ./older.sh
-rw-rw-r-- 1 ubuntu ubuntu  0 Mar  8 18:00 ./foo
```

- After that, I changed the user permission for the shell file using `chmod` command. Doing this will enable us to execute the shell script. Then i ran the script using `./older.sh` command.
- We can see that our script is shwng the oldest file in the current directory.

```
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ pwd
/home/ubuntu/dataviz/busa8090_assignment_1
```

- Next, I added the current directory path to the \$PATH, as the current path is not in the path where shell script are checked, to execute the shell script directly.

```
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ $PATH
-bash: /usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin: No such file or directory
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ PATH=$PATH:$(pwd)
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ $PATH
-bash: /usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/home/ubuntu/dataviz/busa8090_assignment_1: No such file or directory
```

- You can see that `older.sh` executed directly from the directory.

```
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ older.sh
-rw-rw-r-- 1 ubuntu ubuntu  0 Mar  8 18:00 ./foo
```

Question 2

- For executing the second question, I created a new shell script `funny.sh` using `vi` editor

```
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ vi funny.sh
```

- Followed the same step as in first question for the first and second line of the script.
- Used flow control `if else` statement to check whether there is any argument passed to the script.
- `[$# -gt 0]` checks if there is arguments passed to the script. `$#` calculates no of arguments. `-gt` checks if it is greater than `0` then it means there are arguments passed to the script. So it will print `this is funny` . If there is no arguments, it will print `this is not funny`

```
#!/bin/bash
#Checking if the argument is passed or not
if [ $# -gt 0 ]; then
    echo "This is funny"
else
    echo "This is not funny"
fi
~
```

- We can see the execution of the above written script. I have tested three case scenario. With one arguments, without arguments and with multiple arguments.
- With one argument

```
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ ./funny.sh param1
This is funny
```

- With no arguments

```
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ funny.sh
This is not funny
```

- With multiple arguments.

```
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ funny.sh param1 param2 param3
This is funny
```

Question -3

Question 3a

- For this question, I tried downloading the file using two options. `-O` directly downloaded the file from the link with the pre defined name as used by the author.

```
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ curl -O https://www.staff.hs-mittweida.de/~wuenschi/data/media
/compbiolbook/chapter-10-shell-programming--case-cp.sh
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total   Spent    Left     Speed
100  321  100  321    0     0   165      0  0:00:01  0:00:01 --:--:--  165
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ ls
README.md  chapter-10-shell-programming--case-cp.sh  foo  funny.sh  goo  hoo  older.sh
```

- `-O` enabled me to download the file with the option of giving my own file name. In the last command, we can see that file is download and being listed in the directory.

```
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ curl -o case-cp.sh https://www.staff.hs-mittweida.de/~wuenschi
/data/media/compbiolbook/chapter-10-shell-programming--case-cp.sh
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total   Spent    Left     Speed
100  321  100  321    0     0   199      0  0:00:01  0:00:01 --:--:--  199
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ ls
README.md  case-cp.sh  foo  funny.sh  goo  hoo  older.sh
```

Question 3b

- In order to test the chime sound based on minutes, I created a shell script `test-chime.sh`

```
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ vi test-chime.sh
```

```
#!/bin/bash
#testing time and using chime sound
hour=$(date +%I)
minutes=$(date +%M)
if [ "$minutes" -le 19 ]; then
    echo "no chime"
elif [ "$minutes" -le 39 ]; then
    echo -e "\a"
elif [ "$minutes" -le 59 ]; then
    echo -e "\a"
    sleep 1 # sleep for one second
    echo -e "\a"
fi
```

- First two lines of the script is same as the first and second question.
- In the third line, I created a two variable one for `hours` and one for minutes. `%M` will show the minutes of the current time and assigned to the variable `minutes`. `%I` will show the hours in 12 hours format and assigned it to the variable `hours`. As we need just minutes, I used `minutes` alone for this question.
- First, the condition in the `if` statement will check if the minutes are less than 20. In the command, `-le` checks whether `minutes` variable is less than 20. If so, it will

display `no chime` message

- When the first `if` condition fails, Second `elif` checks whether `minutes` variable is less than 40. If so, the system will chime once. I used `echo` command along with `-e` to instruct interpreter to evaluate command after the escape sequence `\a`. Here, `a` after the escape sequence is used to alert the system. Since its used once, the system will alert or chime once.
- When this condition fails as well, we will proceed to the last `elif` statement, to check if the minutes are more than 40. Here I could have used `else` statement as well. But as time sequence is mentioned in the question, I used `elif` itself. In addition to chime, the `sleep` command is used to instruct the system to wait for a second before executing the second `echo` statement for second chime alert

```
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ ./test-chime.sh
no chime
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ time=$(date +%M)
ubuntu@ip-172-31-12-133:~/dataviz/busa8090_assignment_1$ echo $time
05
```

- Executed the script when the time was at five minutes past an hour. As a result, the script executed the first `if` statement and showed `no chime` alert message.

Question wise commands that was typed in the terminal

```
Question - 1:-
49 git clone "https://github.com/mohrizwan1993/busa8090_assignment_1.git"
50 ls
51 busa8090_assignment_1/
52 cd busa8090_assignment_1/
53 ls
54 touch -t 202103081800 foo
55 touch -t 202103081801 goo
56 touch -t 202103081802 hoo

58 vi older.sh
59 chmod u+x older.sh
60 ./older.sh

73 PATH=$PATH":"$(pwd)
74 $PATH
75 older.sh

Adding question 1 to git
80 git status
81 git add foo
82 git add goo
83 git add hoo
84 git add older.sh
85 git commit -m "Finished Question_1"
86 git status
87 git push
```

```
88 git status
89 git log
```

Question 2

```
100 vi funny.sh
101 chmod u+x funny.sh
153 ./funny.sh param1
154 funny.sh
155 funny.sh param1 param2 param3
```

Question 3a

```
158 curl -O https://www.staff.hs-mittweida.de/~wuenschi/data/media/compbiolbook/chapter-10-shell-programmi>
159 ls
160 rm chapter-10-shell-programming--case-cp.sh
161 curl -o case-cp.sh https://www.staff.hs-mittweida.de/~wuenschi/data/media/compbiolbook/chapter-10-shel>
162 ls
```

Question 3b

```
164 vi test-chime.sh
165 chmod u+x test-chime.sh
166 ./test-chime.sh
167 time=$(date +%M)
168 echo $time
```

Adding question 2, 3a and 3b to git

```
205 git status
206 git add case-cp.sh
207 git add funny.sh
208 git add test-chime.sh
209 git commit
210 git commit -m "Finshed 2nd and 3rd question"
211 git push
212 git log
222 history>history.txt
```

Conclusion

All the three questions have been successfully executed and uploaded to git repository. Question files are uploaded with commit message for each questions seperately. Have a great day 😊