Assignment on Linux - Day 5

Create an alias: Add an alias to your .bashrc file that creates a shortcut for a long command that you frequently use. Test the alias by opening a new terminal session and running the alias name.

Set an environment variable: Add an environment variable to your .bashrc file that sets the default text editor to Vim. Test the variable by running the command "echo $EDITOR".

Customize your prompt: Add a custom prompt to your .bashrc file that displays the current directory and time. Use the command "export PS1='[\e[32m]\u@\h [\e[33m]\w [\e[36m]\t [\e[0m]$ "'. Test the prompt by opening a new terminal session and navigating to different directories.

Debugging: Add the "set -x" command to the beginning of your .bashrc file to enable debugging output. Test the debugging by opening a new terminal session and watching the output as the file is executed. Look for any errors or unexpected behavior.

Display all block devices:

Show device partitions:

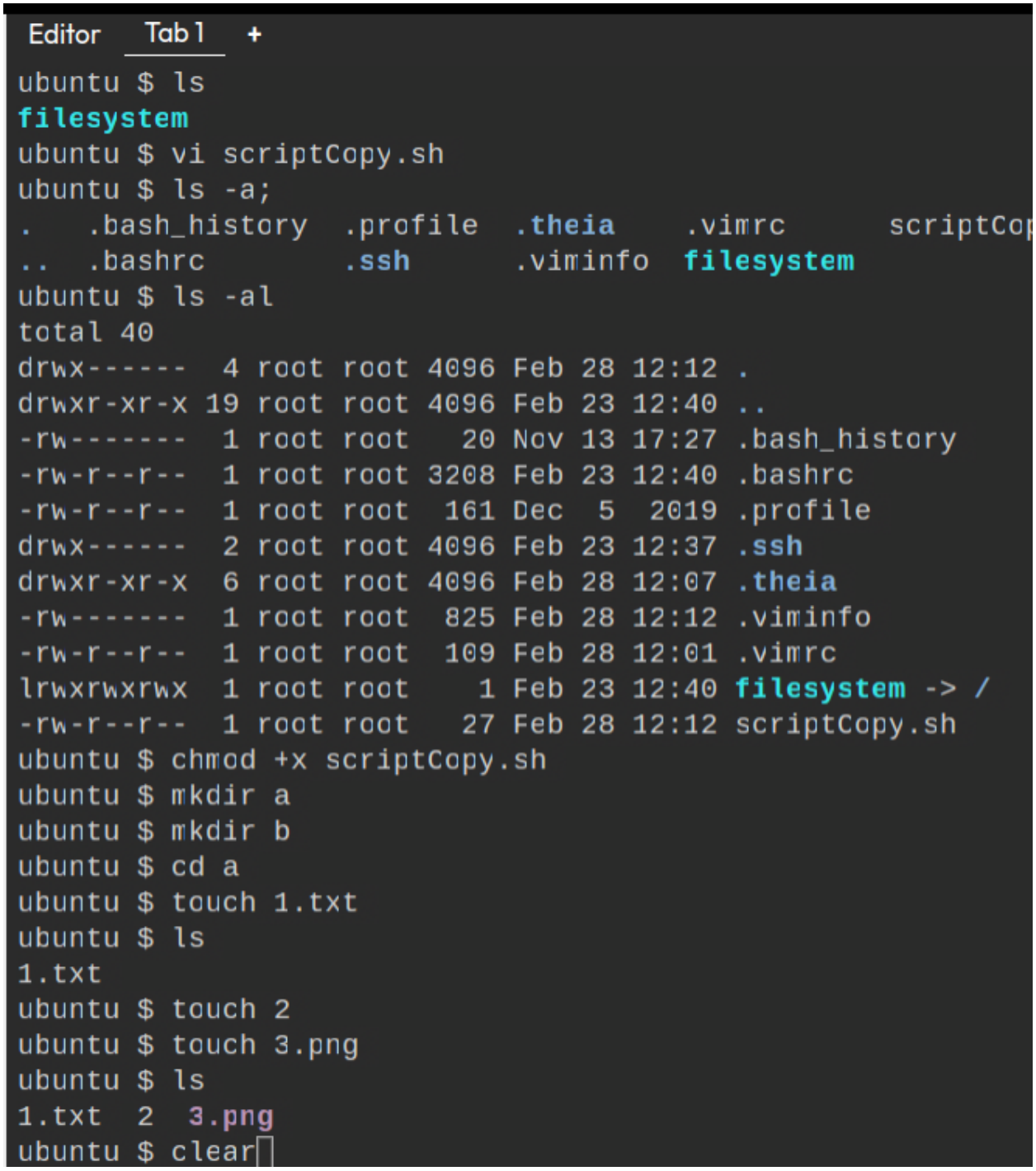
Schedule a script to run daily: Use the crontab command to schedule a script to run at a specific time every day. For example, schedule a script named backup.sh to run at 1:00 AM every day.

Bash Scripting: Day 1 Assignment

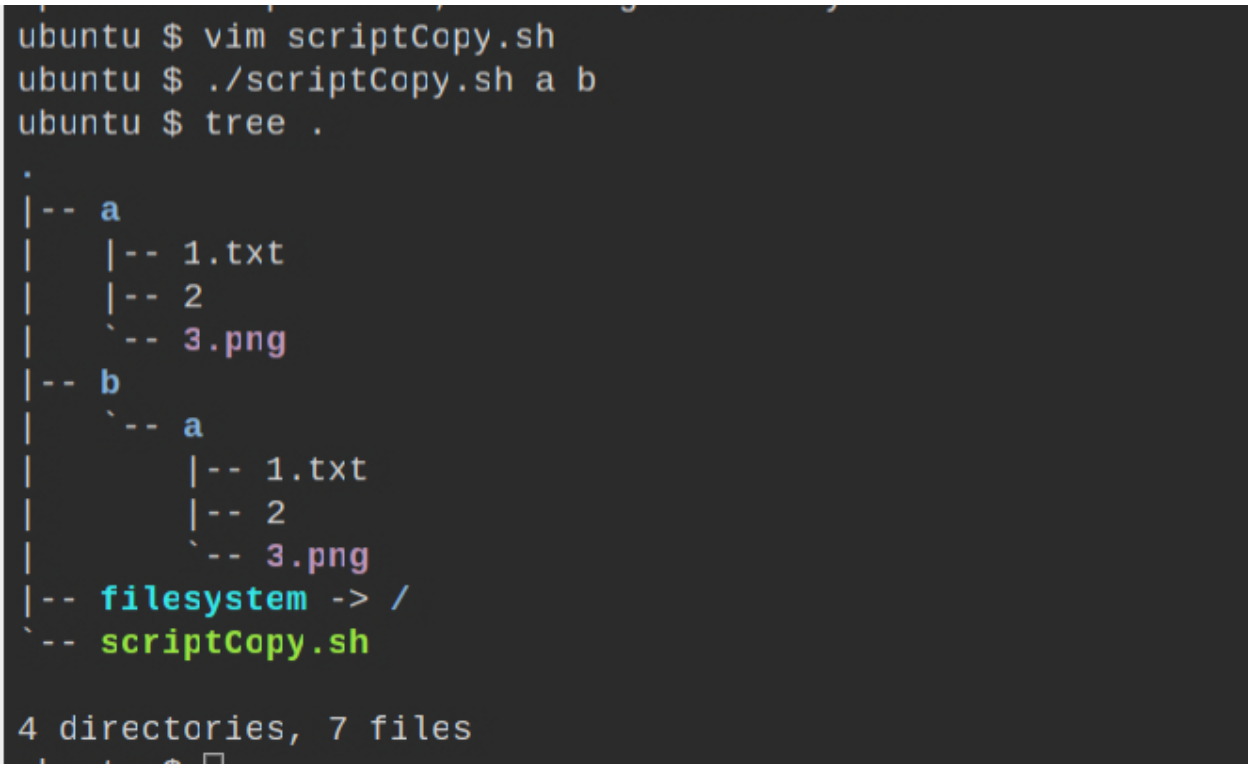
Revise the concepts that were taught in the class

Bash Scripting: Day 2 Assignment

1. A script to loop through a list of files and copy them to a new directory. Take input source and destination directory from the users.


Editor
Tab 1
+
ubuntu $ ls
filesystem
ubuntu $ vi scriptCopy.sh
ubuntu $ ls -a;
. .bash_history .profile .theia .vimrc scriptCop
.. .bashrc .ssh .viminfo filesystem
ubuntu $ ls -al
total 40
drwx------ 4 root root 4096 Feb 28 12:12 .
drwxr-xr-x 19 root root 4096 Feb 23 12:40 ..
-rw------- 1 root root 20 Nov 13 17:27 .bash_history
-rw-r--r-- 1 root root 3208 Feb 23 12:40 .bashrc
-rw-r--r-- 1 root root 161 Dec 5 2019 .profile
drwx------ 2 root root 4096 Feb 23 12:37 .ssh
drwxr-xr-x 6 root root 4096 Feb 28 12:07 .theia
-rw------- 1 root root 825 Feb 28 12:12 .viminfo
-rw-r--r-- 1 root root 109 Feb 28 12:01 .vimrc
lrwxrwxrwx 1 root root 1 Feb 23 12:40 filesystem -> /
-rw-r--r-- 1 root root 27 Feb 28 12:12 scriptCopy.sh
ubuntu $ chmod +x scriptCopy.sh
ubuntu $ mkdir a
ubuntu $ mkdir b
ubuntu $ cd a
ubuntu $ touch 1.txt
ubuntu $ ls
1.txt
ubuntu $ touch 2
ubuntu $ touch 3.png
ubuntu $ ls
1.txt 2 3.png
ubuntu $ clear


Editor
Tab 1
+
ubuntu $ tree .
.
|-- a
| |-- 1.txt
| |-- 2
| `-- 3.png
|-- b
|-- filesystem -> /
`-- scriptCopy.sh


ubuntu $ vim scriptCopy.sh
ubuntu $ ./scriptCopy.sh a b
ubuntu $ tree .
.
|-- a
| |-- 1.txt
| |-- 2
| `-- 3.png
|-- b
| `-- a
| |-- 1.txt
| |-- 2
| `-- 3.png
|-- filesystem -> /
`-- scriptCopy.sh
4 directories, 7 files

```
Editor   Tab 1   +
ubuntu $ cat scriptCopy.sh
#!/bin/bash

cp -r $1 $2


ubuntu $
```

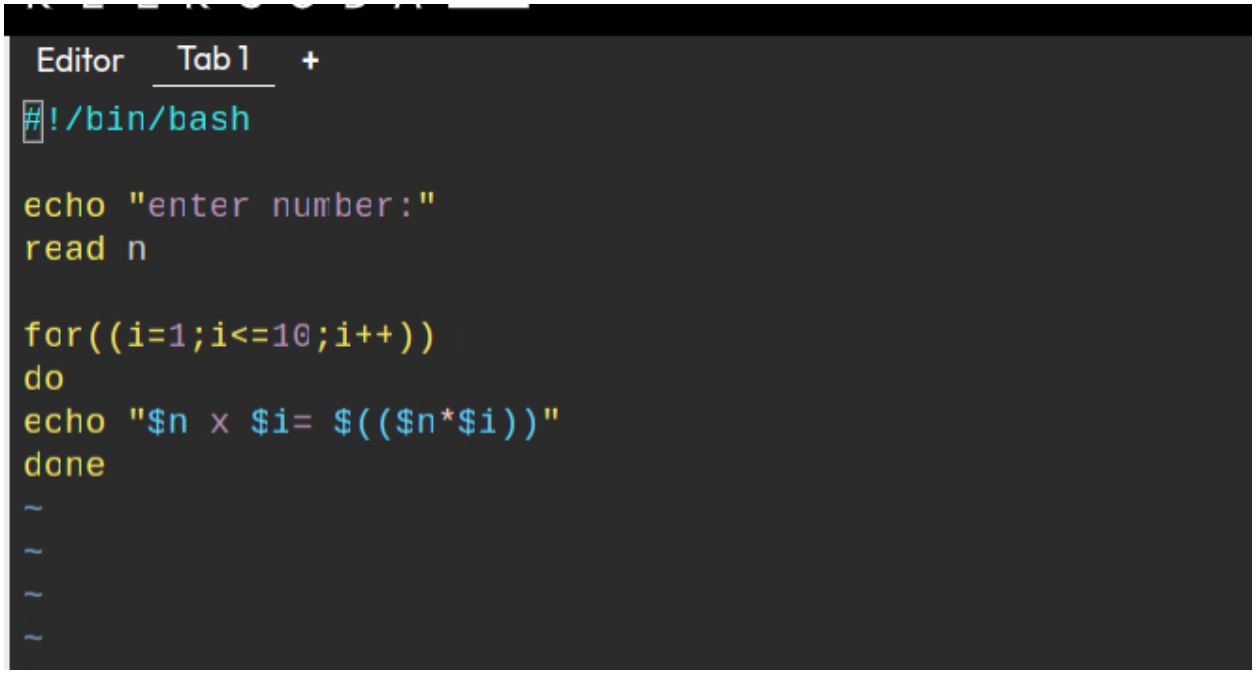
2. A script to prompt the user to enter a number and display whether it is even or odd.

```
Editor   Tab 1   +
#!/bin/bash

echo "enter number:"

read number

if((number%2 == 0))
then
  echo "$number is even"
else
echo "$number is odd"
fi

~
~
~
~
~
~
~
```

```
Editor    Tab 1    +
ubuntu $ vim scriptEvenOdd.sh
ubuntu $ ls -l
total 16
drwxr-xr-x 2 root root 4096 Feb 28 12:14 a
drwxr-xr-x 3 root root 4096 Feb 28 12:17 b
lrwxrwxrwx 1 root root    1 Feb 23 12:40 filesystem -> /
-rwxr-xr-x 1 root root   27 Feb 28 12:16 scriptCopy.sh
-rw-r--r-- 1 root root  129 Feb 28 12:22 scriptEvenOdd.sh
ubuntu $ chmod +x scriptEvenOdd.sh
ubuntu $ ./scriptEvenOdd.sh
enter number:
20
20 is even
ubuntu $ ./scriptEvenOdd.sh
enter number:
7
7 is odd
ubuntu $
```

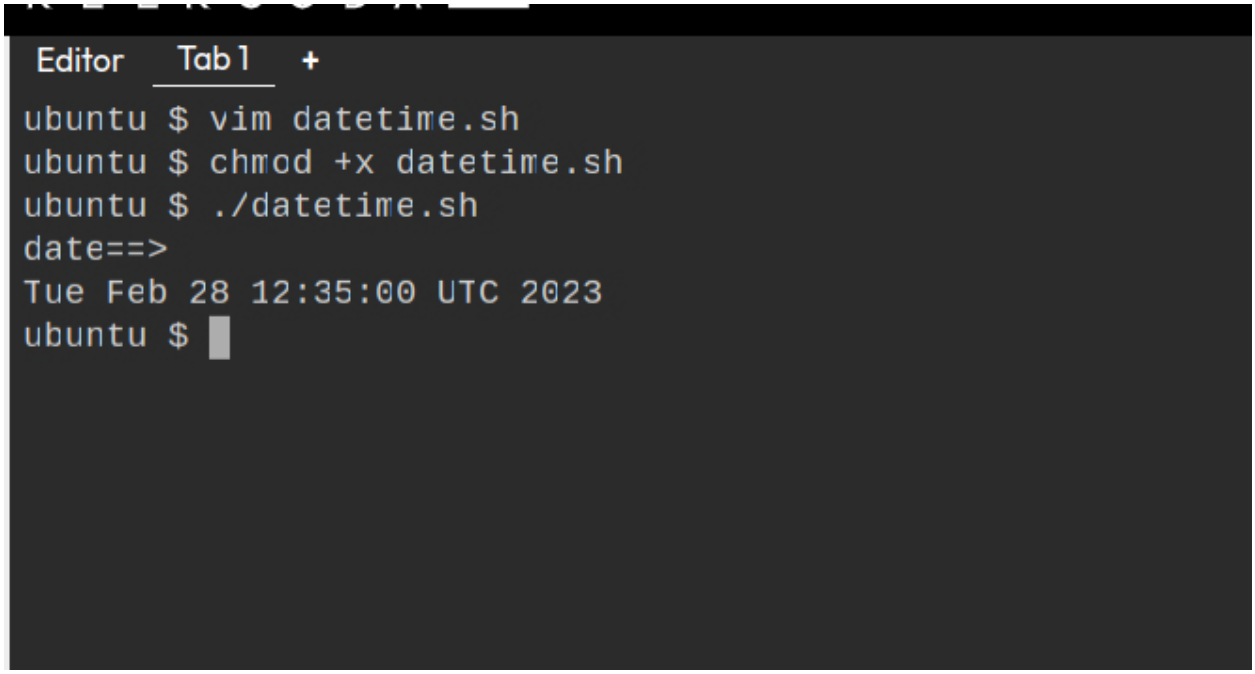
3. A script to generate a multiplication table for a given number.

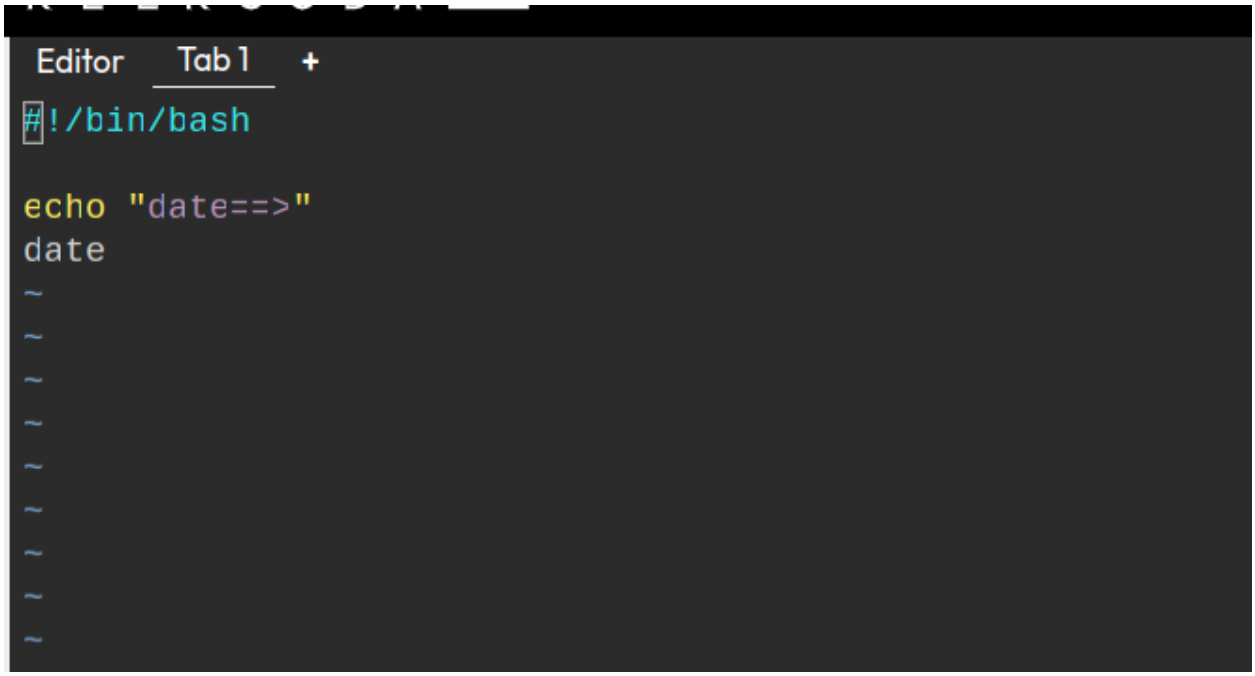
```
ubuntu $ vim table.sh
ubuntu $ ./table.sh
enter number:
5
5 x 1= 5
5 x 2= 10
5 x 3= 15
5 x 4= 20
5 x 5= 25
5 x 6= 30
5 x 7= 35
5 x 8= 40
5 x 9= 45
5 x 10= 50
ubuntu $
```




```
#!/bin/bash

echo "enter number:"
read n

for((i=1;i<=10;i++))
do
echo "$n x $i= $(($n*$i))"
done
```

4. A script to display the current date and time.






```
#!/bin/bash

echo "date==>"
date
```

5. A script to check if a user exists on the system.

```
ubuntu $ vim checkuser.sh
ubuntu $ chmod +x checkuser.sh
ubuntu $ ./checkuser.sh
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
```

6. Write a script to search for a given string in multiple files in a directory using a for loop.

```
ubuntu $ vim finddir.sh
ubuntu $ chmod +x finddir.sh
ubuntu $ vim finddir.sh
ubuntu $ ./finddir.sh a 2
a/2
ubuntu $ ./finddir.sh a 4
ubuntu $ ./finddir.sh a
find: missing argument to `-name'
ubuntu $ ./finddir.sh a 3
ubuntu $ ./finddir.sh a 1
ubuntu $ ./finddir.sh a 1.txt
a/1.txt
ubuntu $
```

```
#!/bin/bash

find $1 -name $2
```

7. Write a script to print the Fibonacci series up to a given number using a while loop.

```
#!/bin/bash

echo "Enter a number:"
read n

a=0
b=1

echo "Fibonacci series up to $n:"
echo -n "$a $b "


while ((b < n)); do
  c=$((a + b))
  echo -n "$c "
  a=$b
  b=$c
done
```

8. Write a script to find the largest and smallest numbers in an array of numbers using a for loop.

```
#!/bin/bash

numbers=(40 20 1 30 13)

min=${numbers[0]}
max=${numbers[0]}


for num in "${numbers[@]}"
do
  if ((num < min)); then
    min=$num
  fi

  if ((num > max)); then
    max=$num
```

```
  fi
done

echo "Minimum number: $min"
echo "Maximum number: $max"
```