

Mohamed Ahmed Mohamed Abd El gani
lab3

```
$ mycase.sh
You, 20 hours ago | 1 author (You)
1  #!/bin/bash
2
3  read -p "Enter a cahr: " char
4
5
6  if [[ ${#char} -eq 1 ]]; then
7      case $char in
8          [A-Z])
9              echo "UPper Case"
10             ;;
11          [a-z])
12              echo "Lower Case"
13              ;;
14          [0-9])
15              echo "Number"
16              ;;
17          "")
18              echo "Nothing"
19              ;;
20          *)
21              echo "Default"
22              ;;
23      esac
24  else
25      echo "enter only one character. your entered \"$char\""
26      echo "closing the script.."
27      sleep 2
28  fi
29
```

```

$ mycase-enhanced.sh
You, 20 hours ago | 1 author (You)
1  #!/bin/bash
2
3  read -p "Enter a char or strng: " input
4
5
6  if [[ ${#input} -eq 1 ]]; then
7      case $input in
8          [A-Z])
9              echo "Upper Case"
10             ;;
11          [a-z])
12             echo "Lower Case"
13             ;;
14          [0-9])
15             echo "Number"
16             ;;
17          "")
18             echo "Nothing"
19             ;;
20          *)
21             echo "Default"
22             ;;
23      esac
24  else
25      case $input in
26          [0-9]*[a-zA-Z]*)
27             echo "Mix"
28             ;;
29          [0-9]*)
30             echo "Numbers"
31             ;;
32          [A-Z]*)
33             echo "Upper Case"
34             ;;
35          [a-z]*)
36             echo "Lower Case"
37             ;;
38          "")
39             echo "Nothing"
40             ;;
41          *)
42             echo "Mix"
43             ;;
44      esac
45  fi
46
47  echo "Closing the script in 2 seconds..."
48
49
50

```

3

```
$ mychmod.sh
You, 20 hours ago | 1 author (You)
1  #!/bin/bash
2
3  for item in `ls ~`
4  do
5      if [[ -f $item ]];then
6          chmod u+x $item
7      fi
8  done
9
10
```

4-

```
$ mybackup.sh
You, 20 hours ago | 1 author (You)
1  #!/bin/bash
2
3  mkdir ~/backupfromscript/
4
5
6
7  for x in ~/.* ~/*; do
8      if [[ -f $x ]]; then
9          cp $x ~/backupfromscript/
10     fi
11 done
12
13 echo "backup finished! in ~/backupfromscript/"
14
```

5

```
$ mymail.sh
You, 20 hours ago | 1 author (You)
1  #!/bin/bash
2
3  #sudo apt install mailutils
4
5  users=`awk -F : '{if($3>=1000){print $1 }}' /etc/p
6
7  body=$(cat ./mtemplate)
8  echo "$body"
9
10 for user in $users
11 do
12     echo "$user"
13     echo "I will Send Email for : $user "
14     echo "$body" | mail -s "subject1" $user
15     echo "I will wait 5 Sec "
16     sleep 2
17 done
18
19 echo "Finish Send Emails"
20
```

6-

```
$ chkmail.sh
You, 20 hours ago | 1 author (You)
1  #!/bin/bash
2
3  checkMails() {
4      username=$(whoami)
5      while true; do
6
7          if [ -e /var/mail/$username ]; then
8              echo "New mails found!"
9              tail /var/mail/$username
10             else
11                 echo "No new mails."
12             fi
13
14             sleep 10
15         done
16     }
17
18     You, 20 hours ago • bash lab3
19     checkMails
```

7- answer in the img _____

```
$ q7.sh
You, 3 hours ago | 1 author (You)
1  #!/bin/bash
2
3  declare -i n1
4  declare -i n2
5  n1=1
6  n2=1
7  while test $n1 -eq $n2
8  do
9      n2=$(( $n2+1 ))
10     echo $n1
11     if [ $n1 -gt $n2 ]
12     then
13         break
14     else
15         continue
16     fi
17     n1=$(( $n1+1 ))
18     echo $n2
19 done
20 | You, 3 hours ago • Uncommitted changes
21 <<com
22 will print the n1 value only one time and then
23 it will continue to the next iter but it will find the n1 != n2
24 because we now have n2+1 so the loop will stop
25 com
26
27 #! the code provided in the lab didnt work on my machine i had to edite it like above
```

8.1

```
$ q8-select.sh
1  #!/bin/bash
2
3  select choice in "a. Press 1 to ls" "b. Press 2 to ls -a" "c. Press 3 to exit"
4  do
5      case $REPLY in
6          1)
7              ls
8              ;;
9          2)
10             ls -a
11             ;;
12          3)
13             echo "exit"
14             break
15             ;;
16          *)
17             echo "InvAlid chice"
18             ;;
19      esac
20  done
21
```

8.2

```
$ q8-while.sh
1  #!/bin/bash
2
3  while true; do
4      echo "please select a option"
5      echo "a. Press 1 to ls"
6      echo "b. Press 2 to ls -a"
7      echo "c. Press 3 to exit"
8
9      read -p "enter a option: " choice
10
11     case $choice in
12         1)
13             ls
14             ;;
15         2)
16             ls -a
17             ;;
18         3)
19             echo "exit"
20             break
21             ;;
22         *)
23             echo "InvAlid chice"
24             ;;
25     esac
26 done
27
```

```

$ myarr.sh
1  #!/bin/bash
2
3  while true; do
4      echo "Enter the number of elements for the array: "
5      read num_elements
6
7      if [[ "$num_elements" =~ ^[1-9][0-9]*$ ]]; then
8          while true; do
9              read -p "enter array as space sparated elements with only $num_elements or less : " -a array
10
11 >          if [ ${#array[@]} -eq $num_elements ] || [ ${#array[@]} -lt $num_elements ]; then...
13              echo "===--> ${#array[@]} Elements entered $(( ${#array[@]} - $num_elements ))
14                  places are empty<===--"
15                  break
16              else
17                  echo "You entered ${#array[@]} enter $num_elements nums as agreed before."
18              fi
19          done
20          break
21      else
22          echo "$num_elements is Invalid num Please enter a num >0 or not dicimal"
23      fi
24  done

```

```

$ myavg.sh
1  #!/bin/bash
2
3  flag=0
4  while [[ $flag < 1 ]] ; do
5  read -p "Enter array as space separated numbers to calculate the avg: " -a array
6      if [ ${#array[@]} -gt 0 ]; then
7          sum=0
8          cause=0
9          for num in "${array[@]"; do
10
11              if [[ "$num" =~ ^[1-9][0-9]*$ ]]; then
12                  sum=$((sum + num))
13              elif ! [[ "$num" =~ ^[1-9][0-9]*$ ]]; then
14                  flag=$((flag + 1))
15                  cause=$num
16              else
17                  echo "$num is not a valid number"
18                  break
19              fi
20          done
21
22      done
23
24      if [[ $flag < 1 ]];then
25          average=$((echo "$sum / ${#array[@]}" | bc))
26          echo "----->This is the avg $average <-----"
27          break
28      else
29          echo "plss enter a number this--> '$cause' is not a num"
30      fi
31
32  else
33      echo "Array is empty"
34  fi
35 done

```



```

mysql.sh
1  #!/bin/bash
2  mysql() {
3      num=$1
4
5      if ! [[ "$num" =~ ^[1-9][0-9]*$ ]]; then
6          read -p "write number to get its sqr: " num
7      fi
8
9      if [[ "$num" =~ ^[1-9][0-9]*$ ]]; then
10         local square=$((num * num))
11         echo "=====>the sqr of '$num' is <-- $square --> <-----="
12     else
13         echo "plss enter a number this---> '$num' ins not a num"
14     fi
15 }
16
17
18
19 mysql "$1"

```


