LAB 10

1.What test command should be used to test that /usr/bin is a directory or a File?

```
#!/bin/bash
file_or_directory="/usr/bin"
if [ -d "$file_or_directory" ]; then
  echo "$file_or_directory is a directory."
elif [ -f "$file_or_directory" ]; then
  echo "$file_or_directory is a file."
else
  echo "$file_or_directory is neither a directory nor a file."
fi
```

Output:

```
syedabdulbasit@Ubuntu:~/Desktop$ ./lab10.sh
/usr/bin is a directory.
```

2.Write a script that takes two strings as input compares them and depending upon the results of the comparison prints the results.

```
#!/bin/bash

if [ "$1" -gt "$2" ]; then
  echo "$1 is greater than $2."
elif [ "$1" -lt "$2" ]; then
  echo "$1 is less than $2."
else
  echo "$1 is equal to $2."
fi
```

Output:

```
syedabdulbasit@Ubuntu:~/Desktop$ ./lab10.sh 2 6
2 is less than 6.
syedabdulbasit@Ubuntu:~/Desktop$ ./lab10.sh 2 2
2 is equal to 2.
syedabdulbasit@Ubuntu:~/Desktop$ ./lab10.sh 2 0
2 is greater than 0.
```

3. Write a script that takes a number (parameter) from 1-3 as input and uses case to display the name of corresponding month.

```
#!/bin/bash
if [ "$#" -ne 1 ]; then
  echo "Usage: $0 <month>"
  exit 1  # Exit with an error code.
fi

month="$1"
case "$month" in
1)

echo "January"
;;
2)
echo "February"
;;
3)
echo "March"
;;
*)
echo "Enter choice  from 1 to 3"
;;
esac
```

Output:

```
syedabdulbasit@Ubuntu:~/Desktop$ ./lab10.sh 1
January
syedabdulbasit@Ubuntu:~/Desktop$ ./lab10.sh 5
Enter choice  from 1 to 3
```

4. Write a script that calculates the average of all even numbers less than or equal to your roll number and prints the result.

```
#!/bin/bash

rollno=89
# Function to check if a number is even
```

```
is_even() {
  if [ "$(($1 % 2))" -eq 0 ]; then
    return 0  # Even
  else
    return 1  # Not even
  fi
}

# Calculate the average of even numbers less than or equal to the roll number
sum=0
count=0

for ((i = 2; i <= $rollno; i += 2)); do
  if is_even "$i"; then
    sum=$((sum + i))
    count=$((count + 1))
  fi
done

# Check if there are even numbers before calculating the average
if [ "$count" -gt 0 ]; then
  average=$((sum / count))
  echo "The average of even numbers less than or equal to $rollno is:
$average"
else
  echo "There are no even numbers less than or equal to $rollno."
fi
```

Output

```
syedabdulbasit@Ubuntu:~/Desktop$ ./lab10.sh
The average of even numbers less than or equal to 89 is: 45
```

5. Write a function that displays the name of the week days starting from Sunday if the user passes a day number. If a number provided is not between 1 and 7 an error message is displayed.

```
#!/bin/bash
if [ "$#" -ne 1 ]; then
  echo "Usage: $0 <weekday>"
  exit 1  # Exit with an error code.
```

```
fi

weekday="$1"
case "$weekday" in
1)

echo "Monday"
;;
2)
echo "Tuesday"
;;
3)
echo "Wednesday"
;;
4)
echo "Thursday"
;;
5)
echo "Friday"
;;
6)
echo "Saturday"
;;
7)
echo "Sunday"
;;
*)
echo "Enter choice  from 1 to 7"
;;
esac
```

Output:

```
syedabdulbasit@Ubuntu:~/Desktop$ ./lab10.sh 5
Friday
syedabdulbasit@Ubuntu:~/Desktop$ ./lab10.sh 3
Wednesday
syedabdulbasit@Ubuntu:~/Desktop$ ./lab10.sh 9
Enter choice  from 1 to 7
```

6. Write scripts that displays the parameters passed along with the parameter number using while and until statements.

```
#!/bin/bash

# Check if at least one command-line argument is provided
if [ "$#" -eq 0 ]; then
  echo "Usage: $0 <parameter1> <parameter2> ..."
  exit 1
fi

# Using a while loop to display parameters and their numbers
count=1
while [ "$#" -gt 0 ]; do
  echo "Parameter $count: $1"
  count=$((count + 1))
  shift
done

# Using a until loop to display parameters and their numbers
count=1
until [ "$#" -eq 0 ]; do
  echo "Parameter $count: $1"
  count=$((count + 1))
  shift
done
```

Output

```
syedabdulbasit@Ubuntu:~/Desktop$ ./lab10.sh hello world C
Parameter 1: hello
Parameter 2: world
Parameter 3: C
```

7. Write a script that displays the following menu:
Quotient
Remainder

Depending on user's choice, the result of division must be displayed and the loop breaks. The two numbers (dividend and divisor) must be supplied at runtime as command line arguments. If user chooses an item

that is not in the list, he must be prompted to make proper choice and the loop must
restart (or continue).

```
#!/bin/bash

# Check if exactly two command-line arguments are provided
if [ "$#" -ne 2 ]; then
  echo "Usage: $0 <dividend> <divisor>"
  exit 1
fi

dividend="$1"
divisor="$2"

while true; do
  # Display the menu
  echo "Menu:"
  echo "1. Quotient"
  echo "2. Remainder"
  echo "3. Quit"

  # Prompt the user for choice
  read -p "Enter your choice (1-3): " choice

  case "$choice" in
    1)
      # Calculate and display the quotient
      quotient=$((dividend / divisor))
      echo "Quotient: $quotient"
      break
      ;;
    2)
      # Calculate and display the remainder
      remainder=$((dividend % divisor))
      echo "Remainder: $remainder"
      break
      ;;
    3)
      # Quit the script
```

```
      echo "Exiting the script."
      exit 0
      ;;
    *)
      # Invalid choice, prompt the user to make a proper choice
      echo "Invalid choice. Please choose a number between 1 and 3."
      ;;
  esac
done
```

Output:

```
syedabdulbasit@Ubuntu:~/Desktop$ ./lab10.sh 8 4
Menu:
1. Quotient
2. Remainder
3. Quit
Enter your choice (1-3): 1
Quotient: 2
syedabdulbasit@Ubuntu:~/Desktop$ ./lab10.sh 8 4
Menu:
1. Quotient
2. Remainder
3. Quit
Enter your choice (1-3): 2
Remainder: 0
syedabdulbasit@Ubuntu:~/Desktop$ ./lab10.sh 8 4
Menu:
1. Quotient
2. Remainder
3. Quit
Enter your choice (1-3): 3
Exiting the script.
syedabdulbasit@Ubuntu:~/Desktop$ ./lab10.sh 8 4
Menu:
1. Quotient
2. Remainder
3. Quit
Enter your choice (1-3): 5
Invalid choice. Please choose a number between 1 and 3
```