1 Write a shell script which will generate the O/P as follows

```
*
**
***
****
[admin@hostname01 Desktop]$ ./patterns.sh
*
**
***
****
```

2 Accept the first name, middle name, and last name of a person in variables fname, mname and lname respectively. Greet the person (take his full name) using appropriate message.

Ans :#!/bin/bash

```
echo "Enter first name"
read fname
echo "Enter mid name"
read mname
echo "Enter last name"
read lname
full_name="$fname $mname $lname"
echo "Hello,$full_name ! Welcome!"

[admin@hostname01 Desktop]$ chmod +x greet.sh
[admin@hostname01 Desktop]$ ./greet.sh
Enter first name
Tanu
Enter mid name
singh
Enter last name
Raghuwanshi
Hello,Tanu singh Raghuwanshi ! Welcome!
```

3 Display the name of files in the current directory along with the names of files with maximum & minimum size. The file size is considered in bytes.

Ans. [admin@hostname01 ~]$ #!/bin/bash

```
# Display all files in the current directory with their sizes
echo "Files in the current directory:"
ls -lh | awk '{print $9, $5}' | tail -n +2
# Check if there are any files
if [[ $(ls -l | wc -l) -le 1 ]]; then
 echo "No files in the current directory."
 exit 1
fi
# Find the file with the maximum size
max_file=$(ls -S | head -1)
max_size=$(ls -lS | awk 'NR==2 {print $5}')
# Find the file with the minimum size
min_file=$(ls -Sr | head -1)
min_size=$(ls -lSr | awk 'NR==2 {print $5}')

# Display results
echo -e "\nFile with the maximum size: $max_file ($max_size bytes)"
echo "File with the minimum size: $min_file ($min_size bytes)"
Files in the current directory:
add.c 0
chap1 0
Desktop 6
Documents 6
Downloads 50
Music 6
newdir 6
Pictures 6
Public 6
Templates 6
Vedios 6
File with the maximum size: Downloads (50 bytes)
File with the minimum size: chap1 (0 bytes)
```

4 Write a script which when executed checks out whether it is a working day or not?

(Note: Working day Mon-Fri)

Ans :#!/bin/bash

```
day=$(date +%u)
if [ $day -ge 1 ] && [ $day -le 5 ]; then
        echo "Weekday"
else
        echo "Weekend"
fi
```

```
[admin@hostname01 Desktop]$ vim days.sh
[admin@hostname01 Desktop]$ ./days.sh
Weekend
```

5 Write a script that accepts a member into HP health club, if the weight of the person is withing the range of 30-250 Kgs.

```
#!/bin/bash

accept_member(){
    if [ $user_weight -ge 30 ] && [ $user_weight -le 250 ]; then
        echo "Welcome to health club"
     else
            echo "You cannot admit to the health club"
    fi
}
 read -p "Enter weight" $user_weight
accept_member $user_weight
```

```
[ admin@hostname01 Desktop]$ ./hp_club.sh
Enter weight300
You cannot admit to the health club
```

6 Write a shell script that greets the user with an appropriate message depending on the system time.

```
#!/bin/bash

HOUR=$(date +"%H")

if [ $HOUR -lt 12 ]; then
   GREETING="GOOD MORNING"
elif [ $HOUR -lt 18 ]; then
   GREETING="GOOD AFTERNOON"
else
   GREETING="GOOD EVENING"
fi

USER_NAME=$(whoami)
echo "$GREETING, $USER_NAME!"

[admin@hostname01 Desktop]$ vim greet_members.sh
[admin@hostname01 Desktop]$ ./greet_members.sh
GOOD MORNING, admin!
```

7 A data file file has some student records including rollno, names and subject marks. The fields are separated by a ":". Write a shell script that accepts roll number from the user, searches it in the file and if the roll number is present - allows the user to modify name and marks in 3 subjects.
If the roll number is not present, display a message "Roll No Not Found". Allow the user to modify one record at a time.

```
#!/bin/bash/
FILE="students.txt"
echo "Enter Roll Number to modify:"
read rollno
record=$(grep "^$rollno:" $FILE)
if [ -n "$record" ]; then
   echo "Record found: $record"
   current_name=$(echo $record | cut -d ':' -f2)
   current_marks1=$(echo $record | cut -d ':' -f3)
```

```
    current_marks2=$(echo $record | cut -d ':' -f4)
    current_marks3=$(echo $record | cut -d ':' -f5)
    echo "Current Name: $current_name"
    echo "Current Marks in Subject 1: $current_marks1"
    echo "Current Marks in Subject 2: $current_marks2"
    echo "Current Marks in Subject 3: $current_marks3"
    echo "Enter new name:"
    read new_name
    echo "Enter new marks for Subject 1:"
    read new_marks1
    echo "Enter new marks for Subject 2:"
    read new_marks2
    echo "Enter new marks for Subject 3:"
    read new_marks3
    sed -i "s/^$rollno:.*/$rollno:$new_name:$new_marks1:$new_marks2:$new_marks3/"
$FILE
    echo "Record updated successfully!"
else
    echo "Roll No Not Found"
fi
[admin@hostname01 ~]$ ./students.sh
Enter Roll Number to modify:
45
```

8 Modify program 7 to accept the RollNo from the command line.

Ans. [admin@hostname01 ~]$ nano modfify_stu.sh

```
# To Accept roll number
read -p "Enter the roll number to search: " rollno
[admin@hostname01 ~]$ ./modfify_stu.sh
Enter the roll number to search: 7
Record found: 7:Isha:90:92:95
```

9 Modify the program 7 to accept the RollNo and display the record and ask for delete confirmation. Once confirmed delete the record and update the data file.

Ans.[admin@hostname01 ~]$ nano del_stu.sh

```
#!/bin/bash
file="studentrecord.txt"
if [ -z "$1" ]; then
read -p "Enter the roll number to search: " rollno
else
 # Use the command line argument for roll number
 rollno=$1
fi
record=$(grep "^$rollno:" "$file")
if [ -n "$record" ]; then
 echo "Record found: $record"
 read -p "want to delete record type y : " confirm
if [ "$confirm" == "y" ] || [ "$confirm" == "Y" ]; then

 sed -i "/^$rollno:/d" "$file"
 echo "Record deleted successfully!"
 else
 echo "Deletion aborted."
 fi
else

 echo "Roll No Not Found"
fi
[admin@hostname01 ~]$ chmod +x del_stu.sh
[admin@hostname01 ~]$ ./del_stu.sh
Enter the roll number to search: 8
Record found: 8:Rani:75:80:85
want to delete record type y : y
Record deleted successfully!
```

10 Write a script that takes a command line argument and reports on its file type (regular file, directory file, etc.). For more than one argument generate error message.

```
#!/bin/bash/
```

```
if [ $# -ne 1 ]; then
   echo "Error: Please provide exactly one file or directory as an argument."
   exit 1
fi
file_path="$1"

if [ ! -e "$file_path" ]; then
   echo "Error: The path '$file_path' does not exist."
   exit 1
fi
if [ -f "$file_path" ]; then
   echo "The path '$file_path' is a regular file."
elif [ -d "$file_path" ]; then
   echo "The path '$file_path' is a directory."
elif [ -L "$file_path" ]; then
   echo "The path '$file_path' is a symbolic link."
Else

   echo "The path '$file_path' is of an unknown type."
fi

[admin@hostname01 ~]$ vim file_type.sh
[admin@hostname01 ~]$ chmod +x file_type.sh
[admin@hostname01 ~]$ ./file_type.sh
Error: Please provide exactly one file or directory as an argument.
```

11 Add some student records in the “student” file manually. The fields to be considered are “RollNo”, “Name”, “Marks_Hindi”, “Marks_Maths”, “Marks_Physics”.
Write a script which does the following

a If the roll number already exists, then store the record and the following message “roll number exists” in a log file “log1”.

b If the marks in the subjects is not in the range of 1 – 99 then store such a record followed by a message “marks out of range” in “log1”

c If the data is valid, the calculate total, percentage, grade and display on the terminal

```
#!/bin/bash

process_record(){

rollno=$1
name=$2
marks_hindi=$3
marks_maths=$4
marks_physics=$5

if grep -q "^$rollno:" student; then
	echo "$rollno:$name:$marks_hindi:$marks_maths:$marks_physics - rRoll number exists" >
log1
	echo"Roll number exists. Entry logged"
	return
fi

if [ "$marks_hindi" -lt 1 ] || [ "$marks_hindi" -gt 99 ] || \
	[ "$marks_maths" -lt 1 ] || ["$marks_maths" -gt 99] || \
	[ "$marks_physics" -lt 1 ] || [ "$marks_physics" -gt 99  ]; then
	echo"$rollno:$name:$marks_hindi:$marks_maths:$marks_physics - Marks out of range" >
log1
	echo"Marks out of range. Entry logged"
	return
fi

total=$((marks_hindi + marks_maths + marks_physics))
percentage=$(echo "scale=2; $total / 3" | bc)

if[ "$(echo "$percentage >= 90" | bc)" -eq 1 ]; then
	grade="A"
elif [ "$(echo "$percentage >= 75" | bc)" -eq 1 ]; then
	grade=B"
elif [ "$(echo "$percentage >= 50" | bc)" -eq 1 ]; then
	grade=C"
else
	grade="D"
fi

echo "$rollno:$name:$marks_hindi:$marks_maths:$marks_physics" > student

echo "Record added successfully:"
echo "RollNo: $rollno"
echo "Name: $name"
echo "Total marks: $total"
echo "Percentage: $percentage%"
```

```
echo "Grade: $grade"
}

echo "Enter Rol Number: "
read rollno
echo "Enter Name: "
read name
echo"Enter Marks in Hindi: "
read marks_hindi
echo "Enter marks in Maths: "
read marks_maths
echo "Enter Marks in Physics: "
read marks_physics


process_record $rollno" "$name" "$marks_hindi" "$marks_maths" "$marks_physics"
```