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

```
*
**
***
****
```

**Command :**

```
for i in {1..4}; do
  for j in $(seq 1 $i); do
    echo -n "*"
  done
  echo
done
```

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.

```
read -p "Enter first name: " fname
read -p "Enter middle name: " mname
read -p "Enter last name: " lname
echo "Hello, $fname $mname $lname!"
```

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.

```
ls -l | awk '{print $9, $5}' > file_sizes.txt
max_size_file=$(sort -k2 -n file_sizes.txt | tail -n 1)
min_size_file=$(sort -k2 -n file_sizes.txt | head -n 1)
echo "Files in current directory:"
cat file_sizes.txt
echo "Max size file: $max_size_file"
echo "Min size file: $min_size_file"
```

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

(Note: Working day Mon-Fri)

```
day=$(date +%u)
if [ "$day" -ge 1 ] && [ "$day" -le 5 ]; then
  echo "Today is a working day."
else
  echo "Today is not a working day."
fi
```

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.

```
read -p "Enter weight (in Kgs): " weight
if [ "$weight" -ge 30 ] && [ "$weight" -le 250 ]; then
  echo "Welcome to HP Health Club!"
else
  echo "Weight not in acceptable range."
fi
```

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

```
hour=$(date +%H)
if [ "$hour" -ge 5 ] && [ "$hour" -lt 12 ]; then
  echo "Good Morning!"
elif [ "$hour" -ge 12 ] && [ "$hour" -lt 18 ]; then
  echo "Good Afternoon!"
else
  echo "Good Evening!"
fi
```

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.

```
read -p "Enter roll number: " rollno
if grep -q "$rollno" file; then
  sed -i "/$rollno/ s/\(.*\)/Enter new name and marks (3 subjects):/;w tempfile" file
else
  echo "Roll No Not Found"
fi
```

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

```
rollno=$1
if grep -q "$rollno" file; then
  sed -i "/$rollno/ s/\(.*\)/Enter new name and marks (3 subjects):/;w tempfile" file
else
  echo "Roll No Not Found"
fi
```

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.

```
rollno=$1
record=$(grep "$rollno" file)
echo "Record found: $record"
read -p "Do you want to delete this record? (y/n): " choice
if [ "$choice" == "y" ]; then
  sed -i "/$rollno/d" file
  echo "Record deleted."
else
  echo "Record not deleted."
fi
```

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.

```
if [ $# -gt 1 ]; then
  echo "Error: More than one argument provided."
else
  filetype=$(file $1)
  echo "File type: $filetype"
fi
```

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

```
read -p "Enter RollNo: " rollno
if grep -q "$rollno" student; then
  echo "$rollno exists" >> log1
else
  read -p "Enter marks for 3 subjects: " marks1 marks2 marks3
  if [[ "$marks1" -lt 1 || "$marks1" -gt 99 || "$marks2" -lt 1 || "$marks2" -gt 99 || "$marks3" -lt 1 || "$marks3" -gt 99 ]]; then
    echo "marks out of range" >> log1
  else
    total=$((marks1 + marks2 + marks3))
    percentage=$((total / 3))
    grade="A"  # Sample grade
    echo "$rollno,$marks1,$marks2,$marks3,$total,$percentage,$grade"
  fi
fi
```