

**Name= shaik Raakin**  
**Class =3rd sem**  
**Branch = cybersecurity**  
**usn=ENG24CY0192**  
**Roll no:64**  
**section:A**  
**Assignment : 7**

**1Q) What is a bash shell script? Give one example**  
**sol)**

A bash shell script is a plain text file containing a sequence of commands written for the Bash shell, a popular command-line interpreter for Linux and Unix systems. These scripts are used to automate repetitive tasks, configure systems, process files, and simplify complex workflows by executing multiple commands in order.

Example

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

```
echo hi
```

#!/bin/bash tells the system to use bash to interpret the script.

echo command prints a message.

**2Q)Write a simple shell script to print "Hello World".**

**Sol:**

The simple shell script is :

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

```
echo "hello world"
```

**3Q) What is the purpose of comments (#) in a shell script?**  
**sol)**

Comments can be included in shell scripts simply by placing a pound sign (#) before the comment; comments will be made until the end of the line and are ignored by the shell. Comments are useful in identifying the meaning of code statements, instructions, author information, and making the script understandable to users and future maintainers of the script. Adequate comments also assist during debugging and to support programming as part of a team.

**4Q)How do you declare variables (int, float, double, string, Boolean, and char in a shell script?**  
**sol)**

In bash, all variables are treated as strings unless used in mathematical contexts. Examples of declarations include,

Integer (e.g. num=10)

Float/Double (e.g. Bash does not support floats natively; we do ncould write num=10.5 to store 10.5 in a variable and use a calculator tool in bash like bc or an inline expression in awk to apply math to the variable).

String (e.g. name="Alice")

Boolean -(e.g. there is no actual boolean within bash, we simply have a flag=something statement. flag=true).

Character - (e.g. Just a one character string; char='A')

It should be noted at this point there is no typing. The only guidance that is offered is to enclose strings/one character with quotes, and the number could interpreted if needed.

**5Q)Write a shell script to display the current date and time of the system.**

**sol)**

The shell script is:

```
#!/bin/bash
echo "The current date and time is: $(date)"
```

Output:

```
raakin@DESKTOP-0FFMSPJ:~/uni_linx$ ./date_time.sh
The current date and time is: Sun Oct 12 18:21:53 UTC 2025
```

**6Q)Explain the difference between a constant and a variable in bash script**

**sol)**

A variable, in programming terms, is a character or word that holds a value that may change as the script runs. A constant is a value that cannot be changed once it is assigned.

In bash, you can simulate constants using the readonly command:

```
readonly PI=3.14
```

**7Q)Write a shell script to read two integer number from the user and compute the sum of both the number.**

**sol)**

Shell script:

```
#!/bin/bash
echo "Enter the first number: "
read num1
echo "Enter the second number: "
read num2
sum=$((num1 + num2))
echo "Sum is $sum"
```

**8Q)What is the use of source command in shell scripting?**

**sol)**

The source command, which is symbolized by the . (dot), is a shell builtin command used to execute commands from a file in the current shell instead of starting a new shell. The source command is typically used to load environment variables, aliases, and code snippets that can be run/reused in one session.

**9Q)How can you debug a shell script? Give two methods.**

**sol)**

To initiate the debugging process, run the script with -x (xtrace) option by typing `bash -x script.sh`. This will print each command to the terminal window as it is executed, along with any variable expansion.

You can also use ``set -x`` and then use ``set +x`` within your script to turn on or off debug mode, around specific blocks of code. You can also use echo statements to print out variable values and track progress.

**10Q)Write a bash script to create and delete a file.**

**sol)**

The bash script to create and delete a file is:

```
#!/bin/bash
touch newfile.txt
echo "File created: newfile.txt"
rm newfile.txt
echo "File deleted: newfile.txt"
```