Task 1:

To create a bash script with comments explaining what the script does.

For suppose just write a script to create a directory and file.

#!/bin/bash

This script creates a directory named "pooja" and a file named "sample.txt" inside it.

Create a directory named "pooja"

mkdir pooja

Create a file named "sample.txt" inside the "pooja" directory

touch Sample.txt

The **#!/bin/bash** line specifies that the script should be interpreted by the Bash shell. Lines beginning with **#** are comments in the script, providing explanations or notes for human readers and are ignored during execution.

```
root@ip-172-31-82-211:/# cd /home
root@ip-172-31-82-211:/home# touch file2.txt
root@ip-172-31-82-211:/home# ls
Vardhan  file1.txt  file2.txt  krishna  ssm-user  ubuntu  username
root@ip-172-31-82-211:/home# vi file2.txt
root@ip-172-31-82-211:/home# chmod 700 file2.txt
root@ip-172-31-82-211:/home# ls
Vardhan  file1.txt  file2.txt  krishna  ssm-user  ubuntu  username
root@ip-172-31-82-211:/home# ./file2.txt
root@ip-172-31-82-211:/home# ls
Sample.txt  Vardhan  file1.txt  file2.txt  krishna  pooja  ssm-user  ubuntu  username
```

Task 2:

To create a bash script that uses echo to print a message of your choice.
Let's add the echo to print for the above script with little change, now you need to create a file inside that directory which you created.

#!/bin/bash

This script creates a directory named "pooja" and a file named "sample.txt" inside it.

echo "Proceeding to create a directory and file"

Create a directory named "pooja"

mkdir pooja

Create a file named "sample.txt" inside the "pooja" directory

touch pooja/sample.txt

echo "The directory pooja and file sample.txt are created successfully"

```
root@ip-172-31-82-211:/home# ls
Sample.txt  Vardhan  file1.txt  file2.txt  krishna  ssm-user  ubuntu  username
root@ip-172-31-82-211:/home# cat file2.txt
#!/bin/bash
# This script creates a directory named "pooja" and a file named "sample.txt" inside it.
echo "Proceeding to create a directory and file"
# Create a directory named "pooja"
mkdir pooja
# Create a file named "sample.txt" inside the "pooja" directory
touch pooja/sample.txt
echo "The directory pooja and file sample.txt are created successfully"

root@ip-172-31-82-211:/home# ./file2.txt
Proceeding to create a directory and file
The directory pooja and file sample.txt are created successfully
root@ip-172-31-82-211:/home# ls
Sample.txt  Vardhan  file1.txt  file2.txt  krishna  pooja  ssm-user  ubuntu  username
root@ip-172-31-82-211:/home# cd pooja
root@ip-172-31-82-211:/home/pooja# ls
sample.txt
root@ip-172-31-82-211:/home/pooja#
```

Task 3:

Create a bash script that takes two variables (numbers) as input and prints their sum using those variables.

#!/bin/bash

echo “Starting the sum of two numbers”

num1=$1

num2=$2

sum=$((num1 + num2))

echo "The sum of two numbers is:" $sum

```
root@ip-172-31-82-211:/home# ls
Sample.txt  Vardhan  file1.txt  file2.txt  krishna  pooja  ssm-user  ubuntu  username
root@ip-172-31-82-211:/home# cat file2.txt
#!/bin/bash
echo “Starting the sum of two numbers”
num1=$1
num2=$2
sum=$((num1 + num2))
echo "The sum of two numbers is:" $sum

root@ip-172-31-82-211:/home# ./file2.txt 10 20
“Starting the sum of two numbers”
The sum of two numbers is: 30
root@ip-172-31-82-211:/home#
```

Task 5:

To create a bash script that utilizes at least three different built-in variables to display relevant information.

Built-in variables in Bash are predefined variables that hold information about the environment or the shell itself. These variables are provided by the shell and are available for your use within Bash scripts. Some commonly used built-in variables include:

$USER: This variable contains the username of the current user.

$HOME: Represents the home directory of the current user.

$PATH: Specifies a colon-separated list of directories in which the shell looks for executable files.

$PWD: Holds the present working directory.

$HOSTNAME: Stores the hostname of the system.

$SHELL: Indicates the path to the current shell.

$RANDOM: Generates a random number between 0 and 32767.

$UID: Contains the user ID of the current user.

$OLDPWD: Represents the previous working directory.

$IFS: Internal Field Separator, used for word splitting after expansion and to split lines into fields.

#!/bin/bash

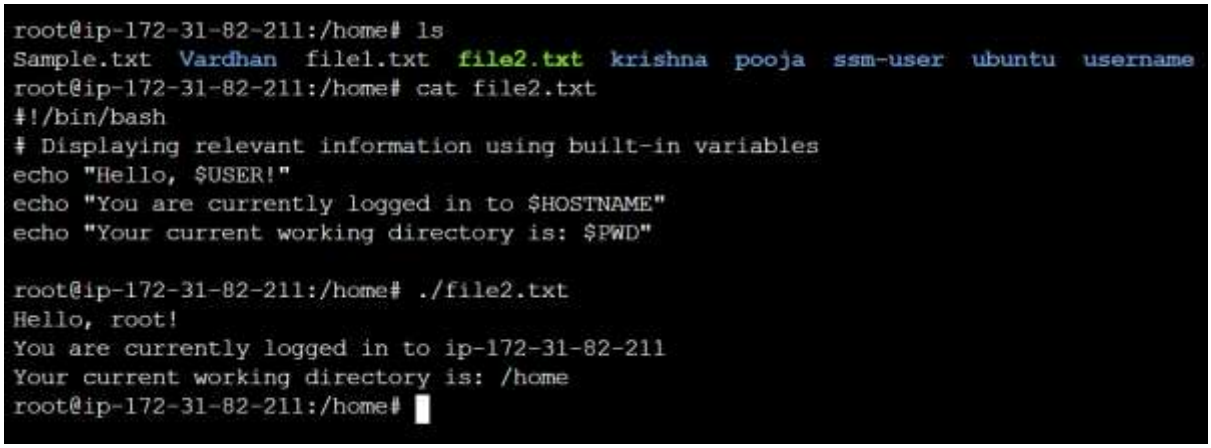
Displaying relevant information using built-in variables

echo "Hello, $USER!"

echo "You are currently logged in to $HOSTNAME"

echo "Your current working directory is: $PWD"

When you run this script, it will greet you, display the hostname of the machine you're logged into, and show your current working directory.



Task 6:

To create a bash script that utilizes wildcards to list all the files with a specific extension in a directory.

Wildcards are special characters used in shell commands to represent patterns in filenames or strings.

They allow you to select multiple files or strings based on a pattern rather than specifying each file or string individually. Some common wildcards in Bash are:

* (asterisk): Matches zero or more characters.

? (question mark): Matches any single character.

[ ] (square brackets): Matches any one character within the specified range or set.

! (exclamation mark): Negates the matching pattern.

#!/bin/bash

echo "Files with .txt extension:"

ls *.txt

```
root@ip-172-31-82-211:/home# cat file2.txt
#!/bin/bash

echo "Files with .txt extension:"
ls *.txt

root@ip-172-31-82-211:/home# ./file2.txt
Files with .txt extension:
Sample.txt  file1.txt  file2.txt
root@ip-172-31-82-211:/home#
```

Task 7:

Create a single bash script that completes all the Tasks mentioned above.

#!/bin/bash

Task 1: Create a directory and file

echo "Task 1: Creating a directory named 'pooja' and a file named 'sample.txt' inside it."

Print a message before proceeding

echo "Proceeding to create a directory and file."

Create a directory named "pooja"

mkdir pooja

Create a file named "sample.txt" inside the "pooja" directory

```
touch pooja/sample.txt
echo "The directory 'pooja' and file 'sample.txt' are created successfully."
echo "Task 1 completed successfully."
echo "--------------------------------------------------"
# Task 3: Create a bash script that takes two variables (numbers) as input and prints their sum
echo "Task 3: Calculating the sum of two numbers."
echo "Starting the sum of two numbers."
num1=$1
num2=$2
sum=$((num1 + num2))
echo "The sum of $num1 and $num2 is: $sum"
echo "Task 3 completed successfully."
echo "--------------------------------------------------"
# Task 4: Create a bash script that takes two variables (numbers) as input and prints their sum
echo "Task 4: Calculating the sum of two numbers as input parameters."
echo "Starting the sum of two numbers from input parameters."
echo "Enter the first number:"
read num1
echo "Enter the second number:"
read num2
sum=$((num1 + num2))
echo "The sum of $num1 and $num2 is: $sum"
echo "Task 4 completed successfully."
echo "--------------------------------------------------"
# Task 5: Create a bash script that utilizes at least three different built-in variables to display relevant information
echo "Task 5: Displaying relevant information using built-in variables."
echo "Hello, $USER!"
echo "You are currently logged in to $HOSTNAME."
echo "Your current working directory is: $PWD."
echo "Task 5 completed successfully."
```

echo "--"

Task 6: Create a bash script that utilizes wildcards to list all the files with a specific extension in a directory

echo "Task 6: Listing all files with '.txt' extension using wildcards."

echo "Files with .txt extension:"

ls *.txt

echo "Task 6 completed successfully."

echo "--"

echo "All tasks completed."

```
root@ip-172-31-82-211:/home# ./file2.txt
Task 1: Creating a directory named 'pooja' and a file named 'sample.txt' inside it.
Proceeding to create a directory and file.
mkdir: cannot create directory ‘pooja’: File exists
The directory 'pooja' and file 'sample.txt' are created successfully.
Task 1 completed successfully.
--------------------------------------------------
Task 3: Calculating the sum of two numbers.
Starting the sum of two numbers.
The sum of  and  is: 0
Task 3 completed successfully.
--------------------------------------------------
Task 4: Calculating the sum of two numbers as input parameters.
Starting the sum of two numbers from input parameters.
Enter the first number:
12
Enter the second number:
13
The sum of 12 and 13 is: 25
Task 4 completed successfully.
--------------------------------------------------
Task 5: Displaying relevant information using built-in variables.
Hello, root!
You are currently logged in to ip-172-31-82-211.
Your current working directory is: /home.
Task 5 completed successfully.
--------------------------------------------------
Task 6: Listing all files with '.txt' extension using wildcards.
Files with .txt extension:
Sample.txt  file1.txt  file2.txt
Task 6 completed successfully.
--------------------------------------------------
All tasks completed.
root@ip-172-31-82-211:/home#
```

.