



Assignment 2 – Script Execution & Explanation



Objective

Understand how existing scripts in the repository work by running and explaining them.



Tasks Overview

- Select two scripts from `Scripts/` (e.g., `print_numbers.sh`, `eight_script.sh`)
- Run each script and observe the output
- Provide script name, purpose, line-by-line explanation, and example run with input/output



Script 1: `print_numbers.sh`



Purpose

Prints numbers from 1 to 5 in the terminal.



Script Content

```
#!/bin/bash
for i in {1..5}
do
    echo "Number:$i"
done
```



Line-by-Line Explanation

1. `#!/bin/bash`

Declares the script should run using the Bash shell.

2. `for i in {1..5}`

Loops through numbers 1 to 5, assigning each to variable `i`.

3. `do`

Begins the loop block.

4. `echo "Number:$i"`

Prints the current value of `i` with the label "Number:".

5. **done**

Ends the loop.

▶ Example Run

```
$ bash print_numbers.sh
Number:1
Number:2
Number:3
Number:4
Number:5
```

```
ujjwal@tyagi@ujjwal-tyagi:~/scripts$ nano fifth_script.sh
ujjwal-tyagi@ujjwal-tyagi:~/scripts$ chmod 777 fifth_script.sh
ujjwal-tyagi@ujjwal-tyagi:~/scripts$ ./fifth_script.sh
Number:1
Number:2
Number:3
Number:4
Number:5
```

📁 Script 2: **eight_script.sh**

📄 Purpose

Prints each element of an array.

🔗 Script Content

```
#!/bin/bash
fruits=("apple" "banana" "cherry")

echo "First fruit: ${fruits[0]}"

for fruit in "${fruits[@]"; do
    echo "Fruit: $fruit"
done
```

📄 Line-by-Line Explanation

1. `#!/bin/bash`

Declares the script should run using the Bash shell.

2. `fruits=("apple" "banana" "cherry")`

Defines an array named `fruits` with three elements.

3. `echo "First fruit: ${fruits[0]}"`

Prints the first element of the array.

4. `for fruit in "${fruits[@]}"; do`

Loops through each element in the array, assigning it to `fruit`.

5. `echo "Fruit: $fruit"`

Prints the current array element with the label "Fruit:".

6. `done`

Ends the loop.

▶ Example Run

```
$ bash eight_script.sh
First fruit: apple
Fruit: apple
Fruit: banana
Fruit: cherry
```

```
ujjwal-tyagi@ujjwal-tyagi:~/scripts$ touch eight_script.sh
ujjwal-tyagi@ujjwal-tyagi:~/scripts$ nano eight_script.sh
ujjwal-tyagi@ujjwal-tyagi:~/scripts$ chmod 777 eight_script.sh
ujjwal-tyagi@ujjwal-tyagi:~/scripts$ ./eight_script.sh
First fruit: apple
Fruit: apple
Fruit: banana
Fruit: cherry
```

? Extra Questions

1] What is the purpose of `#!/bin/bash` at the top of a script?

It tells the system to use the Bash shell to interpret the script, ensuring compatibility and correct execution.

2] How do you make a script executable?

Use the command:

```
chmod +x scriptname.sh
```

This adds execute permissions, allowing you to run the script with `./scriptname.sh`.

✦ End of Assignment 2 – Script Execution & Explanation