LAB2 : Script Execution & Explanation

Objective: Understand how existing scripts in repo work

Script that I chose : `array_length.sh`

Line-by-Line EXPLANATION:

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

- **Shebang line**: Specifies that the script should be run using the Bash shell.

```
number=(0 9 8 7 6)
```

- **Array declaration**: Creates an indexed array named `number` with five integer elements.

```
length="${#number[@]}"
```

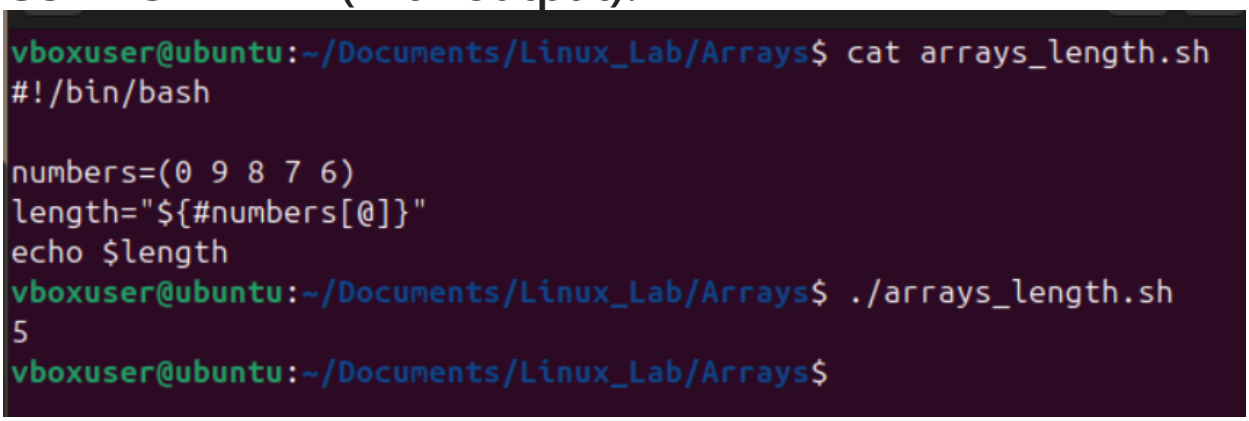
- **Length calculation and output**:
  - `${number[@]}` accesses all elements of the array.
  - `${#number[@]}` returns the total number of elements.

```
echo "$length"
```

- **Printing the variable** `length`

- `echo` prints the result: `5`.

CODE SNIPPET (with output):



Script that I chose : `print_with_range.sh`

Line-by-Line EXPLANATION:

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

- **Shebang line**: Tells the system to execute the script using the Bash shell.

```
for i in {0..7}
```

- **Loop initialization**: Starts a `for` loop that iterates over the numbers 1 through 5 using brace expansion `{1..5}`.

```
do
```

- **Loop body begins**: Marks the start of the commands to be executed in each iteration.

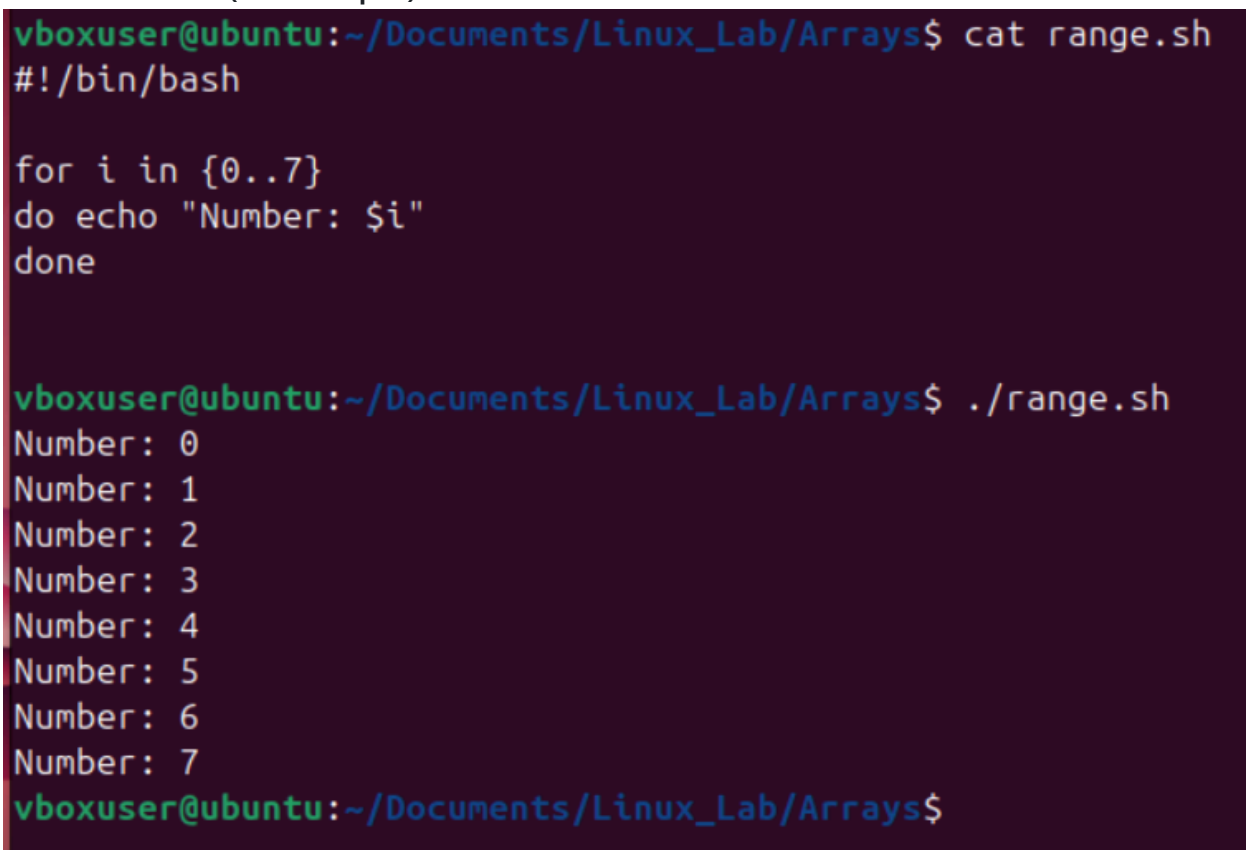
```
  echo "Number: $i"
```

- **Output statement**: Prints the current value of `i` with the label `"Count:"`.

```
done
```

- **Loop body ends**: Signals the end of the `for` loop.

CODE SNIPPET (with output):



Extra Questions:

1.What is the purpose of #!/bin/bash at the top of a script?
Answer:
-The line `#!/bin/bash` is called a **shebang**.
-It tells the system to execute the script using the Bash shell, ensuring consistent behavior across environments.

2. How do you make a script executable?
   Answer: