**First Shell scripting** is a method of using **command-line interpreters**, known as shells, to control operating systems through command-line interfaces. It allows users to communicate directly with their systems.

**Shell scripting** provides a simple way to automate tasks using Linux or Unix commands. It offers conditional and loop control structures for executing commands repeatedly. Different shells have different names, such as**( Sh, Bash, csh, or tesh)**, depending on the operating system.

Shells can be categorized as **command-line** or **graphica**l. Command-line shells use human-readable commands for input and display output through command-line interfaces. **Graphical shells** employ a graphical user interface **(GUI)** for interactions, including file operations like opening, closing, and saving.

**after we know what a shell scripting is, let me talk about its elements**

**A shell script comprises following elements**

⁃ Shell Keywords — **if, else, break etc.**

⁃ Shell commands — **cd, ls, echo, pwd, touch etc.**

⁃ Functions

⁃ Control flow — **if..then..else, case and shell loops etc.**

**• What is #!/bin/bash? Can we write #!/bin/sh as well..?**

> The line **"#!/bin/bash"** is known as the **shebang** in Unix. It is placed at the beginning of a script and specifies the interpreter to be used to run the script, in this case, Bash. The shebang informs the operating system which shell should be used for interpreting the commands in the script.

>Yes, it is possible to use **"#!/bin/sh"** instead of **"#!/bin/bash"**. The shebang can be set to "#!/bin/sh" to indicate that the script should be executed using the default shell, which may vary depending on the operating system. In many Unix-like systems, **"#!/bin/sh"** is a symbolic link or a compatibility layer that points to the default shell, which is often Bash or a similar shell.

Example: *// Using shebang to specify the OS to*

*// use bash shell*

*//* [*myScript.sh*](http://myScript.sh)

**Cmd:**

#!/bin/bash

# Bash script

echo “Hello World!”

**• Write a Shell Script which prints “I will complete#90DaysOofDevOpschallenge”..?**

after writing a shell script you must make it excusable to run it by using a command

chmod +X script.sh

**A black background with purple text

Description automatically generatedA screen shot of a computer

Description automatically generated**

**Write a Shell Script to take user input, input from arguments and print the variables..?**

**A screenshot of a computer

Description automatically generatedA screen shot of a computer

Description automatically generated**

**Write an Example of If else in Shell Scripting by comparing 2 number**

**A screen shot of a computer code

Description automatically generatedA screen shot of a computer

Description automatically generated**

**In the end** we should know that **shell scripting** is a powerful tool in the **DevOps** toolkit, enabling automation, infrastructure management, deployment, and monitoring. It helps streamline processes, improve efficiency, and maintain consistency in software development and operations workflows.

thanks for your time <3

i hope this article will be useful for you <3