* **What is shell?**
* Shell is the command line interface (CLI) that interprets the Linux commands and executes them.
* Keeping all the commands you need in a file with variables, conditions, loops, functions is called shell.
* Shell interprets and executes and executes shell script.

**Shell script Basics**

**1.shebang (#! /bin/bash)**

The first line in many shell scripts is the shebang. It tells the system which interpreter to use to execute the script.

#! /bin/bash

* **What is shebang?**

The shebang is the combination of the # (pound key) and ! (exclamation mark) this character combination has a special meaning when it is used in the very first line of the script. It is used to specify the interpreter with which the given script will run by default

So, if the first line of a script is

#! /bin/bash

**2.comments**

Comments are lines that the shell ignores. They start with #

# This is a comment.

**3. variables:**

Variables are values and are referenced using the $ symbol.

#! /bin/bash

Name= “john”

echo “Hello, $name”

**4. Reading user input:**

You can prompt the user for input and store it in a variable.

#! /bin/bash

echo “enter your name:”

read name

echo “Hello, $name”

**5.Basic commands:**

Shell scripts can executes any command that you can run in the terminal

#! /bin/bash

Echo “listing all files in the directory:”

Ls

**6.conditional statements:**

You can use if, else and elif to execute code based on conditions.

!#/ bin/bash

echo “enter a number:”

read number

If [ $number -gt 10 ]: then

echo “the number is greater then 10”

else

echo “the number is 10 or less”

fi

**7. loops:**

Loops are used to repeat commands.

**For loop**

#! /bin/bash

For I in 1 2 3 4 5

do

Echo “number: $i”

done

**While loop:**

#! /bin/bash

Count=1

While [ $count -le 5 ]

do

echo “count : $count”

Count=$((count + 1))

done

**8 functions:**

functions are you to group commands into reusable blocks.

#! /bin/bash

greet() {

echo “Hello, $1”

}

greet “Alice”

greet “bob”

**9 file operations:**

You can create, read, write, and delete files within a script.

**Create a file:**

#! /bin/bash

echo “ this is a sample file. “> sample.txt

**Append to a file:**

#! /bin/bash

echo “ this is a appended text.”>>sample.txt

**Read from a file.**

#! /bin/bash

While read line

do

echo $line

done < sample.txt

**delete a file:**

#! /bin/bash

rm sample.txt

**10 script execution:**

Make your script executable with the **chmod** command.

chmod +x your\_script.sh

**run the script:**

./your-script.sh

**Example script:**

Here’s an example script that combines some of these basics.

#!/bin/bash

# Greet the user

echo "Enter your name:"

read name

echo "Hello, $name!"

# List files in the current directory

echo "Listing files in the current directory:"

ls

# Loop through numbers and print them

for i in 1 2 3 4 5

do

echo "Number: $i"

done

# Create and read a file

echo "This is a sample file." > sample.txt

echo "Contents of sample.txt:"

cat sample.txt

# Delete the file

rm sample.txt

echo "sample.txt has been deleted."

**Some sample scripts.**

**1.hello world script**

#! /bin/bash

echo “hello, world”

**2.user input and greeting script**

#! /bin/bash

echo “ enter your name:”

read name

echo “Hello, $name!”

**3.file listing script**

#! /bin/bash

echo “listing files in the current directory:”

Ls

**4. sample calculation script:**

#! /bin/bash

Echo “enter two numbers”

Read num1

Read num2

Sum=$((num1 + num2))

Echo “sum of $num1 and $num2 is: $sum”

**5.file backup script:**

#! /bin/bash

echo “enter the file to backup:”

Read filename

backup\_dir=~/backup

cp $filename $backup\_dir

echo “backup of $filename completed in $backup\_dir”

**6 check the file exists**

#! /bin/bash

echo “enter a file name:”

read file name

If [ -f “$filename” ]; then

echo “$filename exists.’

else

Echo “$filename does not exist.”

fi

**7 Loop through numbers and print:**

#! /bin/bash

Echo “looping through numbers 1 to 5:”

For I in {1..5}

do

echo “$i”

done

**8.display system information:**

#!/bin/bash

echo "System Information:"

echo "Hostname: $(hostname)"

echo "CPU: $(grep 'model name' /proc/cpuinfo | head -n 1 | cut -d ':'

-f 2 | sed 's/^[ \t]\*//')"

echo "Memory: $(free -h | awk '/^Mem:/ {print $2}')"

echo "Disk Usage: $(df -h / | awk '/\// {print $5}')"

**9. Simple Menu Selection**

#!/bin/bash

echo "Menu Selection:"

echo "1. Display Date and Time"

echo "2. Display Calendar"

echo "3. Display Disk Usage"

read choice

case $choice in

1) echo "Current date and time: $(date)";;

2) cal;;

3) df -h;;

\*) echo "Invalid choice";;

esac

**10. Check Internet Connectivity:**

#! /bin/bash

echo "Checking internet connectivity..."

ping -c 1 google.com > /dev/null

if [ $? -eq 0 ]; then

echo "Internet is up!"

else

echo "Internet is down."

fi