Name : Ayushi Bindal **Roll no : 1/23/SET/BCS/341** **Class & Section : 4CSF2**

Experiment – 5

Aim: If-Else statement in bash

Bash scripting allows conditional execution using if-else statements. These statements enable decision-making within a script, executing different commands based on whether a condition evaluates to true or false.

1. Simple if Statement: This script checks if 1 is equal to 1 and prints a message.

```
cat > testif1.sh
#!/bin/bash
if [ 1 -eq 1 ];
then
   echo "1 is equal to 1"
fi
```

Save as : testif1.sh
Execute using : bash testif1.sh

```
localhost:~/Ayushi# cat > testif1.sh
#!/bin/bash
if [ 1 -eq 1 ];
then
    echo "1 is equal to 1"
fi
localhost:~/Ayushi# bash testif1.sh
1 is equal to 1
```

2. If-else statement:
This script compares two numbers.

```
cat > testif2.sh
#!/bin/bash
if [ 1 -eq 1 ];
then
     echo "1 is equal to 1"
else
     echo "The numbers are not equal"
fi
```

Save as: testif2.sh
Execute using: bash testif2.sh

```
localhost:~/Ayushi# cat > testif2.sh
#!/bin/bash
if [ 1 -eq 1 ];
then
    echo "1 is equal to 1"
else
    echo "The numbers are not equal"
fi
localhost:~/Ayushi# bash testif2.sh
1 is equal to 1
localhost:~/Ayushi#
```

3. Check if a file exists: This script checks if a file named data.txt exists.

```
cat > testif3.sh
#!/bin/bash
if [ -f data.txt ]
then
     echo “File exists”
else
     touch data.txt
     echo “New file created”
fi
```

Save as: testif3.sh
Execute using: bash testif3.sh

```
localhost:~/Ayushi# cat > testif3.sh
#!/bin/bash
if [ -f data.txt ]
then
    echo "File exists"
else
    touch data.txt
    echo "New file created"
fi
localhost:~/Ayushi# bash testif3.sh
New file created
localhost:~/Ayushi#
```

4. String Comparison:
This script compares two string variables.

```
cat > testif4.sh
#!/bin/bash
user=”admin”
if [ “$user” = “administrator” ];
then
      echo “The string matches”
else
      echo “The strings do not matches”
fi
```

Save as: testif3.sh
Execute using: bash testif3.sh

```
localhost:~/Ayushi# cat > testif4.sh
#!/bin/bash
user="admin"
if [ "$user" = "administrator" ];
then
    echo "The string matches"
else
    echo "The strings do not match"
fi
localhost:~/Ayushi# bash testif4.sh
The strings do not match
localhost:~/Ayushi#
```

5. User Authentication Check: This script prompts the user for a name and verifies if they are an administrator.

```
cat > testif5.sh
#!/bin/bash
auth_users="admin"
read -p "What's your name? " user
if [[ "$user" == "$auth_users" ]];
then
    echo "You are an administrator"
elif [[ -z "$user" ]];
then
    echo "Please enter a username"
    read user
    echo "Hello $user, Greetings!"
else
    echo "You are a standard user"
fi
```

Save as: testif3.sh
Execute using: bash testif3.sh

```
localhost:~/Ayushi# cat > testif5.sh
#!/bin/bash
auth_users="admin"
read -p "What's your name? " user

if [[ "$user" == "$auth_users" ]];
then
    echo "You are an administrator"
elif [[ -z "$user" ]];
then
    echo "Please enter a username"
    read user
    echo "Hello $user, Greetings!"
else
    echo "You are a standard user"
fi
localhost:~/Ayushi# bash testif5.sh
What's your name? Ayushi
You are a standard user
```