CONDITIONAL STATEMENTS

>> These are used to execute different code blocks/statements based on some conditions, i.e., True or False.

Types:

1. if statement
2. if-else statement
3. elif

1. if statement:

if statement executes the block of code, IF the condition is TRUE.

Syntax:

```
if [  condition  ];
then
    #statement/code
fi
```

(( )) → Operartion

[ ] → Condition

SCRIPT: WRITE A SCRIPT TO CHECK IF THE NUMBER IS GREATER THAN 10 0R NOT

```
#!/bin/bash
read -p "Enter number: "  num
if [  $num -gt 10  ];
then
   echo "$num is greater than 10"
fi
```

2. if-else statement:

If the conditions is 'True', 'if' statement block will execute BUT if the condition is 'False' then 'else' block will get executed.

Syntax:

```
if [  condition  ];
then
    #statement/code
else
    #statement
fi
```

SCRIPT: WRITE A SCRIPT TO CHECK IF THE NUMBER IS GREATER THAN 10 0R NOT, IF NOT,
PRINT "Number is less than 10, TRY AGAIN"

3. elif statement:

If we have multiple conditions needed to be checked, then we make use of elif statements.

```
cond1:
          #statement 1
cond2:
          #statement 2
cond3:
          #statement 3
```

Syntax:

```
if [  condition  ];
then
    #statements
elif [  condition  ];
then
    #statements
elif [  condition  ];
then
    #statements
elif [  condition  ];
then
    #statements
elif [  condition  ];
then
    #statements
else
    #statements
fi
```

SCRIPT: WRITE A SCRIPT TO CHECK WHETHER THE NUMBER IS GREATER THAN 10

-IF YES, PRINT "NUMBER IS GREATE THAN 10"

-IF NOT, PRINT "NUMBER IS LES THAN"

-IF IT IS EQUAL TO 10, PRINT "NUMBERS ARE EQUAL"

-IF THE ENTERED NUMBER IS LESS THAN 0, THEN PRINT "NUM IS LESS THAN1/ NUM IS ZERO"

NUM >15
NUM = 15
NUM < 1
else: num is less than 15