

PROGRAMMING CONSTRUCTS

The if ... else ... fi :-

The if-else-fi statement is the next form of control statement that allows shell to execute statements in a controlled way and make the right choice.

Syntax :-

if [expression]

then

Statement(s) to be executed if expression is true

else

Statement(s) to be executed if expression is not true

fi

The shell expression is evaluated in the above syntax. If the resulting value is true, given statement(s) are executed. If the expression is false, then no statement will be executed.

Eg:- a=10

b=20

if [\$a == \$b]

then

echo "a is equal to b"

else

echo "a is not equal to b"

fi

Nested if :-

If specified condition is not true in if part then else if part will be execute, or else else part will be executed. If none of the condition is true then it processes else part.

Syntax :-

if [expression]
then

statement 1

else if [expression 2]

then

statement 2

...

else

statement n

fi

if conditional constructor :-

If condition checks for the cond expression if the expression or condition is true it executes the statement or else it will come out of loop.

if [expression]

then

statement

fi.

While Condition :-

The while loop enables you to execute a set of commands repeatedly until some condition occurs. It is usually used when you need to manipulate the value of a variable repeatedly.

Syntax :-

while command

do

Statement(s) to be executed if command is true

done.

Eg:-

a=0

```
while [ $a -lt 10 ]
```

```
do
```

```
  echo $a
```

```
  a=$((a + 1))
```

```
done
```

Continue Condition :-

Continue is a command which is used to skip the current iteration in for, while and until loop. It takes one more parameter [N], if N is mentioned then it continues from the nth enclosing loop.

Continue

or

Continue [N]

Eg:- for i in 'seq 1 10'

> do

> if((\$i == 5))

> then

> Continue

> fi

> echo \$i

> done

1

2

3

4

6

7

8

9

10

Variable Types:-

1) Local Variables:- A local variable that is present within the current instance of the shell. It is not available to programs that are started by the shell. They are set at the command prompt.

2) Environment Variables:-

An environment variable is available to any child process of the shell. Some programs need environment variables in order to function correctly. Usually, a shell script defines only those environment variables that are needed by the programs that it runs.

3) Shell Variables:-

A shell variable is a special variable that is set by the shell and is required by the shell in order to function correctly. Some of these variables are environment variables whereas others are local variables.