Programming constructs

The shell programming language has several constructs that give added flexibility to your program.

⇒ Comments / lets you document the functi of the program

⇒ The exit commands let the user termin a program at a point other than the end of the program and use return codes

⇒ The looping constructs, for and while allow a program ~~and~~ to iterate through the commands in a loop.

⇒. The conditional command, if and case execute a group of command only if a ~~test~~ particular set of conditions are met.

⇒ The break command allows the user ~~to~~ exit from the program unconditionally from a loop.

if statement

```
if [expression ]
then
        statement
fi
```

Example

```
a=10     } initialising two variables
b=20.

if [$a == $b].
then
     ~~cat~~ echo "a is equal to b"
fi

# if a & b are not equal.
if [$a != $b]
then
       echo " a is not equal to b"
fi.
```

If else. statement

```
    if [expression].
        then
                statement.
else
        ~~ech~~ statement
fi.
```

Example

```
a= 10    } initialising two variables
b=20     }

  if [ $a = = $b ]
        echo " a is equal to b"
  else
        echo " a is not equal to b"
  fi.
```

Switch statement

```
Cars="bmw"

Case "$cars" in

  "merc") echo "Head quaters, Germany";;

  "Maruti") echo "Head quaters, India";;

  "bmw") echo "Head quater, Kerela";;

esac.
```

Output

```
Headquarter, Kerela.
```

=> **While statement**

```
      i=1
     while (( $i<=10))
   do
     if (( $i==5))
     then
     break
    fi
       echo $i
       ((++i))
       done
```

=> **break statement**

```
      i=0
     until (($i==10))
      do.
      if (($i==5)).
       then
       break
       fi
       echo $i
        ((++i))
```