DECISION MAKING & LOOPING

'while' Statement :

```
initialization;
while (Condition)
{
        loop body;
}
```

The Condition is tested at entry point of the while loop. So, While is called entry-Controlled loop.

'do' Statement :

```
intialization;
do
{
    loop body;
}
while (Condition);
```

The Condition is tested at exit point of the loop. So, do-while is called exit-Controlled loop. Therefore, it executes at-least once even if Condition is false.

'for' Statement :

```
for (initialization ; test condition ; increment)
{
        loop body ;
}
```

Looping process includes following steps:

(i) Initialization of a Counter
(ii) Specifying loop breaking Condition
(iii) Statements inside the loop
(iv) Increment / Decrement of the Counter.

Additional features of FOR Loop:

(i) More than one variable can be initialized at a time in the **FOR** statement.

e.g. `for( x=1 , y=1 ; condition; increment)`

(ii) Like initialization, more than one variable can be increased/decreased at the same time.

e.g. `for(x=1, y=1 ; condition; x++ , y++)`

(iii) Test Condition need not be limited only to the loop control variable.

(iv) If necessary more than one section can be omitted.

e.g.
```
for( ; condition; )
for(initialization;  ;)
```

break: When break statement is encountered then the immediate loop exits and the program continues with the statement, present after the loop.

Continue: When Continue Statement is encountered then for while and do-while, the control directly go to check the loop breaking condition and continues the iteration process. But for FOR loop, the control directly go to increment section and it continues with iteration process.

BREAK is used to jump out of the loop and CONTINUE is used to skip a part of the loop.