

Looping meant, directs a program to perform a set of operations again and again until a specified condition is achieved, which causes the termination of the loop. Programming language Solidity contains three statements for looping:

- while loop
- do... while loop
- for loop

while loop

While loop construct contains the condition first. If the condition is satisfied, the control executes the statements following the while loop else, it ignores these statements. The general form of while loop is:

```
while(condition)
{
    statement1;
    statement2;
    ....
}
```

do... while loop

do-while loop construct is another method used in Solidity programming. do-while loop ensures that the program is executed atleast once and checks whether the condition at the end of the do-while loop is true or false. As long as the test condition is true, the statements will be repeated. The control will come out from the loop, only when the test condition is false. The do-while loop has the following form:



```
do
{
statement1;
statement2;
.....
}
while(condition);
```

The blocks of statements with in double braces { } following the word do are executed at least once. Then the condition is evaluated. If the condition is true, the block of statements are executed again until the value of condition tested is false.

for loop

for loop construct is used to execute a set of statements for a given number of times. Thus, it is a shorthand method for executing statements in a loop. The syntax is:

```
for(initial condition; test condition; incrementer or decrementer)
{
statement1;
statement2;
}
```



for loop construct requires to specify three characteristics. These are:

1. The initial value of the loop counter;
2. Testing the loop counter value to determine whether its current value has reached the number of repetitions desired;
3. Increasing or decreasing the value of loop counter by a specified number, each time the program segment is executed

YouTube Link:



<https://www.youtube.com/embed/rW5KYqWMn7k>

Solidity Source File

{CODE}

