for loop

Used to iterate through the loop.

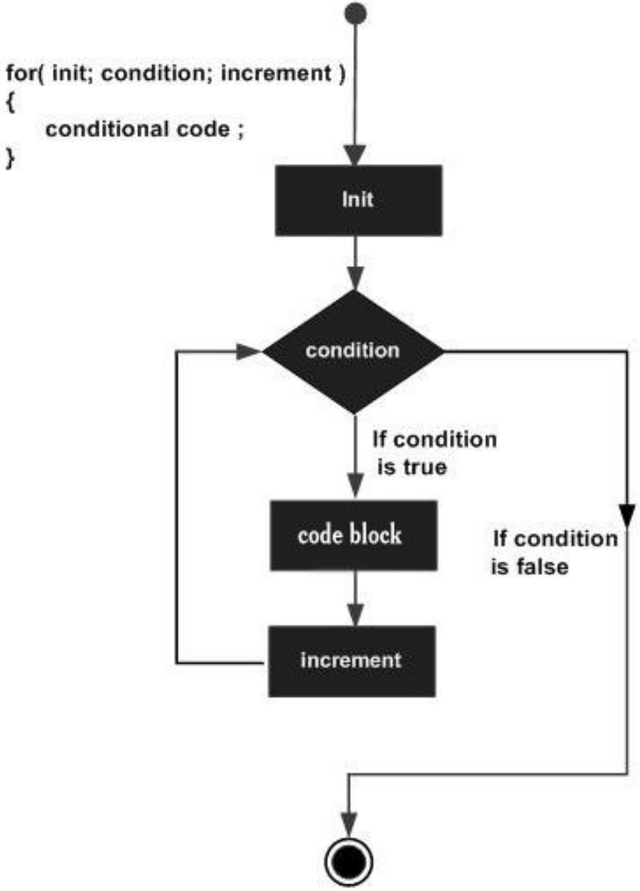
Useful when the number of times to loop is known.

Commonly used to process an array:

```
for (int i=0; i < arrayname.length; i++) {

     // i will go from 0 to last element #
     // reference an element: arrayname[i]

}
```



```
for( init; condition; increment )
{
     conditional code ;
}
```

while loop

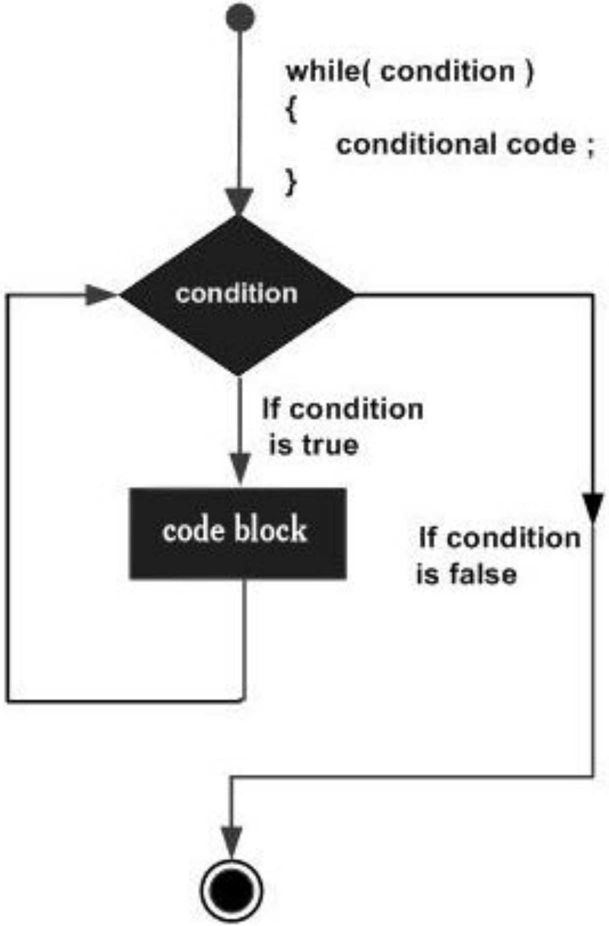
Loop while a condition is true.

May loop zero times since condition is checked before loop

Somewhere in the code-block the condition must be made false at some point in the processing.

To processing an array using a while loop rather than a for loop:

```
int i = 0;    // define an index variable
// loop while index variable is inside the array
while (i < arrayname.length) {
   // process and element using arrayname[i];
i++;  // increment index variable
}
```



do-while loop

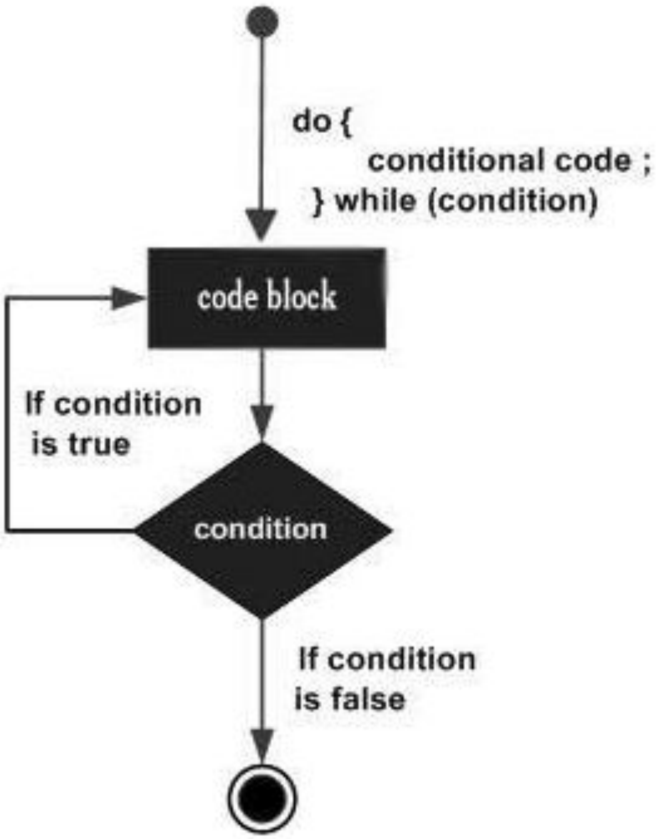
Loop while a condition is true.

Always loop at least once since condition is checked at end of loop

Somewhere in the code-block the condition must be made false at some point in the processing.

To processing an array using a do-while loop rather than a for loop:

```
int i = 0;    // define an index variable
do {
// process and element using arrayname[i];
i++;  // increment index variable
// loop while index variable is inside the array
while (i <= arrayname.length-1) {
}
```



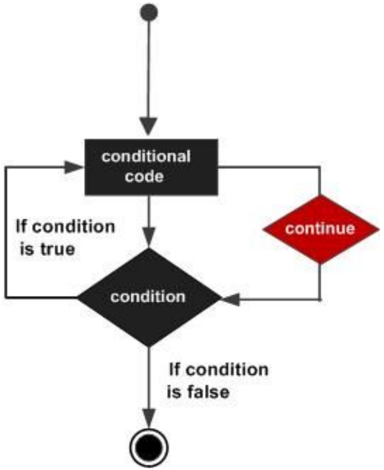
continue / break statement

continue will cause processing to skip to the end of the loop:

In a for loop: the increment

In a while / do while: the condition

break statement will immediately exit the loop.


conditional code
If condition is true
continue
condition
If condition is false