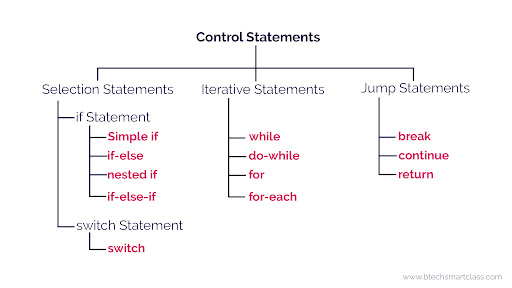
# Control Statements:

* **Control the execution flow of the program**



= 🡪 initialization

== 🡪 condition checking

&& 🡪 both conditions are must satisfied

|| 🡪 any 1 condition

>= greaterthan or equal

<=lesserthan or equal

### IF Statement:

* **It will go inside the block only if the condition is true otherwise, it will not execute the block.**

**if**(condition) {

statement 1; [true]

}

### If-Else Statement

* **If the condition is true then, it will execute the If block. Otherwise, it will execute the Else block.**

**if**(condition) {

statement 1; [true]

}

**else**{

statement 2; [false]

}

### Nested if-statement

* **It Contain an if or if-else statement inside another if or else-if statement.**

**if**(condition 1) {

statement 1; [ condition 1 is true]

**if**(condition 2) {

statement 2; [ condition 2 is true]

}

**else**{

statement 2; [condition 2 is false]

}  }

else{

statement 1; [ condition 1 is false]

}

If-else-if

* **if-statement followed by multiple else-if statements**

**if**(condition 1) {

statement 1; [condition 1 is true]

}

**else** **if**(condition 2) {

statement 2; [ condition 2 is true]

}

**else** **if**(condition 3) {

statement 3; [ condition 2 is true]

}

**else** {

statement 2; [all conditions are false]

}

Switch Statement:

* **select one of many code blocks to be executed**

initialize

int a =10;

**switch(**expression**){**

**case 1:**

**condition;**

**break;**

**case 2:**

**condition;**

**break;**

**default:**

**condition;**

**}**

We can print here.

ITERATIVE STATEMENT:

* While
* Do-while
* For
* For-each

While

Entry loop level condition checking.

**while** (condition){

//code to be executed

increment / decrement statement  }

Do - While

Exit level condition checking

**do**{

//code to be executed / loop body

}**while** (condition);

For Loop

Execute based on the iteration

// Rows : OuterLoop

0 100 //Col : Inner Loop

**for**(initialization; condition; increment/decrement){

//statement or code to be executed

}

For Each or Enhanced For Loop

print each element of the numbers array one by one [i🡺eachitem array🡪target]

datatype d[]={};

int array[] = {10, 22, 31,34,15};

**for**(datatype s :array){

//statement or code to be executed

}

Limitation of for-each Loop;

Not a general-purpose loop. Example: Hello.

JUMP STATEMENT:

Break; It will exit from the loop

Continue; It will skip the particular iteration

Interview question:

How many times 'Hello' is printed?

public class Hello {

public static void main(String[] args){

for(int i = 0; i<5; i++)

{

System.out.println("Hello");

break; / i++;

}

}

}