ASSIGNMENT

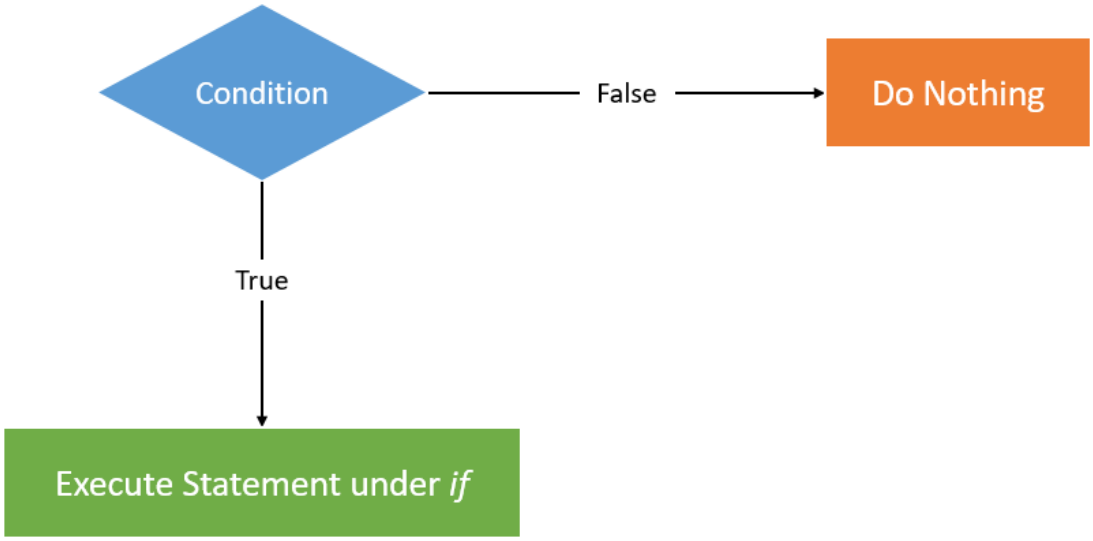
Conditional statements in java

Conditional statements in Java are the executable block of code (or branch to a specific code) dependent on certain conditions. These statements are also known as decision statements or selection statements in Java.

Following are the statements covered under conditional statements in Java:

IF STATEMENT

If a statement executes a set of statements based upon certain conditions. The condition statement follows the if keyword.



Syntax:

```
if (condition)
{
//Statements to be executed
}
```

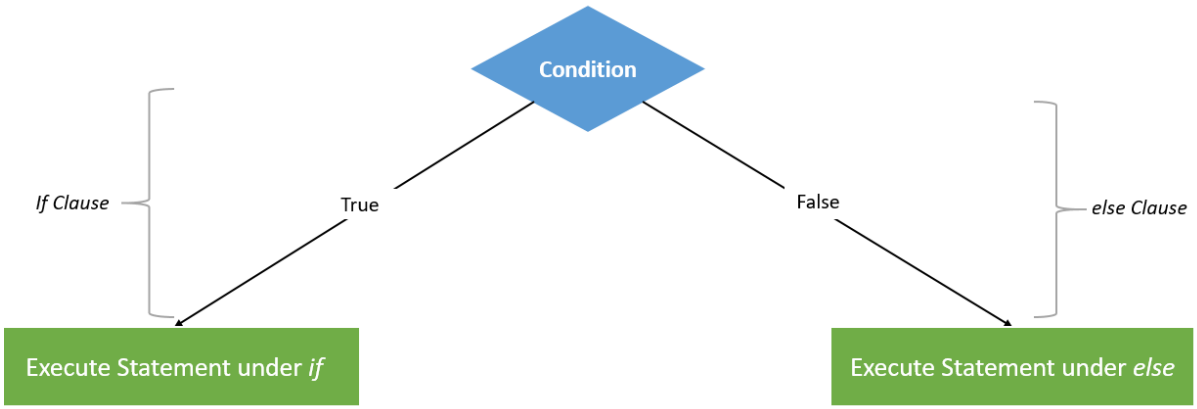
Example-

```
//Java Program to demonstate the use of if statement.
public class IfExample {
public static void main(String[] args) {
   //defining an 'age' variable
   int age=20;
   //checking the age
   if(age>18){
      System.out.print("Age is greater than 18");
   }
}
}
```

If-Else Statement

If-Else statement is a control structure that selects or chooses a set of statements depending upon certain conditions.

If statements are like a subset of if-else statements.



Syntax:

```
if (condition)
{
//Statements to be executed if condition satisfies
}
else
{
//Statements to be executed if the condition is not satisfied
}
```

Example-

```
//A Java Program to demonstrate the use of if-else statement.
//It is a program of odd and even number.
public class IfElseExample {
public static void main(String[] args) {
   //defining a variable
   int number=13;
   //Check if the number is divisible by 2 or not
```

```
    if(number%2==0){
        System.out.println("even number");
    }else{
        System.out.println("odd number");
    }
}
}
```

Switch Statement

Unlike the if-else statement, the switch has multiple paths of execution. Moreover, it evaluates the expression based on some of the primitive types or class types and matches their value with its cases.

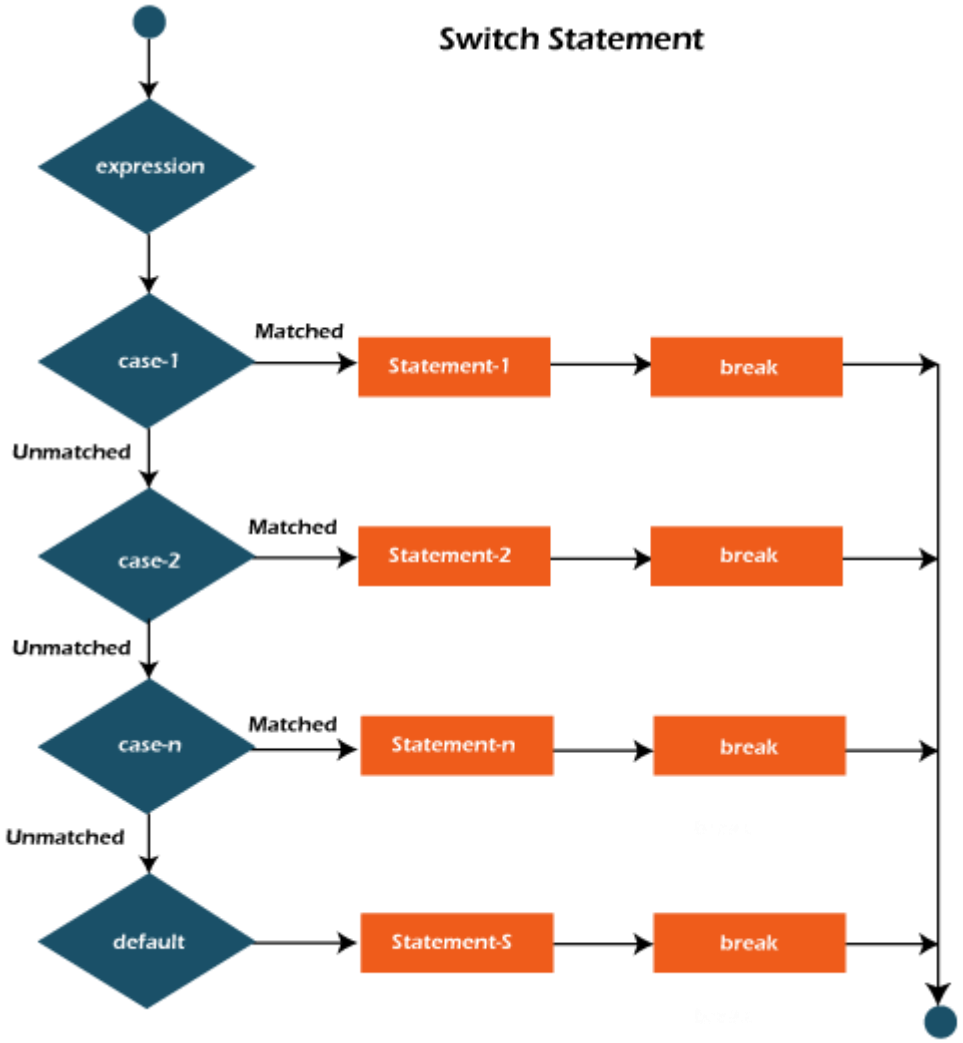
A switch can work well with a byte, char, short, and int primitive data types.

Syntax:

```
switch (Expression)
{
case value 1: Statement 1;
case value 2: Statement 2;
case value 3: Statement 3;
.
.
.
case value n: Statement n;
```

Default: default statement;

}



Example-

public class SwitchExample {

public static void main(String[] args) {

//Declaring a variable for switch expression

int number=20;

//Switch expression

```
    switch(number){
    //Case statements
    case 10: System.out.println("10");
    break;
    case 20: System.out.println("20");
    break;
    case 30: System.out.println("30");
    break;
    //Default case statement
    default:System.out.println("Not in 10, 20 or 30");
    }
}
}
```

One line answers on conditional statements

Ques1 – What are conditional statements?

Ans- The conditional statement is a branch of code that can be executed depending on another condition.

Ques2- If the condition of an IF-statement is false, which is true below.
Ans- Else block is executed.

Ques3- Conditional statements are also called?
Ans- Decision-making statements.

Ques4- How does if statement check the condition?
Ans- firstly, it checks the Boolean value whether it is true or false, if it is true the if block code will be executed otherwise the else block code will be executed.

Ques5- Explain construction of *if*,*else-if* and *else*.
Ans- Use *if* to specify a block of code to be executed, if a specified condition is true. Use else to specify a block of code to be executed, if the same condition is false. Use *else if* to specify a new condition to test, if the first condition is false.

Ques6- Explain *nested-if* statements.
Ans- it means one *If* function inside of another, allows you to test multiple criteria and increases the number of possible outcomes.

Ques7- Another name of *if..else if ladder?*
Ans- Multipath decision statement.

Ques8- What do you mean by *if..else if ladder*?
Ans- The if-else-if statement is used to execute one code from multiple conditions. It is a chain of if..else statements in which each if statement is associated with an else if statement and the last would be an else statement.

Ques9- Which statement can be considered as a replace of *if..else if* ladder?
Ans- Switch case statement.

Ques10- What is *switch-case* statement?
Ans- It is used to test a list of cases. A *switch* statement contains one or more case labels that are tested against the switch expression. When the expression match to a case then the associated statements with that case would be executed.

Ques11- List some constraints while using *switch-case*
Ans-

- The switch statement must be an integral type.
- Case labels must be constants.
- Case labels must be unique.
- Case labels must end with a colon.
- The break statement transfers the control out of the switch statement.
- The break statement is optional.

Ques12- Why we use break statement in switch-case?
Ans- Break statement is used to terminate switch statement execution, once we have achieved our result.

Ques13- What is default statement in *switch-case*?
Ans- When none of the case values are equal to the expression of switch statement then default case is executed. Default case is optional. If default case is absent and no case values match then none of the statements from switch are executed.

Ques14- The conditional statement, ……. can only test for equality, whereas ……. can evaluate any type of Boolean expression.
Ans- switch, if.

Ques15- Is there any limit for maximum lines of code that can be written inside a Java style IF, ELSE or IF-ELSE block?

Ans- There is no such limit on the number of lines of code in any block or statement in Java.