

# Java Conditional Statements | If, If-Else, If-Else-If Ladder, Switch

## What is Conditional Statement?

Conditional statements are used to **make decisions in Java**.

Example real life:

- If it rains → take umbrella
- Else → go normally

Same in Java.

## 1. IF Statement

IF means: **Check condition. If true → run code**

### Syntax

```
if (condition) {  
    // code  
}
```

### Example

```
int age = 20;  
  
if (age >= 18) {  
    System.out.println("You can vote");  
}
```

### Explanation

- $age \geq 18 \rightarrow \text{true}$
- So code runs.

## 2. IF-ELSE Statement

If condition is true → run IF

Else → run ELSE

### Syntax

```
if (condition) {  
    // true block  
} else {  
    // false block  
}
```

## Example

```
int age = 15;  
  
if (age >= 18) {  
    System.out.println("You can vote");  
} else {  
    System.out.println("You cannot vote");  
}
```

## 3. IF-ELSE-IF Ladder

Used when **multiple conditions** are there.

### Syntax

```
if (condition1) {  
}  
else if (condition2) {  
}  
else if (condition3) {  
}  
else {  
}
```

## Example (Marks System)

```
int marks = 85;  
  
if (marks >= 90) {  
    System.out.println("Grade A");  
}  
else if (marks >= 75) {  
    System.out.println("Grade B");  
}  
else if (marks >= 50) {  
    System.out.println("Grade C");  
}
```

```
}

else {
    System.out.println("Fail");
}
```

## How Ladder Works?

Java checks **top to bottom**  
First true condition → stop checking

## 4. SWITCH Statement

Used when **many fixed choices** exist  
Better than many else-if

### Syntax

```
switch (variable) {
    case value1:
        // code
        break;

    case value2:
        // code
        break;

    default:
        // code
}
```

### Example

```
int day = 3;

switch(day) {
    case 1:
        System.out.println("Monday");
        break;
    case 2:
        System.out.println("Tuesday");
        break;
    case 3:
        System.out.println("Wednesday");
```

```

        break;
    default:
        System.out.println("Invalid day");
}

```

## Why break is Important?

If you don't use break → Java will run all next cases (fall-through)

## Example Without break (Danger)

```

int x = 1;

switch(x) {
    case 1:
        System.out.println("One");
    case 2:
        System.out.println("Two");
}

```

Output:

One  
Two

## Switch vs If-Else Ladder

If-Else Ladder	Switch
Works with ranges	Works with fixed values
Complex conditions	Simple equality
Slower	Faster

## Important Notes

IF

Used for one condition

IF-ELSE

Two choices

IF-ELSE-IF

Multiple conditions

SWITCH

Multiple fixed options

## Memory Trick

IF = check

ELSE = otherwise

ELSE IF = multiple checks

SWITCH = menu system

## Switch vs If-Else-If Ladder

### 1. Switch Statement

Switch can check **only equal values**.

**Example:**

```
int day = 2;
```

```
switch(day) {  
    case 1: System.out.println("Monday"); break;  
    case 2: System.out.println("Tuesday"); break;  
}
```

You cannot write:

```
case day > 2: // not allowed
```

### 2. If-Else-If Ladder

If-else-if can check **all types of conditions**.

**Example:**

```
int marks = 80;
```

```
if(marks > 90) {  
    System.out.println("A");
```

```
}

else if(marks > 60) {
    System.out.println("B");
}
```

Switch	If-Else-If
Checks only equality (==)	Can check >, <, ==, !=
Good for menu options	Good for ranges
Faster for many cases	Slower if many conditions

Switch = menu

If-Else = decision logic