

## Java Switch Statement

In Java, the switch statement is used for executing one statement from multiple conditions. It is similar to an if-else-if ladder.

Switch statement consists of conditional based cases and a default case.

In a switch statement, the expression can be of byte, short, char and int type.

From JDK-7, enum, String can also be used in switch cases.

Following are some of the rules while using the switch statement:

There can be one or N numbers of cases.

The values in the case must be unique.

Each statement of the case can have a break statement. It is optional.

- Syntax:

```
switch(expression)
{
    case value1:
        //code for execution;
        break; //optional
    case value2:
        // code for execution
        break; //optional
```

.....

.....

.....

.....

Case value n:

`// code for execution`

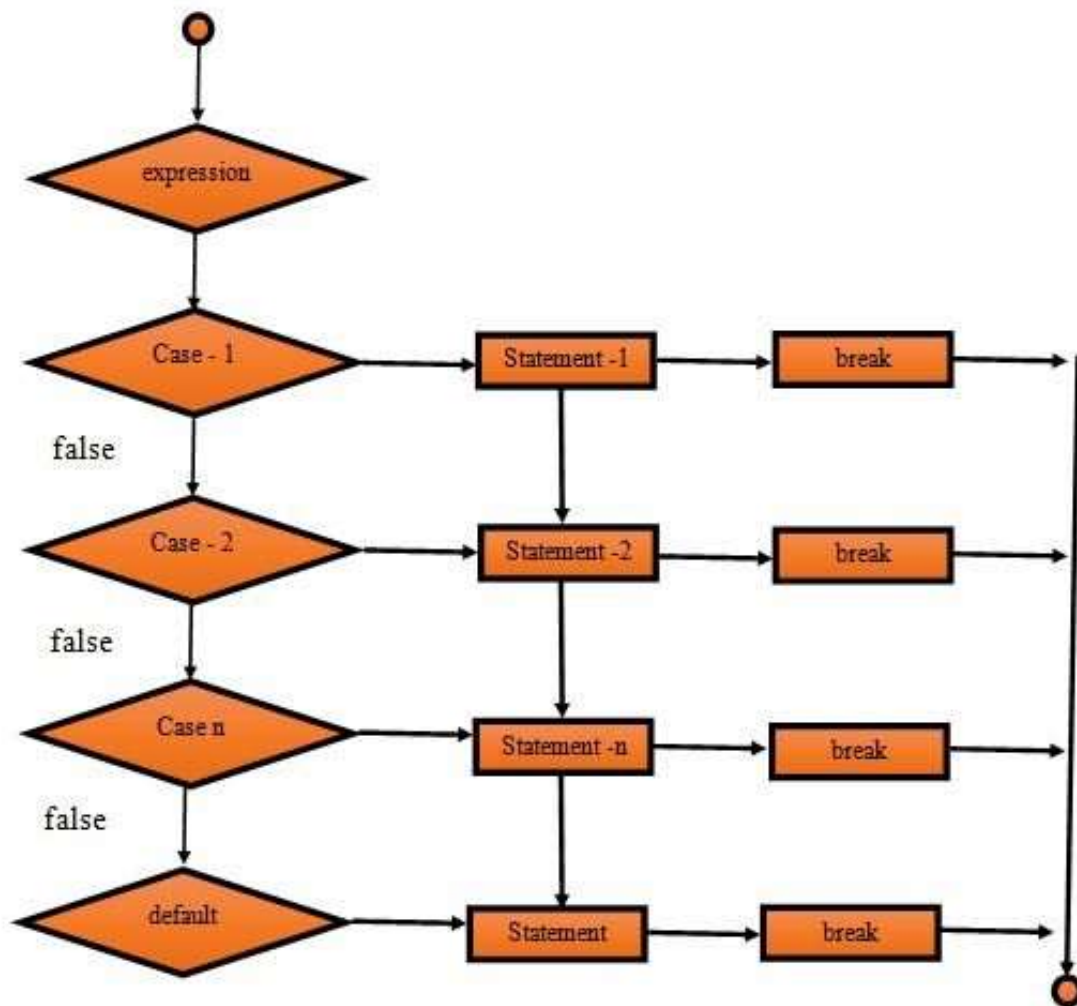
`break; //optional`

default:

`code for execution when none of the case is true;`

}

- Data Flow Diagram Of switch Block



- Example: Using integer value

In this example, we are using int type value to match cases. This example returns day based on the numeric value.

```
class SwitchDemo1{  
    public static void main(String[] args)  
    {  
        int day = 3;  
        String dayName;  
        switch (day) {  
            case 1:  
                dayName = "Today is Monday";  
                break;  
            case 2:  
                dayName = "Today is Tuesday";  
                break;  
            case 3:  
                dayName = "Today is Wednesday";  
                break;  
            case 4:  
                dayName = "Today is Thursday";  
                break;  
            case 5:  
                dayName = "Today is Friday";  
                break;  
            case 6:  
                dayName = "Today is Saturday";  
                break;  
        }  
    }  
}
```

case 7:

dayName = "Today is Sunday";

break;

default:

dayName = "Invalid day";

}

System.out.println(dayName);

}

}

- Example: String in switch case

Since Java has allowed to use string values in switch cases, so we are using string to create a string based switch case example.

```
class Switch1{
```

```
public static void main(String[] args) {
```

```
    String name = "Mango";
```

```
    switch(name){
```

```
        case "Mango":
```

```
            System.out.println("It is a fruit");
```

```
            break;
```

```
        case "Tomato":
```

```
            System.out.println("It is a vegetable");
```

```
            break;
```

```
        case "Coke":
```

```
            System.out.println("It is cold drink");
```

```
    }
```

```
}
```

```
}
```

- Example: without break switch case

Break statement is used to break the current execution of the program. In switch case, break is used to terminate the switch case execution and transfer control to the outside of switch case.

Use of break is optional in the switch case. So let's see what happens if we don't use the break.

```
class Demo
```

```
{
```

```
    public static void main(String[] args) {
```

```
        String name = "Mango";
```

```
        switch(name){
```

```
            case "Mango":
```

```
                System.out.println("It is a fruit");
```

```
            case "Tomato":
```

```
                System.out.println("It is a vegetable");
```

```
            case "Coke":
```

```
                System.out.println("It is cold drink");
```

```
        }
```

```
    }
```

```
}
```