GeeksforGeeks
A computer science portal for geeks

Custom Search

Write an Article

Login

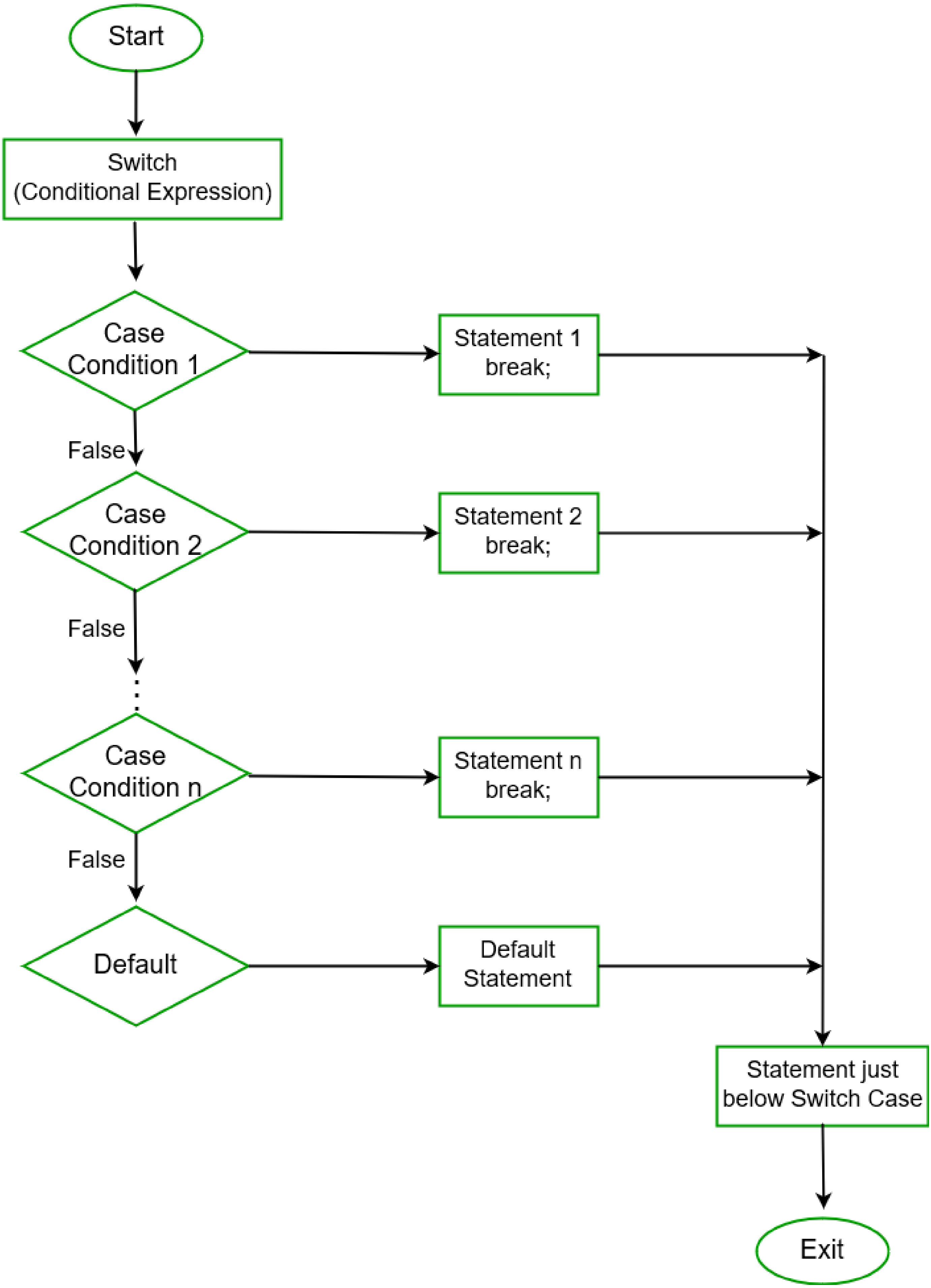
Switch Statement in Java

The switch statement is a multi-way branch statement. It provides an easy way to dispatch execution to different parts of code based on the value of the expression. Basically, the expression can be byte, short, char, and int primitive data types. Beginning with JDK7, it also works with enumerated types (Enums in java), the String class and Wrapper classes.

Syntax of Switch-case :

```
// switch statement 
switch(expression)
{
   // case statements
   // values must be of same type of expression
   case value1 :
      // Statements
      break; // break is optional
   
   case value2 :
      // Statements
      break; // break is optional
   
   // We can have any number of case statements
   // below is default statement,used when none of the cases is true. 
   // No break is needed in the default case.
   default : 
      // Statements
}
```

Flow Diagram of Switch-case :



Some Important rules for switch statements :

- Duplicate case values are not allowed.
- The value for a case must be the same data type as the variable in the switch.
- The value for a case must be a constant or a literal.Variables are not allowed.
- The break statement is used inside the switch to terminate a statement sequence.
- The break statement is optional. If omitted, execution will continue on into the next case.
- The default statement is optional, and it must appear at the end of the switch.

Examples:

Consider the following java program, it declares an int named day whose value represents a day(1-7). The code displays the name of the day, based on the value of day, using the switch statement.

```java
// Java program to demonstrate switch case
// with primitive(int) data type
public class Test
{
    public static void main(String[] args)
    {
        int day = 5;
        String dayString;

        // switch statement with int data type
        switch (day)
        {
            case 1:  dayString = "Monday";
                     break;
            case 2:  dayString = "Tuesday";
                     break;
            case 3:  dayString = "Wednesday";
                     break;
            case 4:  dayString = "Thursday";
                     break;
            case 5:  dayString = "Friday";
                     break;
            case 6:  dayString = "Saturday";
                     break;
            case 7:  dayString = "Sunday";
                     break;
            default: dayString = "Invalid day";
                     break;
        }
        System.out.println(dayString);
    }
}
```

Run on IDE

Output:

```
Friday
```

Omitting the break statement

As break statement is optional. If we omit the break, execution will continue on into the next case. It is sometimes desirable to have multiple cases without break statements between them. For example, consider the updates version of above program, it also display whether a day is a weekday or a weekend day.

```java
// Java program to demonstrate switch case
// with multiple cases without break statements
public class Test
{
    public static void main(String[] args)
    {
        int day = 2;
        String dayType;
        String dayString;

        switch (day)
        {
            case 1:  dayString = "Monday";
                     break;
            case 2:  dayString = "Tuesday";
                     break;
            case 3:  dayString = "Wednesday";
                     break;
            case 4:  dayString = "Thursday";
                     break;
            case 5:  dayString = "Friday";
                     break;
            case 6:  dayString = "Saturday";
                     break;
            case 7:  dayString = "Sunday";
                     break;
            default: dayString = "Invalid day";
        }

        switch (day)
        {
          //multiple cases without break statements

            case 1:
            case 2:
            case 3:
            case 4:
            case 5:
                dayType = "Weekday";
                break;
            case 6:
            case 7:
                dayType = "Weekend";
                break;

            default: dayType= "Invalid daytype";
        }

        System.out.println(dayString+" is a "+ dayType);
    }
}
```

Run on IDE

Output:

```
Tuesday is a Weekday
```

Nested Switch Case statements

We can use a switch as part of the statement sequence of an outer switch. This is called a nested switch. Since a switch statement defines its own block, no conflicts arise between the case constants in the inner switch and those in the outer switch. For example:

```
// java program to demonstrate
// nested switch case statement
public class Test
{
    public static void main(String[] args)
    {
        String Branch = "CSE";
        int year = 2;

        switch(year)
        {
            case 1 :
                System.out.println("elective courses : Advance english, Algebra");
                break;
            case 2:
                switch (Branch) // nested switch
                {
                    case "CSE":
                    case "CCE":
                        System.out.println("elective courses : Machine Learning, Big Data");
                    break;

                    case "ECE":
                        System.out.println("elective courses : Antenna Engineering");
                    break;

                    default:
                        System.out.println("Elective courses : Optimization");
                }
        }
    }
}
```

Run on IDE

Output:

```
elective courses : Machine Learning, Big Data
```

This article is contributed by **Gaurav Miglani**. If you like GeeksforGeeks and would like to contribute, you can also write an article using contribute.geeksforgeeks.org or mail your article to contribute@geeksforgeeks.org. See your article appearing on the GeeksforGeeks main page and help other Geeks.

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

Java School Programming java-basics

Login to Improve this Article

Please write to us at contribute@geeksforgeeks.org to report any issue with the above content.

Recommended Posts:

String in Switch Case in Java

Do we need forward declarations in Java?
Decision Making in Java (if, if-else, switch, break, continue, jump)
Comments in Java
Classes and Objects in Java
Java Math copySign() method with Examples
Java | Converting an Image into Grayscale using cvtColor()
Java | Removing whitespaces using Regex
Java System.nanoTime() vs System.currentTimeMillis
Java toDegrees() method with Example

(Login to Rate)

1.5 Average Difficulty : **1.5/5.0**
Based on **18** vote(s)

Basic Easy Medium Hard Expert

□ Add to TODO List
□ Mark as DONE

Writing code in comment? Please use ide.geeksforgeeks.org, generate link and share the link here.

Load Comments Share this post!

A computer science portal for geeks

710-B, Advant Navis Business Park,
Sector-142, Noida, Uttar Pradesh - 201305
feedback@geeksforgeeks.org

COMPANY
About Us
Careers
Privacy Policy
Contact Us

LEARN
Algorithms
Data Structures
Languages
CS Subjects
Video Tutorials

PRACTICE
Company-wise
Topic-wise
Contests
Subjective Questions

CONTRIBUTE
Write an Article
GBlog
Videos