switch Statement

In this lesson, an introduction of the switch statement, its basic syntax, and how it is written using an example is provided.

We'll cover the following

- The switch case construct
  - Explanation

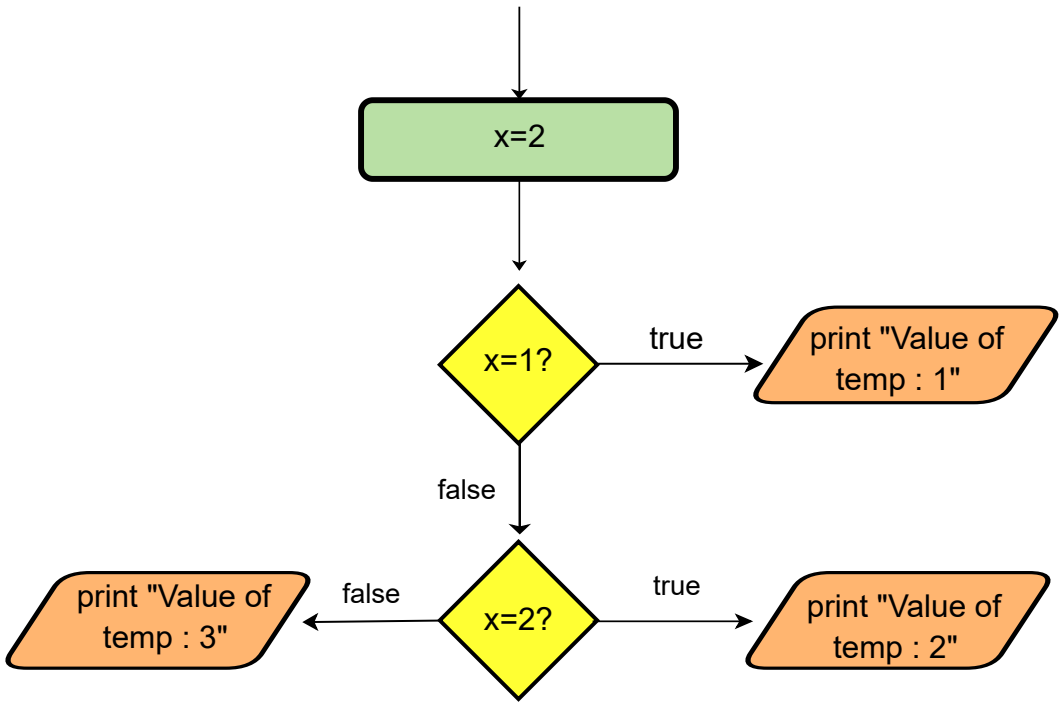
The switch case construct

This `switch` clause tests an input variable for equality with any number of cases and then executes the corresponding code.

The `switch` statement is very similar to the `if-else if` statement, described in the previous lesson. The only difference is that the `switch` case construct **only** uses `int`, `short` values, and character *constants* **or** character *literals*. For this reason, the **switch-case** can come in handy, but it is limited to these circumstances.

```
class switch_statement {
    public static void main(String[] args) {
        int x = 2;
        int temp;

        switch (x) {
            case 1:
                temp = 1;
                break;
            case 2:
                temp = 2;
                break;
            default:
                temp = 3;
                break;
        }
        System.out.println("Value of temp: " + temp);
    }
}
```



Explanation

Here is a line-by-line explanation:

- **Line 6:** The `switch` will take the value of `x` in its input and then compare this value with each case value. If the value of `x` matches the value of the case, then the statements under this case will be executed.
- **Line 7:** Since `x` is not equal to 1. Therefore, statements under this case won't be executed.
- **Line 10:** Since `x` is equal to 2. Therefore, statements under this case will be executed.
- **Line 12:** `break` allow us to come out of the `switch` block immediately. If you don't use the `break`, then all the statements following the correct case will be executed.

> **Note:** `break` is used to break the normal execution of code, skipping everything in a conditional or a loop block and moving to the statement right after the conditional or loop block.

- **Line 13:** The `default` block is similar to the `else` block in a normal conditional. It is chosen if **none** of the *previous* cases matched the value of the *condition* variable.

Sometimes the same statement(s) should be executed in multiple conditions. The following example shows how such a scenario can be implemented using a `switch` case statement.

```
class switch_statement {
    public static void main(String[] args) {
        int x = 2;
        int temp;
        switch (x) {

            case 1:
            case 2:
            case 3:
                temp = 0;
                break;
            case 4:
                temp = 4;
                break;
            default:
                temp = 5;
                break;
        }
        System.out.println("Value of temp: " + temp);
    }
}
```

Try changing the value of `x` above to **1**, **3**, **4**, or **any other number** and then see the value of `temp` in output.

Now let's look at the *conditional expressions* in the next lesson.