Decision making structures have one or more conditions to be evaluated or tested by the program, along with a statement or statements that are to be executed if the condition is determined to be true, and optionally, other statements to be executed if the condition is determined to be false.

Following is the general form of a typical decision making structure found in most of the programming languages:

![Decision Making Structure Diagram](image)

Java programming language provides following types of decision making statements. Click the following links to check their detail.

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>if statement</td>
<td>An if statement consists of a boolean expression followed by one or more statements.</td>
</tr>
<tr>
<td>if...else statement</td>
<td>An if statement can be followed by an optional else statement, which executes when the boolean expression is false.</td>
</tr>
<tr>
<td>nested if statements</td>
<td>You can use one if or else if statement inside another if or else if statements.</td>
</tr>
<tr>
<td>switch statement</td>
<td>A switch statement allows a variable to be tested for equality against a list of values.</td>
</tr>
</tbody>
</table>

**The ? : Operator:**

We have covered conditional operator ? : in previous chapter which can be used to replace if...else statements. It has the following general form:

```
Exp1 ? Exp2 : Exp3;
```

Where Exp1, Exp2, and Exp3 are expressions. Notice the use and placement of the colon.
To determine the value of whole expression, initially \( \text{exp1} \) is evaluated

- If the value of \( \text{exp1} \) is true, then the value of \( \text{Exp2} \) will be the value of the whole expression.
- If the value of \( \text{exp1} \) is false, then \( \text{Exp3} \) is evaluated and its value becomes the value of the entire expression.

**What is Next?**

Next chapter discusses about the Number class \textit{in}\texttt{thejava.langpackage} and its subclasses in Java Language.

We will be looking into some of the situations where you would use instantiations of these classes rather than the primitive data types, as well as classes such as formatting, mathematical functions that you need to know about when working with Numbers.