CONDITIONAL LOGIC

Java programming is a *sequential programming*. In essence its code is executed from top to bottom. It's very linear, in that each and every line of code will be read, starting with the first line of code you write and ending at the last line.

Sometimes, you would want certain conditions to be met before you execute some code. For example, you might want to display a message if a user is below the age of 18. In essence, you want to control the flow of the program for yourself. This can be achieved with conditional logic.

Conditional logic is mainly about the 'if' and 'else' keywords.

If Statement

The `if` statement executes a code when a condition is met. The structure of the `if` statement in Java is like this:

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

As exemplified above, the condition to be met is contained in the round bracket; the code to be executed when the condition is met is contained in the curly brackets. For instance:

```
int userAge= 17;
if (userAge < 18) {
      System.out.println (“User is less than
18”);
}
```

Explaining the instance above, we set up an integer variable to contain the age of the user. Thereafter is our `if` statement. Our `if` statement checks if the age of our user is less than 18. And if the user is below the age 18, the code in the curly brackets, which prints out the user’s age, is executed.

If … Else Statement

With only an if statement, you can only execute codes on one condition what if when the condition is not met, you would still want to execute some other code. For something like this, we use the `if` and `else` statement. Its structure is like this:

```
if (condition)  {
      //code_one;
}
else {
      //code_two;
}
```

The first line starts with `if`, followed by the condition you want to test for. The condition is contained in the round brackets. If the condition is met the code in the curly brackets would be executed, but if the condition is not the code in the curly brackets of the else section is executed. For instance:

```
int usersAge= 17;
if (usersAge < 18) {
      System.out.println (“User is younger than
18”);
```

```
}
else {
      System.out.println (“User is 18 or
older”);
}
```

If … Else if … Else Statement

If you wanted to test for more than two choices, `if … else if … else` statement can be used. The structure of the `if … else if … else` statement is like this:

```
if (condition_one) {
      //code one;
}
else if {
      //code two;
}
else {
      //code three;
}
```

Nested If Statements

Nesting an `if` statement just means putting one if statement inside of another. You may want to check if a person is a pupil, and thereafter check if he is a pupil in class one, so to execute a different code. For such you would need to put the `if` statement that checks for the class of the pupil inside the `if` statement that checks if the user is a pupil.