Nuances of working with a conditional statement

1. Sequence of `if` statements

Sometimes a program needs to perform many different actions depending on the value of a variable or the value of an expression.

Let's say our task is something like this:

- If the temperature is greater than `20` degrees, then put on a shirt
- If the temperature is greater than `10` degrees and less than (or equal to) `20`, then put on a sweater
- If the temperature is greater than `0` degrees and less than (or equal to) `10`, then put on a raincoat
- If the temperature is less than `0` degrees, then put on a coat.

Here's how this can be represented in code:

```
int temperature = 9;

if (temperature > 20) {
   System.out.println("put on a shirt");
} else { // Here the temperature is less than (or equal to) 20
   if (temperature > 10) {
      System.out.println("put on a sweater");
   } else { // Here the temperature is less than (or equal to) 10
      if (temperature > 0) {
         System.out.println("put on a raincoat");
      } else // Here the temperature is less than 0
         System.out.println("put on a coat");
   }
}
```

`If-else` **statements can be nested within one another**. This makes it possible to implement rather complex logic in a program.

However, programmers usually write this construct a little differently:

```
int temperature = 9;

if (temperature > 20) {
   System.out.println("put on a shirt");
} else if (temperature > 10) { // Here the temperature is less than (or equal to) 20
   System.out.println("put on a sweater");
} else if (temperature > 0) { // Here the temperature is less than (or equal to) 10
   System.out.println("put on a raincoat");
} else { // Here the temperature is less than 0
   System.out.println("put on a coat");
}
```

The two examples given are equivalent, but the second one is easier to understand.

---

2. Nuances of the `else` block

An important point:

If don't use curly braces in an `if-else` construct, then the `else` refers to the closest previous `if`.

Example:

Our code	How will it work

`int age = 65;`
 	
`if (age < 60)`
`  if (age > 20)`
`  System.out.println("You must work");`
`else`
`  System.out.println("You don't have to work");`
	`int age = 65;`
	
	`if (age < 60) {`
	`  if (age > 20)`
	`  System.out.println("You must work");`
	`  else`
	`  System.out.println("You don't have to work");`
	`}`

If you look at the code on the left, it seems that the screen output will be "You don't have to work". But that isn't the case. In reality, the `else` block and the "You don't have to work" statement are associated with the second (the closer) `if` statement.

In the code on the right, the associated `if` and `else` are highlighted in red. Additionally, the curly braces are placed unambiguously, clearly showing what actions will be performed. The string `You don't have to work` is never displayed when `age` is greater than `60`.

3. Example of using an `if-else` statement

Since we explored the `if-else` statement so well, let's give an example:

```
import java.util.Scanner;
public class Solution {
   public static void main(String[] args) {
     Scanner console = new Scanner(System.in); // Create a Scanner object
     int a = console.nextInt(); // Read the first number from the keyboard
     int b = console.nextInt(); // Read the second number from the keyboard
     if (a < b)                 // If a is less than b
       System.out.println(a);   // we display a
     else                       // otherwise
       System.out.println(b);   // we display b
   }
}
```

Displaying the minimum of two numbers