

Conditional Logic

If – Else Statements

- So far we have only been doing sequential programming, meaning the code is executed from top to bottom.
- This type of programming is very linear, meaning that each and every line of code will be read, starting with the first line of code and ending at the last line.
- However, sometimes we want code to be executed only if certain conditions are met.
- That's where Conditional Logic comes into play.
- Conditional Logic includes the **IF** and **ELSE** Statements.
- The **if** statement will execute some code if a condition is true.
- The **else** statement will execute some code if the condition in the **if** statement is false.

IF Statements

- The structure of the **IF Statement** in Java is this:
- **if (Statement or condition goes here) { Code goes here }**
- 1. You start with the word IF (in lowercase) and a pair of round brackets.
- 2. 3. A pair of curly brackets are used to section off a chunk of code.
- 4. This chunk of code is code that you only want to execute **IF** your condition is met.
- Whatever you type between the curly brackets will only be executed **IF** the condition is met.

If Statement

- **Example:** `if (user < 21) {System.out.println("User cannot buy alcohol"); }`
- Our condition says "IF user is less than 21".
- If the user is not less than 21
- The code between the curly brackets will be skipped,
- The program continues downwards towards the last line of code.
- We can use the right-pointing angle bracket which means "greater than".
- Our IF Statement above can check for users who are greater than 21:
- **Example:** `if (user > 21) {System.out.println("User can buy alcohol"); }`

If Statement

- What if we want check for people who are exactly 21.
- If we want to check for those who are 18 or over, we can say "greater than or equal to".
- The symbols for this are **>=**
- **Example:** `if (user >= 21) {System.out.println("User can buy alcohol"); }`
- You can also check for "**<=**" in a similar way:

Practice Project: Conditional Logic

- `public class IFStatements {`
- `public static void main(String[] args) {`
- `int user = 17;`
- `if (user < 18) {`
- `System.out.println (" user is less than 18");`
- `}`
- `}`
- `}`

If Statement

- `public class IFStatement2 {`
- `public static void main(String[] args){`
- `int user = 18;`
- `if (user < 18) {`
- `if (user > 18) {`

If Else Statement

- Instead of using two IF Statements, you can use an IF ... ELSE Statement
- Here's the structure of an IF ... ELSE statement:
- **if (condition_to_test) {**
- **}**
- **else {**
- **}**

If Else Statement

- `public class IFStatement2 {`
- `public static void main(String[] args){`
- `int user = 17;`
- `if (user < 18) {`
- `System.out.println("User is 18 or younger");`
- `}`
- `else {`
- `System.out.println("User is older than 18");`
- `}`
- `}`
- `}`

Switch Statements

- So far we have used IF ...Else to control the flow of our program.
- Another way to control the flow of your programs is to use a switch statement.
- A switch statement gives you the option to test for a range of values for your variables.
- They can be used instead of long, complex **If ... Else If** statements.
- For every value that you want to check, you need the word **case**.

Switch Statements

- The structure of the switch statement is this:

```
switch ( variable_to_test ) {  
case value: ← Value to be evaluated.  
code_here; ← What we want the program to do.  
break; ← break out of each case.  
case value:  
code_here;  
break; break out of each case  
default:  
values_not_caught_above;
```

Switch Example

- `public static void main(String[] args){`
- `int user = 18;`
- `switch (user) {`
- `case 18:`
- `System.out.println("You are 18");`
- `break;`
- `case 19:`
 - `System.out.println("You are 19");`
- `break;`
- `case 20:`
 - `System.out.println("You are 20");`
- `break;`
- `default:`
 - `System.out.println("You are not 18, 19 or 20");`

(value to test for is 18)

(The switch statement will check the user variable)

(It will then go through each of the case statements)

(When it finds one that matches, it will stop and execute)

(It will then break out of the switch statement)

Practice

- A clothing store offer a discount that depends on the quality of the goods based on class. Write a code using a Switch statement.
- Class "A" goods are not discounted at all.
- Class "B" goods are discounted 10%.
- Class "C" goods are discounted 20%.
- Anything else is discounted 30%.

