



Coding-bootcamps.com

Introduction to Java Programming Language

By Jim Sullivan
from [Coding Bootcamps](https://coding-bootcamps.com)



Flow Control

Session 4



Brief Recap

- Re-hash what we've covered in our last class

Objective - Discuss The Following

- Conditional Constructs
- Looping Constructs
 - Counter-Controlled Repetition
 - Sentinel-Controlled Repetition
- Nested Control Construct
- break and continue Statements
- Structured Programming Best Practices

Materials

- These Powerpoint Slides

OOP - Conditional Constructs

- Java uses boolean variables to evaluate conditions.
 - The boolean values true and false are returned when an expression is compared or evaluated. For example:

```
int a = 4;
```

```
if (a == 4) {  
    System.out.println("Ohhh! So a is 4!");  
}
```

OOP - Conditional Constructs

- There aren't that many operators to use in conditional statements and most of them are pretty straightforward:

OOP - Conditional Constructs

```
int a = 4;
```

```
int b = 5;
```

```
boolean result;
```

```
result = a < b; // true
```

```
result = a > b; // false
```

```
result = a <= 4 // a smaller or equal to 4 - true
```

```
result = b >= 6 // b bigger or equal to 6 - false
```

```
result = a == b // a equal to b - false
```

```
result = a != b // a is not equal to b - true
```

```
result = a > b || a < b // Logical or - true
```

```
result = 3 < a && a < 6 // Logical and - true
```

```
result = !result // Logical not - false
```


OOP - Conditional Constructs

```
int a = 4;
```

```
int b = 5;
```

```
boolean result;
```

```
result = a < b; // true
```

```
result = a > b; // false
```

```
result = a <= 4 // a smaller or equal to 4 -  
true
```

```
result = b >= 6 // b bigger or equal to 6 - false
```

```
result = a == b // a equal to b - false
```

```
result = a != b // a is not equal to b - true
```

```
result = a > b || a < b // Logical or - true
```

```
result = 3 < a && a < 6 // Logical and - true
```

```
result = !result // Logical not - false
```

OOP - Conditional Constructs

- The if, else statement in java is pretty simple.

```
if (a == b) {  
    System.out.println("Another line Wow!");  
} else {  
    System.out.println("Double rainbow!");  
}
```

OOP - Looping Constructs (For)

- There are two kind of loops in Java, for and while.
- For
The for loop has three sections:

```
for (int i = 0; i < 3; i++) {}
```

OOP - Looping Constructs (For)

- First section runs once when we enter the loop.
- Second section is the gate keeper,
 - If it returns true, we run the statements in the loop
 - If it returns false, we exit the loop.
 - It runs right after the first section for the first time, then every time the loop is finished and the third section is run.
- The third section is the final statement that will run every time the loop runs.

OOP - Looping Constructs (While)

- The syntax is very similar to the previous for we looked at:

```
while (condition) {}
```

OOP - Looping Constructs (Foreach)

- Another version of for, is the foreach. The keyword we use is still for, but when we want to iterate on the elements inside an array we can simply use it

```
int[] arr = {2, 0, 1, 3};  
  
for (int el : arr) {  
  
    System.out.println(el);  
  
}
```

OOP - break and continue Statements

- Another version of for, is the foreach. The keyword we use is still for, but when we want to iterate on the elements inside an array we can simply use it

```
int i;

for (i = 0; i < 5; i++) {
    if (i >= 2) {
        Break;
    }
    System.out.println("Yuhu");
}

System.out.println(i);
```

// Output:
// Yuhu
// Yuhu
// 2

OOP - Structured Programming Best Practices

- Use Strings carefully
 - If two Strings are concatenated using “+” operator in a “for” loop, then it creates a new String Object, every time.
- One of the most expensive operations (in terms of Memory Utilization) in Java is Object Creation. Thus it is recommended that Objects should only be created or initialized if necessary
-

Assignment

- Using if statements, else if, and else statements, make a program which displays a different message depending on the age given.

age	message
less than 16	"You can't drive."
16 to 17	"You can drive but not vote."
18 to 24	"You can vote but not rent a car."
25 or older	"You can do pretty much anything."

Assignment

Reminder, this is how you read and accept keyboard input

- ```
Scanner keyboard = new Scanner(System.in);
System.out.println("enter an integer");
int myint = keyboard.nextInt();
```

# Summary

# Live private coaching sessions for Java

- [Private tutoring sessions for software design and engineering- Weekly and monthly plans](#)
- [Java programming language- Private tutoring sessions](#)



Coding-bootcamps.com

Thank You



**Coding**  
Bootcamps