

Lesson 37: Conditionals #1 (W11D2)

Balboa High School

Michael Ferraro

October 27, 2015

- Create a new project, Lesson37
- Import the starter code for class SumOfSquaresFor from [here](#).
- Complete the `getSumOfSquares()` method using a `for()` loop.
 - If you get confused, write it first using a `while()` loop and then change it to a `for()` loop.
 - To test your method, add a `main()` that creates an instance of the `SumOfSquares` class, as described in the comments.
- Your code will be tested shortly using an automated tester!¹

¹...just like code you write for PS #6!

Students will learn about conditionals and boolean expressions in Java.

Conditionals

- Conditionals provide a means by which you can write software that makes decisions
- if a **condition** is met, then...
- while a **condition** is met, do...

if() Statements

- We've already used these on a limited basis
- Syntax:

```
if ( <condition> ) {  
    statement1;  
    statement2;  
    ...;  
}
```

→ *If* <condition> is true, run the statements.

if() Statements

- We've already used these on a limited basis
- Example:

```
int age = 6;
```

```
if ( age < 21 ) {  
    System.out.println("Too young to drink!");  
}
```

if () Statements

If there's only one statement to run, you may omit the curly braces, but it's not recommended!

```
int age = 6;  
  
if ( age < 21 )  
    System.out.println("Too young to drink!");
```

if () Statements

Good programming practice: Use a symbolic constant (in case the legal drinking age changes!)

```
final int DRINKING_AGE = 21;
int age = 6;

if ( age < DRINKING_AGE ) {
    System.out.println("Too young to drink!");
}
```


if()/else Statements

Introducing else():

```
final int DRINKING_AGE = 21;
int age = 6;

if ( age < DRINKING_AGE ) {
    System.out.println("Too young to drink!");
} else {
    System.out.println("Too old for the prom!");
}
```

booleans

- A boolean variable can hold two values:
true or false

booleans

- A `boolean` variable can hold two values:
`true` or `false`
- When we've used `ints` set to 0 or 1 to indicate a binary state, we could have used a `boolean` variable

- A boolean variable can hold two values:
true or false
- When we've used ints set to 0 or 1 to indicate a binary state, we could have used a boolean variable
- Example:

```
int windy = 0;
```

```
if ( windy == 1 ) {  
    System.out.println("Wear a windbreaker.");  
} else {  
    System.out.println("The air is calm.");  
}
```

- A boolean variable can hold two values:
true or false
- When we've used ints set to 0 or 1 to indicate a binary state, we could have used a boolean variable
- Example:

```
boolean isWindy = false;
```

```
if ( isWindy ) {  
    System.out.println("Wear a windbreaker.");  
} else {  
    System.out.println("The air is calm.");  
}
```

Another Example:

```
int myNum = 7;

public boolean isEven(int n) {
    if ( n % 2 == 0 ) {
        return true;
    } else {
        return false;
    }
}

boolean a = isEven(myNum);

if ( a ) {
    System.out.println("myNum is even!");
} else {
    System.out.println("myNum is odd!");
}
```

Slicker Way:

```
int myNum = 7;

public boolean isEven(int n) {
    if ( n % 2 == 0 ) {
        return true;
    } else {
        return false;
    }
}

if ( isEven(myNum) ) {
    System.out.println("myNum is even!");
} else {
    System.out.println("myNum is odd!");
}
```

Relational Operators

- You already know most of them!
- Java evaluates their value to true or false — so you can use them in `if()` and `while()` statements

| Operator | Meaning |
|----------|--------------|
| > | greater than |
| < | less than |
| >= | ≥ |
| <= | ≤ |
| == | equals |
| != | ≠ |

Relational Operators

Predict what happens:

```
int a = 5, b = -2;
boolean areEqual = ( a == b );

if ( areEqual ) {
    System.out.println("a = b");
} else {
    System.out.println("a != b");
}
```

Relational Operators

Predict what happens:

```
int a = 5, b = -2;
boolean areEqual = ( a == b );

if ( areEqual ) {
    System.out.println("a = b");
} else {
    System.out.println("a != b");
}
```

→ *What would happen if you removed the ()s from the "boolean areEqual..." line?*

Complete PS #6, §1 & (most of) §2.