

Java Control: Conditionals

**CPSC 233: Introduction to Computer Science for Computer Science
Majors II
Winter 2022**

Jonathan Hudson, Ph.D.
Instructor
Department of Computer Science
University of Calgary

Thursday, 4 November 2021

Copyright © 2021



Relational/Boolean Operators

Relational/Boolean Difference Python to Java

Java	Python
&&	and
	or
!	not
<,<=,>,>=	<,<=,>,>=

Python may read

```
if ((not sunny and cold ) or raining)
```

Java reads

```
if ((!sunny && cold) || raining)
```

This is c++ inherited syntax (unlike python that uses more human syntax)

Comparing primitives versus objects

Java	Python
<code>x == y, x != y</code> (compare primitives by data)	No primitives in python
<code>x == y, x != y</code> (compares objects by id only)	Same – python <code>__eq__</code> default (check's <code>id(x) == id(y)</code>)
<code>x.equals(y)</code> (compares objects, id or data)	Same – python <code>__eq__</code>

```
Person x = new Person("Alice");
Person y = new Person("Alice");
z = x;
```

```
x == x ;      is true
x == z ;      is true
```

```
x == y;      is false
x.equals(y); is true as the data is the same (if equals() is re-defined as comparing Person names to decide equality)
x.equals(y); is false as different objects (if equals() has not been re-defined to compare data in each Person)
```

Comparing primitives versus objects

Java	Python
<code>x == y, x != y</code> (compare primitives)	No primitives in python
<code>x == y, x != y</code> (compares objects by id only)	Same – python <code>__eq__</code> default (check's <code>id(x) == id(y)</code>)
<code>x.equals(y)</code> (compares objects, id or data)	Same – python <code>__eq__</code>

```
Person x = new Person("Alice")
Person y = new Person("Alice")
z = x;
```

```
x == x;      is true
x == z;      is true
```

```
x == y;      is false
x.equals(y); is true as the data is the same
x.equals(y); is false as different objects
```

Return To
This Later!

*(comparing Person names to decide equality)
(to compare data in each Person)*

Conditionals

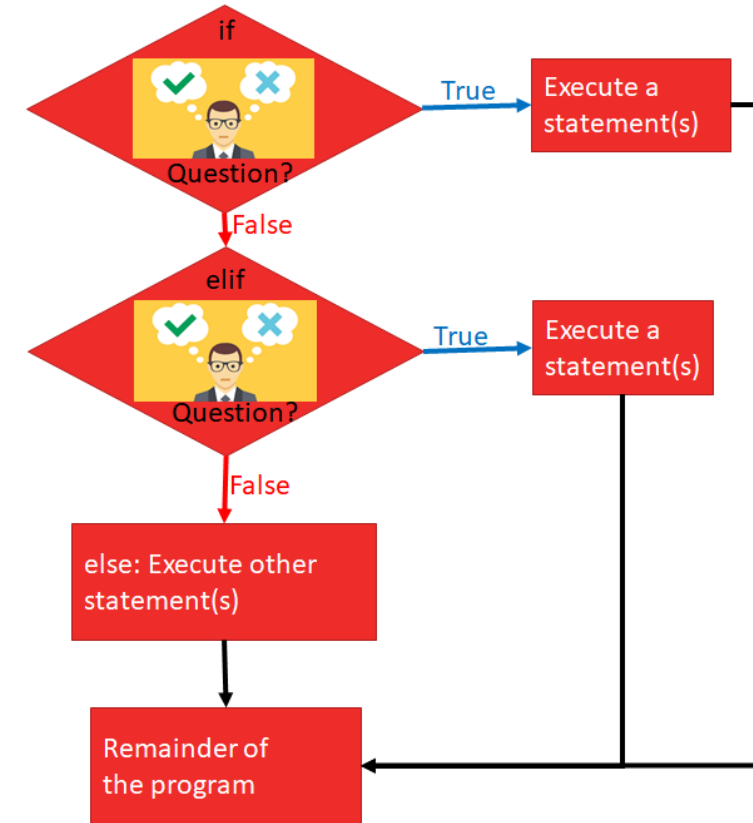
Modifications

// JAVA

```
if (expression){  
    body of 'if'  
} else if (expression){  
    body of 'elif'  
} else {  
    body of 'else'  
}
```

PYTHON

```
if expression:  
    body of 'if'  
elif expression:  
    body of 'elif'  
else:  
    body of 'else'
```



Modifications

// JAVA

```
if (expression){  
    body of 'if'  
} else if (expression){  
    body of 'elif'  
} else {  
    body of 'else'  
}
```

PYTHON

```
if expression:  
    body of 'if'  
elif expression:  
    body of 'elif'  
else:  
    body of 'else'
```

- We are required to use () around the logical expression
- We replace elif by else if
- We don't use : and indentation to indicate inside conditional
 - We use {} instead
- As before
 - 0 or more - **else if**
 - 0 or 1 - **else** at end

Modifications

```
// JAVA
if (expression){
  body of 'if'
} else if (expression){
  body of 'elif'
} else {
  body of 'else'
}
```

- We are required to

Indentation
Doesn't
Matter!!!!!!

- 0 or more - **else if**
- 0 or 1 - **else** at end

Conditionals - Scope

Scope

Scope is different in Java

{ } designate a scope, if we create a variable inside, when it ends, the variable name is deleted

Ex.

- Here printing x,y works inside the if
- After { } ends for the if, only x will print without error

```
public class Test {  
  
    public static void main(String[] args) {  
        int x = 1;  
        if (true) {  
            int y = 2;  
            System.out.println(x);  
            System.out.println(y);  
        }  
        System.out.println(x);  
        System.out.println(y);  
    }  
}
```

java: cannot find symbol
symbol: variable y
location: class Test

Conditionals - Switches

Switch-Statements (NEW!)

// Annoying chained else-if

```
if (colour == "red"){
```

```
  } else if (colour == "green"){
```

```
  } else if (colour == "blue"){
```

```
  } else {
```

```
}
```

// Straight-forward Java Switch

```
switch (colour) {
```

```
  case "red":
```

```
    break;
```

```
  case "green":
```

```
    break;
```

```
  case "blue":
```

```
    break;
```

```
  default:
```

```
    break;
```

```
}
```

Switch-Statements (NEW!)

// Straight-forward Java Switch

```
switch (colour) {  
    case "red":  
        break;  
    case "green":  
        break;  
    case "blue":  
        break;  
    default:  
        break;  
}
```

- Equality with colour is checked for each case
- Example case “red” is executed if colour==“red”
- Anything we want to do for “red” should be put before the break; for “red” case
- Default acts like our else (otherwise)

Question? What happens if break is omitted?

Switch-Statements (NEW!)

// Straight-forward Java Switch

```
switch (colour) {  
    case "red":  
        System.out.println("red");  
    case "green":  
        System.out.println("green");  
    case "blue":  
        System.out.println("blue");  
    default:  
        System.out.println("black");  
        break;  
}
```

Question? What happens if break is omitted?

- We will fall through from above case body into cases below
- Example "red"==colour will print
- Example "blue"==colour will print

Switch-Statements (NEW!)

// Straight-forward Java Switch

```
switch (colour) {  
    case "red":  
        System.out.println("red");  
    case "green":  
        System.out.println("green");  
    case "blue":  
        System.out.println("blue");  
    default:  
        System.out.println("black");  
    break;  
}
```

Question? What happens if break is omitted?

- **We will fall through from above case body into cases below**
- Example "red"==colour will print
 - red
 - green
 - blue
 - black
- Example "blue"==colour will print
 - blue
 - black
- **Easy mistake to make! One reason python omits switch as often switches are involved in dangerous program bugs/exploits.**

Conditionals – In-Line

In-line if statements (Hard to read!)

// Regular simple if assignment

```
if (colour == "red"){
```

```
    x = 2;
```

```
} else {
```

```
    x = 3;
```

```
}
```

//Inline if assignment (not stylistically recommended)

```
x = colour == "red" ? 2 : 3;
```

In-line if statements (Hard to read!)

// Regular simple if assignment

```
if (colour == "red"){
```

```
    x = 2;
```

```
} else {
```

```
    x = 3;
```

```
}
```

//Inline if assignment (not stylistically recommended)

```
x = colour == "red" ? 2 : 3;
```

//read as → set x, IF colour is red to 2, ELSE to 3

Onward to ... Loops.

Jonathan Hudson
jwhudson@ucalgary.ca
<https://pages.cpsc.ucalgary.ca/~jwhudson/>

