

Conditional Statements

If Statement and Boolean Expressions

Produced
by:

Department of Computing and Mathematics



Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

Topics list

- Conditional Statements
- Boolean conditions and Relational Operators
- Logical Operators

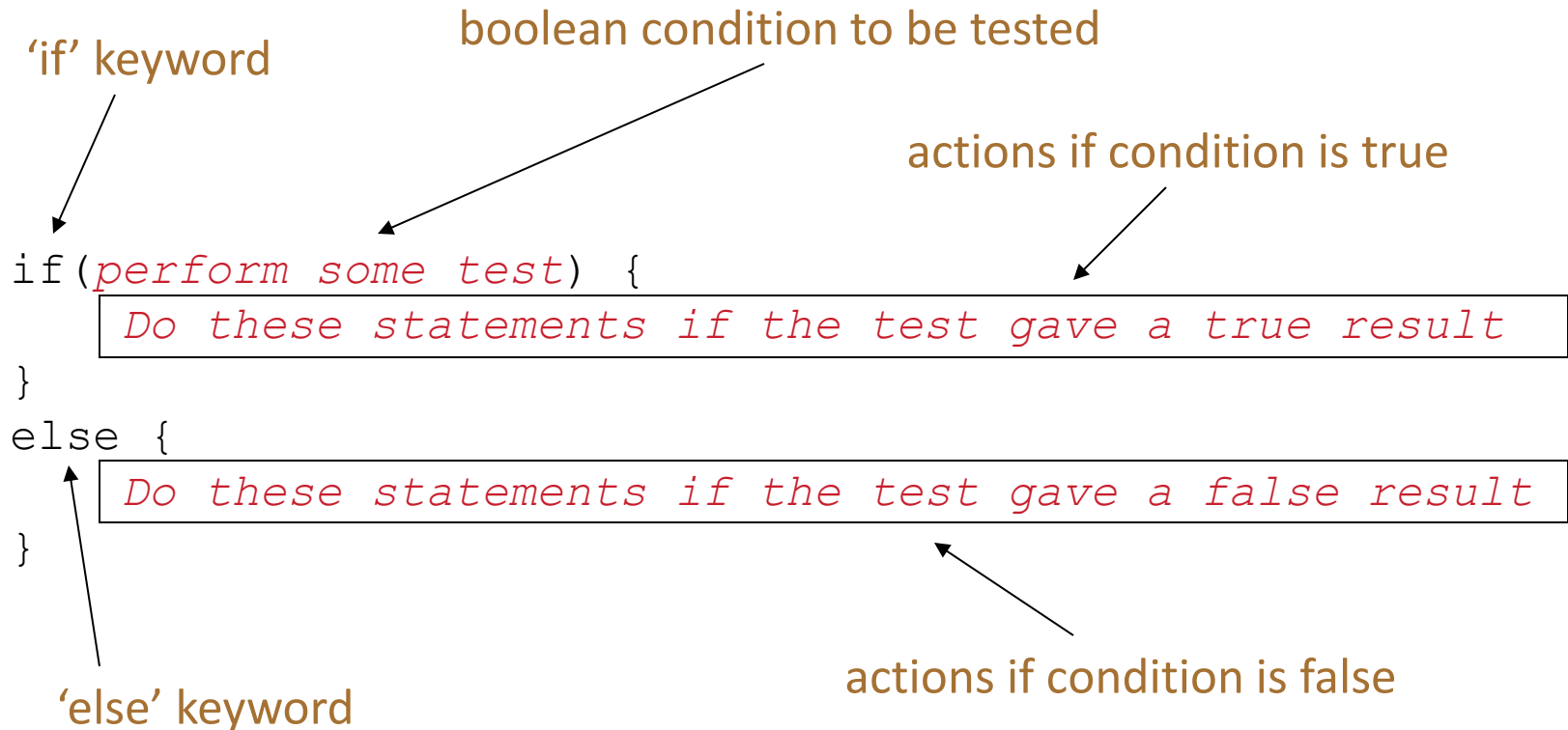
Conditional Statement Syntax (1)

'if' keyword boolean condition to be tested actions if condition is true

```
if (perform some test)  
{  
    Do these statements if the test gave a true result  
}
```

The diagram illustrates the syntax of an if statement. It shows the 'if' keyword, a boolean condition in parentheses, and a block of actions enclosed in curly braces. Arrows point from the labels to the corresponding parts of the code. The condition and the actions are highlighted in red in the original image.

Conditional Statement Syntax (2)



Conditional Statement Syntax (3)

```
if(condition1...perform some test)
```

```
{
```

Do these statements if condition1 gave a true result

```
}
```

```
else if(condition2...perform some test)
```

```
{
```

Do these statements if condition1 gave a false result and condition2 gave a true result

```
}
```

```
else
```

```
{
```

Do these statements if both condition1 and condition2 gave a false result

```
}
```

Topics list

- Conditional Statements
- Boolean conditions and Relational Operators
- Logical Operators

Boolean conditions

- A boolean condition is an expression that evaluates to either true or false e.g.

`mouseX < 50`

- An if statement evaluates a boolean condition and its result will determine which portion of the if statement is executed.

Boolean conditions

```
// Do these statements before.  
  
if (boolean condition)  
{  
    // Perform this clause if the condition  
    //is true.  
}  
  
// Do these statements after.
```


Java Relational Operators

Operator	Use	Returns true if
>	op1 > op2	op1 is greater than op2
>=	op1 >= op2	op1 is greater than or equal to op2
<	op1 < op2	op1 is less than to op2
<=	op1 <= op2	op1 is less than or equal to op2
==	op1 == op2	op1 and op2 are equal
!=	op1 != op2	op1 and op2 are not equal

Some notes on the if statement

- An if statement **IS** a statement; it is only executed once.
- When your if statement only has one statement inside it, you do not need to use the curly braces.
- For example, both of these are the same:

```
if (mouseX < 50)
{
  rect(0, 0, 50, 100);
}
```

```
if (mouseX < 50)
  rect(0, 0, 50, 100);
```

Some notes on the if statement

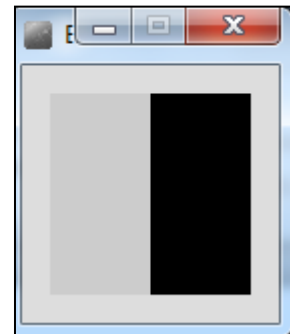
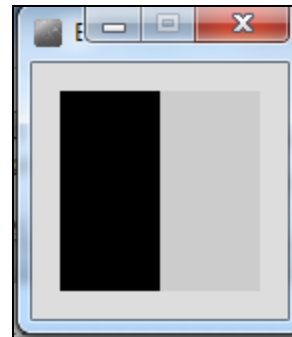
- The semi-colon (;) is a statement terminator.
- One is circled in the code example below:

```
if (mouseX < 50)
{
  rect(0, 0, 50, 100);
}
```

- Your if statement does not need a statement terminator.

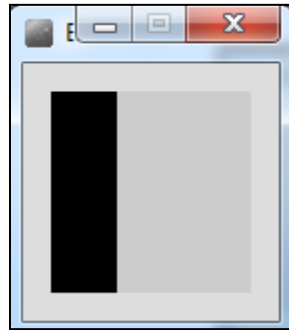
Conditional Example 4.1

```
void setup() {  
  size(100, 100);  
  noStroke();  
  fill(0);  
}  
  
void draw() {  
  background(204);  
  if (mouseX < 50)  
  {  
    rect(0, 0, 50, 100);  
  }  
  else  
  {  
    rect(50, 0, 50, 100);  
  }  
}
```



Conditional Example 4.2

```
void setup() {  
  size(100, 100);  
  noStroke();  
  fill(0);  
}  
  
void draw() {  
  background(204);  
  if (mouseX < 33) {  
    rect(0, 0, 33, 100);  
  }  
  else if (mouseX < 66) {  
    rect(33, 0, 33, 100);  
  }  
  else {  
    rect(66, 0, 33, 100);  
  }  
}
```



Topics list

- Conditional Statements
- Boolean conditions and Relational Operators
- Logical Operators

Logical operators

- Logic operators operate on boolean values.
- They produce a new boolean value as a result.
- The ones that we will use are:

&&	(and)
	(or)
!	(not)

Logical operators

a && b *(and)*

- This evaluates to true if both **a** and **b** are true.
- It is false in all other cases.

a || b *(or)*

- This evaluates to true if either **a** or **b** or both are true, and false if they are both false.

!a *(not)*

- This evaluates to true if **a** is false, and false if **a** is true.

Logical operators - quiz

```
int a = 5;  
int b = 10;  
int c = 7;
```

What is the result of each of these boolean expressions:

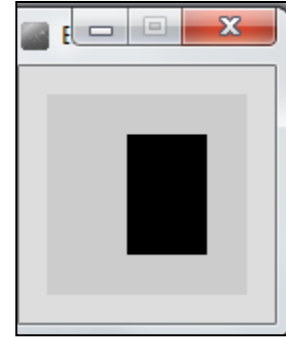
$(a > b) \ \&\& \ (a < c)$

$(a < b) \ || \ (c < a)$

$!(b < a) \ \&\& \ (c > b)$

Conditional Example 4.3

```
void setup() {  
  size(100, 100);  
  noStroke();  
  fill(0);  
}  
  
void draw() {  
  background(204);  
  if ((mouseX > 40) && (mouseX < 80) &&  
      (mouseY > 20) && (mouseY < 80)) {  
    fill(255);  
  } else {  
    fill(0);  
  }  
  rect(40, 20, 40, 60);  
}
```



Conditional Example 4.4

```
void draw() {  
  background(204);  
  if ((mouseX <= 50) && (mouseY <= 50)) {  
    rect(0, 0, 50, 50);      // upper-left  
  } else if ((mouseX <= 50) && (mouseY > 50)) {  
    rect(0, 50, 50, 50);     // lower-left  
  } else if ((mouseX > 50) && (mouseY <= 50)) {  
    rect(50, 0, 50, 50);     // upper-right  
  } else {  
    rect(50, 50, 50, 50);    // lower-right  
  }  
}
```

```
void setup() {  
  size(100, 100);  
  noStroke();  
  fill(0);  
}
```

Questions?



References

- Reas, C. & Fry, B. (2014) Processing – A Programming Handbook for Visual Designers and Artists, 2nd Edition, MIT Press, London.



Except where otherwise noted, this content is licensed under a Creative Commons Attribution-NonCommercial 3.0 License.

For more information, please see <http://creativecommons.org/licenses/by-nc/3.0/>



Waterford Institute of Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

Department of Computing and Mathematics
<http://www.wit.ie/>