

Data Types and Operators

Variables, Data Types & Arithmetic Operators

Produced
by:

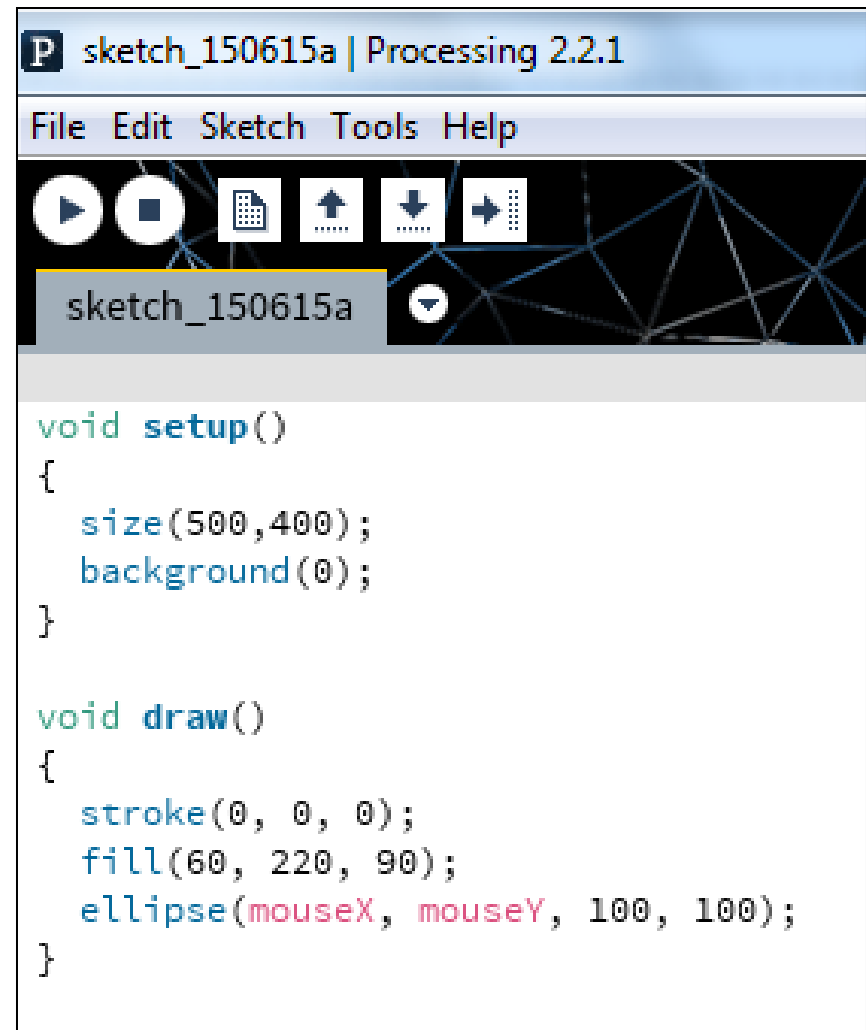
Department of Computing and Mathematics



Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

Variables

- You used **mouseX** and **mouseY** last week.
- These are “system variables”.



```
void setup()
{
  size(500,400);
  background(0);
}

void draw()
{
  stroke(0, 0, 0);
  fill(60, 220, 90);
  ellipse(mouseX, mouseY, 100, 100);
}
```

Variables

- We can create our own variables to store data.
- The data stored in the variable can change over time.
- Variables have a name and a data type.

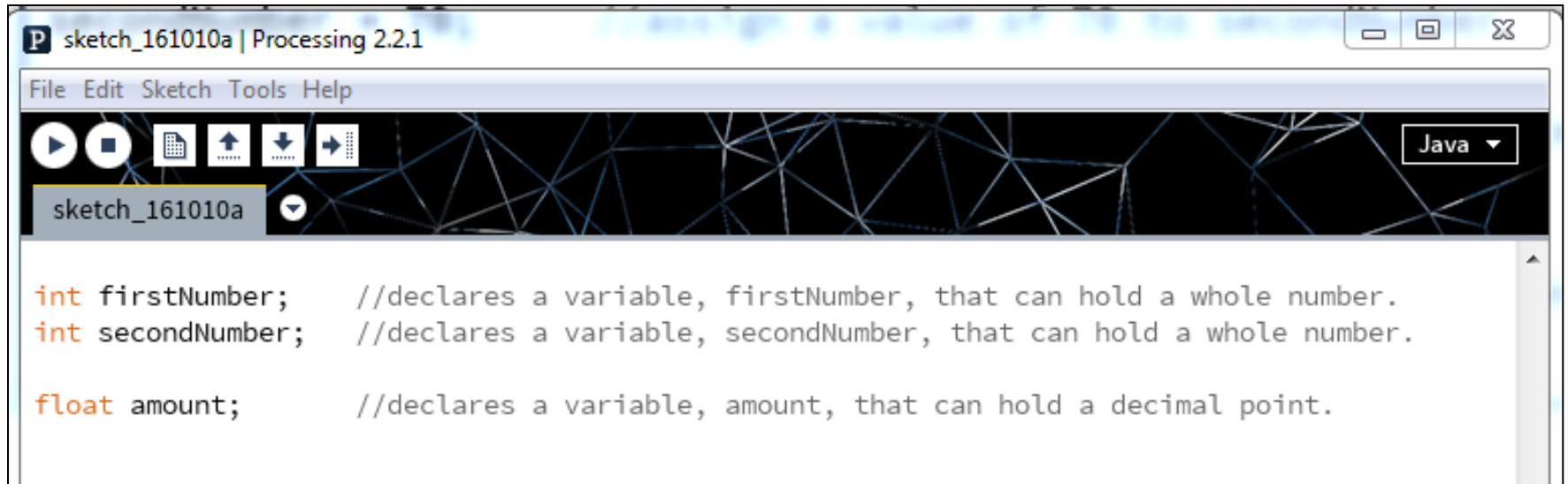
Variable names

- Are case-sensitive.
- Can be any length you choose.
- Cannot contain spaces.
- Examples:
 - firstNumber
 - xCoordinate

Declaring variables (type)

- When creating a variable, we need to tell Java what kind of values will be stored in it.
- We are going to learn how to create **int** and **float** variables.
 - **int** is a whole number e.g. 67
 - **float** is a decimal point number e.g. 67.8987

Declaring variables (type)

A screenshot of the Processing 2.2.1 IDE window. The title bar shows 'P sketch_161010a | Processing 2.2.1'. The menu bar includes 'File', 'Edit', 'Sketch', 'Tools', and 'Help'. Below the menu bar is a toolbar with icons for running, stopping, opening, saving, and other functions. A dropdown menu shows 'Java'. The main text area contains the following code:

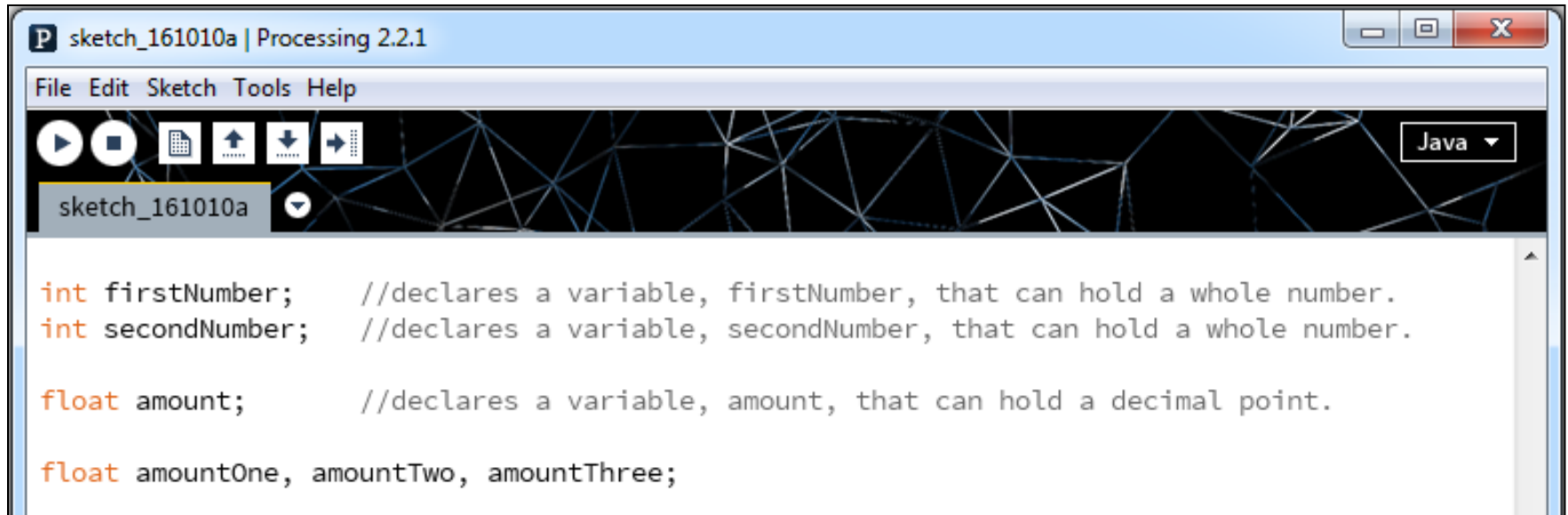
```
int firstNumber;    //declares a variable, firstNumber, that can hold a whole number.
int secondNumber;   //declares a variable, secondNumber, that can hold a whole number.

float amount;       //declares a variable, amount, that can hold a decimal point.
```

The variable types 'int' and 'float' are highlighted in red in the original image.

Note: the type is highlighted red when it is typed into Processing.

Declaring variables (type)

A screenshot of the Processing 2.2.1 IDE window. The title bar reads "P sketch_161010a | Processing 2.2.1". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". Below the menu bar is a toolbar with icons for running, stopping, opening, saving, and other functions. A dropdown menu shows "Java". The main text area contains the following code:

```
int firstNumber;    //declares a variable, firstNumber, that can hold a whole number.
int secondNumber;   //declares a variable, secondNumber, that can hold a whole number.

float amount;       //declares a variable, amount, that can hold a decimal point.

float amountOne, amountTwo, amountThree;
```

Note: several variables of the same type
can be declared on one line.

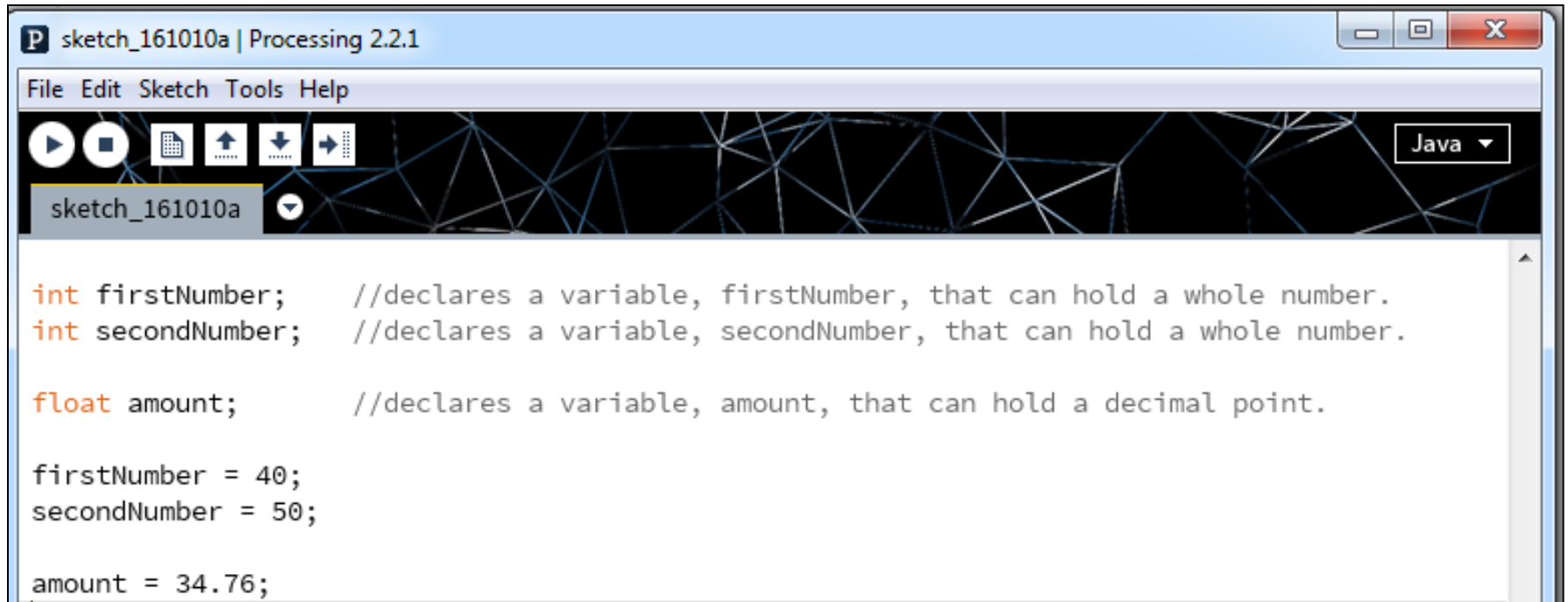
Storing values in Variables

- Values are stored in variables using the = sign e.g.:

```
firstNumber = 40;
```

- The value of 40 is stored in the firstNumber variable.
- A variable stores a single value, so any previous value is lost.

Storing values in Variables



The screenshot shows the Processing 2.2.1 IDE interface. The title bar reads "p sketch_161010a | Processing 2.2.1". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". The toolbar contains icons for running, stopping, opening, saving, and other functions. The sketch name "sketch_161010a" is displayed in the top left of the editor area. The language is set to "Java". The code in the editor is as follows:

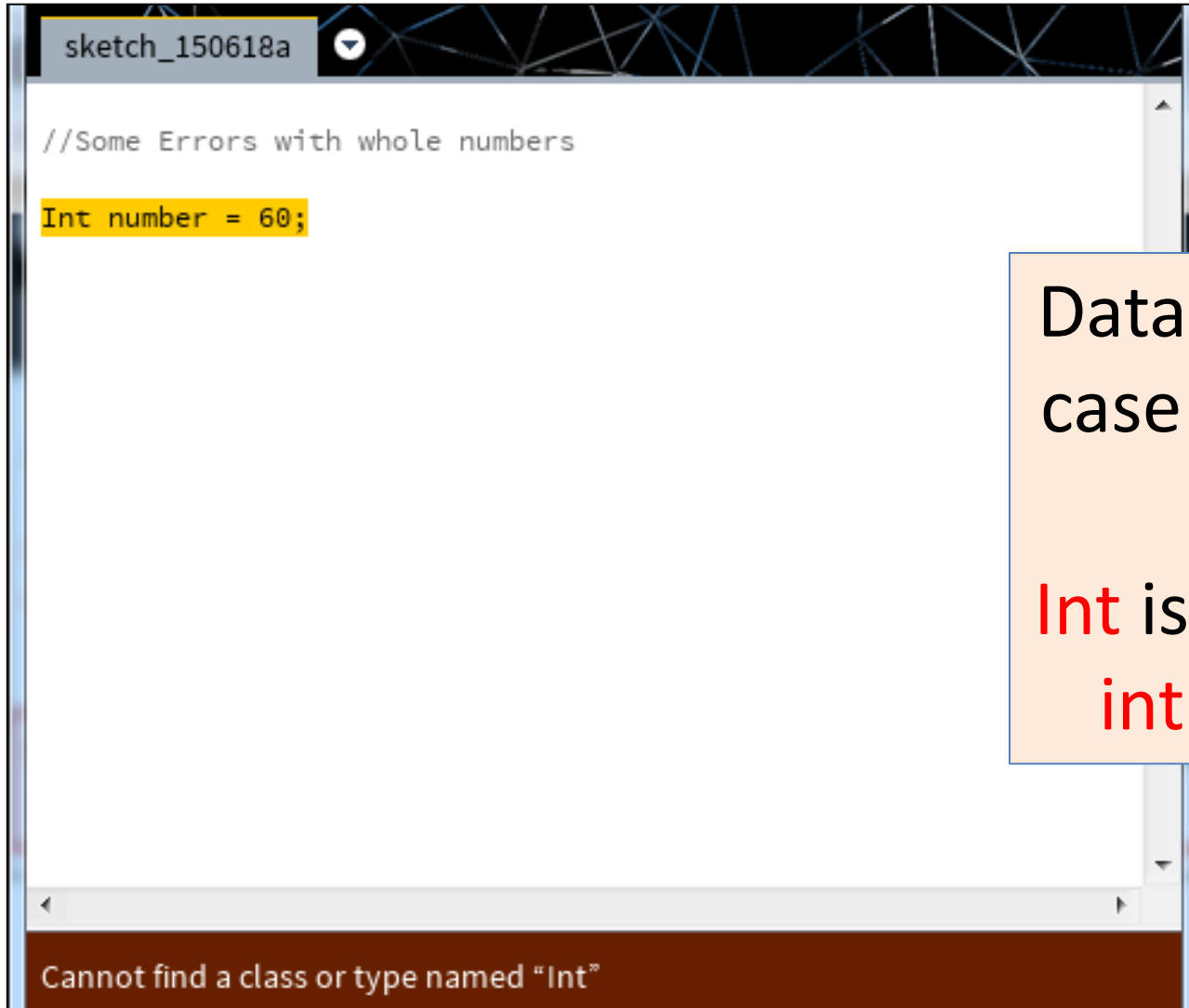
```
int firstNumber;    //declares a variable, firstNumber, that can hold a whole number.
int secondNumber;   //declares a variable, secondNumber, that can hold a whole number.

float amount;       //declares a variable, amount, that can hold a decimal point.

firstNumber = 40;
secondNumber = 50;

amount = 34.76;
```

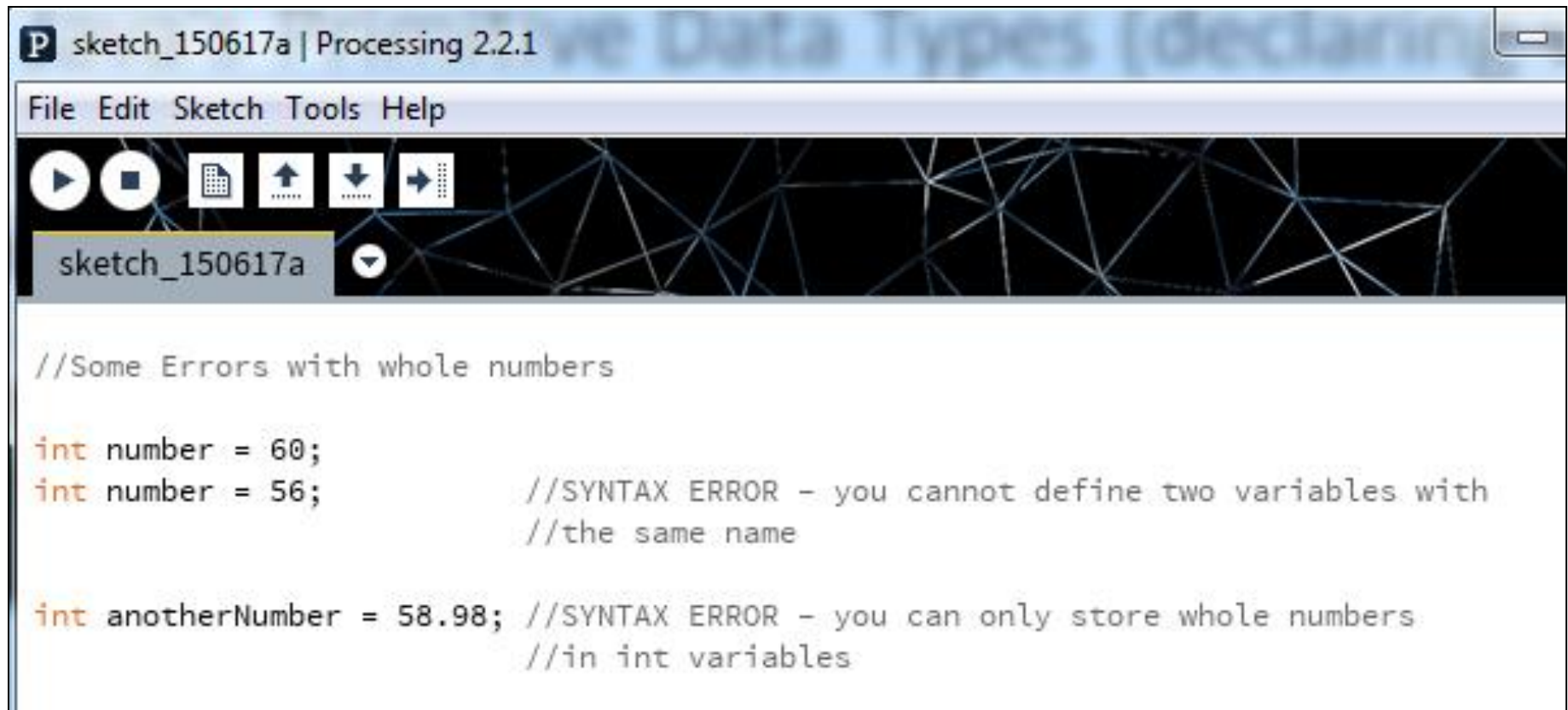
Declaring variables – some errors



Data types are
case sensitive.

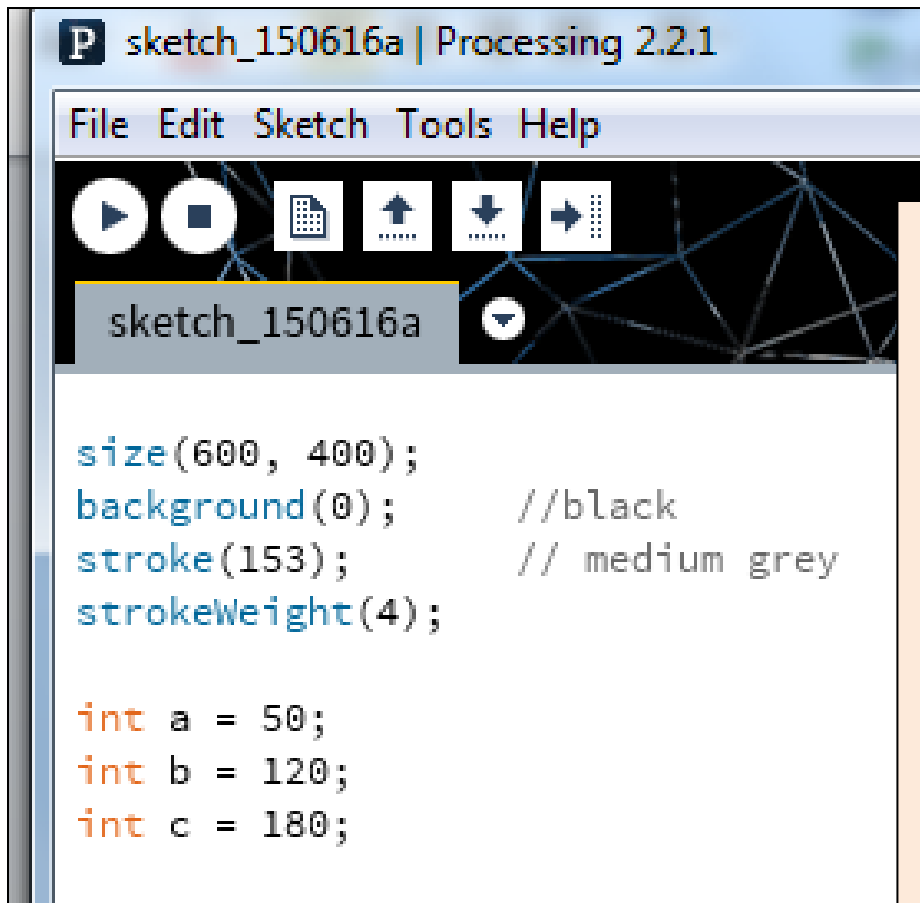
Int is not valid.
int is valid.

Declaring variables – some errors

A screenshot of the Processing 2.2.1 IDE window. The title bar shows 'P sketch_150617a | Processing 2.2.1'. The menu bar includes 'File', 'Edit', 'Sketch', 'Tools', and 'Help'. Below the menu bar is a toolbar with icons for running, stopping, opening, saving, and other functions. The main text area contains the following code:

```
//Some Errors with whole numbers  
  
int number = 60;  
int number = 56;           //SYNTAX ERROR - you cannot define two variables with  
                           //the same name  
  
int anotherNumber = 58.98; //SYNTAX ERROR - you can only store whole numbers  
                           //in int variables
```

int data type: Example 4.1



The screenshot shows the Processing IDE interface. The title bar reads 'sketch_150616a | Processing 2.2.1'. The menu bar includes 'File', 'Edit', 'Sketch', 'Tools', and 'Help'. Below the menu bar is a toolbar with icons for running, stopping, opening, saving, and other functions. The main text area contains the following code:

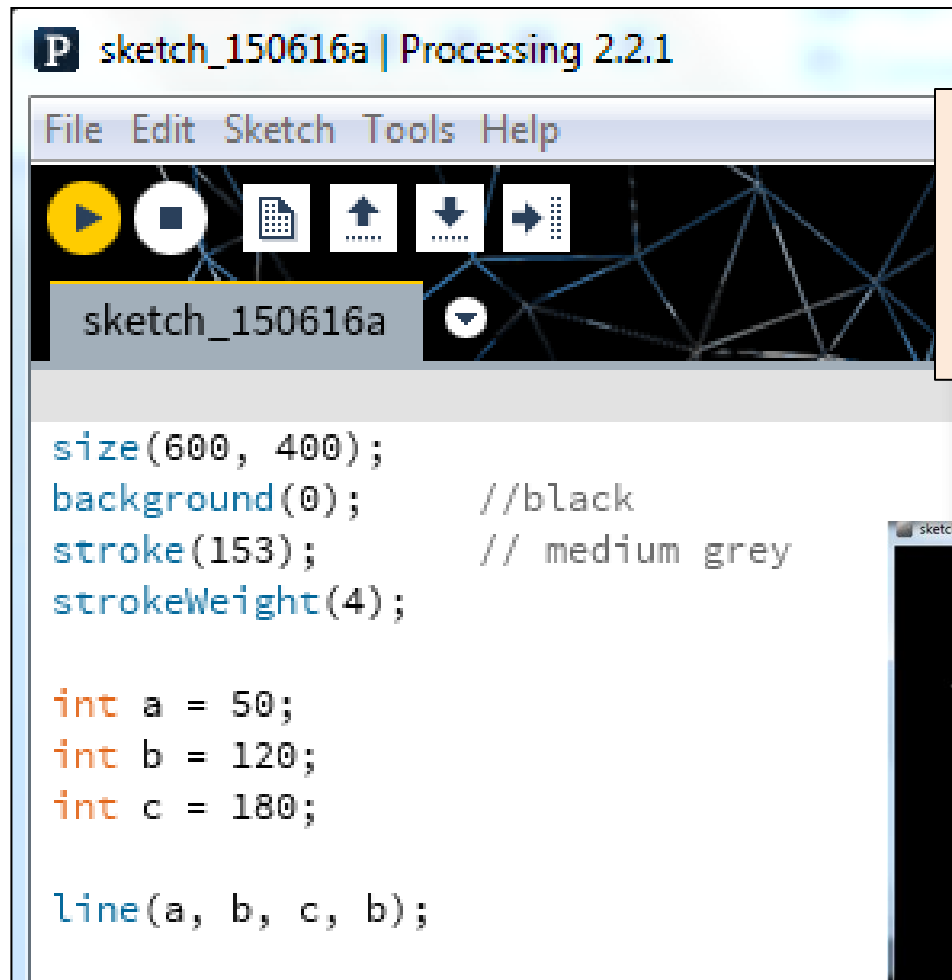
```
size(600, 400);
background(0);    //black
stroke(153);      // medium grey
strokeWeight(4);

int a = 50;
int b = 120;
int c = 180;
```

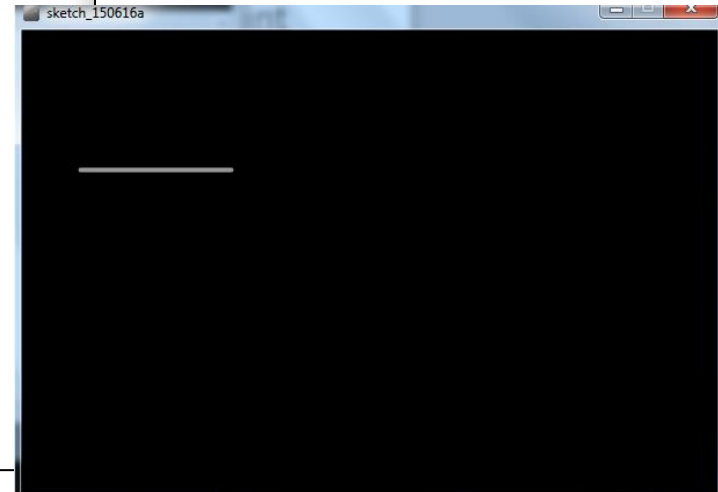
In this example, we have:

- defined three variables (a, b and c)
- that can hold whole numbers (int).
- and are set with a starting value.

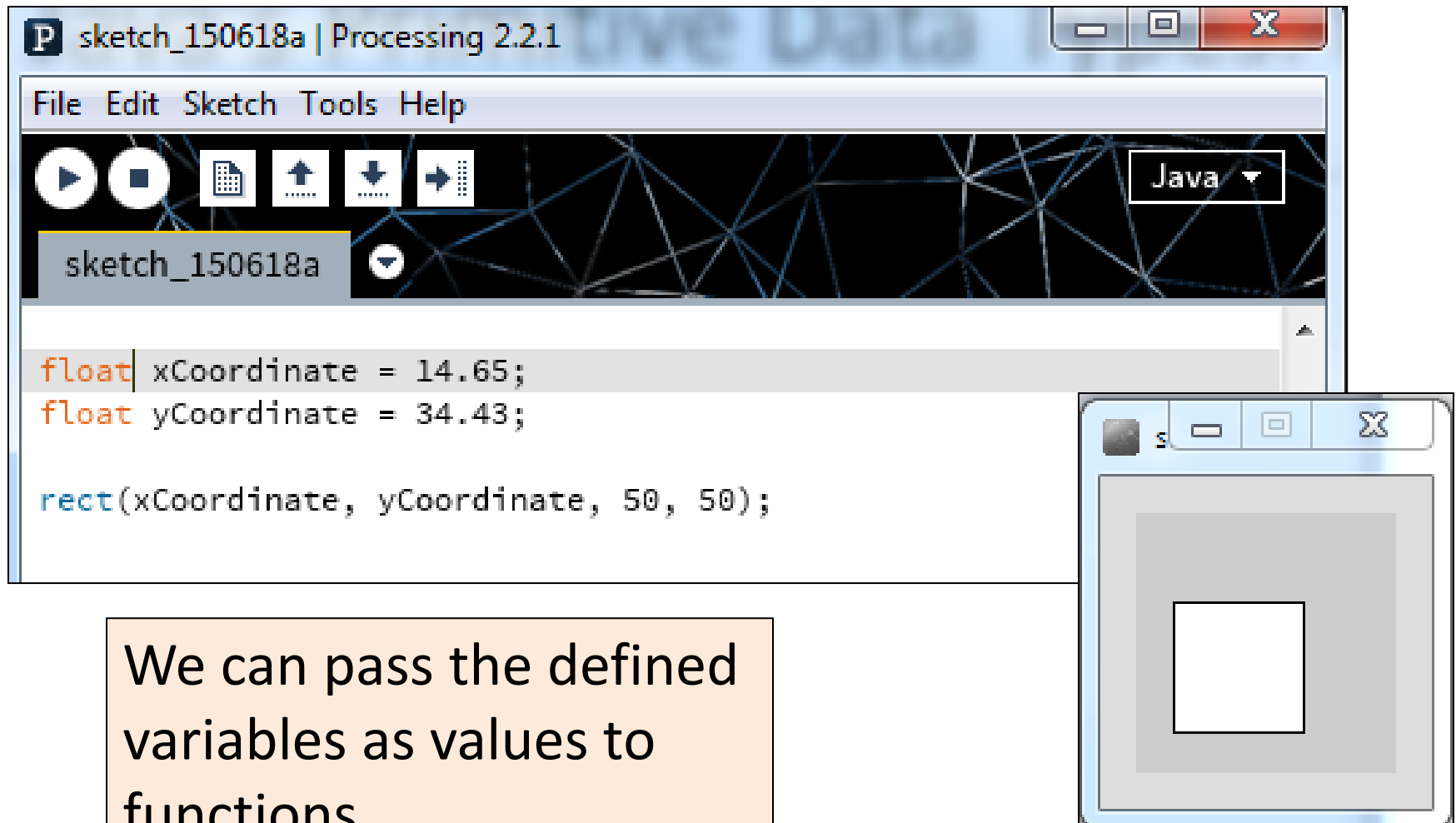
int data type: Example 4.2



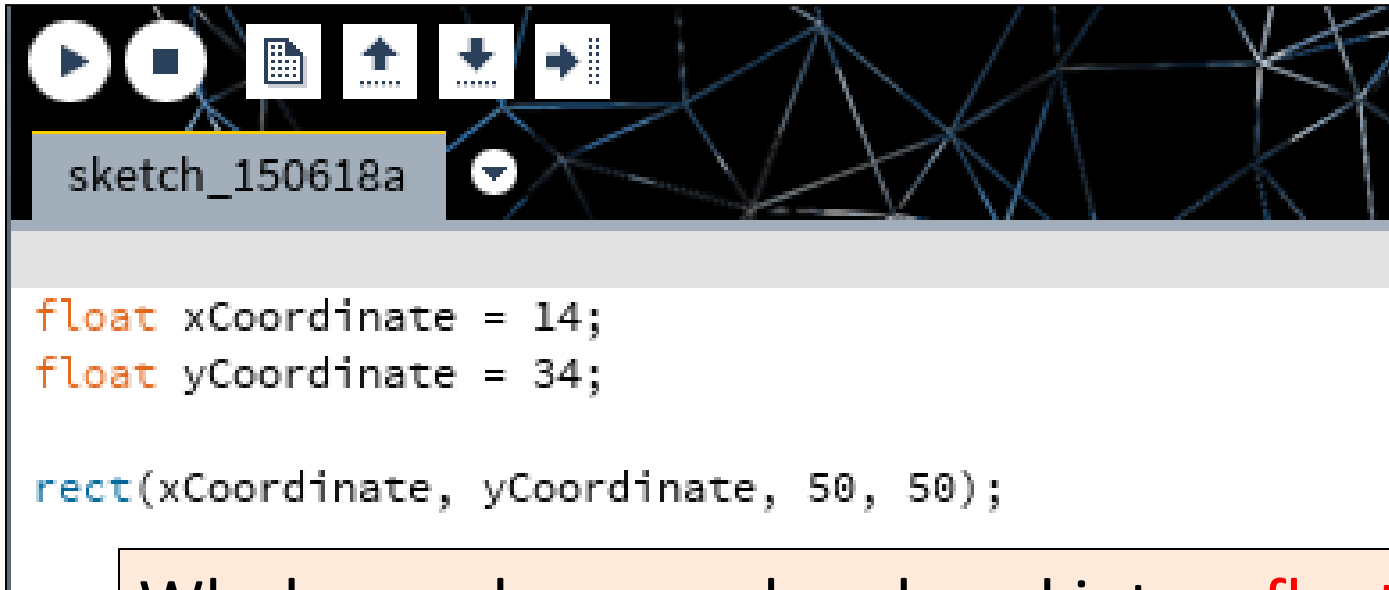
We can pass the defined variables as values to functions.



float data type: Example 4.3



float data type: Example 4.4

A screenshot of a code editor window. The title bar shows a play button, a square icon, and a document icon. Below the title bar, the sketch name 'sketch_150618a' is displayed. The code area contains three lines: 'float xCoordinate = 14;', 'float yCoordinate = 34;', and 'rect(xCoordinate, yCoordinate, 50, 50);'. The 'float' keyword is highlighted in orange, and 'rect' is highlighted in blue.

```
float xCoordinate = 14;
float yCoordinate = 34;

rect(xCoordinate, yCoordinate, 50, 50);
```

Whole numbers can be placed into a **float** variable.

Q: Why?

A: There is no loss of precision. We are not losing any data.

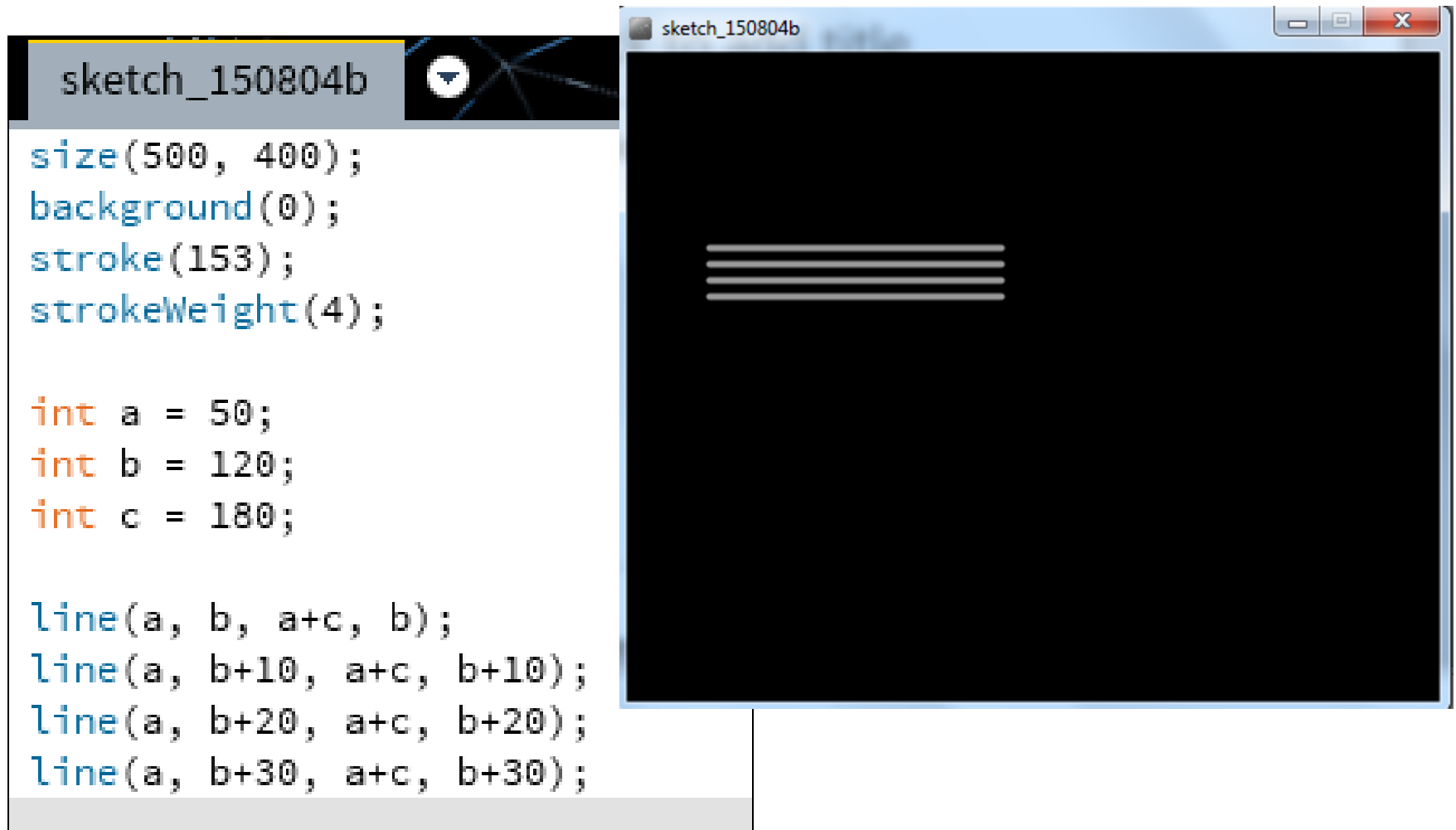
Performing calculations on our variables

and the Order of Evaluation

Arithmetic Operators

Arithmetic Operator	Explanation	Examples
+	Addition	$6 + 2$
		<code>amountOwed + 10</code>
-	Subtraction	$6 - 2$
		<code>amountOwed - 10</code>
*	Multiplication	$6 * 2$
		<code>amountOwed * 10</code>
/	Division	$6 / 2$
		<code>amountOwed / 10</code>

Arithmetic operators: Example 4.5



Arithmetic operators: Example 4.6

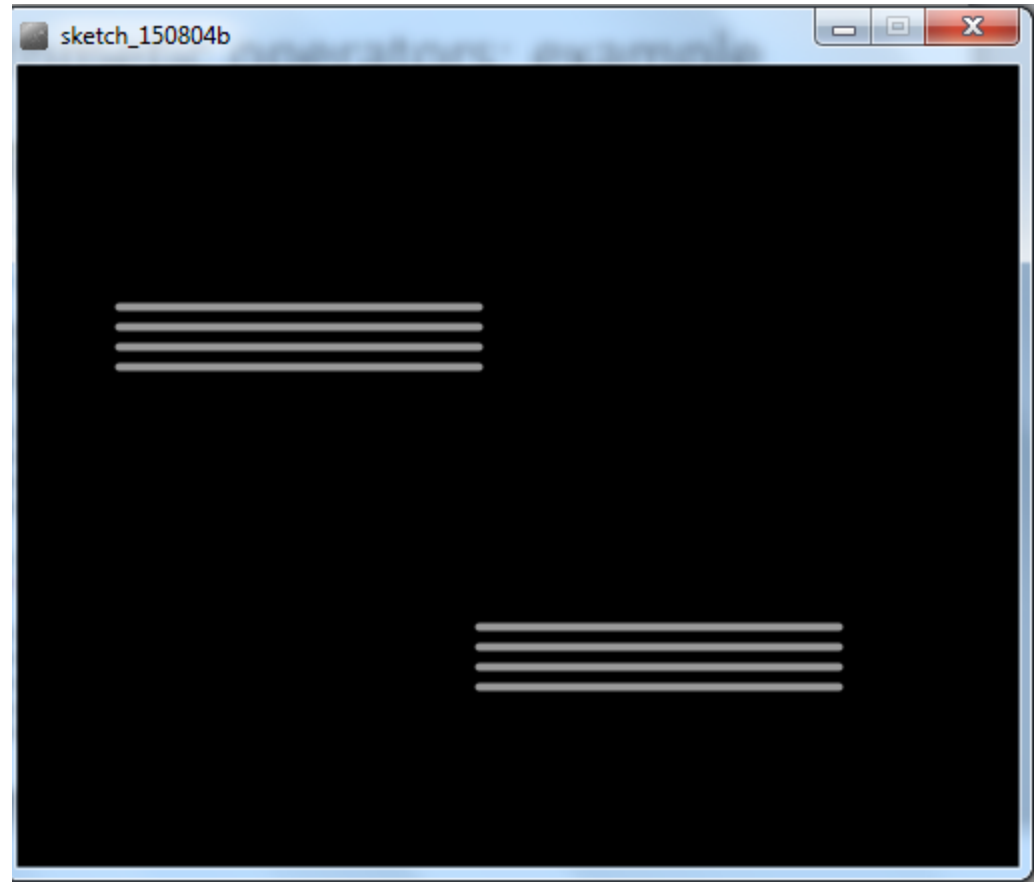
```
sketch_150804b
size(500, 400);
background(0);
stroke(153);
strokeWeight(4);

int a = 50;
int b = 120;
int c = 180;

line(a, b, a+c, b);
line(a, b+10, a+c, b+10);
line(a, b+20, a+c, b+20);
line(a, b+30, a+c, b+30);

a = a + c;
b = height-b;

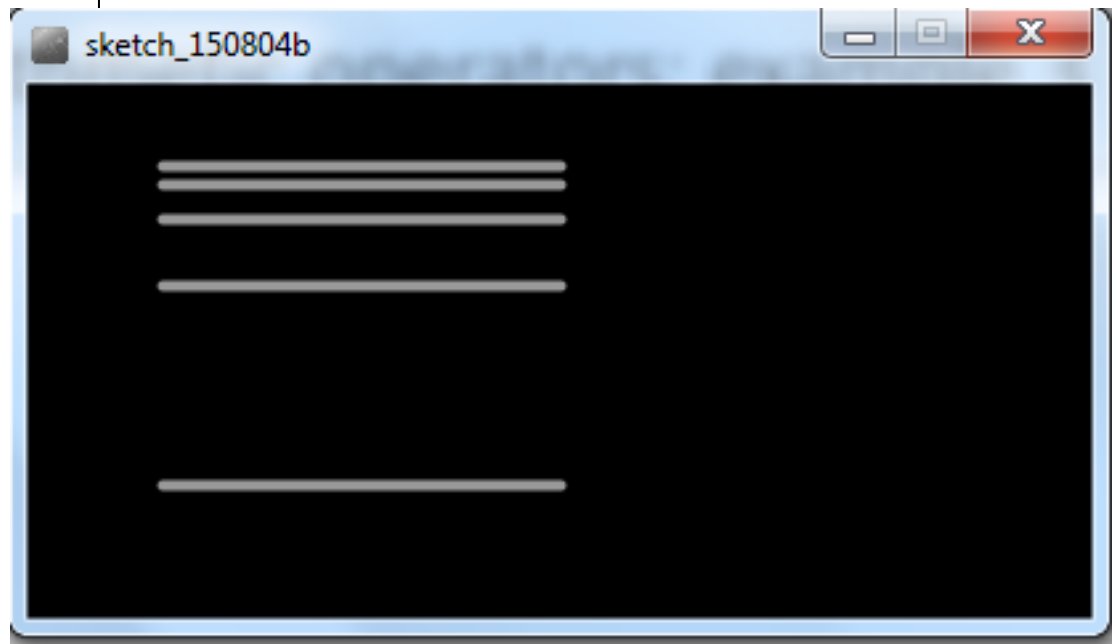
line(a, b, a+c, b);
line(a, b+10, a+c, b+10);
line(a, b+20, a+c, b+20);
line(a, b+30, a+c, b+30);
```



Based on the Processing Example: Basics → Data → Variables

Arithmetic operators: Example 4.7

```
sketch_150804b  
  
size(400, 200);  
background(0);  
stroke(153);  
strokeWeight(4);  
  
int a = 50;  
int b = 1500;  
int c = 4;  
  
line(a, b/10, a*c, b/10);  
line(a, b/20, a*c, b/20);  
line(a, b/30, a*c, b/30);  
line(a, b/40, a*c, b/40);  
line(a, b/50, a*c, b/50);
```



Arithmetic Operators

- These examples are straightforward uses of the arithmetic operators.
- However, we typically want to do more complex calculations involving many arithmetic operators.
- To do this, we need to understand the **Order of Evaluation**.

Order of Evaluation

- Brackets ()
- Multiplication (*)
- Division (/)
- Addition (+)
- Subtraction (-)

BoMDAS

Beware My Dear Aunt Sally

Order of Evaluation - Quiz

What are the results of these calculations?

Q1: $3+6*5-2$

Q2: $3+6*(5-2)$

Q3: $(3+6)*5-2$

Questions?





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/>