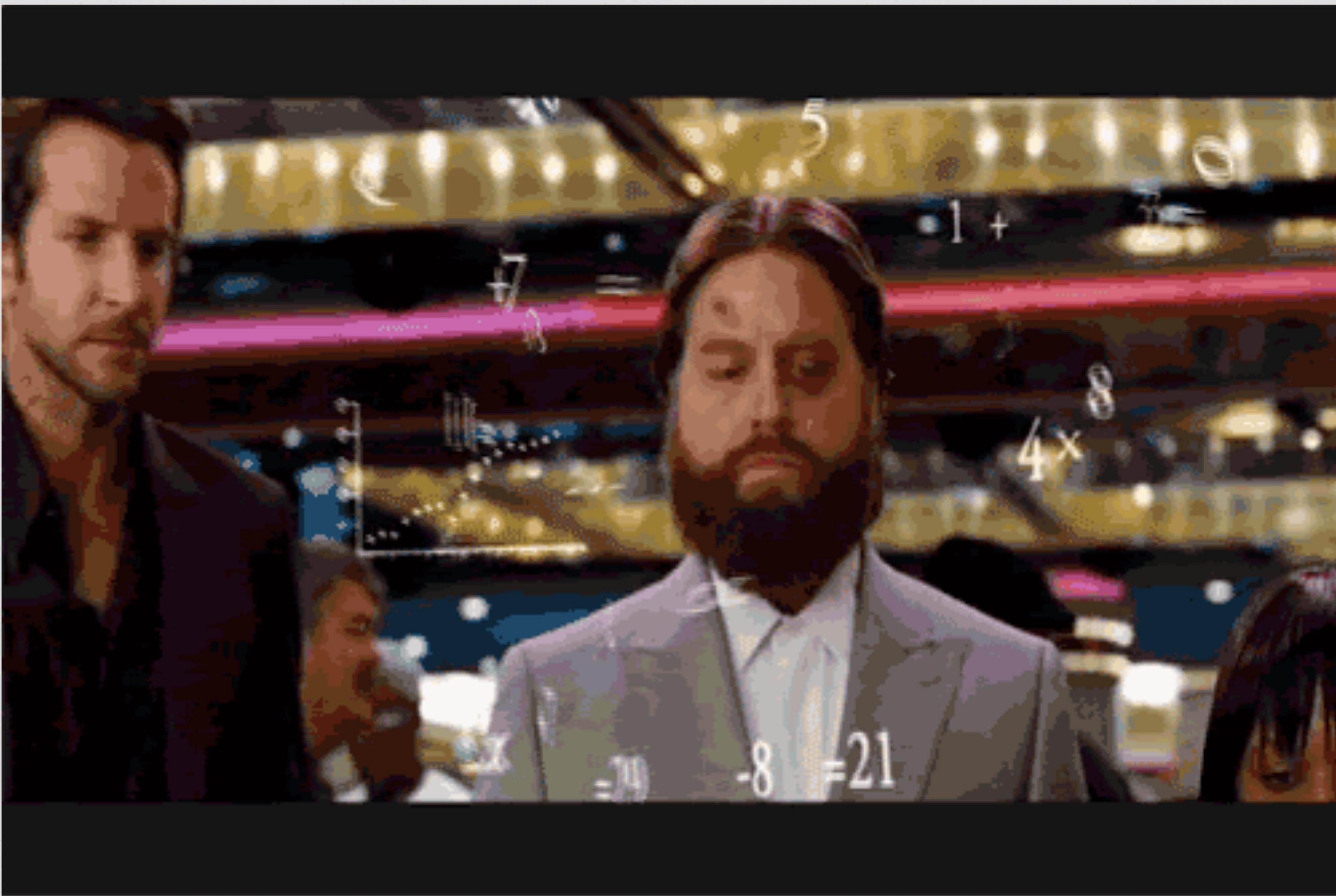


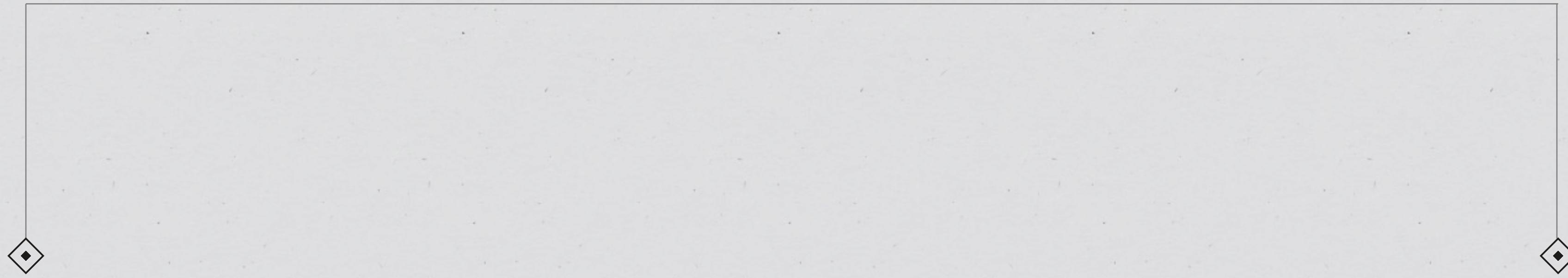
---

# CODE: CLASS 3



Datatypes, Variables, and Basic Math





# What is a Datatype?

## Why Do We Need it?

- Datatype: the classification of a piece of information
- Reason: As in real life, the computer has to know what something is before it can store it, or perform actions on it, or whatever.

# Datatypes

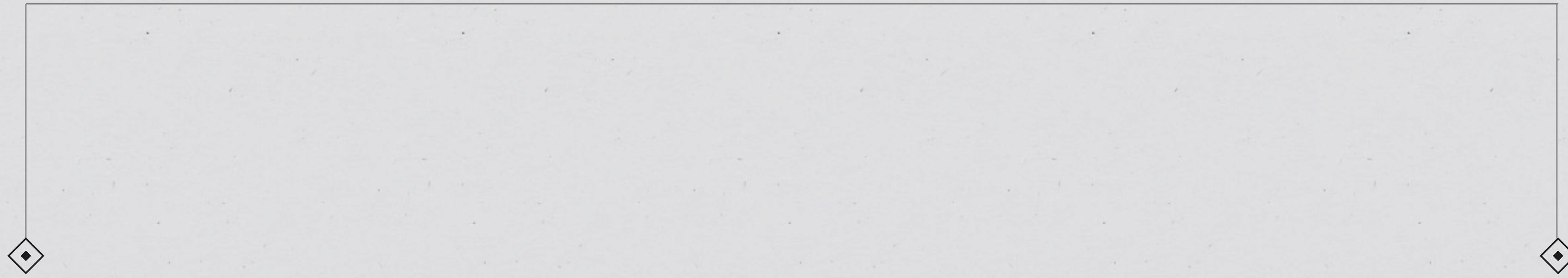
Name	Use	Code	Real life
<i>Integer</i>	Whole numbers	int	I have 5 dogs.
<i>Float</i>	Numbers with a decimal	float	I'm 24.5 years old.
<i>String</i>	Words or letters	String	“Get off my lawn!”
<i>Character</i>	A single letter	char	‘A’
<i>Boolean</i>	True or false	boolean	It's true that I want coffee.

# Datatypes

**How would you classify these things?**

*Choose: integer, float, string, boolean, char*

- \* The number of kids you have
- \* Your name
- \* Your middle initial
- \* Your apartment number
- \* Whether you ate breakfast or not
- \* The street you live on
- \* The temperature outside
- \* Whether the lights are on or not



# What is a Variable?

## Why Do We Need it?

- Variable: A container that holds a piece of information (integer, float, string, boolean, etc.) that can change its value.
- Reason: If you can't change the value of something, it stays static. Forever. Not very interactive, huh?

# Variables

## Datatype

What kind of thing are we dealing with here? Can we add it or subtract it? Is it text? Or is it a true or false condition?

## Name

What are we calling this thing?  
We need a way to refer to it within our code.

## Value

What number, text, or condition is this variable representing?

## Syntax:

datatype name = value;

- Requirements for computer to hold anything in its memory:
  - What kind of thing are we dealing with here?
  - What are we going to call it?
  - What is its value?

# Variables

How would you create a variable that represents this condition?

The number of years Jane has left in her program is 1.

# Variables

The number of years Jane has left in her program is 1.

Requirements	Definition	Here
<b>Datatype</b>	The category of data	A whole number (integer, or <i>int</i> )
<b>Name</b>	A name for the thing we're referring to	Something relating to the value (like <i>years</i> )
<b>Value</b>	What amount, condition, or text the variable actually contains	The years left ( <i>l</i> )

Syntax:

`datatype name = value;`

# Variables

The number of years Jane has left in her program is 1.

## Real Life:

The **number** of **years** Jane has  
left in her program is **1**.

*datatype name = value;*



## Code:

**int years = 1;**

# Variable Notes

- \* Name your variables something that makes sense
  - \* Good: playerHealth
  - \* Bad: herpDerp
- \* Use underscores or alternating caps to make things readable
  - \* Good: userAge, player\_speed
  - \* Bad: timeelapsesincethegamestarted

**what the fuck kind of variable name is “data”  
you should be incarcerated**



[theprofoundprogrammer.tumblr](http://theprofoundprogrammer.tumblr.com)

# Changing values

Use a single-equals (=) to make the thing on the *left* equal the thing on the *right*.

*name* initially equals “Jane”  
We change *name*’s value to “Rebecca”  
We print the value to see if it worked...

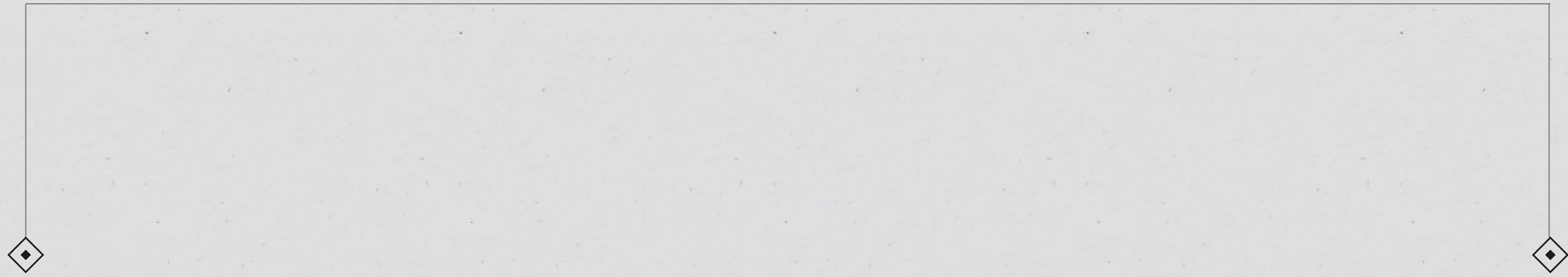


The image shows the Arduino IDE interface. In the top menu bar, there is a "sketch\_jul12a" tab and a "STANDARD" button. The code area contains the following Java-like pseudocode:

```
String name = "Jane";
name = "Rebecca";
println(name);
```

The bottom half of the screen shows the serial monitor window. It displays the word "Rebecca" followed by the number "4".

... and it did!



# What is Concatenation?

# Why Do We Need It?

Monday, July 30, 12

## CONCATENATION:

- Concatenation: Basically, gluing two pieces of info end-to-end.
- Reason: It lets you label the things you print out, so you don't have an unintelligible string of letters and numbers.

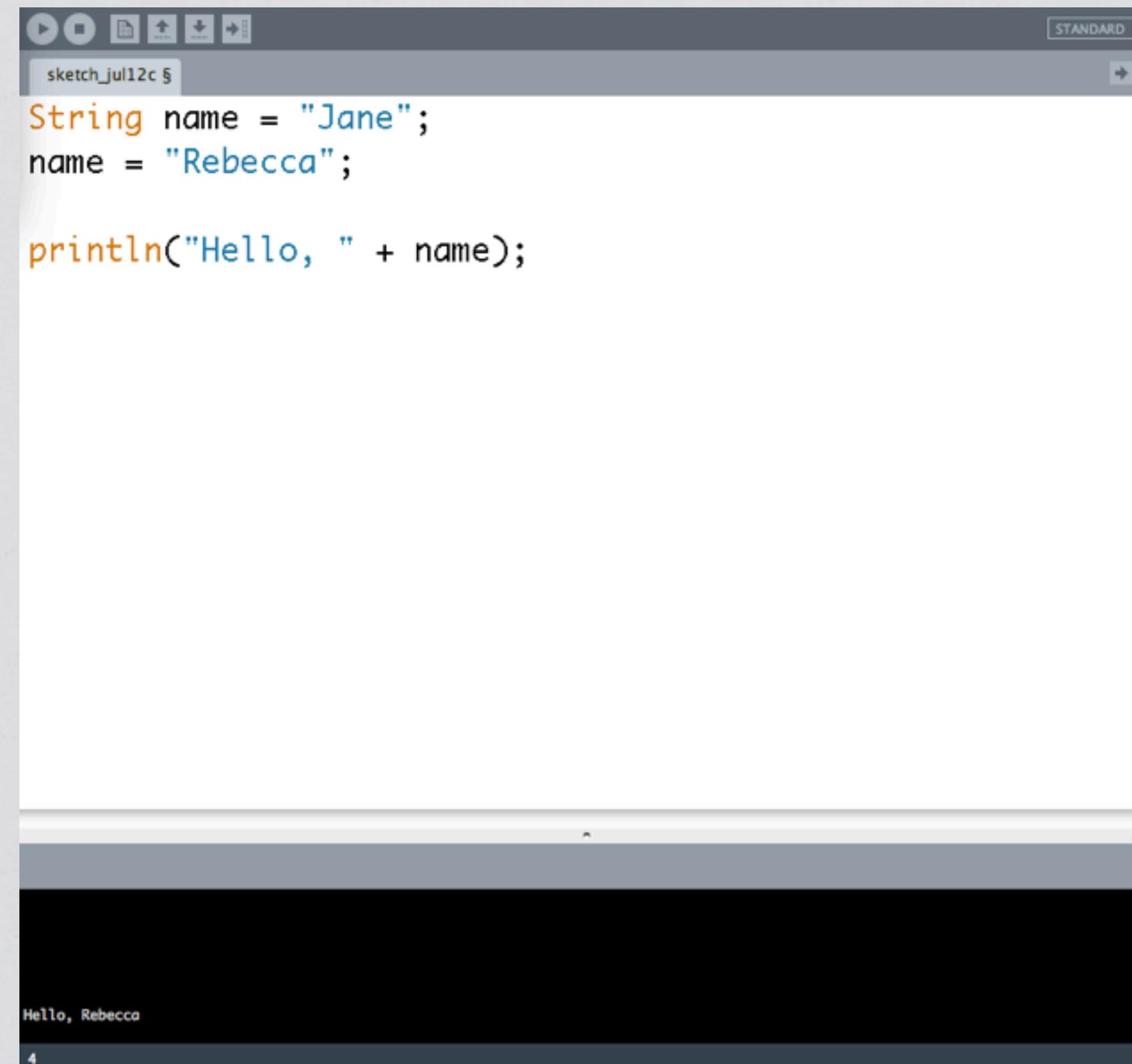
# Concatenation

*name* initially equals “Jane” →

We change *name*’s value to “Rebecca” →

We print the value to see if it worked, but →  
glue “Hello, ” in front of it...

... and it did! →



The screenshot shows the Arduino IDE interface. The top bar displays "sketch\_jul12c §" and "STANDARD". The code area contains the following Java-like pseudocode:

```
String name = "Jane";
name = "Rebecca";

println("Hello, " + name);
```

The bottom window shows the serial monitor output:

```
Hello, Rebecca
```

# Math

(We need some way to manipulate numbers, right?)

Operator	Code
Addition	+
Subtraction	-
Division	/
Multiplication	*

*We'll talk about another operator, modulo (%), soon.*

# Datatypes

❖ Try using `println` and the math symbols to print the result of the following equations.

\*Five plus 3

\*Five minus 3

\*Five times 3

\*Five divided by three

\*Five-point-zero divided  
by three-point-zero



## Notice something weird?

```
sketch_jul12d §
```

```
println(5+3);  
println(5-3);  
println(5*3);  
println(5/3);  
println(5.0/3.0);
```

The output window displays the following results:

Calculation	Result
$5 + 3$	8
$5 - 3$	2
$5 * 3$	15
$5 / 3$	1
$5.0 / 3.0$	1.6666666

A red dashed circle highlights the result of  $5.0 / 3.0$ , which is 1.6666666.

Monday, July 30, 12

To a computer,  $5/3$  is not the same as  $5.0/3.0$ !

It goes back to **datatypes**. The computer sees whole numbers, so it assumes everything is an integer, and spits out a whole number--even when that's not really correct.

# Putting It All Together

Try defining an integer variable called `xPos`, and make it equal 5.

Define another integer variable called `yPos`, and make it equal 7.

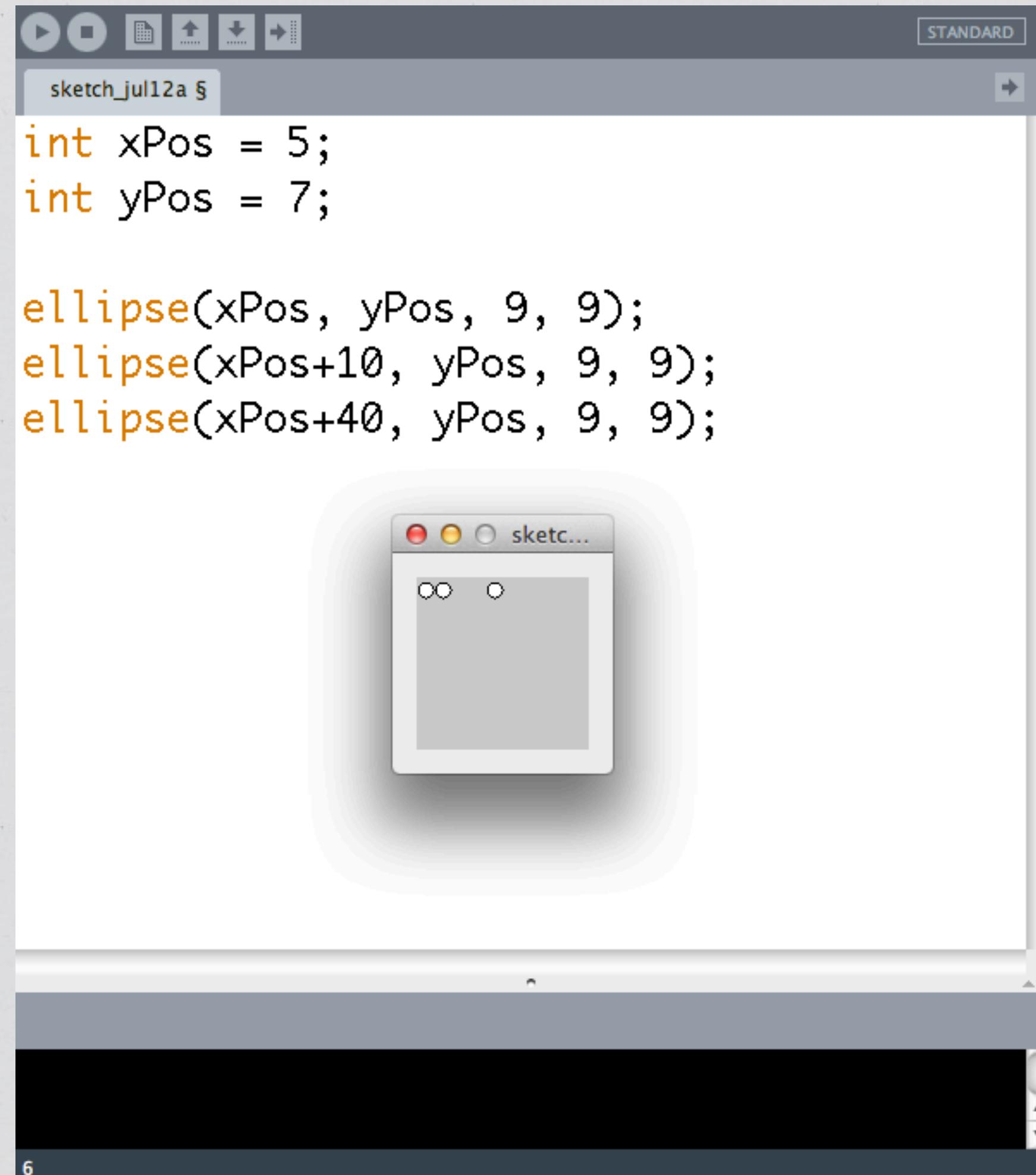
Draw a circle at `xPos` and `yPos`.

Draw another circle at `xPos + 10` and `yPos`.

Draw another circle at `xPos + 40` and `yPos`.

*(They should have a width and height of 9.)*

# Putting it All together



The screenshot shows the Processing IDE interface. At the top, there's a toolbar with various icons. Below the toolbar, the title bar displays "sketch\_jul12a §" and "STANDARD". The main area contains the following code:

```
int xPos = 5;
int yPos = 7;

ellipse(xPos, yPos, 9, 9);
ellipse(xPos+10, yPos, 9, 9);
ellipse(xPos+40, yPos, 9, 9);
```

Below the code is a preview window titled "sketch...". Inside the preview window, there is a gray square containing two small white circles, one near each corner. The preview window has a red, yellow, and green close button at the top left. The entire Processing window is centered on a light gray background.



**I'm going to need a lot of  
booze to handle this shit**