

SCHULICH IGNITE 2019

PROGRAM OVERVIEW



- Six interactive and fun sessions
- Exercise and project oriented approach
- Creativity is highly valued
- Workshops are meant to give students a hands on experience with programming

CONTENT OVERVIEW

Lecture 1: Variables, Mouse Movement, **setup()** and **draw()**

Lecture 2: Operators, Moving objects

Lecture 3: If-Statements, Logical Operators

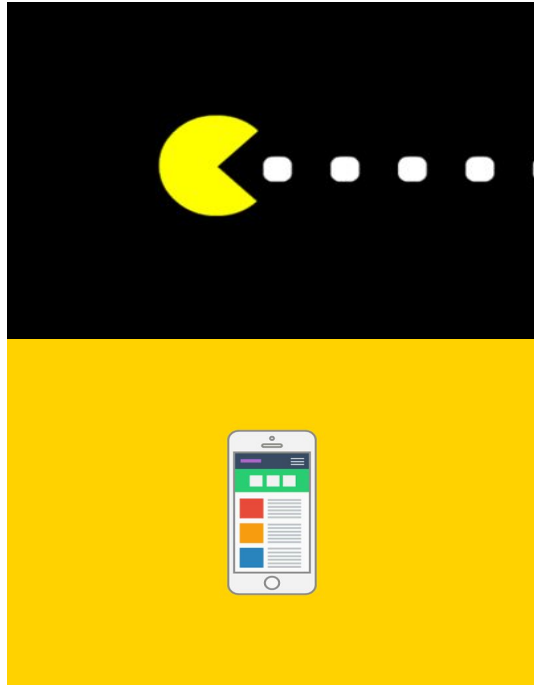
Lecture 4: Arrays, Loops part 1

Lecture 5: Loops part 2

Lecture 6: Functions



WHAT IS SOFTWARE ENGINEERING AND ITS BENEFITS?



COURSE CONTENT!

- Website: <https://schulichignite.com/>
 - Here's where you can see all the content slides for all of our sessions
 - Examples also live here!



ICEBREAKER
TIME!

ICEBREAKER ACTIVITY WITH YOUR GROUP!

Time for introductions! With your group answer the following questions:

- What's your name?
- What is your favorite smell?
- Name something about programming that interests you
- Talk to your mentors?



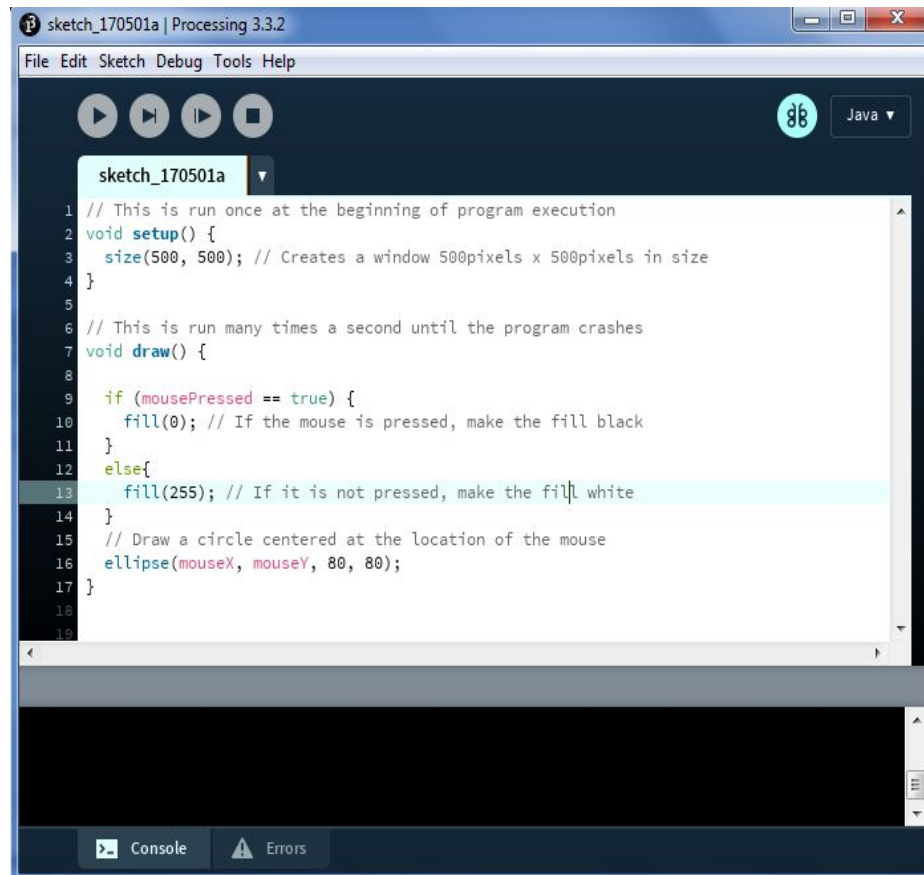
WHAT IS THE DIFFERENCE BETWEEN
HUMANS AND COMPUTERS?

LET'S GET STARTED!



WHAT IS PROCESSING?

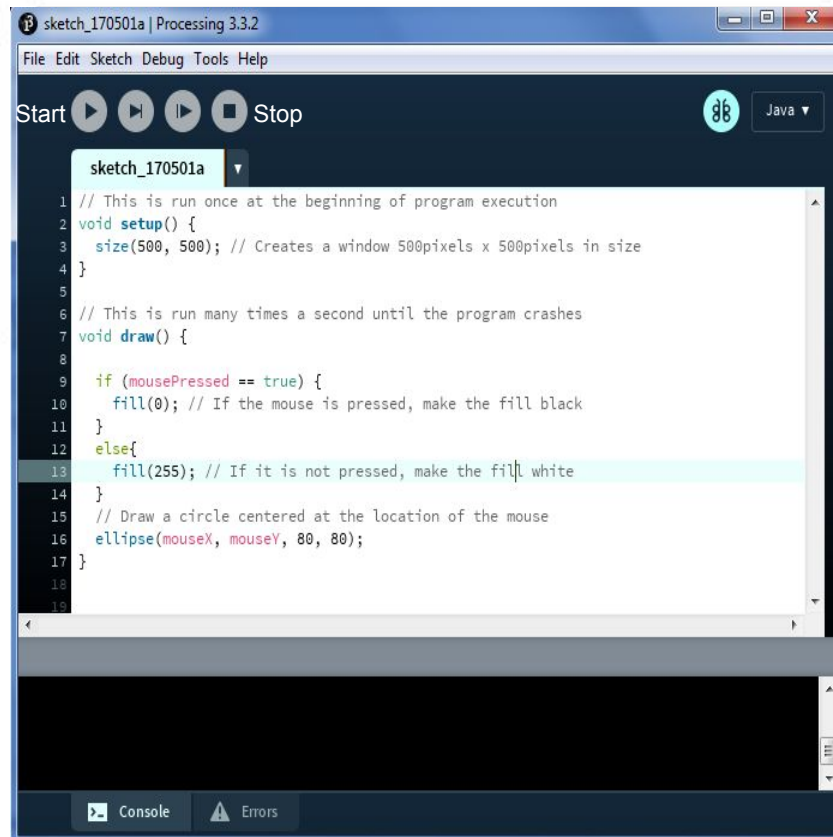
- Beginner-friendly programming language
- Free to download!
- Java Based
- Lots of references available at processing.org/reference/



PROCESSING'S STRUCTURE



Display Window



Menu

Toolbar

Tabs

Text Editor

Message Area

Console

THE BASICS!

VOID SETUP

- For now, all code goes inside the `void setup()` { curly braces }

```
void setup() {  
    // Type your code in here...  
}
```

TELL ME SOMETHING!

- `println()`; prints words to the console
(the little black box)
- Very useful to see that's going on in your program.
- Comments: Anything after `//` is ignored as a comment until the end of the line. Very useful for programmers!

LET'S DRAW SOMETHING!

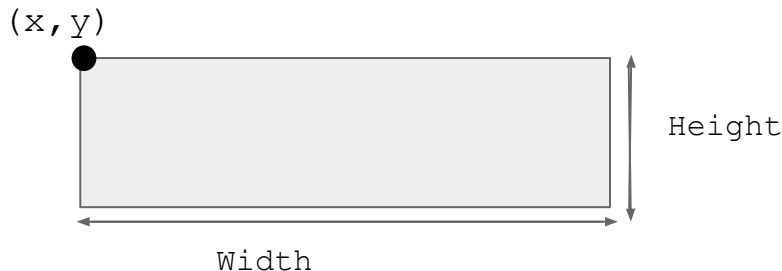
```
rect(x, y, w, h);
```

x : x-coordinate of the top left corner

y : y-coordinate of the top left corner

w : width of rectangle

h : height of rectangle



ANOTHER SHAPE

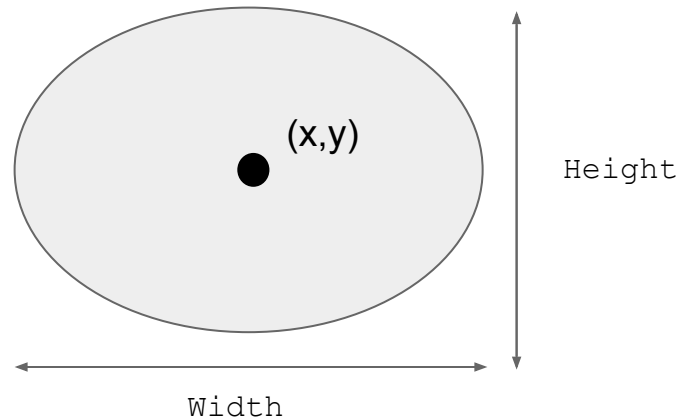
`ellipse(x, y, w, h)`

x : x-coordinate of the **centre** of the ellipse

y : y-coordinate of the **centre** of the ellipse

w : width of the ellipse

h : height of the ellipse



THE OTHER SHAPES

```
point(x, y);  
line(x1, y1, x2, y2);  
triangle(x1, y1, x2, y2, x3, y3);
```

Ask your mentor to know more!

COLOURS AND EXERCISE

```
fill(r, g, b);
```

r : Red amount

g : Green amount

b : Blue amount

NOTE: All colour amounts must be between 0 and 255

TRY IT YOURSELF!

Choose one of the following to draw (10 mins):

- Draw a smiley face
- Draw a snowman
- Draw a car
- Draw a bear/teddy bear
- Draw a robot
- Draw anything you like

VARIABLES

VARIABLES

Has a **type**, what it contains (Number or text)

Has a **name**, what you're calling it

It stores a **value**, the thing it contains

<code>int</code>	<code>age</code>	<code>=</code>	<code>23;</code>
<u> </u>	<u> </u>		<u> </u>
Type	Name		Value

INTEGER

As you guessed an integer is just an... `integer`
(Therefore, does **not** include decimals)

This includes positive and negative numbers

SYNTAX:

`int`

FLOAT

Use this for all of your decimal dreams

This also include positive and negative decimals

SYNTAX:

`float`

STRING

Fancy name for text (or a string of letters)

(Sadly, this does not include negative texts because it doesn't even exist)

SYNTAX:

String

EXAMPLES

// No decimals allowed!

```
int age = 23;
```

// Don't forget the double quotes!

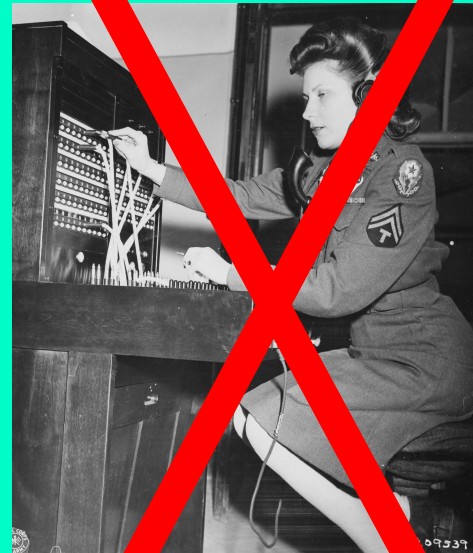
```
String name = "Judy";
```

// Decimals used here!

```
float salary = 3000.29;
```

OPERATORS*

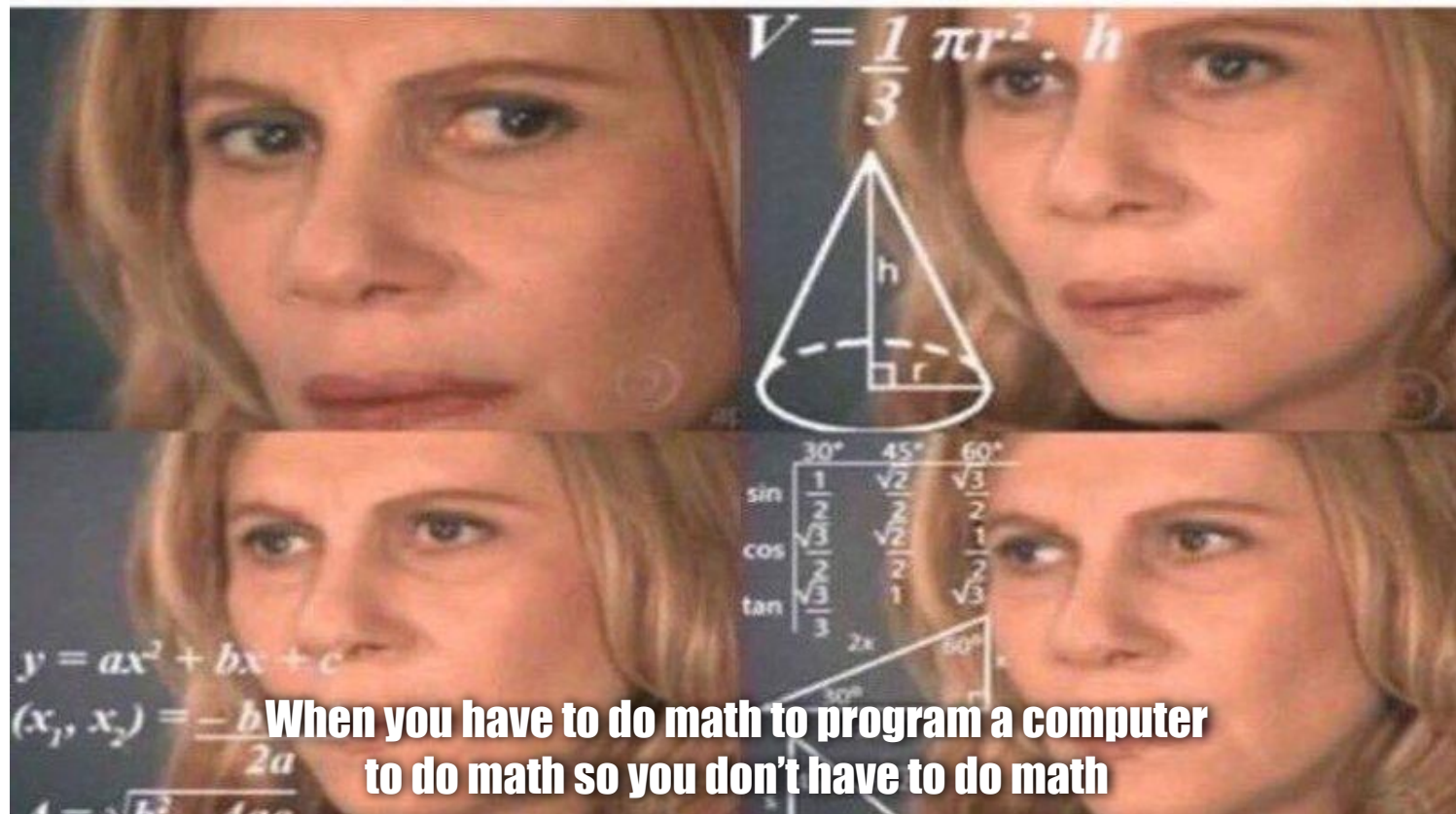
*Not a phone operator



ARITHMETIC OPERATORS

Operator	Meaning	Example
+	Addition	$3 + 4$
-	Subtraction	$7 - 5$
*	Multiplication	$2 * 3$
/	Division	$12 / 3$
%	Modulus	$5 \% 2$

LET'S DO SOME QUICK MATHS



LETS PUT THESE VARIABLES TO USE!

Let's make a moving ball!

```
void draw() {  
    //Code goes here  
}
```

Interesting variables already given:

mouseX: your mouse's X coordinate

mouseY: your mouse's Y coordinate

END OF SESSION EXERCISES (DO WHAT YOU CAN)

- Go back and finish your drawing
- Use both `mouseX` and `mouseY` to make the move the ball to the mouse's coordinates
- Draw a rectangle with the top-left corner at the coordinates (100, 200), and make the bottom-right corner follow the mouse