

An abstract graphic on the left side of the slide, consisting of a network of thin, light-blue lines and small circles, resembling a circuit board or a neural network diagram. The lines are vertical and horizontal, with some diagonal connections, and the circles are small and white, scattered along the lines.

# VARIABLES

JANUARY 14, 2022

# WHAT DOES A VARIABLE DO?

- Stores a value in memory so it can be used later in the program
- Value can be easily changed while the program is running

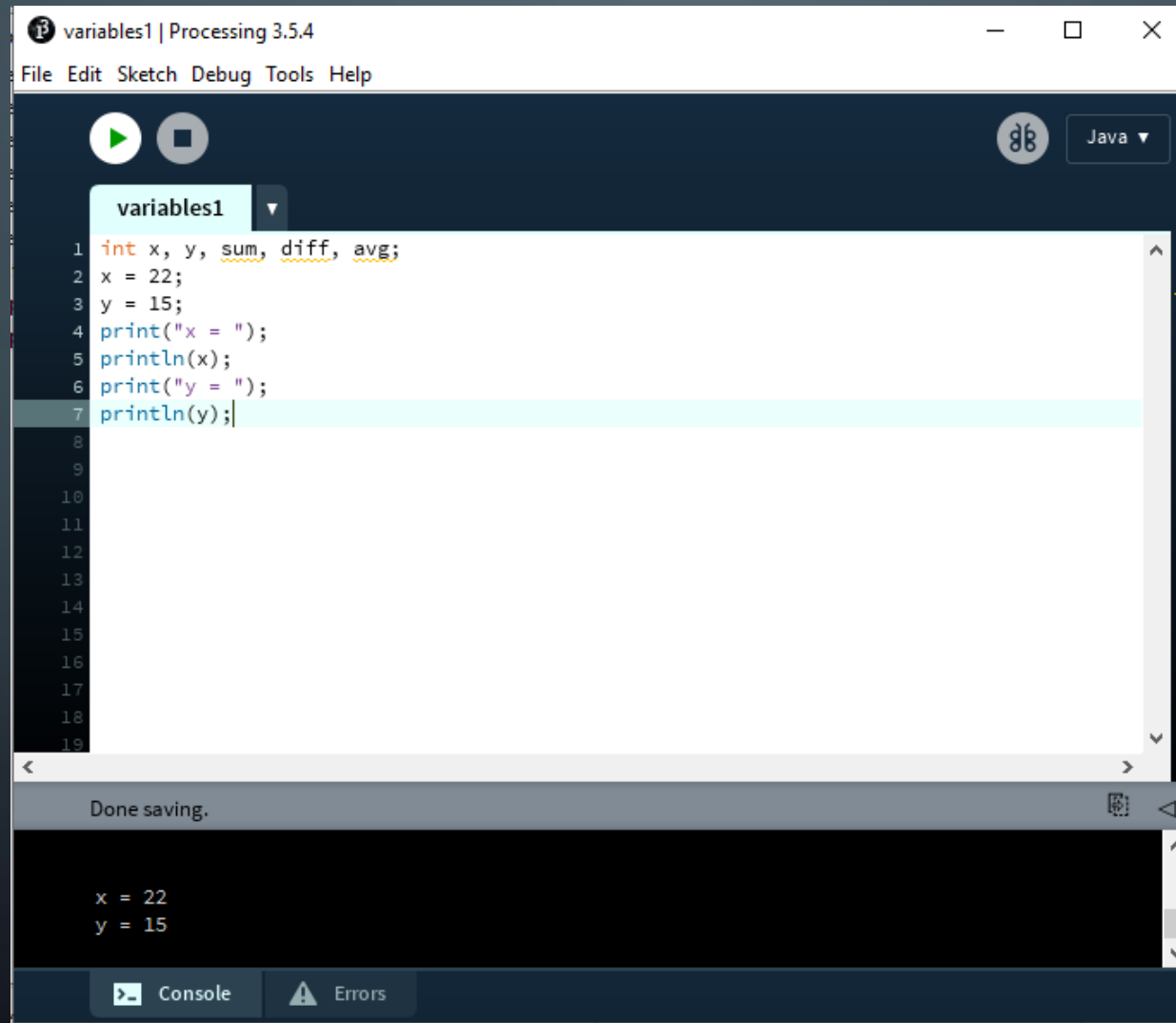
# WHY DO WE USE VARIABLES?

- Avoid repeating ourselves
- Much easier to update code
- Much more efficient

# WHAT DO WE NEED TO DO?

- Determine the name, data type, and value
- Convention is to use lower case for variable names
- Data types include (but are certainly not limited to)
  - int (integer)
  - float (short for floating-point, which can include decimals)
  - Array (holds a list of data)

# EXAMPLE FROM LAB 1



The screenshot shows the Processing 3.5.4 IDE window titled "variables1 | Processing 3.5.4". The menu bar includes File, Edit, Sketch, Debug, Tools, and Help. The toolbar contains a play button, a stop button, a Java language selector, and a "variables1" dropdown menu. The code editor displays the following code:

```
1 int x, y, sum, diff, avg;  
2 x = 22;  
3 y = 15;  
4 print("x = ");  
5 println(x);  
6 print("y = ");  
7 println(y);
```

The console output at the bottom shows:

```
x = 22  
y = 15
```

# BUILT-IN VARIABLES

- width – automatically stores width of window
- height – automatically stores height of window
- mouseX – automatically stores current x-position of the mouse
- mouseY – automatically stores current y-position of the mouse

# LOOKING AT EXAMPLE 2-2 AGAIN

```
void setup() {  
  size(480, 120);  
}  
  
void draw() {  
  if (mousePressed) {  
    fill(0);  
  } else {  
    fill(255);  
  }  
  ellipse(mouseX, mouseY, 80, 80);  
}
```

We still don't have the tools to understand *all* of this code...but how about “size”? And “ellipse”? We can understand those now!