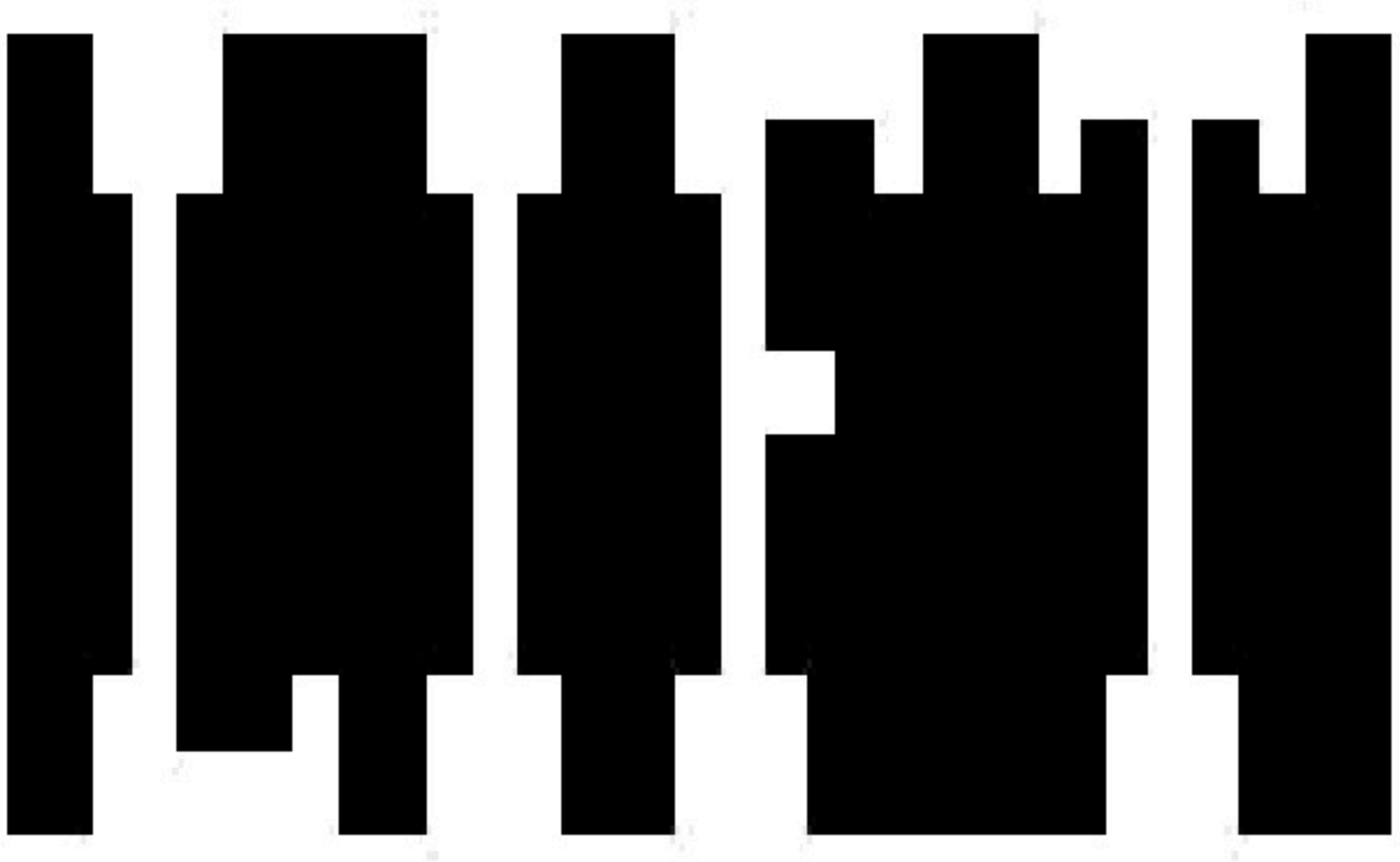
INTRO TO JAVASCRIPT

INTRO TO JS



WEEKLY OVERVIEW

WEEK 4	Forms / Intro to JS
WEEK 5	jQuery Basics / jQuery Lab

LEARNING OBJECTIVES

- Practice programmatic thinking by writing pseudocode.
- Describe event-based web site behavior and list practical uses of JavaScript on a website's front end.
- Predict DOM output / changes by reading JavaScript code.
- Define variables and identify best cases to use them.
- Describe strings, numbers, and boolean variable types.
- Use comparison operators to evaluate and compare statements.
- Apply conditionals to change the program's control flow

AGENDA

- Intro to Programming
- Intro to Pseudo Code
- Intro to JS
- Reading JS
- Variables
- Conditionals
- Lab

INTRO TO PROGRAMMING

INTRO TO JS

DO NOW

DIRECTIONS

1. Find a new partner. Someone you haven't yet worked with!

2. Identify one person to DRAW and one person to TALK.

3. The person TALKING chooses a shape or object (circle, star, heart, slice of pizza, motorcycle etc).

4. Using explicit step-by-step directions, the talker must get their partner to draw the correct shape without ever stating what that shape is.

5. The person drawing must follow directions exactly, and **may not draw anything that their partner did not say exactly.**

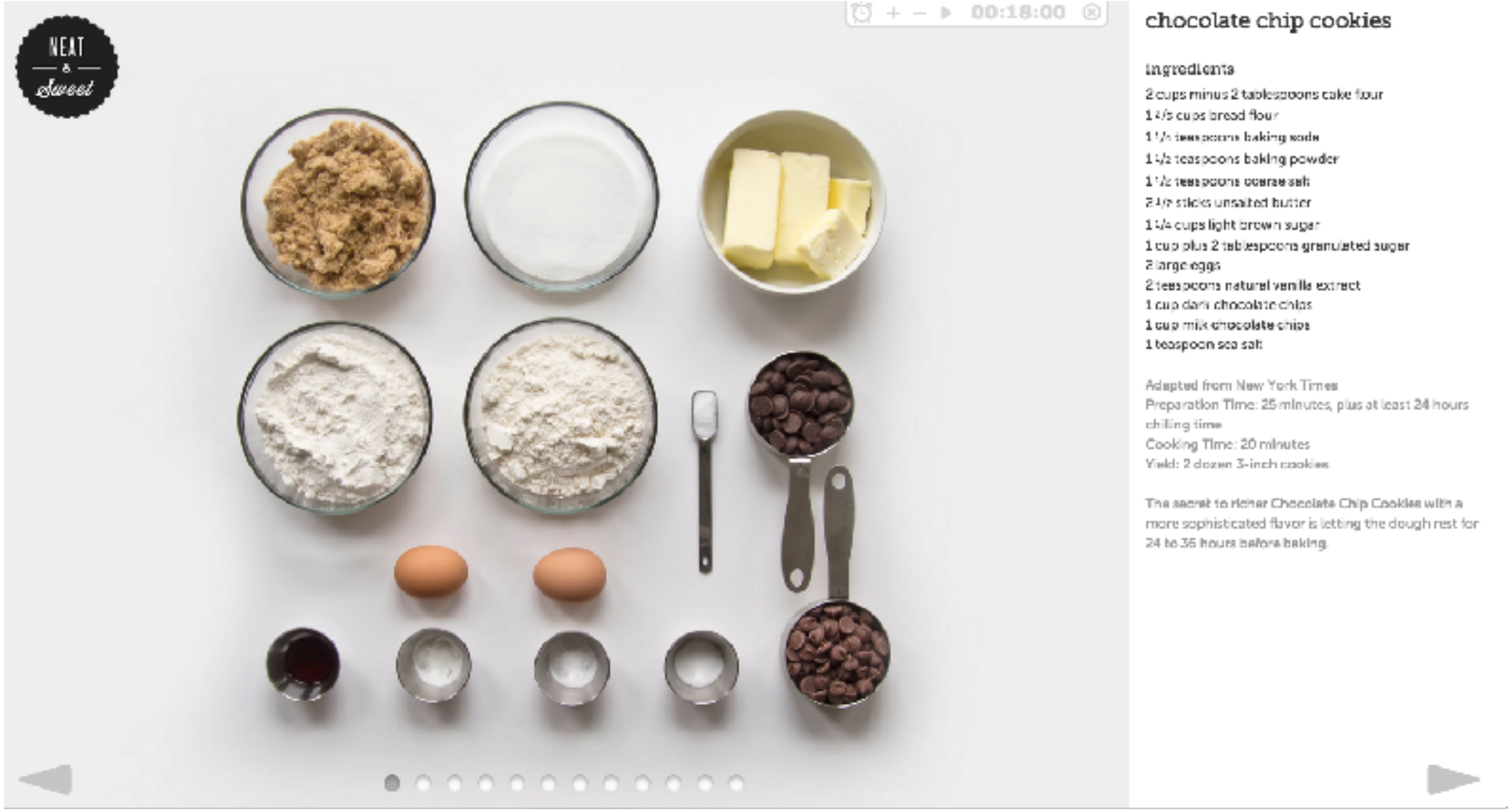
WHAT IS A PROGRAM?

- A program is a set of instructions that you write to tell a computer what to do

WHAT IS PROGRAMMING?

- Programming is the task of writing those instructions in a language that the computer can understand.

WHAT IS A PROGRAM?



chocolate chip cookies

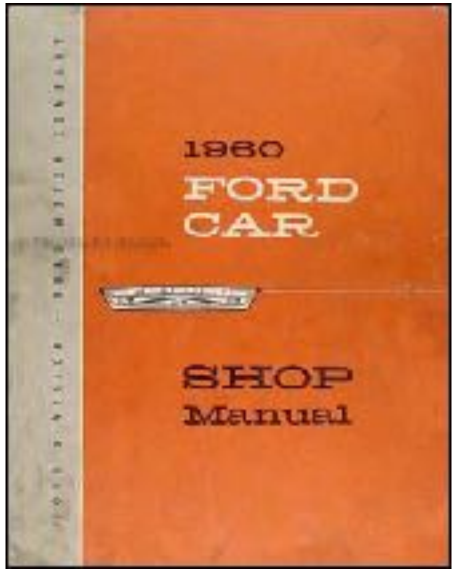
ingredients

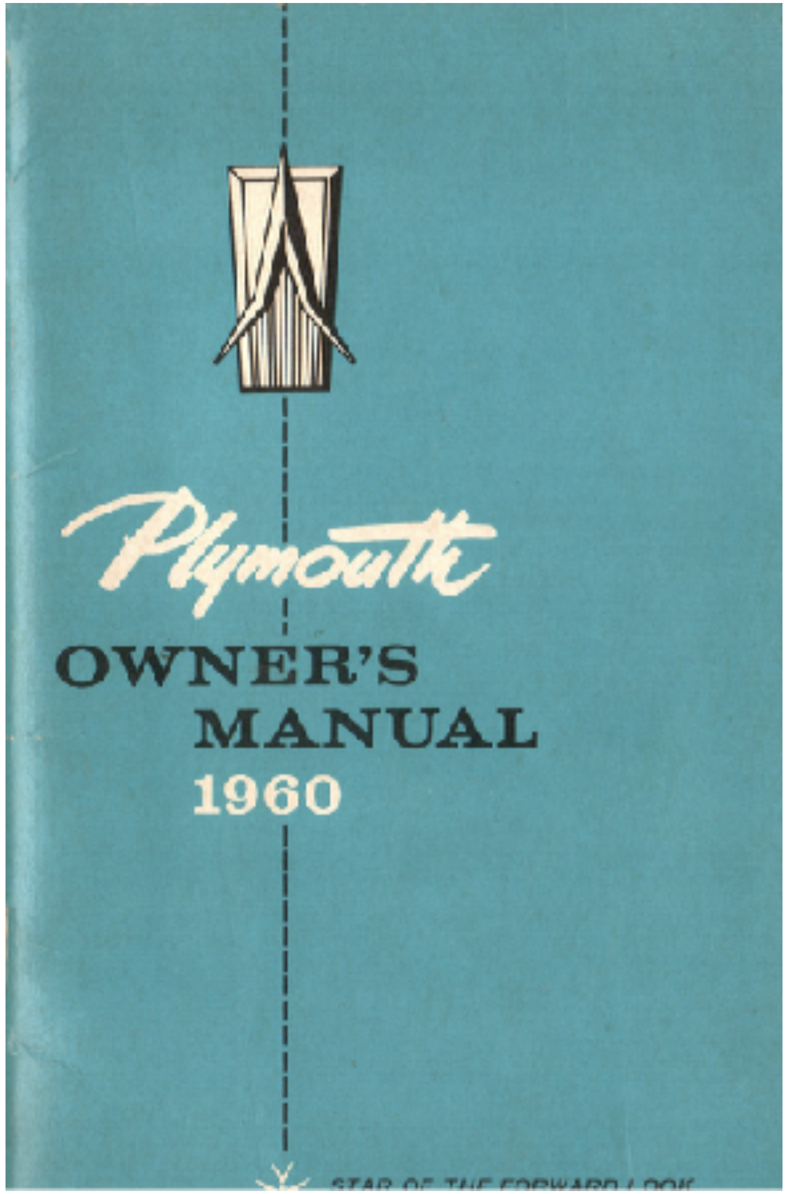
2 cups minus 2 tablespoons cake flour
1 2/3 cups bread flour
1 1/4 teaspoons baking soda
1 1/2 teaspoons baking powder
1 1/2 teaspoons coarse salt
2 1/2 sticks unsalted butter
1 1/4 cups light brown sugar
1 cup plus 2 tablespoons granulated sugar
2 large eggs
2 teaspoons natural vanilla extract
1 cup dark chocolate chips
1 cup milk chocolate chips
1 teaspoon sea salt

Adapted from New York Times
Preparation Time: 25 minutes, plus at least 24 hours chilling time
Cooking Time: 20 minutes
Yield: 2 dozen 3-inch cookies

The secret to richer Chocolate Chip Cookies with a more sophisticated flavor is letting the dough rest for 24 to 36 hours before baking.

WHAT IS A PROGRAM?





SO HOW DO COMPUTERS THINK?

- Short answer — they don’t think!
- While computers don’t think, they *act as if they do*, by sequentially executing simple instructions.
- The only things a computer knows are the things we tell it.
- A computer doesn't learn to perform tasks like you and I — it needs to follow instructions every time it performs the task.

INTRO TO PSEUDO CODE

PSEUDO CODE

- When we write a program, we need to figure out a way to translate the ideas that are in our heads into code
- Pseudo code is a way to 'plan out' your program before coding it
- **Pseudo code** is a *detailed yet readable description* of what a computer program must do
- Expressed in plain English rather than in a programming language

THE IMPORTANCE OF PLANNING

PSEUDO CODE — HEIGHT COMPARISON

</>

CODE ALONG

PSEUDO CODE — PASSING SCORE

LAB — SURVEY

EXERCISE

KEY OBJECTIVE

- Practice programmatic thinking by writing pseudo code to solve a basic problem

TYPE OF EXERCISE

- Group of 2

TIMING

5 min

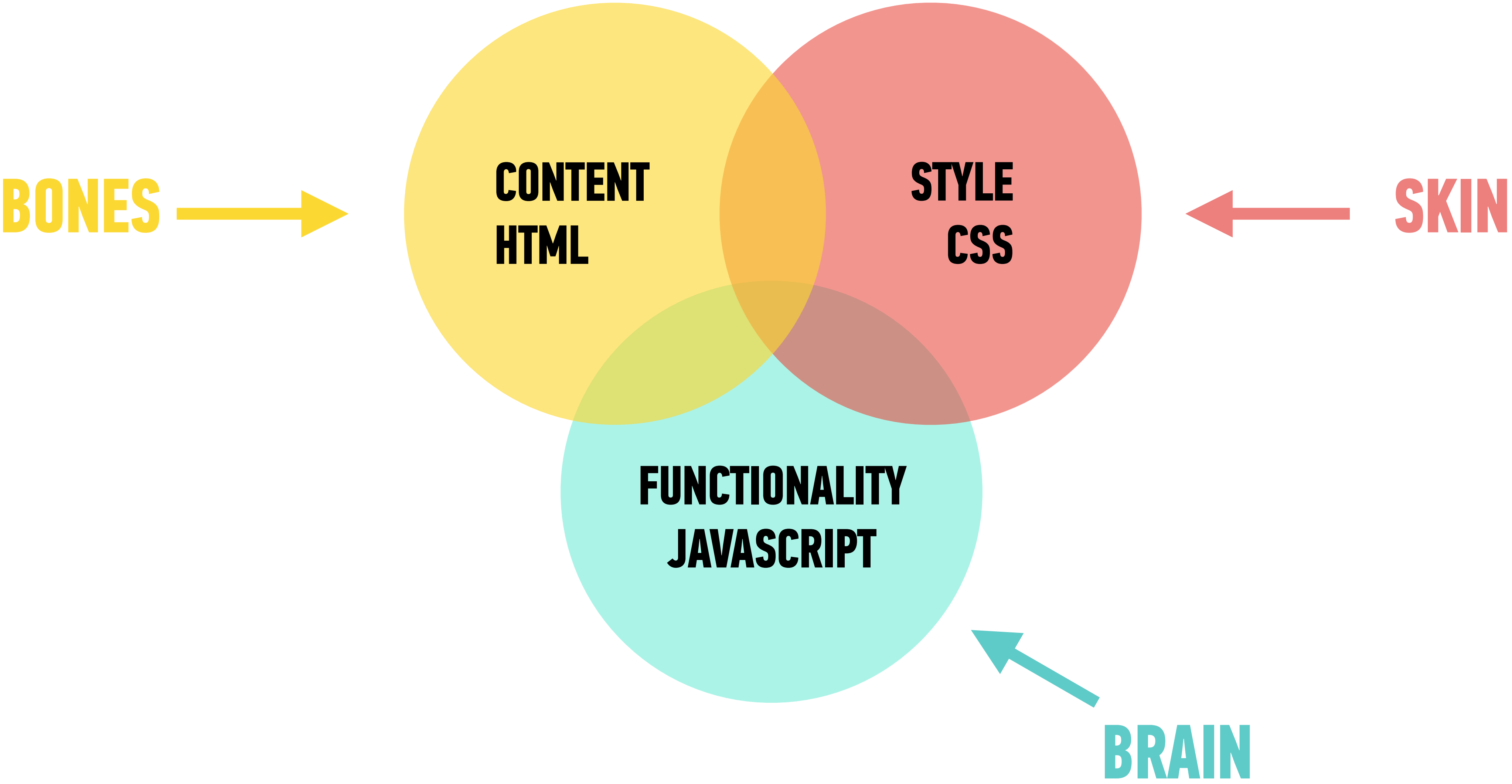
1. Image we have a simple webpage with a button, a ring, and Frodo.
2. Every time users click on the button, the Ring of Power gets put on Frodo's finger, which makes Frodo disappear. If users click on the button again, the Ring of Power gets taken off of Frodo's finger, and Frodo reappears.

LEARNING OBJECTIVES

- Practice programmatic thinking by writing pseudocode.
- Describe event-based web site behavior and list practical uses of JavaScript on a website's front end.
- Predict DOM output / changes by reading JavaScript code.
- Define variables and identify best cases to use them.
- Describe strings, numbers, and boolean variable types.
- Use comparison operators to evaluate and compare statements.
- Apply conditionals to change the program's control flow

INTRO TO JS

THE TRIPLE SCOOP: STRUCTURE, STYLE, BEHAVIOR



JAVA VS. JAVASCRIPT

Just a quick note! We're learning ***JavaScript*** *in this class, not Java. Java and JavaScript are actually two different languages.*

!==

WHAT JAVASCRIPT CAN DO!

1 Access Content

2 Modify Content

3 Program Rules

4 React to Events

WHAT JAVASCRIPT CAN DO!

1
Access Content

2
Modify Content

3
Program Rules

4
React to Events

You can use JS to select any element, attribute or text from an HTML page.

For example:

- Find out what the user entered into a text input when they submit a form
- Find out whether the user checked a checkbox

WHAT JAVASCRIPT CAN DO!

1
Access Content

2
Modify Content

3
Program Rules

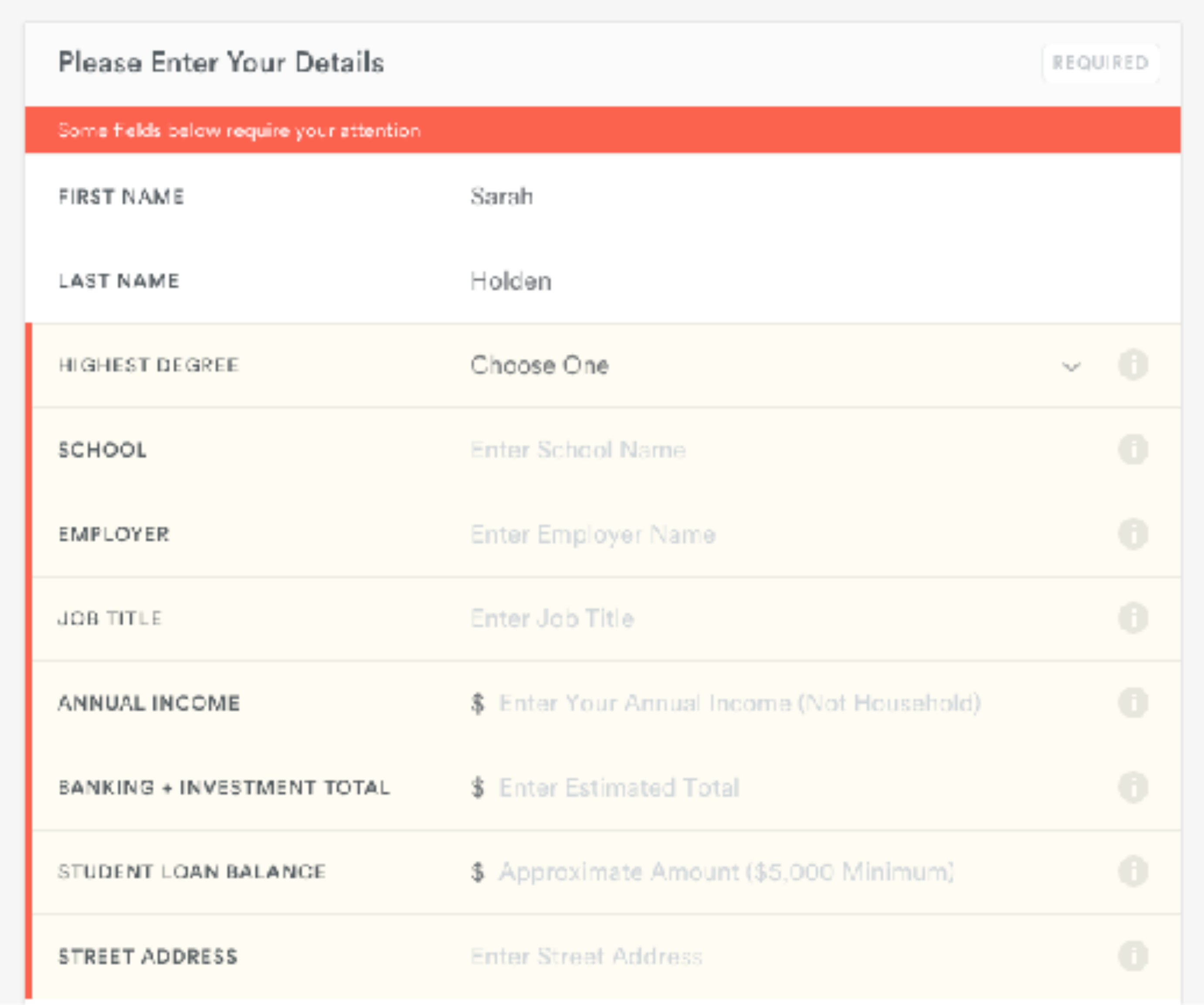
4
React to Events

You can use JS to add elements, attributes and text to the page (or remove them)

For example:

- Add an error message below a form
- Change the size, position, color, or other styles for an element

WHAT JAVASCRIPT CAN DO – MODIFYING CONTENT



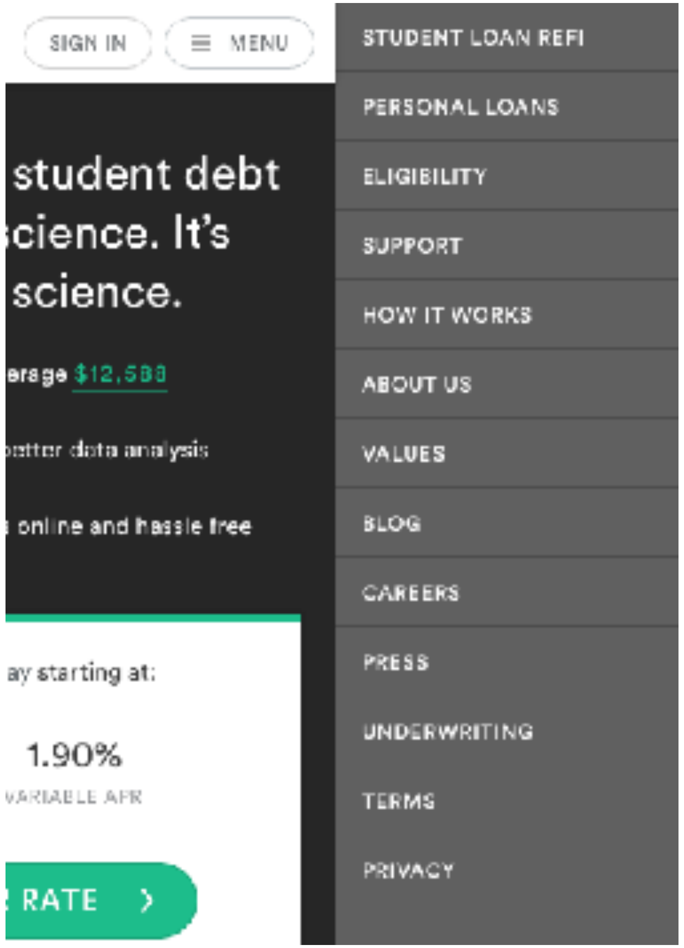
Please Enter Your Details

Some fields below require your attention

FIRST NAME	Sarah
LAST NAME	Holden
HIGHEST DEGREE	Choose One
SCHOOL	Enter School Name
EMPLOYER	Enter Employer Name
JOB TITLE	Enter Job Title
ANNUAL INCOME	$ Enter Your Annual Income (Not Household)
BANKING + INVESTMENT TOTAL	$ Enter Estimated Total
STUDENT LOAN BALANCE	$ Approximate Amount ($5,000 Minimum)
STREET ADDRESS	Enter Street Address

Add an error message (and styles) to a form

WHAT JAVASCRIPT CAN DO – MODIFYING CONTENT



Change the size, position, color, or other styles for an element

WHAT JAVASCRIPT CAN DO!

1 Access Content

2 Modify Content

3 Program Rules

4 React to Events

You can specify a set of steps (instructions) for the browser to follow.

For example:

- Have images/text fade in **if** the user has scrolled to a certain portion of the page
- Check to make sure the user has entered a valid email address into a form and display an error message if not

WHAT JAVASCRIPT CAN DO!

1 Access Content

2 Modify Content

3 Program Rules

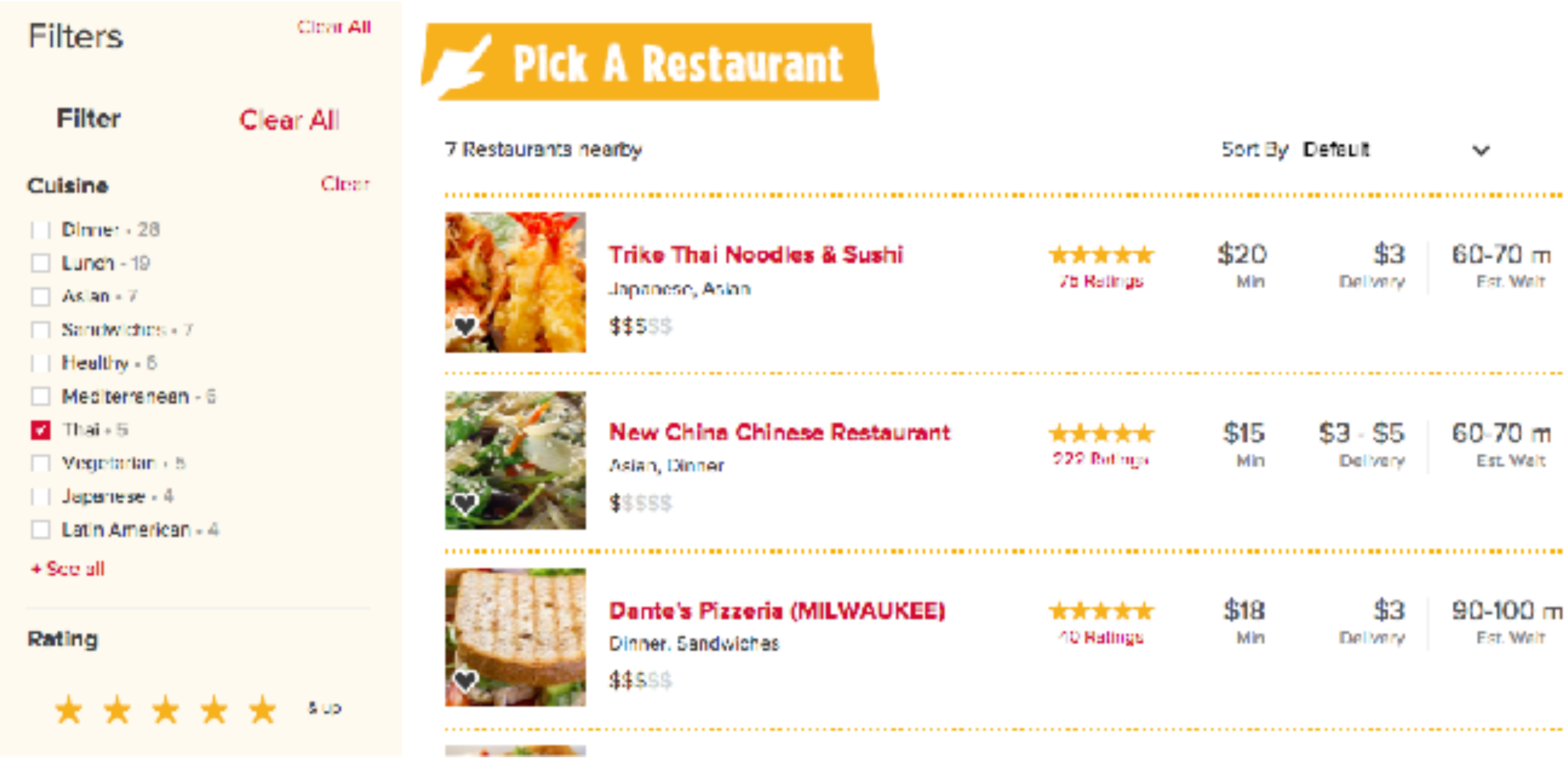
4 React to Events

You can specify that a script should run when an event occurs

For example:

- When a button is clicked
- When the cursor hovers over an element
- When the user types information into a form

WHAT JAVASCRIPT CAN DO – PROGRAM RULES



□ Dinner - 28
□ Lunch - 19
□ Asian - 7
□ Sandwiches - 7
□ Healthy - 6
□ Mediterranean - 6
☑ Thai - 5
□ Vegetarian - 5
□ Japanese - 4
□ Latin American - 4

★ ★ ★ ★ ★ & up

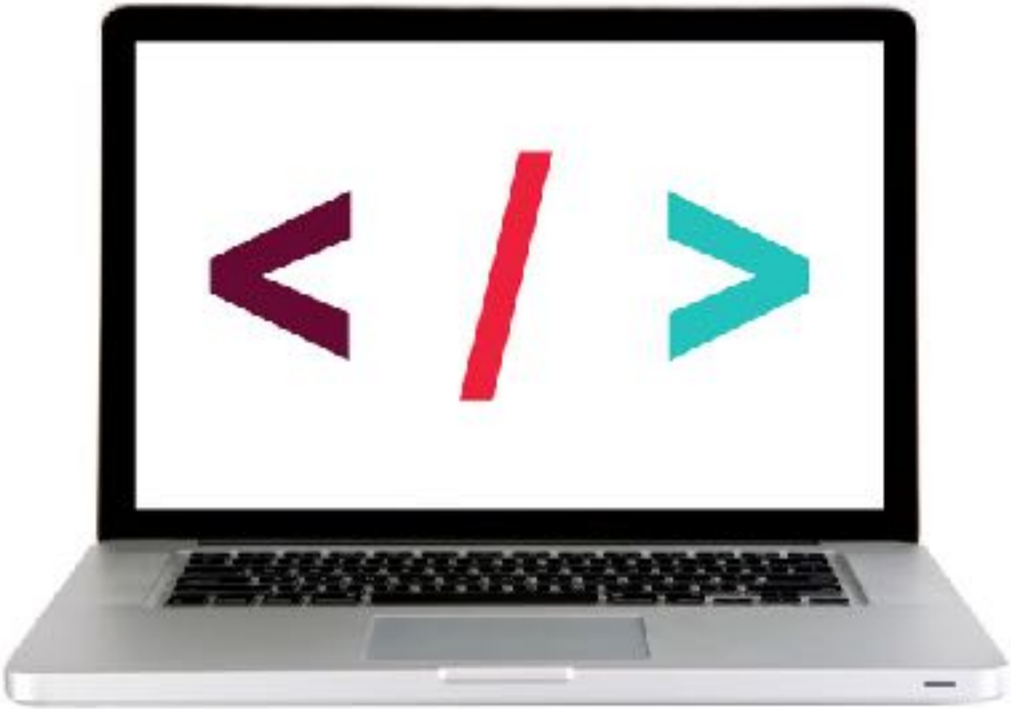
7 Restaurants nearby

Sort By Default

Trike Thai Noodles & Sushi Japanese, Asian $$$$$	★★★★★ 75 Ratings	$20 Min	$3 Delivery	60-70 m Est. Wait
New China Chinese Restaurant Asian, Dinner $$$$$	★★★★★ 229 Ratings	$15 Min	$3 - $5 Delivery	60-70 m Est. Wait
Dante's Pizzeria (MILWAUKEE) Dinner, Sandwiches $$$$$	★★★★★ 40 Ratings	$18 Min	$3 Delivery	90-100 m Est. Wait

Filter data when the user checks a checkbox

LET'S TAKE A LOOK



https://kinhr.com/

READING JS

READING JS

- When you are a child you learn to speak and read before you learn to write
- We learned to 'speak' JS with the discussion and pseudo code

READING JS — COLOR SWITCHER WALK THROUGH

Color Switcher CodePen

LAB — TRAFFIC LIGHT

LAB — TRAFFIC LIGHT

EXERCISE

KEY OBJECTIVE

- Predict DOM output / changes by reading JS code.

TYPE OF EXERCISE

- Partner

TIMING

30 min

1. Take a look at the Traffic Light code in Codepen
2. The yellow button changes the bulb to purple and the green light does not work.
3. Make some minor changes to the code so that the traffic light works correctly.

LEARNING OBJECTIVES

- Practice programmatic thinking by writing pseudocode.
- Describe event-based web site behavior and list practical uses of JavaScript on a website's front end.
- Predict DOM output / changes by reading JavaScript code.
- Define variables and identify best cases to use them.
- Describe strings, numbers, and boolean variable types.
- Use comparison operators to evaluate and compare statements.
- Apply conditionals to change the program's control flow

VARIABLES

EXERCISE — READING AND GUESSING

EXERCISE

KEY OBJECTIVE

- Read a sample JavaScript file and see if you can guess what will happen.

TYPE OF EXERCISE

- Reading exercise (Groups of 3 - 4)

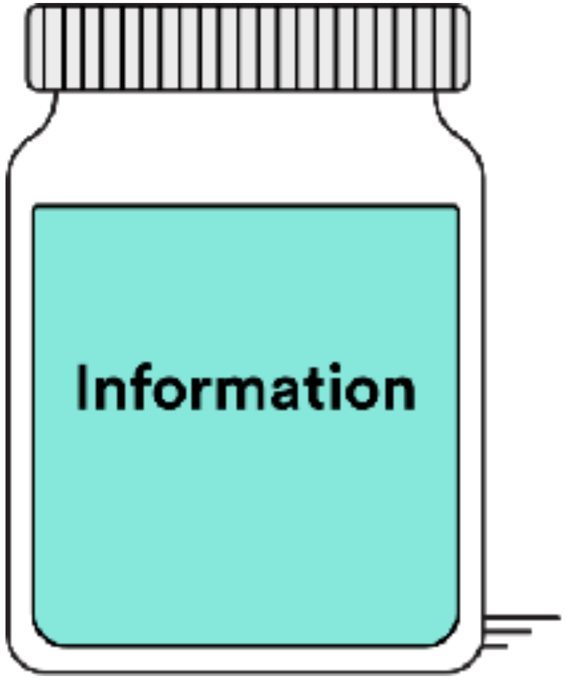
EXECUTION

1 min

1. Follow the instructions in this JS Bin
2. After reading the instructions, hit the "Run" button in the console tab to confirm your guesses.

WHAT ARE VARIABLES?

- We can tell our program to remember (store) values for us to use later on.
- The 'container' we use to store the value is called a **variable**



CREATING VARIABLES

EXERCISE

DIRECTIONS

1. We'll be using the console to practice creating variables. It's where JavaScript is interpreted and run. You can use it to practice writing JavaScript!
2. Open a new JS Bin session and select the "Console" tab. You can close any other tabs.

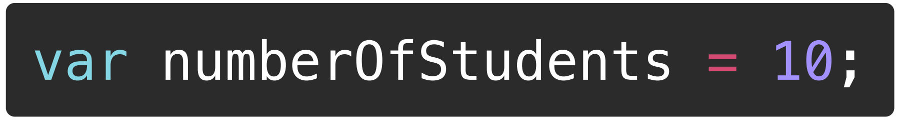
DECLARING A VARIABLE

```
var age = 29;
```

VARIABLE CONVENTIONS

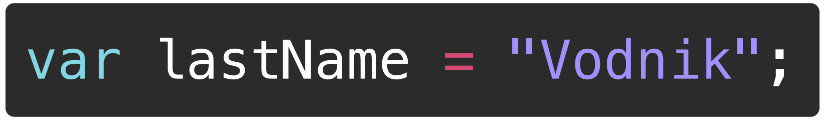
RULES:

1. Should be "camel case" — First word starts with a lowercase letter and any following words start with an uppercase letter.
2. Names can only contain: letters, numbers, $ and _
3. No dashes, no periods.
4. Cannot start with a number
5. Case sensitive - numberofstudents is not the same as numberOfStudents

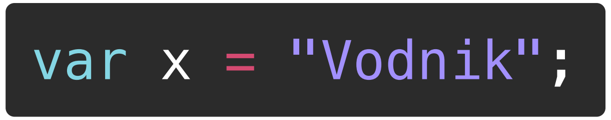
★ 

```
var numberOfStudents = 10;
```

Guideline: Names should be descriptive:

★ 

```
var lastName = "Vodnik";
```

✗ 

```
var x = "Vodnik";
```

JAVASCRIPT — UPDATING THE VALUE OF A VARIABLE

Declaring a variable:

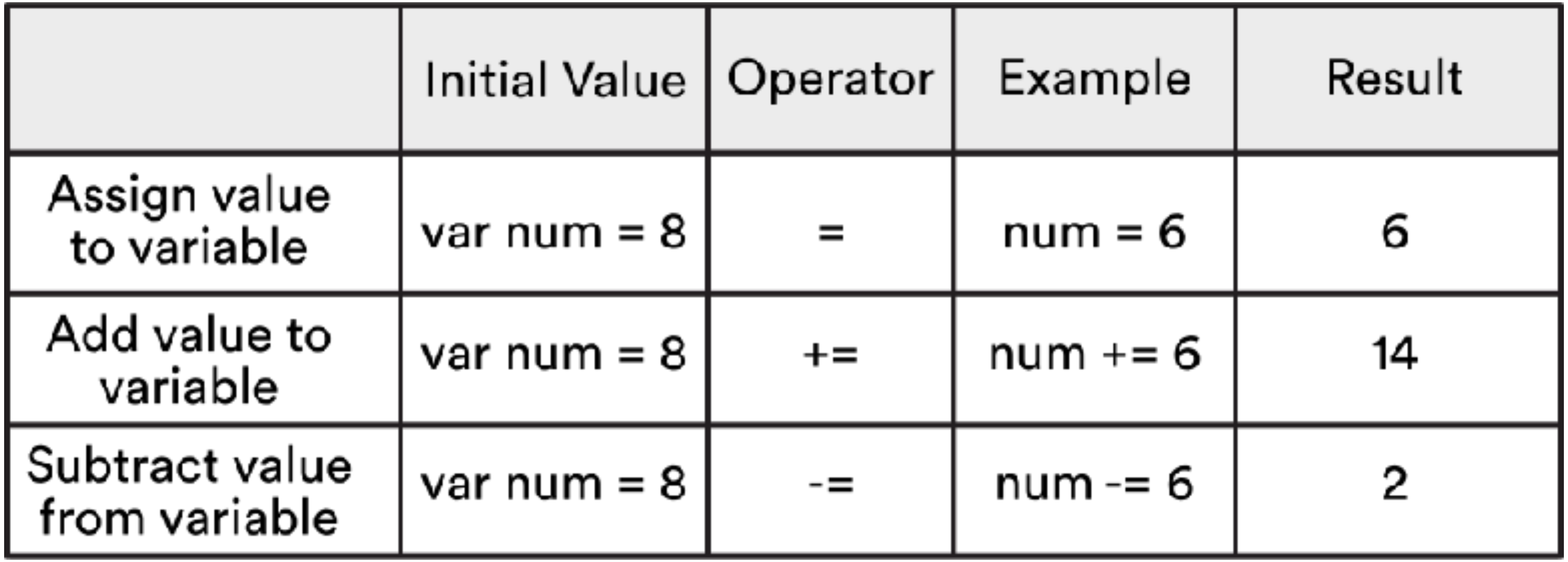
```
var host = "Sasha";
```

Update the value of the variable:

```
host = "TJ";
```

ASSIGNMENT OPERATORS

ASSIGNMENT OPERATORS



	Initial Value	Operator	Example	Result
Assign value to variable	var num = 8	=	num = 6	6
Add value to variable	var num = 8	+=	num += 6	14
Subtract value from variable	var num = 8	-=	num -= 6	2

ASSIGNMENT OPERATORS

```
var totalAmount = 6;
totalAmount += 4;
totalAmount -= 2;
```

What will total amount be equal to?

ASSIGNMENT OPERATORS

```
var score = 6;
score + 2;
```

What will score be equal to?

EXERCISE — VARIABLES

EXERCISE

KEY OBJECTIVE

- Practice declaring and assigning variables

TYPE OF EXERCISE

- Individual/paired

LOCATION

- Score Keeper (Codepen)

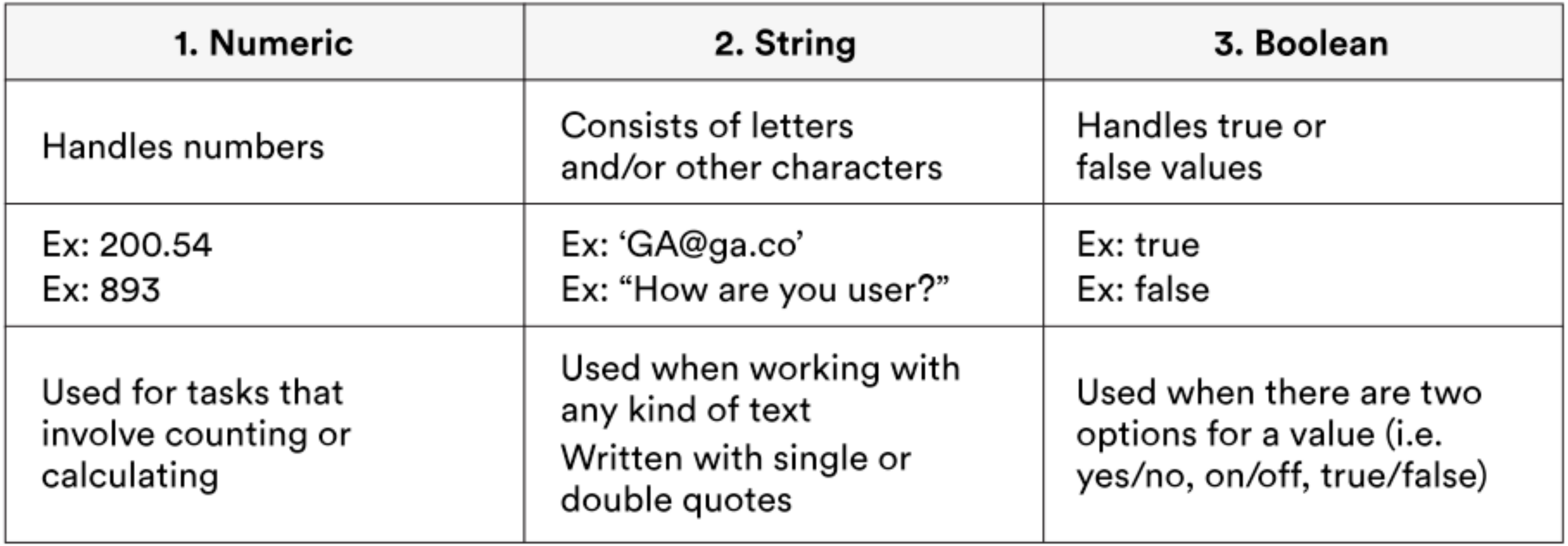
EXECUTION

5 min 1. Hook up the +10, -1 and -5 buttons

DATA TYPES

WHAT CAN BE STORED IN VARIABLES?

DATA TYPES:



1. Numeric	2. String	3. Boolean
Handles numbers	Consists of letters and/or other characters	Handles true or false values
Ex: 200.54 Ex: 893	Ex: ‘GA@ga.co’ Ex: “How are you user?”	Ex: true Ex: false
Used for tasks that involve counting or calculating	Used when working with any kind of text Written with single or double quotes	Used when there are two options for a value (i.e. yes/no, on/off, true/false)

DATA TYPES

NUMBERS

MORE ABOUT NUMBERS

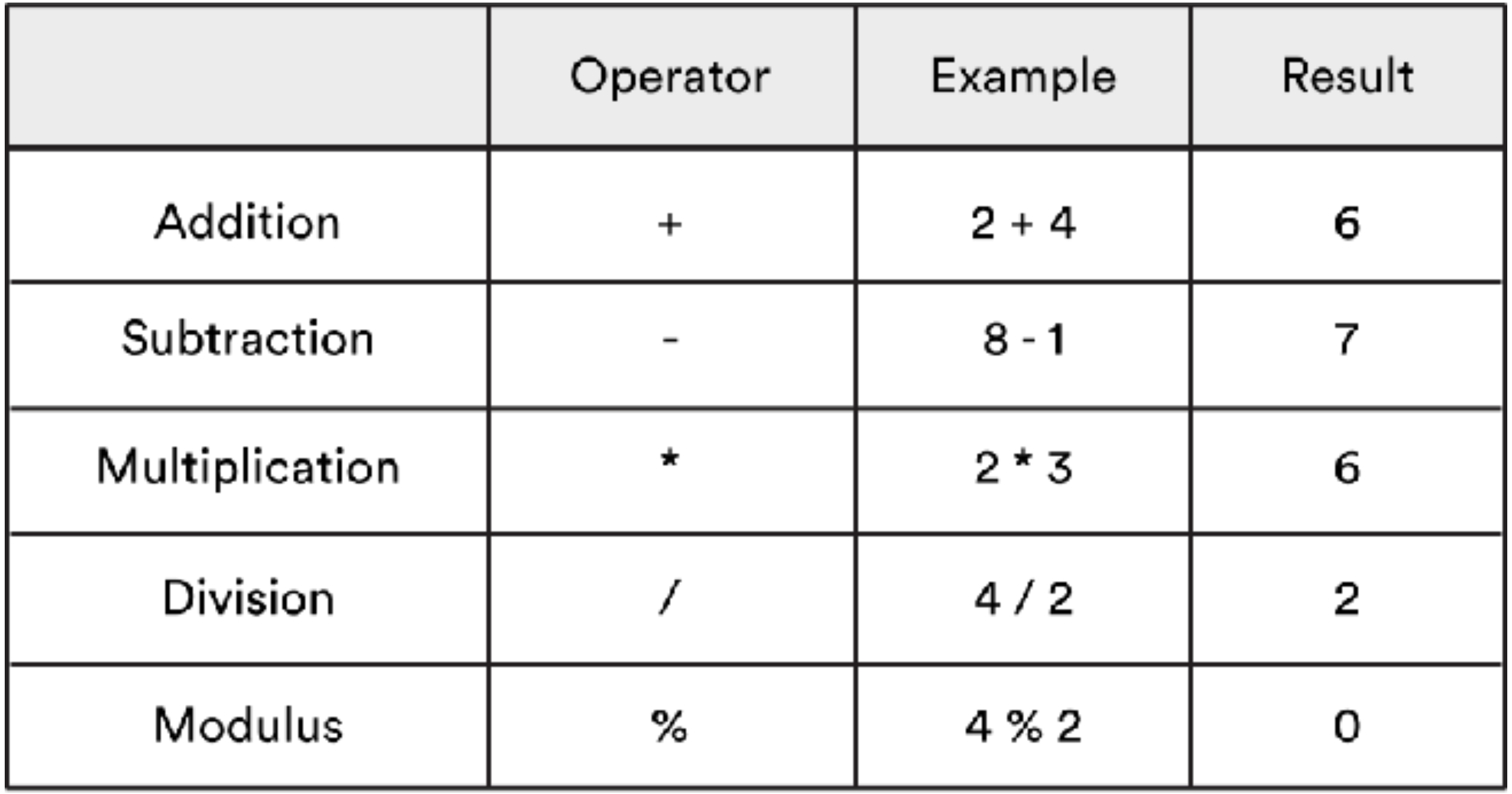
```
10
```

Whole numbers

```
22.75
```

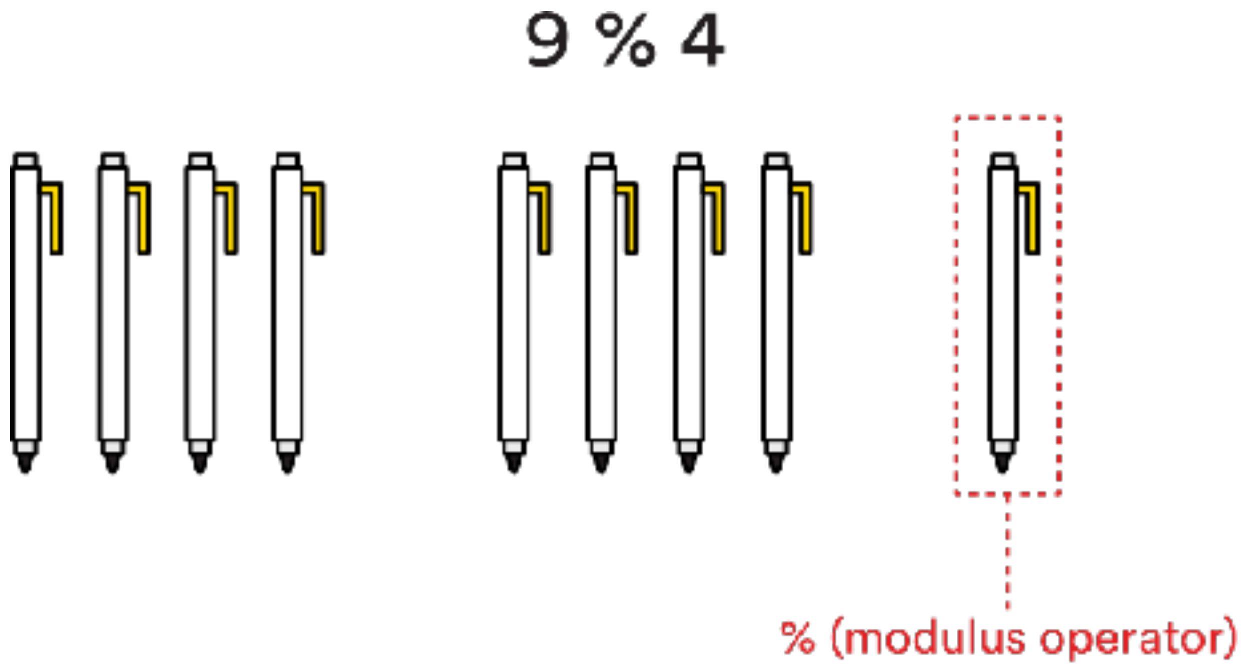
Number that uses a decimal to represent a fraction

ARITHMETIC OPERATORS



	Operator	Example	Result
Addition	+	2 + 4	6
Subtraction	-	8 - 1	7
Multiplication	*	2 * 3	6
Division	/	4 / 2	2
Modulus	%	4 % 2	0

ARITHMETIC OPERATORS



DATA TYPES

STRINGS

MORE ABOUT STRINGS

A STRING:

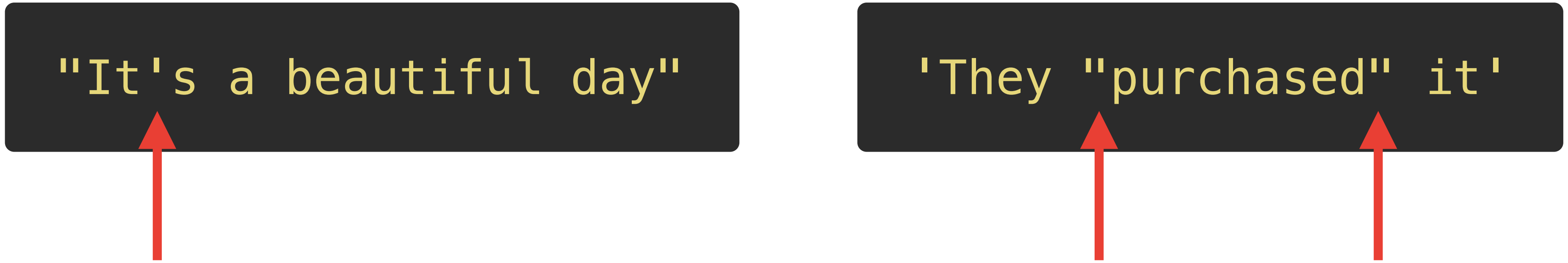
- Stores textual information
- Is surrounded by quotes

```
"How is the weather today?"
```

```
'Cold'
```

STRINGS

DOUBLE QUOTES VS. SINGLE QUOTES



```
"It's a beautiful day"
```

```
'They "purchased" it'
```

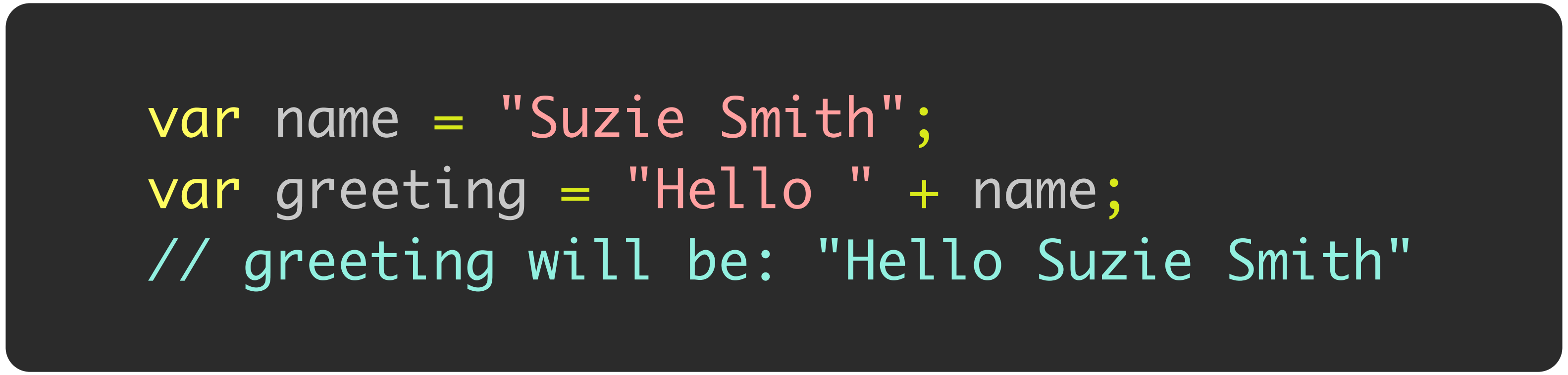
ESCAPING

```
'It\'s a beautiful day'
```

```
"They \"purchased\" it"
```

STRING CONCATENATION

- To take two strings (or a combination of strings and variables) and stick them together, use the + operator.
- This is called **string concatenation**.



```
var name = "Suzie Smith";
var greeting = "Hello " + name;
// greeting will be: "Hello Suzie Smith"
```

QUIZ

COMMON MISTAKES

```
"Bill" = var name;
```

COMMON MISTAKES

```
var name = "Bill";
```

COMMON MISTAKES

```
var total score = 20;
```

COMMON MISTAKES

```
var totalScore = 20;
```

COMMON MISTAKES

```
var fullName = Suzie Smith;
```

COMMON MISTAKES

```
var fullName = "Suzie Smith";
```

COMMON MISTAKES

```
Var fullName = "Bill Smith";
```

COMMON MISTAKES

```
var fullName = "Bill Smith";
```

COMMON MISTAKES

```
var score = "5";
 score += "6";
```

COMMON MISTAKES

```
var score = 5;
score += 6;
```

EXERCISE — VARIABLES

EXERCISE

KEY OBJECTIVE

- Practice writing variables

TYPE OF EXERCISE

- Individual/paired

EXECUTION

6 min

1. Follow the instructions under "Part 1" in this JS Bin.
2. Keep this tab open.

EXERCISE — VARIABLES

EXERCISE

KEY OBJECTIVE

- What are variables? Why would we want to use variables?

TYPE OF EXERCISE

- Turn and Talk

EXECUTION

30 sec

1. Describe variables. Explain why we would want to use variables in our programs.
2. What are the three data types in JS? Can you think of an example of each?

LEARNING OBJECTIVES

- Practice programmatic thinking by writing pseudocode.
- Describe event-based web site behavior and list practical uses of JavaScript on a website's front end.
- Predict DOM output / changes by reading JavaScript code.
- Define variables and identify best cases to use them.
- Describe strings, numbers, and boolean variable types.
- Use comparison operators to evaluate and compare statements.
- Apply conditionals to change the program's control flow

CONTROL FLOW

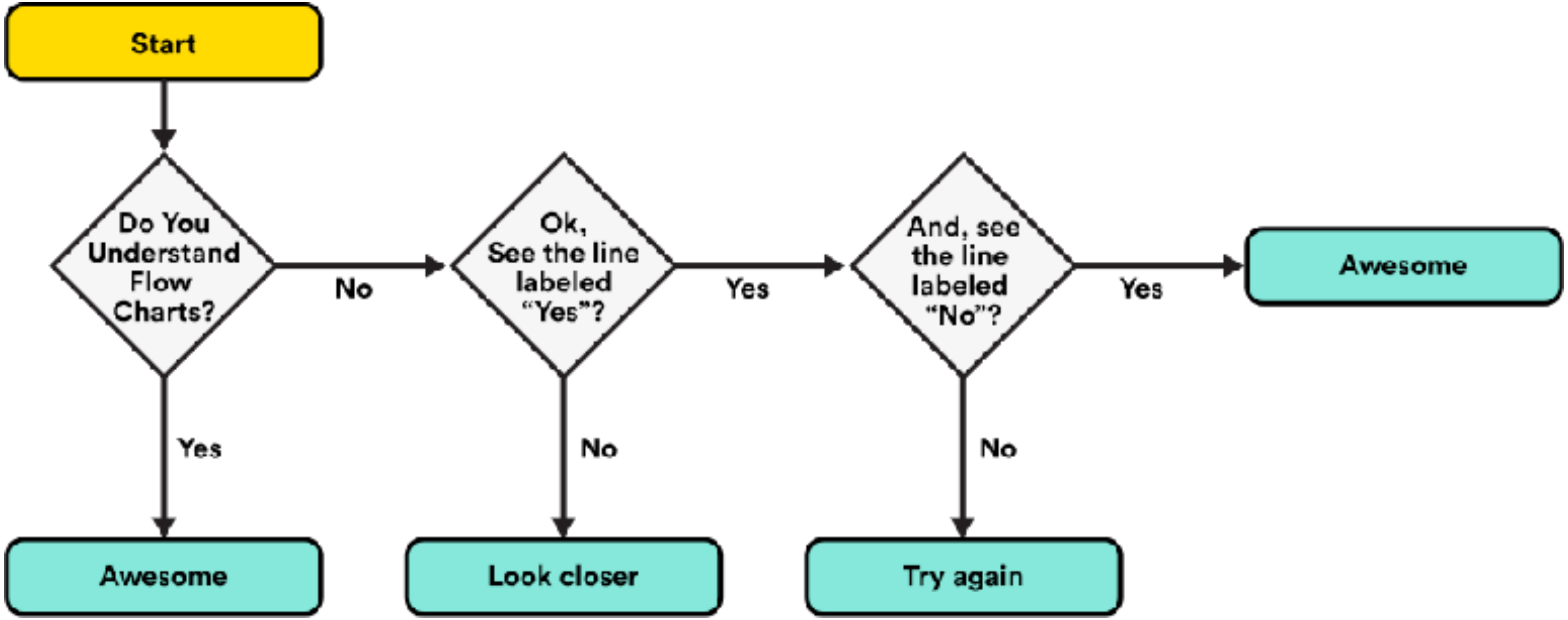
IF STATEMENTS

CONDITIONAL LOGIC

If something is true, do one thing. If it is not, do something else.

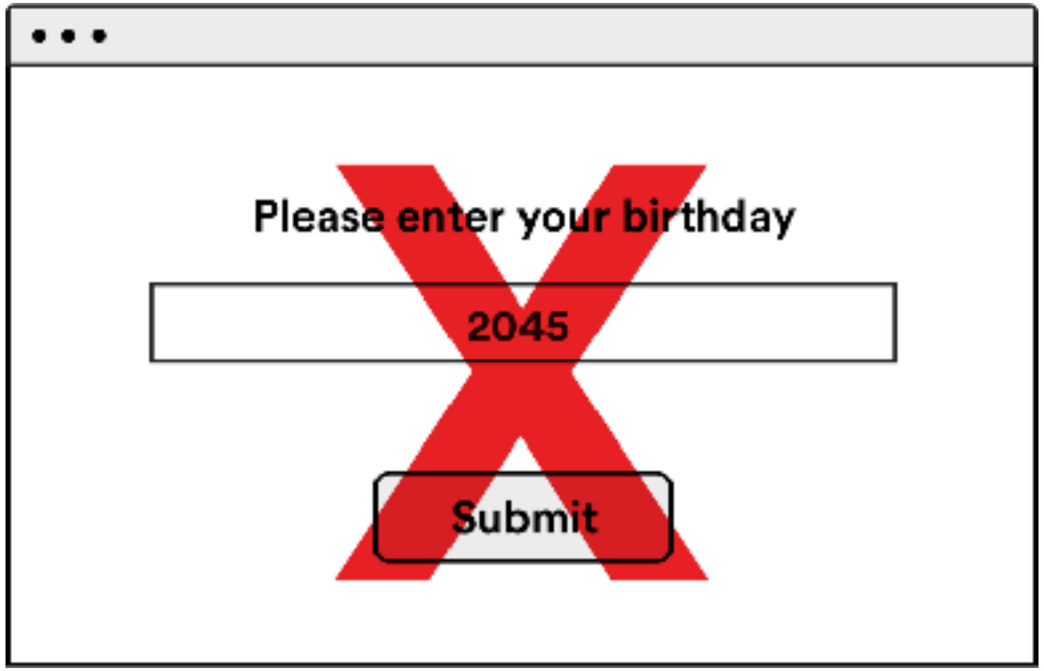
This type of logic or statement is a condition.

CONTROL FLOW

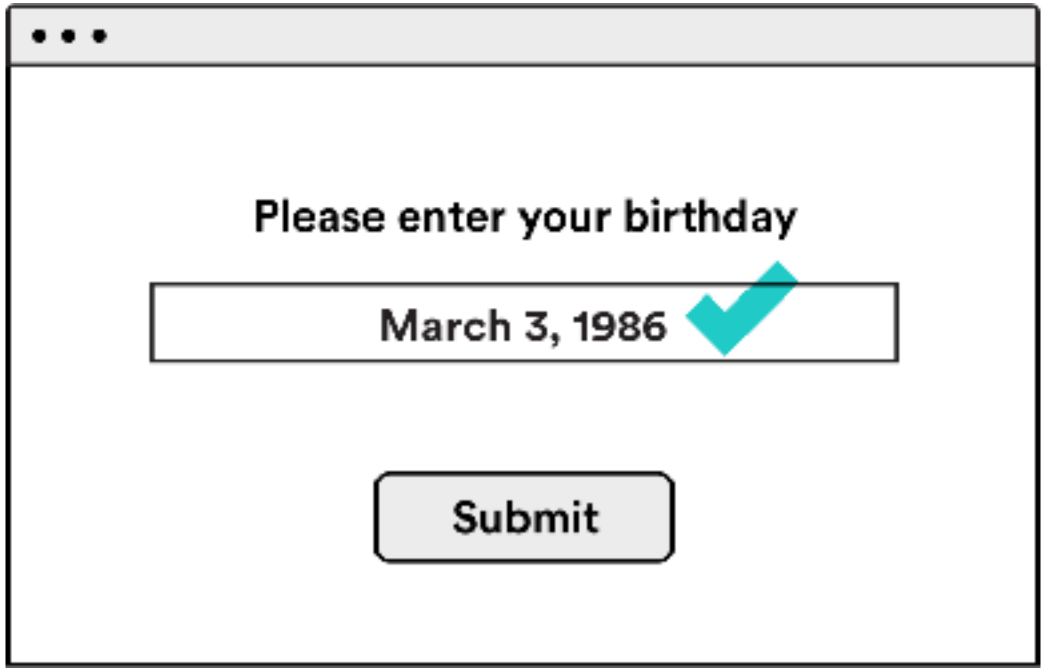

Start
Do You Understand Flow Charts?
No
Ok, See the line labeled "Yes"?
Yes
And, see the line labeled "No"?
Yes
Awesome
Yes
No
No
Awesome
Look closer
Try again

CONDITIONAL LOGIC

In JavaScript (and coding in general) you'll need to make comparisons all the time:



2045



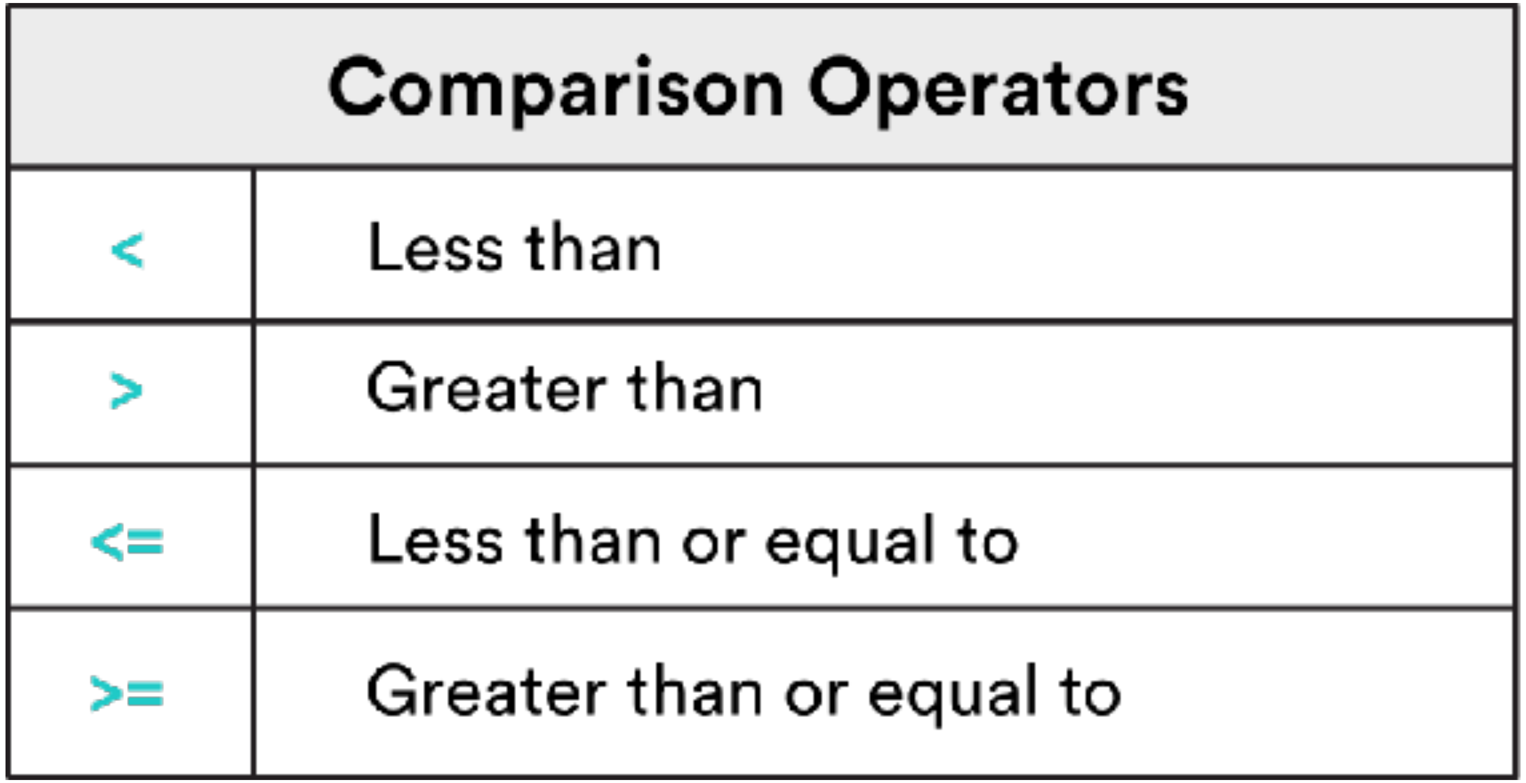
March 3, 1986

Is the year less than or equal to 2016 and more than or equal to 1900? If the answer to this question is "true," then we know the user has entered a valid year, and that he or she was born somewhere between 1900 and 2016.

JS BASICS

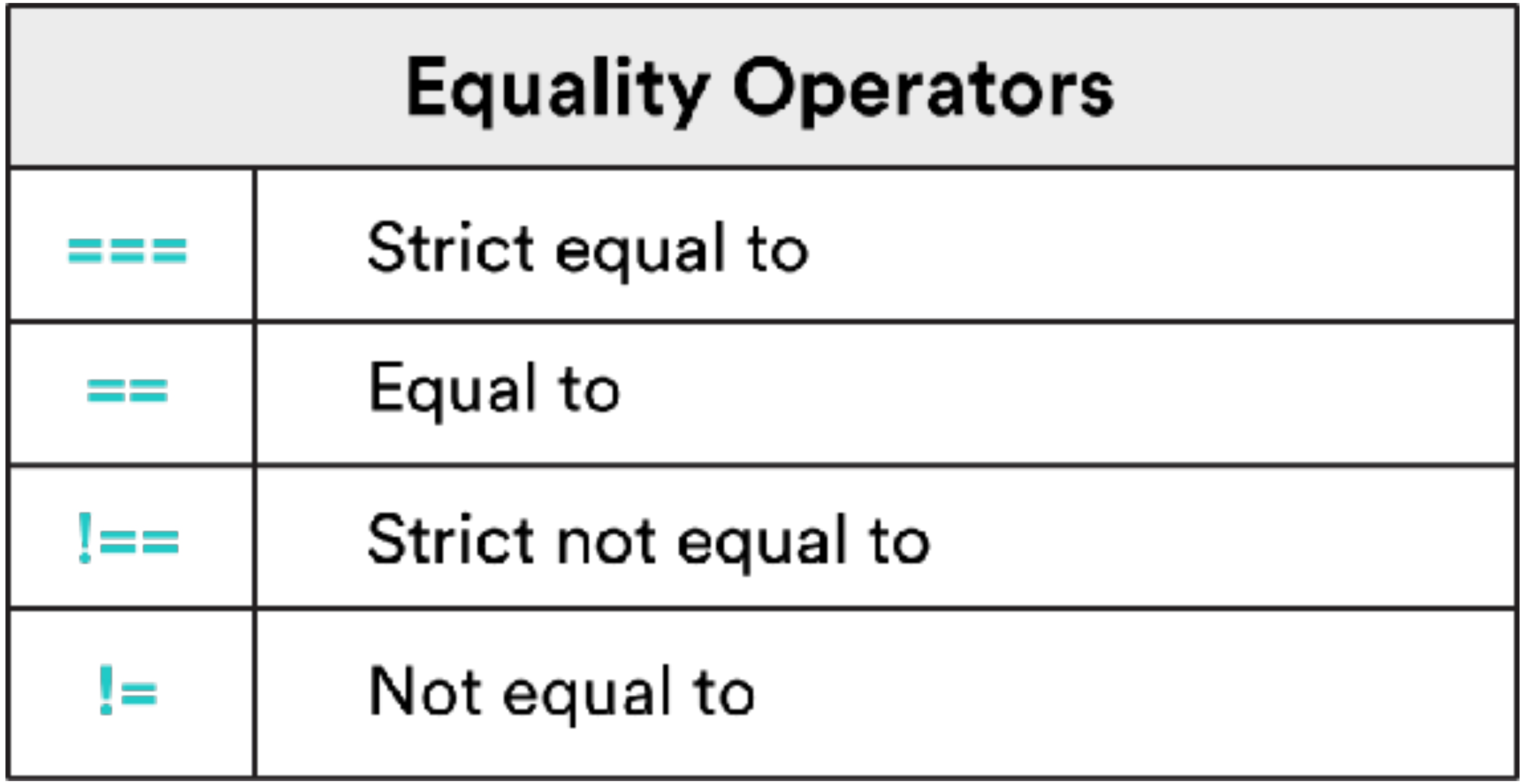
COMPARISON OPERATORS

COMPARISON OPERATORS



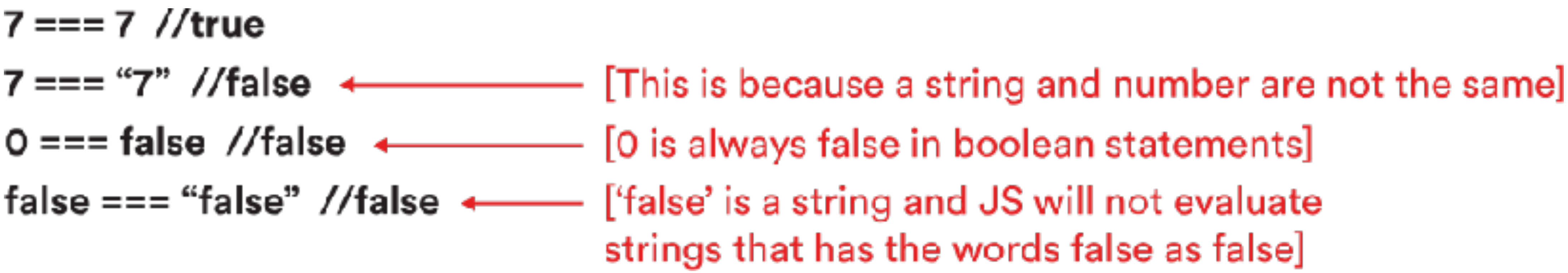
Comparison Operators	
<	Less than
>	Greater than
<=	Less than or equal to
>=	Greater than or equal to

COMPARISON OPERATORS



Equality Operators	
===	Strict equal to
==	Equal to
!==	Strict not equal to
!=	Not equal to

COMPARISON OPERATORS



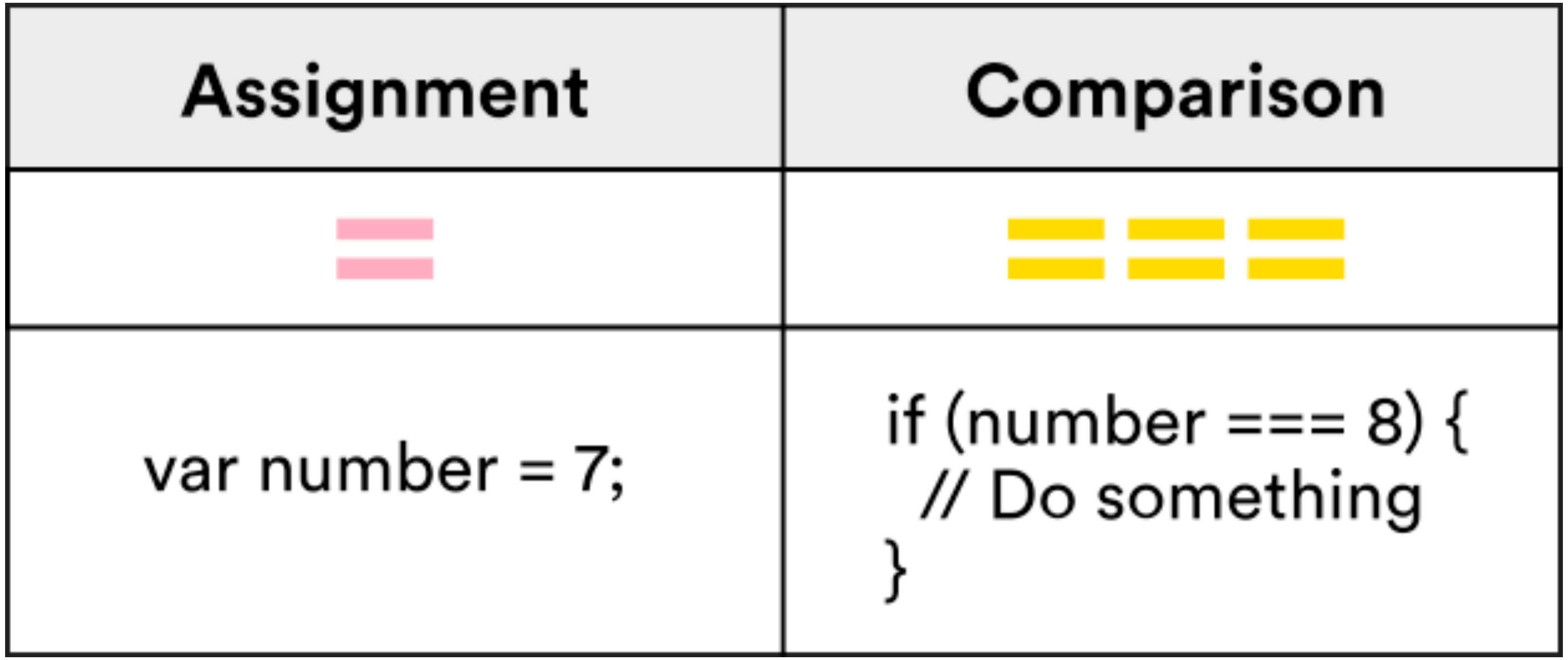
```
7 === 7  //true
7 === “7”  //false
0 === false  //false
false === “false”  //false
```

[This is because a string and number are not the same]

[0 is always false in boolean statements]

[‘false’ is a string and JS will not evaluate
strings that has the words false as false]

ASSIGNMENT VS. COMPARISON — DON'T GET THEM CONFUSED!



Assignment	Comparison
=	===

var number = 7;	if (number === 8) {
	  // Do something
	}

JS BASICS

CONDITIONAL STATEMENTS

JAVASCRIPT — IF STATEMENT

Condition

```
if (answer === 38) {
    // Do something if true
}
```

IF STATEMENTS

```
if (age > 65) {
    $('h1').html("Senior Discount Applied");
}
```

JAVASCRIPT — IF/ELSE STATEMENT

```
if (answer === 38) {
    // Do something if true
} else {
    // Do something if false
}
```

IF STATEMENTS

```
if (age > 65) {
    $('h1').html("Senior Discount Applied");

} else {
    $('h1').html("Sorry, you do not qualify for a discount.");
}
```

JAVASCRIPT — IF/ELSE IF/ELSE

```
if (answer === 38) {
    // Do something if first condition is true
} else if (answer === 30) {
    // Do something second condition is true
} else {
    // Do something if all above conditions are false
}
```

IF STATEMENTS

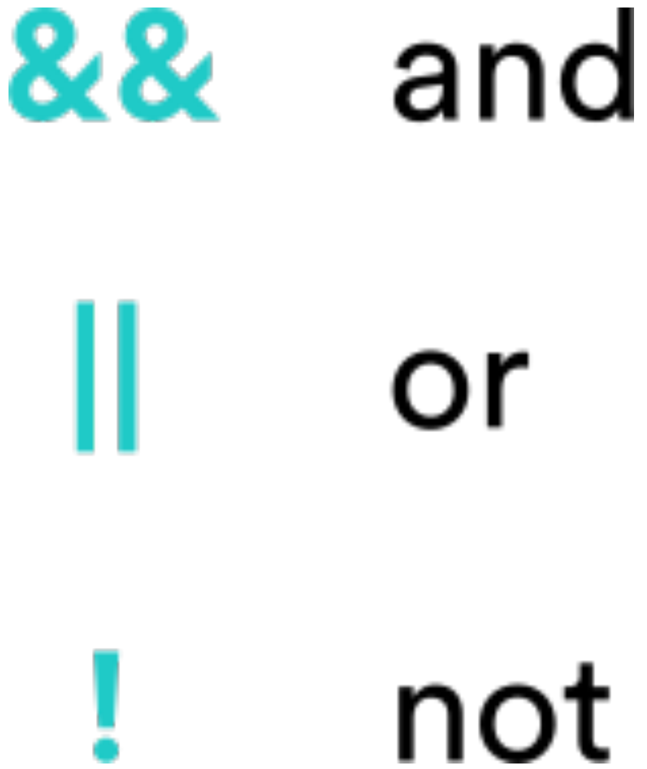
```
if (age > 65) {
    $('h1').html("Senior Discount Applied");

} else if (age < 18) {
    $('h1').html("Student Discount Applied");

} else {
    $('h1').html("Sorry, you don't qualify for a discount");
}
```

LOGICAL OPERATORS

MULTIPLE CONDITIONS



&&	and
||	or
!	not

MULTIPLE CONDITIONS

```
if (name === "GA" && password === "YellowPencil"){
    //Allow access to website
}
```

EXERCISE — CONDITIONALS

EXERCISE

KEY OBJECTIVE

- Practice writing conditionals

TYPE OF EXERCISE

- Individual/paired

EXECUTION

6 min	1. Follow the instructions under "Part 2" in this JS Bin.

LAB

LAB — ICE CREAM

EXERCISE

KEY OBJECTIVE

- Practice writing JS to decide whether or not to buy an ice cream cone.

TYPE OF EXERCISE

- Individual/Partner

EXERCISE

Until 9:20

1. Follow the steps in this JS Bin.

LEARNING OBJECTIVES

- Practice programmatic thinking by writing pseudocode.
- Describe event-based web site behavior and list practical uses of JavaScript on a website's front end.
- Predict DOM output / changes by reading JavaScript code.
- Define variables and identify best cases to use them.
- Describe strings, numbers, and boolean variable types.
- Use comparison operators to evaluate and compare statements.
- Apply conditionals to change the program's control flow

WEEKLY OVERVIEW

WEEK 4	Forms / Intro to JS
WEEK 5	jQuery Basics / jQuery Lab

HOMEWORK

HOMEWORK

Be sure to read the specs on the website.

EXIT TICKETS!