

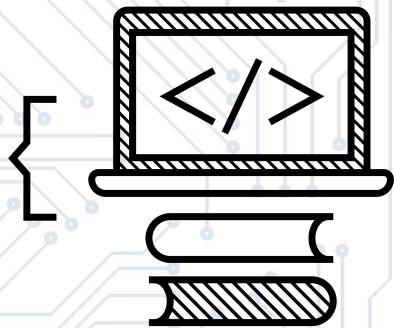


Checkpoint - HTML/CSS/Git

- Check Slack's "general" channel for a link
- Find your name and complete the checkpoint

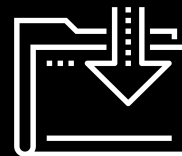
Suggested Time:
10 minutes





JavaScript Juggernauts

Web Development Boot Camp
Lesson 3.3



Today's Class

Objectives

In today's class, we'll cover:



JavaScript Functions



JavaScript Objects



Building Simple JavaScript Applications



Partner Activity:

Loop TV (Array Building)

Suggested Time:
10 minutes



Partner Activity: Array Building

01

Run the program sent to you via Slack.

02

Then, with a partner, fill in the missing comments for each line of code.

03

Make sure both of you can fully explain what each line means.

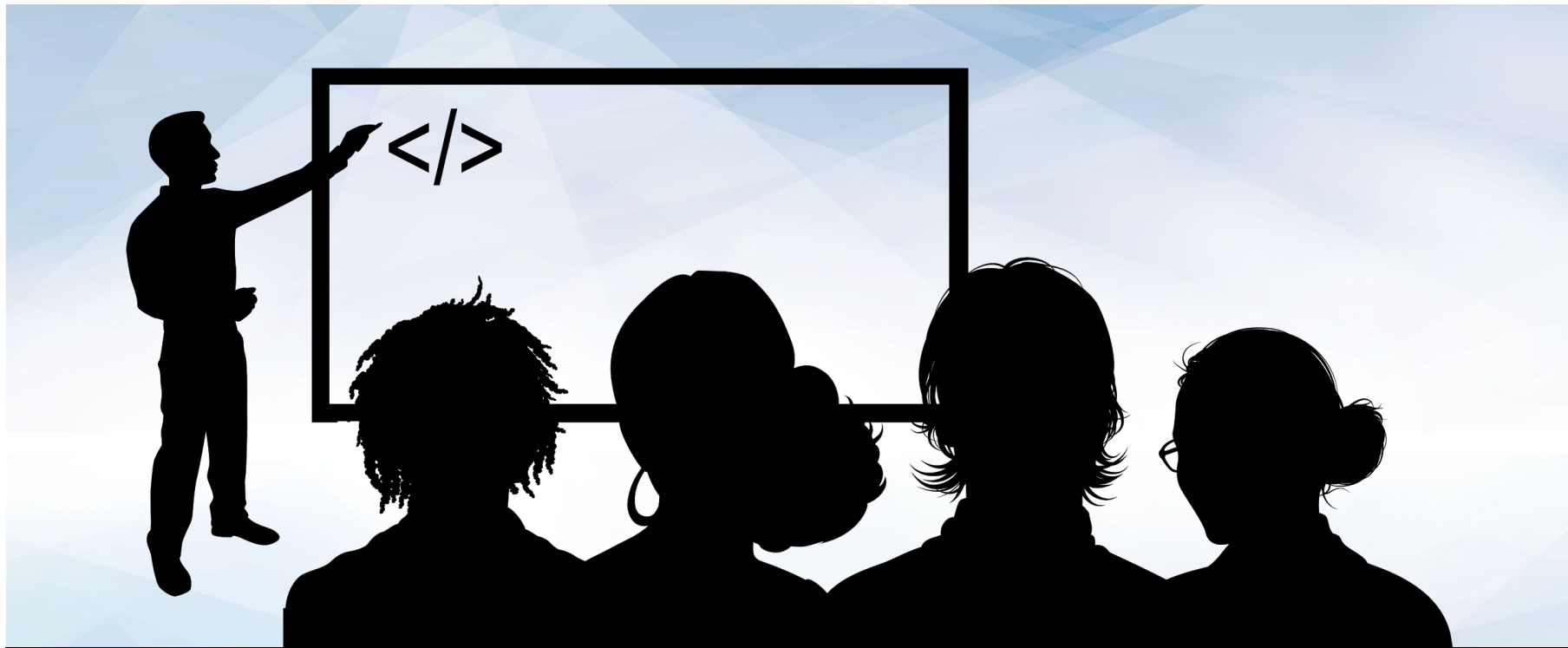
04

Be prepared to share with the class.

Suggested Time: 10 minutes



JavaScript Functions



Instructor Demonstration

Logging: No Functions

Mondo Repetitive

Who wants to maintain this?

What do all of these loops have in common?



```
// For Loop for Brands
for (var i = 0; i < brands.length; i++) {
  console.log(brands[i]);
}
console.log("-----");

// For Loop for Heroes
for (var i = 0; i < heroes.length; i++) {
  console.log(heroes[i]);
}
console.log("-----");

// For Loop for booksOnMyShelf
for (var i = 0; i < booksOnMyShelf.length; i++) {
  console.log(booksOnMyShelf[i]);
}
console.log("-----");

// For Loop for thingsInFrontOfMe
for (var i = 0; i < thingsInFrontOfMe.length; i++) {
  console.log(thingsInFrontOfMe[i]);
}
console.log("-----");

// For Loop for howIFeel
for (var i = 0; i < howIFeel.length; i++) {
  console.log(howIFeel[i]);
}
console.log("-----");
```



Instructor Demonstration

Logging: With Functions

Much Better with Functions!

Squeaky clean code. Minimal repetition.

```
// Here we create a "Function" that allows us to "call" (run) the loop for any array we wish.  
// We pass in an array as a "parameter".  
function consoleInside(arr) {  
  
    // We then loop through the selected array.  
    for (var i = 0; i < arr.length; i++) {  
  
        // Each time we print the value inside the array.  
        console.log(arr[i]);  
    }  
    console.log("-----");  
}
```



Partner Activity:

My First Functions

Suggested Time:
20 minutes



Partner Activity: My First Functions



Working in pairs and using the starter file sent to you via Slack, fill in the missing functions and function calls.



Note: Try to finish all four functions if you can, but don't be distressed if you only get 1 or 2. The important thing is that you get at least one function fully done.



HINT: Look back to the previous example if you need help.

Suggested Time: 20 minutes

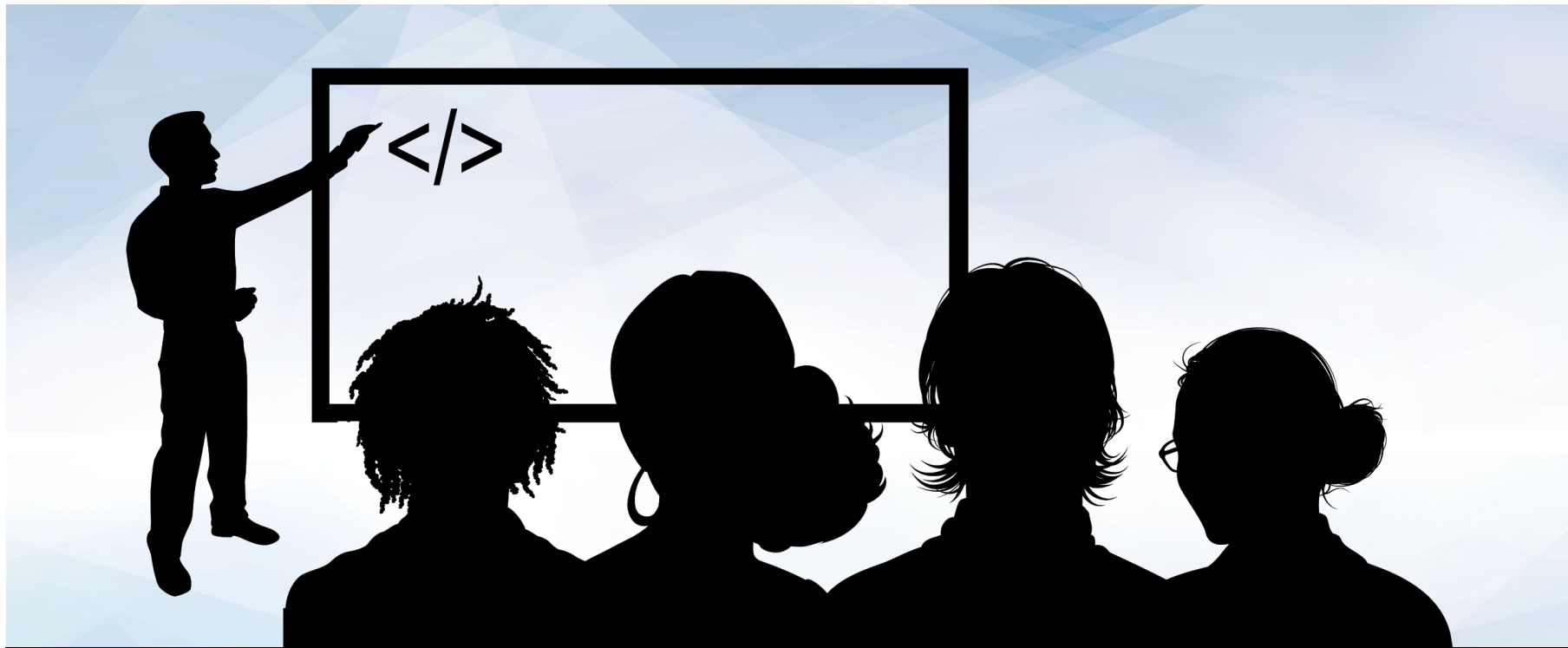


JavaScript Objects



Instructor Demonstration

Good Arrays



Instructor Demonstration

Joan of Arc (Bad Arrays)

Associated Data: Arrays

Relating two separate arrays is not fun.

```
1  // Joan of Arc "properties".
2  var joanOfArcInfoParts = [
3      "Real Name",    (0)
4      "Grew Up Where", (1)
5      "Known For",   (2)
6      "Scars",       (3)
7      "Symbolism"    (4)
8  ];
9
10 // Values for Joan's "properties".
11 var joanOfArcInfoValues = [
12     "Jehanne la Pucelle.", (0)
13     "Domremy, a village in northeastern France.", (1)
14     "Peasant girl, daughter of a farmer, who rose to become Commander of the French army.", (2)
15     "Took an arrow to the shoulder and a crossbow bolt to the thigh while trying to liberate Paris.", (3)
16     "Stands for French unity and nationalism." (4)
17 ];
```



Instructor Demonstration

Gandalf the Grey Objects

Gandalf: The Object

Gandalf's **properties** and **values** are associated in object form, making it easy to recall specific data.

```
11  var gandalf = {
12      "real name": "Gandalf",
13      "age (est)": 11000,
14      "retirementPlan": true,
15      "aliases": [
16          "Greyhame",
17          "Stormcrow",
18          "Mithrandir",
19          "Gandalf the Grey",
20          "Gandalf the White"
21      ]
22  };
23
24  // Object properties can be accessed with "bracket notation"
25  alert("My name is " + gandalf["real name"]);
26
27  // Or with "dot notation" if the property has no spaces
28  if (gandalf.retirementPlan) {
29
30      // Or with a variable that matches the name of the property
31      var ageProperty = "age (est)";
32      var years = gandalf[ageProperty];
33      alert("My 401k has been gathering interest for " + years + " years!");
34  }
35
```

Objects Visualized

This is Gandalf. According to code, Gandalf is an **object**.

var gandalf	=	{
-------------	---	---



}

"real name"	:	"Gandalf"	,
-------------	---	-----------	---

"age (est)"	:	11000	,
-------------	---	-------	---

"hasRetirementPlan"	:	true
---------------------	---	------

Objects Visualized

These are Gandalf's **properties** (like descriptors).

var gandalf	=	{
-------------	---	---



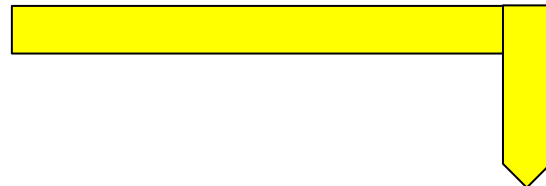
}



"real name"	:	"Gandalf"	,
"age (est)"	:	11000	,
"hasRetirementPlan"	:	true	

Objects Visualized

These are the **values** of Gandalf's properties.



var gandalf	=	{
-------------	---	---



}

"real name"	:	"Gandalf"	,
-------------	---	-----------	---

"age (est)"	:	11000	,
-------------	---	-------	---

"hasRetirementPlan"	:	true
---------------------	---	------

Objects Visualized

Thus: `gandalf["real name"] = "Gandalf"`

<code>var gandalf</code>	<code>=</code>	<code>{</code>
--------------------------	----------------	----------------



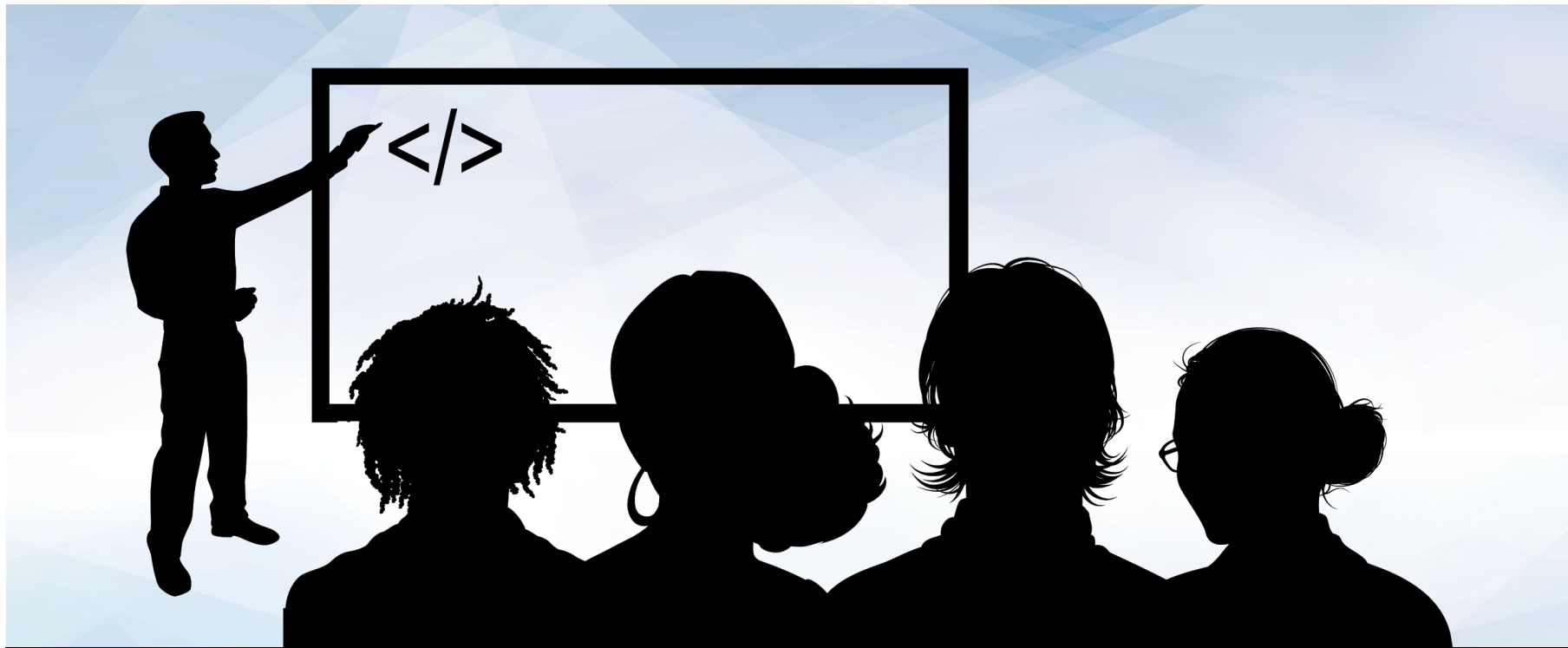
<code>}</code>

<code>"real name"</code>	<code>:</code>	<code>"Gandalf"</code>	<code>,</code>
--------------------------	----------------	------------------------	----------------

<code>"age (est)"</code>	<code>:</code>	<code>11000</code>	<code>,</code>
--------------------------	----------------	--------------------	----------------

<code>"hasRetirementPlan"</code>	<code>:</code>	<code>true</code>
----------------------------------	----------------	-------------------

*** Let's revisit Joan of Arc and update her data into an object



Instructor Demonstration

Gandalf: The Grey Objects (Repeat)



Group Activity (2 people): Basic Objects

Suggested Time:
15 minutes



Group Activity: Basic Objects



With a partner, spend the next few moments studying the code just slacked to you.



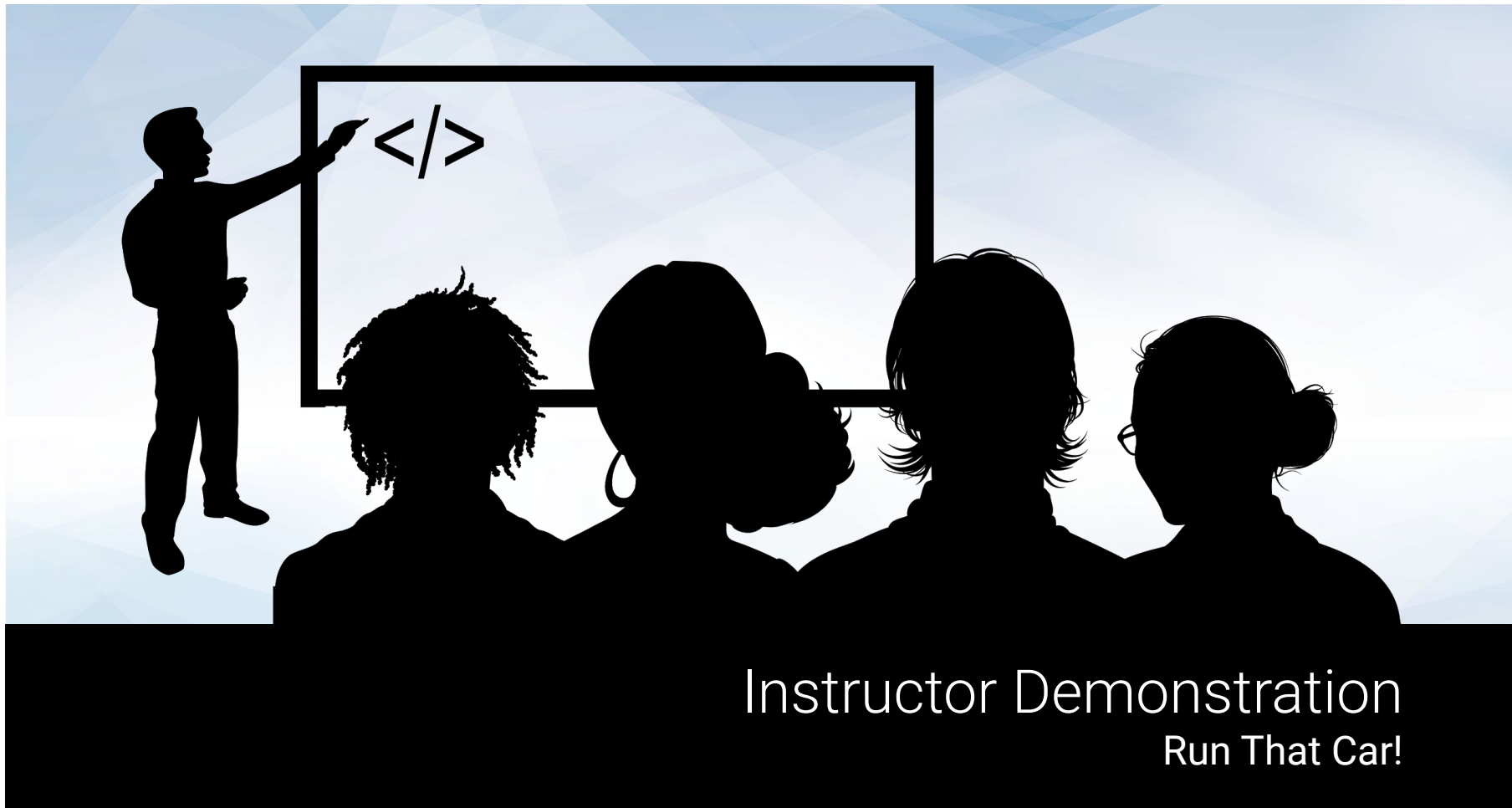
Then, write code below each comment to log the relevant information about the provided `car` object.

Suggested Time: 15 minutes



Take a Break!







Challenge: Run That Car!

Suggested Time:
30 minutes



Challenge: Run That Car!

Using the code from the previous activity as a starting point, create a complete application such that:



Each of the car's methods are assigned to a key.



When the user presses a key, it calls the appropriate function.



These letters also trigger a global function called `reWriteStats()` that logs the car's make, model, color, mileage, and `isWorking` status to the console.



HINT: You will need to use the `document.onkeyup()` function to collect input from the user's keyboard.

Suggested Time: 30 minutes





Instructor Demonstration

Run That Car!



Activity: Scope & Callbacks

Instructions sent via **Slack**

Suggested Time:
20 minutes



Callback Functions

A callback function, also known as a higher-order function, is a **function that is passed to another function as a parameter**, and the callback function is called (or executed) inside the otherFunction.

```
function functionOne(x) { return x; };  
  
function functionTwo(var1) {  
    // some code  
}  
  
functionTwo(functionOne);
```

Let's look at the documentation for **forEach**.

Workbook and Homework



Group Activity: Question Game

Suggested Time:
20 minutes



Group Activity: Question Game

Starting from a blank HTML file:



Create an object with 10 questions. The object should be structured like this:
q1: ["QUESTION", "ANSWER"] **q2:** ["QUESTION", "ANSWER"]



Then create code that will ask the user questions, one by one. The user must answer by hitting **t** (for true) or **f** (for false).



Check the user's answer against the correct answer, and provide them with an alert telling them if they are right or wrong.



Bonus: Keep track of the user's score.



Hint: Don't worry about having DRY code to start with. Just focus on getting working code first.

Suggested Time: 20 minutes





Questions?