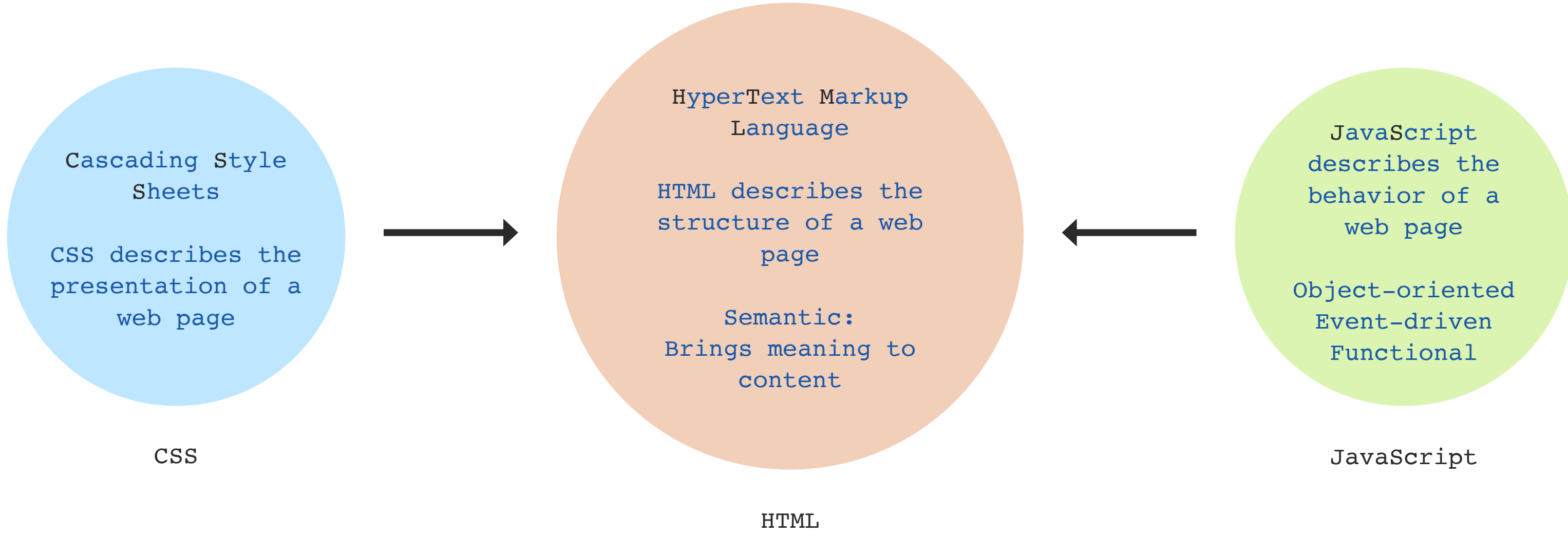


Introduction to Programming

Browser-native languages



HTML, CSS, and JavaScript make up the three core technologies of the World Wide Web. These languages are built into the browser and required nothing additional to function. Websites using technology that is native to the browser can be faster, efficient, and more accessible.

What can JavaScript be used for?

Loading new web page content without reloading the page.

Animating and/or manipulating the appearance of HTML elements, such as fading objects in and out, resizing, or moving them.

Controlling the playback of streaming media.

Generating pop-ups or temporary states.

Manipulating multiple elements at once and instructing the computer to complete tasks that would be tedious by hand.

Much more...



JavaScript

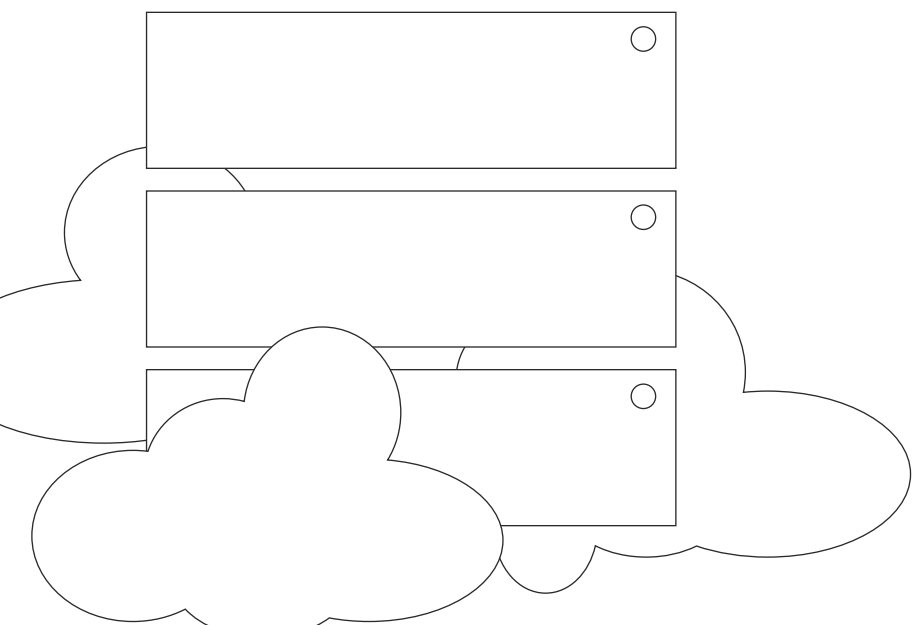
Where does JavaScript run?

Website files:

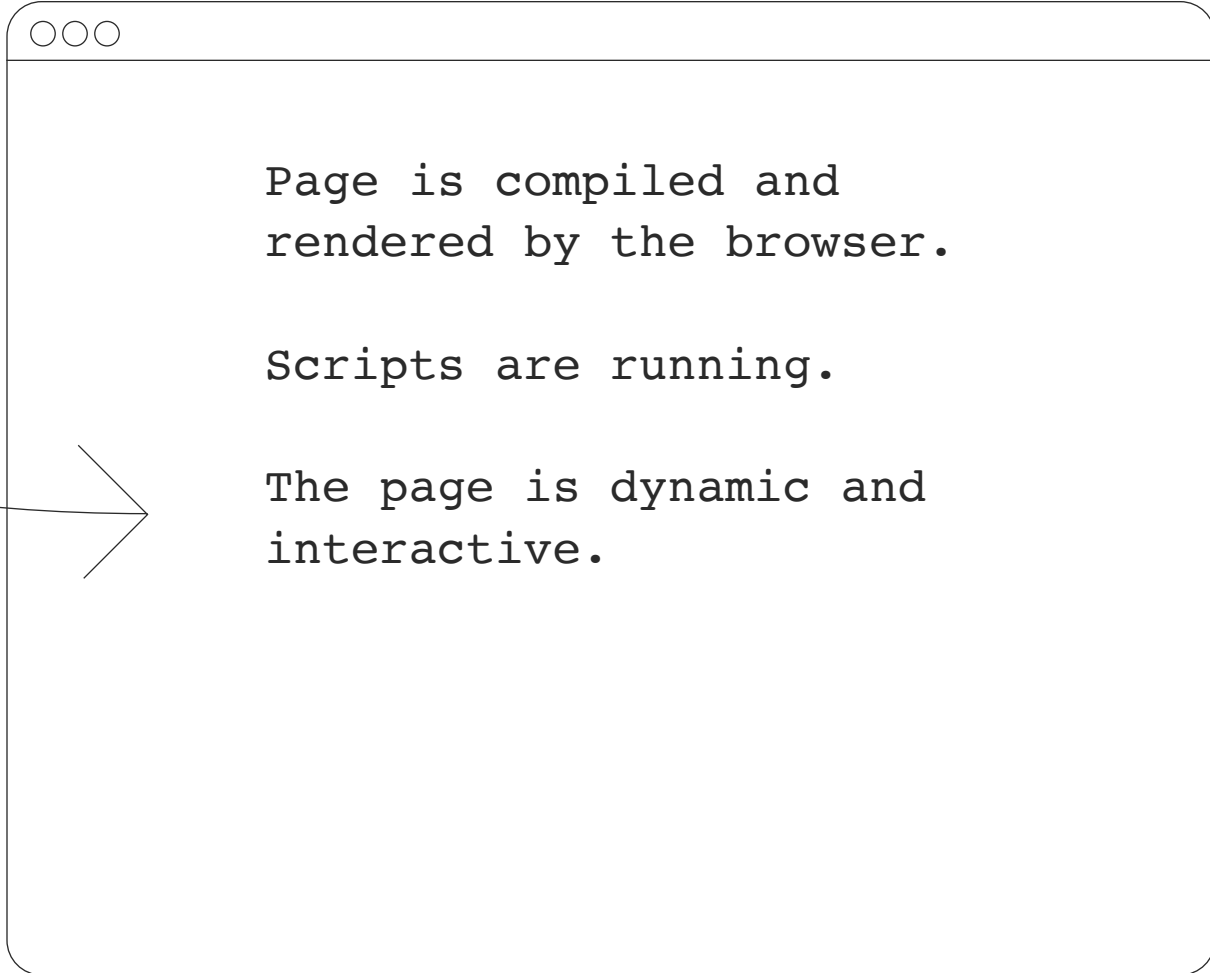
```
public_html/  
|- index.html  
|- style.css  
|- scripts.js
```

The server
is static
and unchanged

SERVER

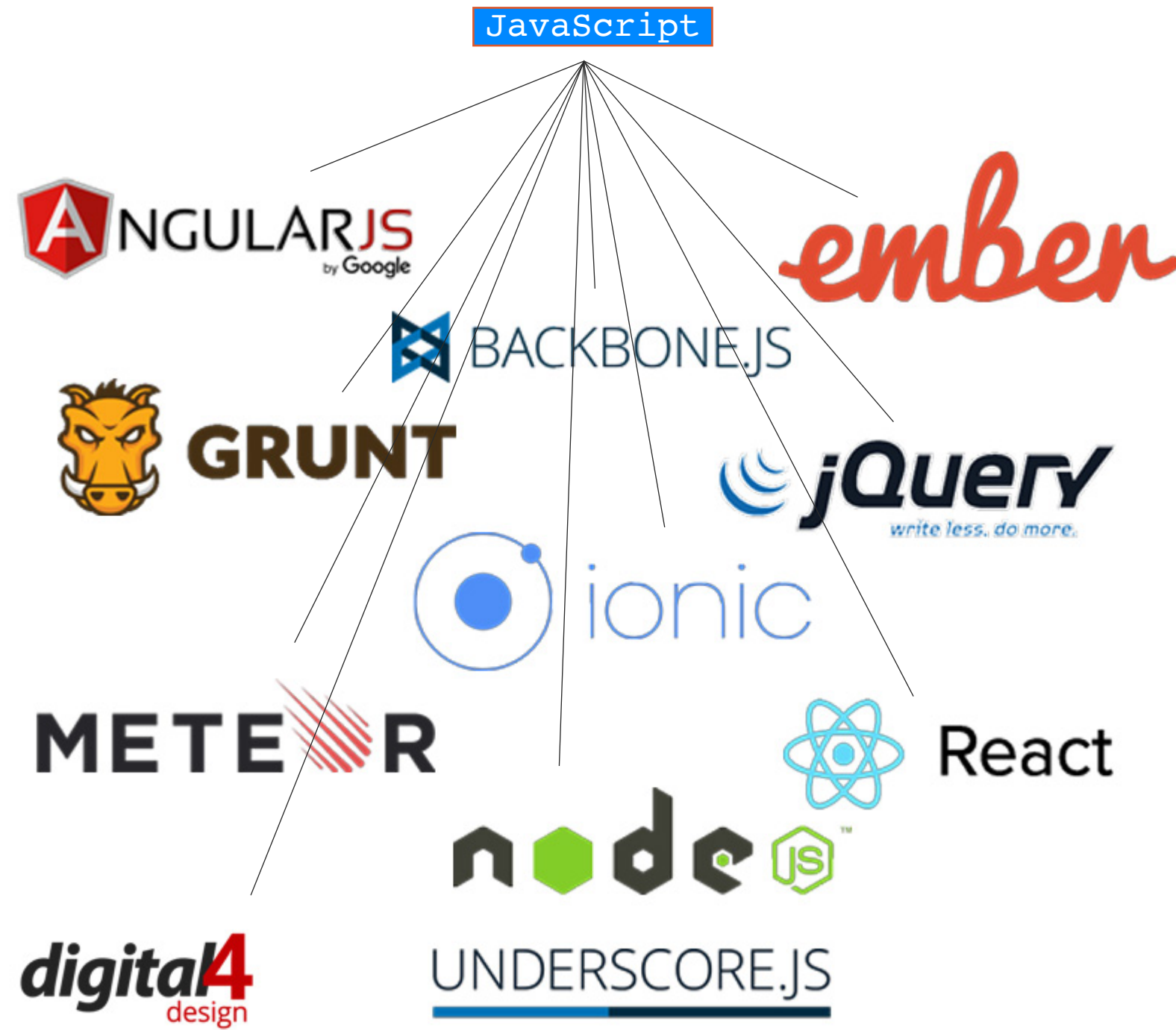


BROWSER



JavaScript is a client-side scripting language, meaning its functions run inside the users browser not on the server.

What are JavaScript libraries and frameworks?



JavaScript is like the Latin of the modern web. There are many extensions and evolutions of the original language, but all of them depend on JavaScript at the root. You may have heard people refer to plain old JavaScript as "Vanilla JS". Only Vanilla JavaScript has the ability to run in the browser without installing additional files on the server.

JavaScript !== Java



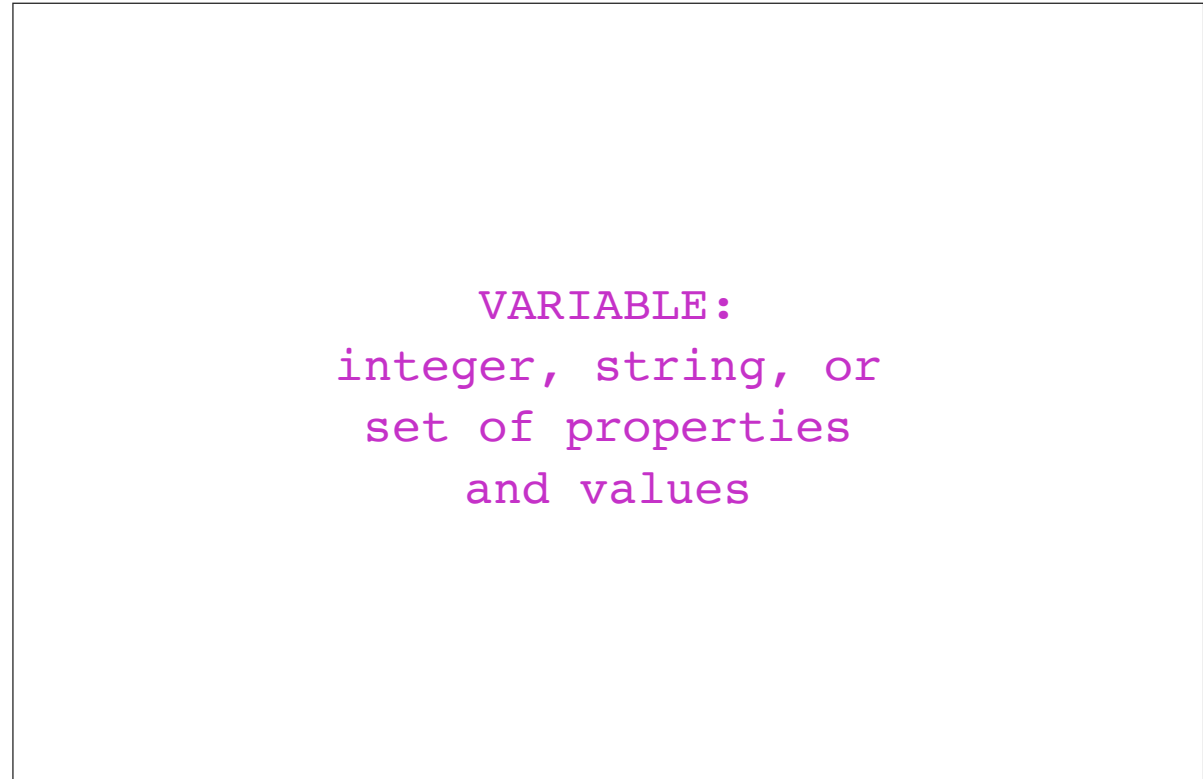
Keep in mind that JavaScript has nothing to do with the programming language Java. "Java is to JavaScript as Ham is to Hamster". When searching for resources online, make sure to include the full term 'javascript' in your search.

Object-Oriented Programming (OOP)

METHODS

METHODS

```
function turnRed() {
  rectangle.backgroundColor = "red";
}
```



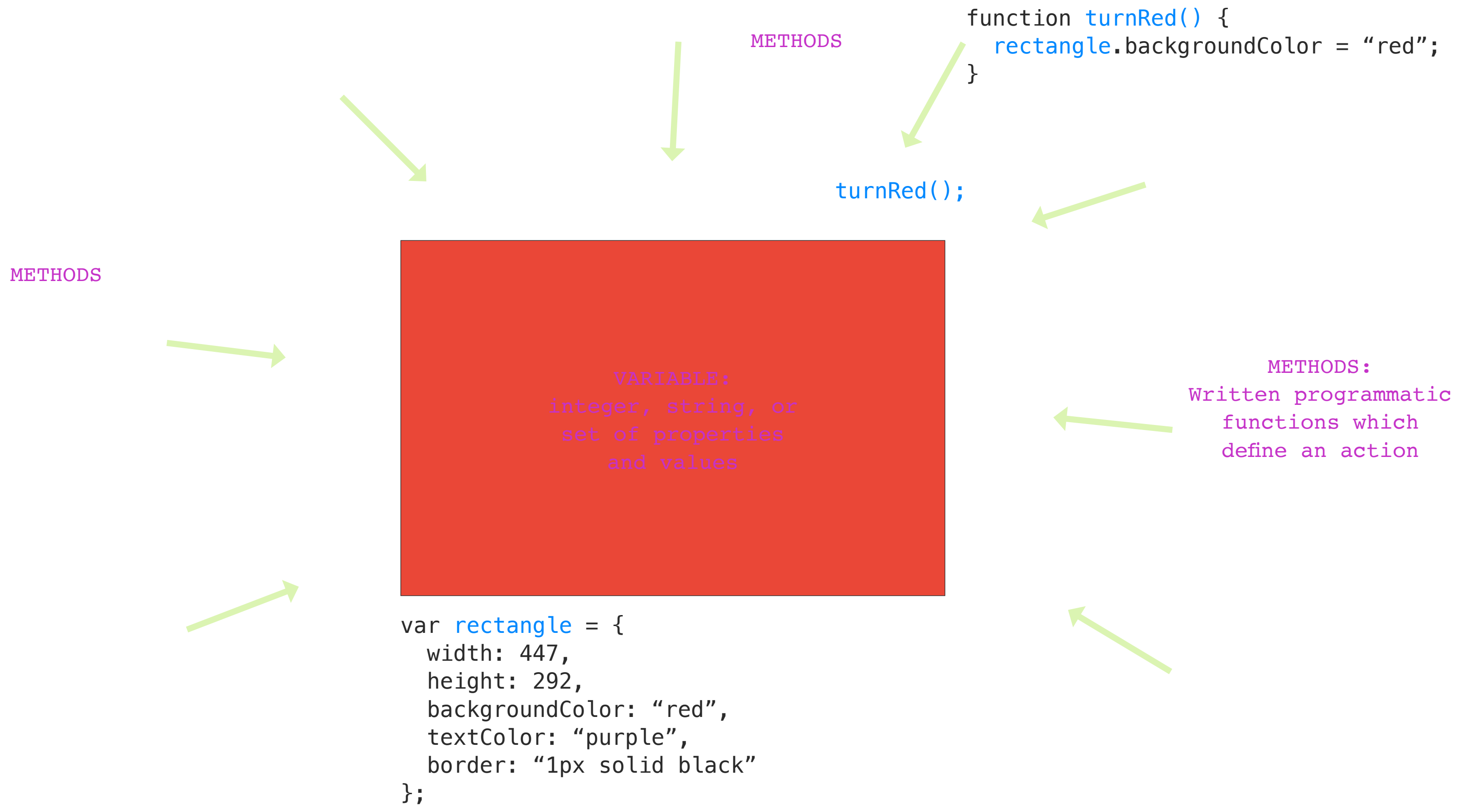
METHODS:
Written programmatic functions which define an action

```
var rectangle = {
  width: 447,
  height: 292,
  backgroundColor: "white",
  textColor: "purple",
  border: "1px solid black"
};
```

JavaScript is an object-oriented programming language; a way of writing computer programs using "objects" to stand for data and methods.

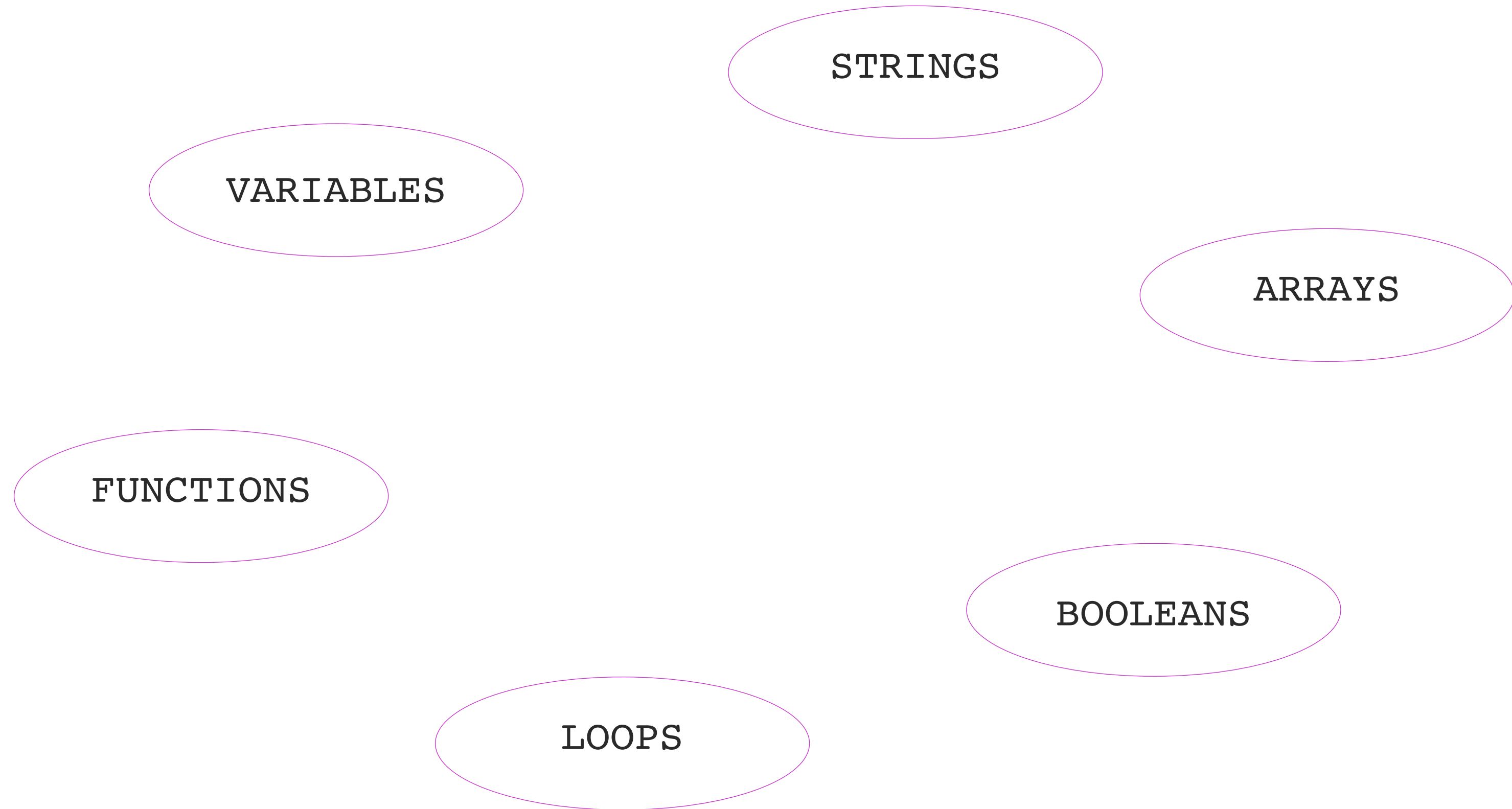
- Variable objects store information formatted in a small number of built-in data types like integers and alphanumeric characters.
- Method objects also known as functions, are lists of instructions telling the computer to take input, do some calculations or change data, and return the output to the user.

Object-Oriented Programming (OOP)

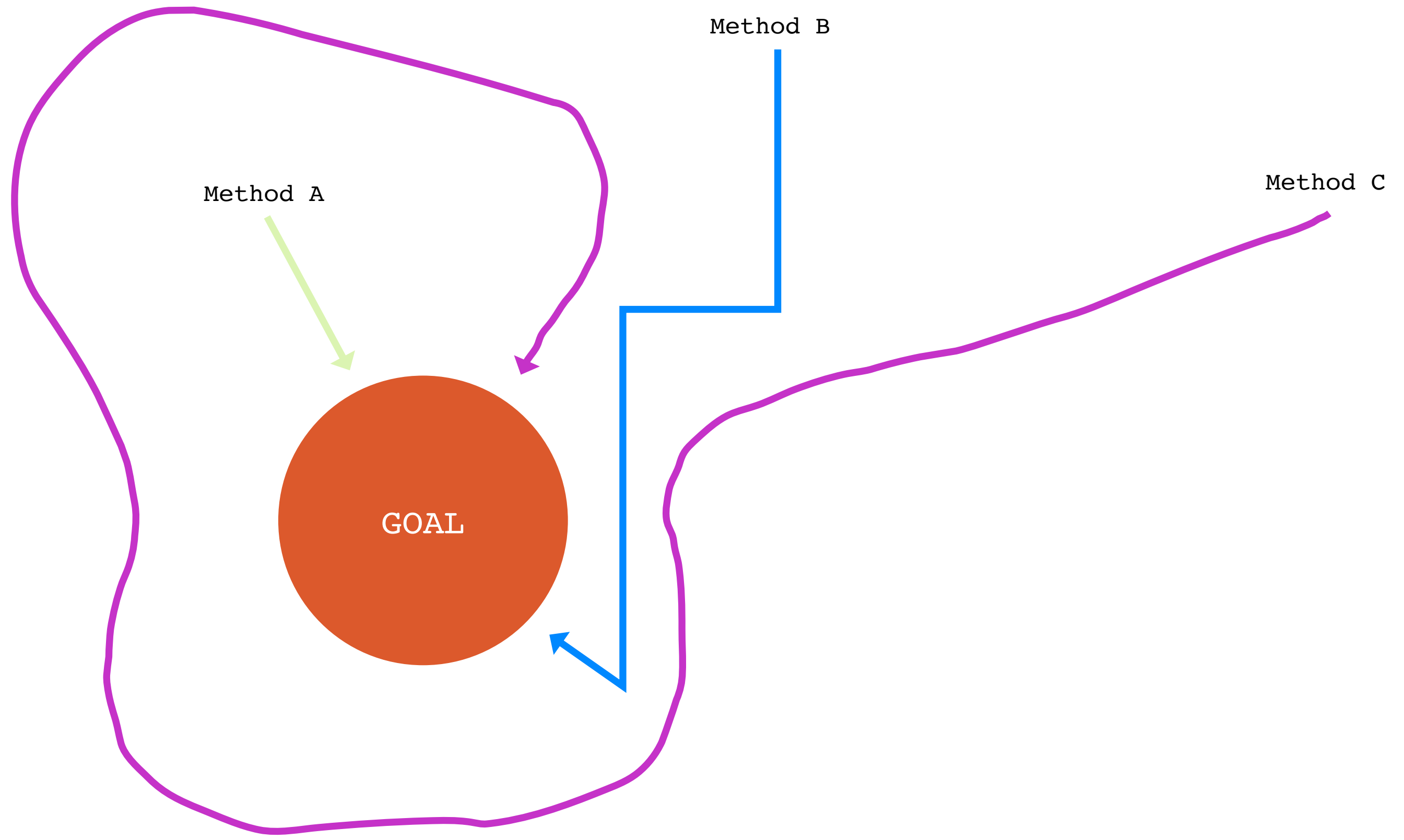


JavaScript is an object-oriented programming language; a way of writing computer programs using "objects" to stand for data and methods.

- Variable objects store information formatted in a small number of built-in data types like integers and alphanumeric characters.
- Method objects also known as functions, are lists of instructions telling the computer to take input, do some calculations or change data, and return the output to the user.



By studying JavaScript you will also get to know the foundations of all object-oriented programming. The knowledge of these concepts can be transferred to proficiency in other programming languages and increase your overall understanding of how to communicate with computers.



There's more than one way to do it (TMTOWTDI or TIMTOWTDI, pronounced Tim Toady) is a Perl programming motto. The language was designed with this idea in mind, in that it "doesn't try to tell the programmer how to program."

There are many different programmable routes all leading to the same perceived result. There are not necessarily right or wrong answers, but there is more and less efficient code.

What is up?

What's up?

Wassup?

Sup?

“I want my whole website to turn black when the light switch button is clicked.”

If you can speak an action in plain English, you can likely translate that behavior into JavaScript. When approaching a problem, try to put the desired outcome into words, then re-write it in less specific terms. Using more general language will help bridge the gap between words and code and also give you a good starting place for search terms if you need to seek answers online.

“I want my whole website to turn black when the light switch button is clicked.”

OR

“Change ‘body’ background color on ‘div’ click.”

If you can speak an action in plain English, you can likely translate that behavior into JavaScript. When approaching a problem, try to put the desired outcome into words, then re-write it in less specific terms. Using more general language will help bridge the gap between words and code and also give you a good starting place for search terms if you need to seek answers online.

“I want my whole website to turn black when the light switch button is clicked.”

OR

“Change ‘body’ background color on ‘div’ click.”

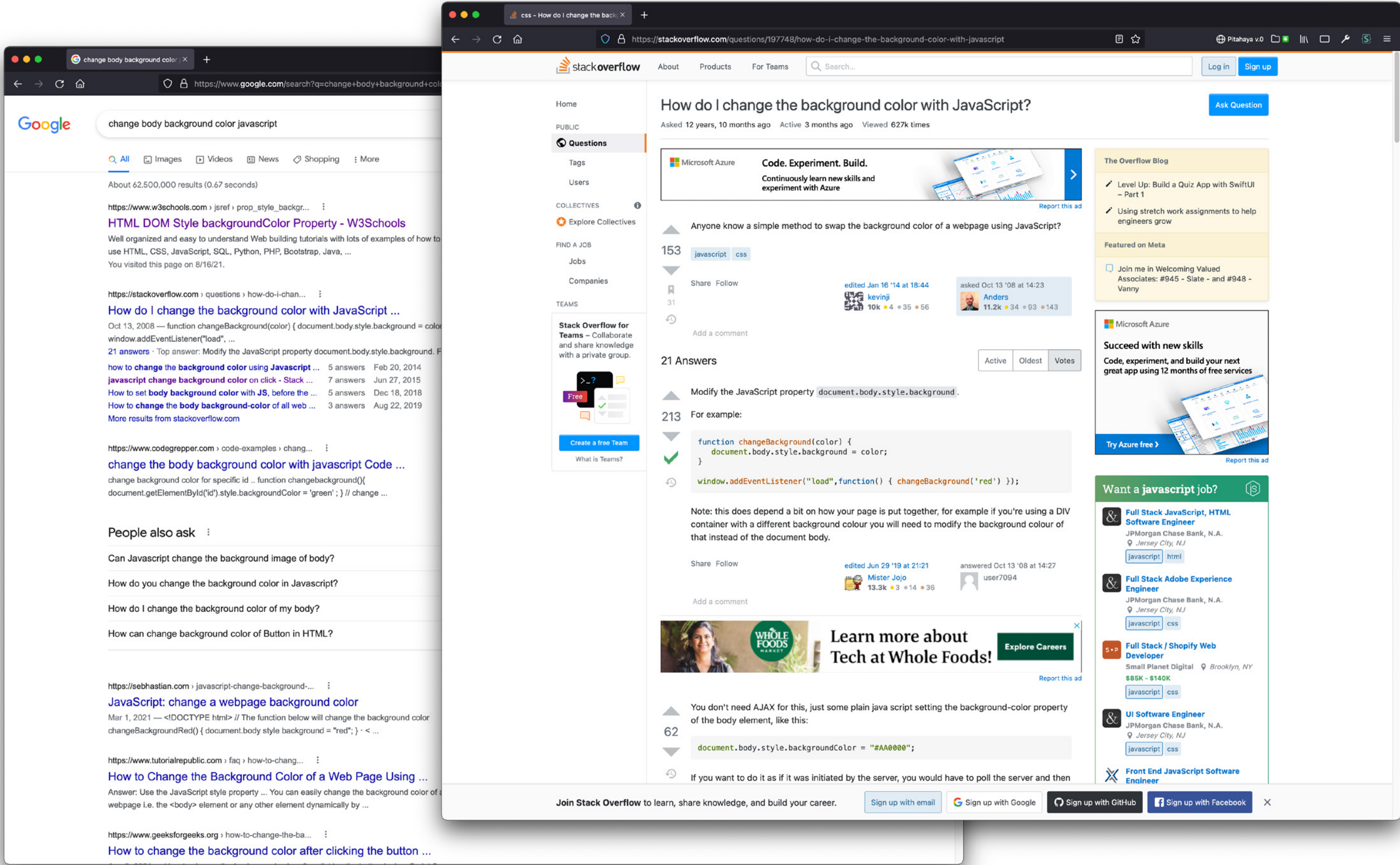
OR

```
var lightSwitch = document.querySelector('div#light-switch');  
var body = document.body;
```

```
lightSwitch.addEventListener('click', function() {  
  body.style.backgroundColor = 'black';  
})
```

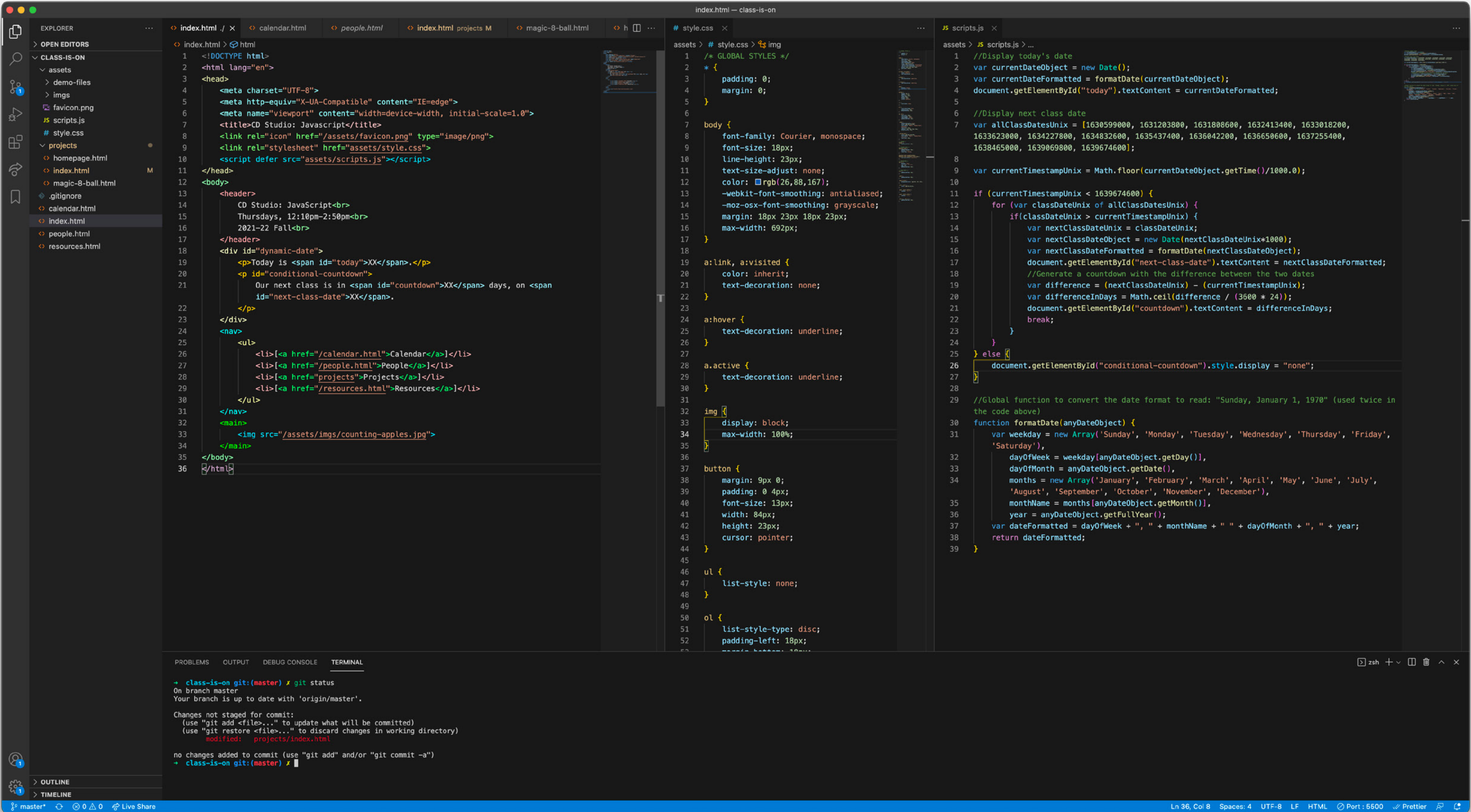
If you can speak an action in plain English, you can likely translate that behavior into JavaScript. When approaching a problem, try to put the desired outcome into words, then re-write it in less specific terms. Using more general language will help bridge the gap between words and code and also give you a good starting place for search terms if you need to seek answers online.

Searching for answers



Programmers of all levels have to look things up. Learning how to search and process the results is just as important as learning the code itself. Using global terms will lead to universal answers for your specific problem. Don't be afraid to copy what you find, but be mindful of how these snippets are functioning so you can integrate them successfully into your own code.

Code environment



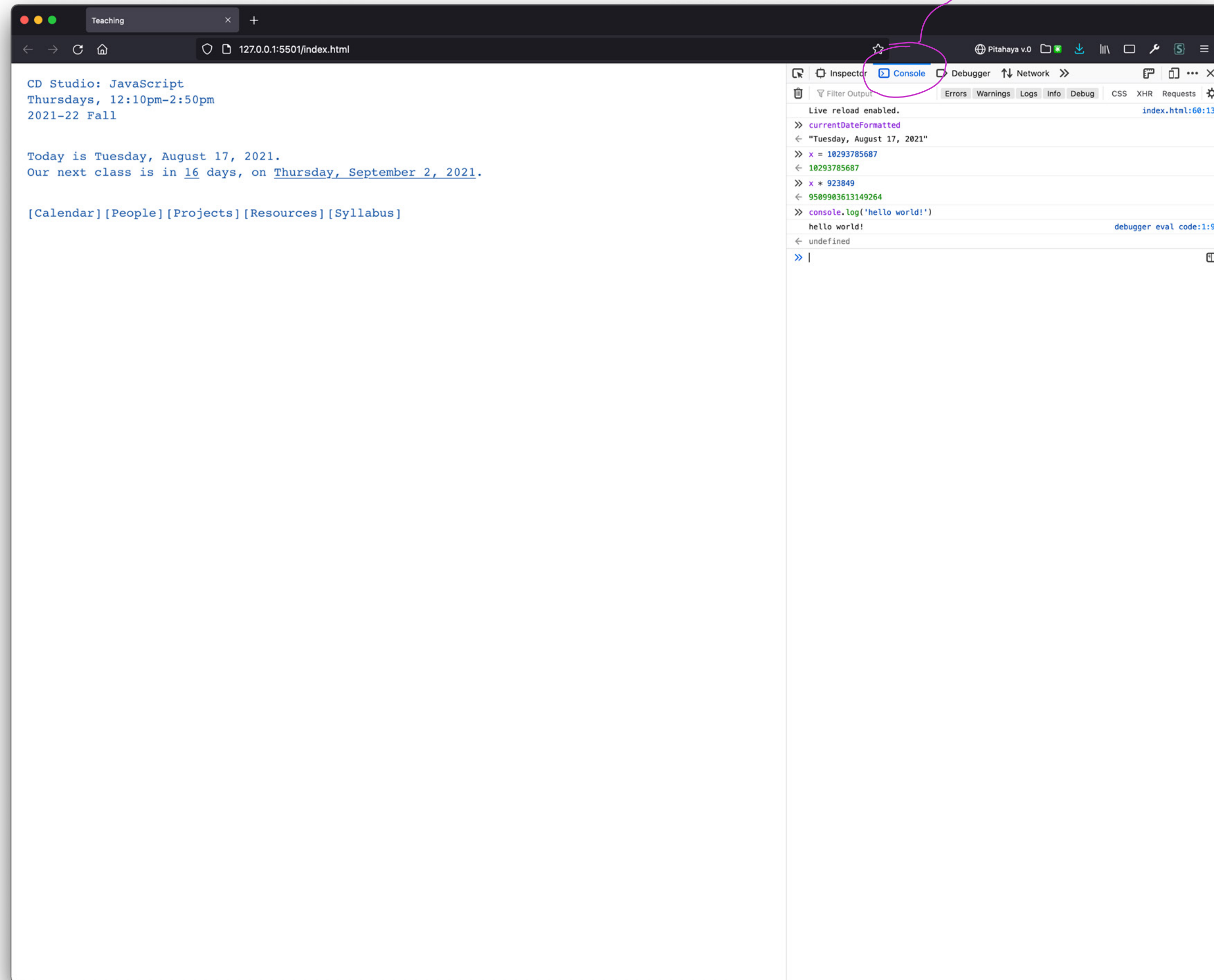
HTML

CSS

Javascript

Split your window into 3 columns so you can see each type of code at once.

Browser tools and console



Console tab

The console is like a powerful calculator for JavaScript built into the browser. Errors will appear here to assist with debugging, and developers often code in log messages to the console to make sure that their JavaScript is working as expected.