

PART I

FUNDAMENTALS



1

WHAT IS JAVASCRIPT?

Computers are incredibly powerful machines, capable of performing amazing feats like playing competitive chess, serving thousands of web pages, or making millions of complex calculations in less than a few seconds. But deep down, computers are actually pretty dumb. Computers can *only* do exactly what we humans tell them to do. We tell computers how to behave using computer programs, which are just sets of instructions for the computers to follow. Without programs, computers can't do anything at all!

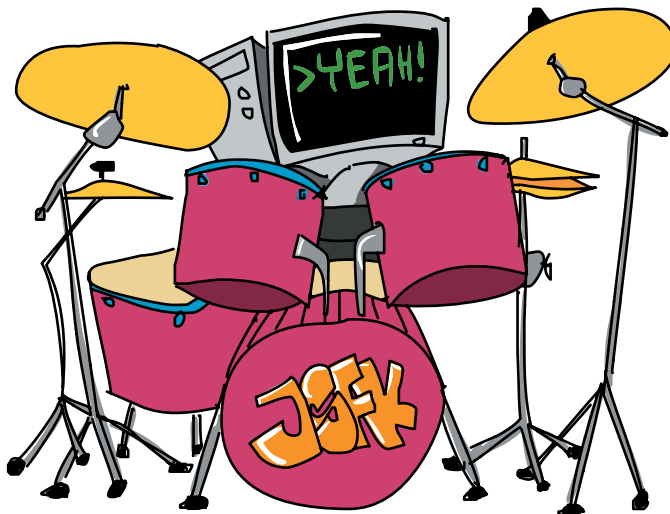
MEET JAVASCRIPT

Even worse, computers can't understand English or any other spoken language. Computer programs are written in a *programming language* like JavaScript. You might not have heard of JavaScript before, but you've certainly used it. The JavaScript programming language is used to write programs that run in web pages. JavaScript can control how a web page looks or make the page respond when a viewer clicks a button or moves the mouse.

Sites like Gmail, Facebook, and Twitter use JavaScript to make it easier to send email, post comments, or browse websites. For example, when you're on Twitter reading tweets from @nostarch and you see more tweets at the bottom of the page as you scroll down, that's JavaScript in action.

You only have to visit a couple of websites to see why JavaScript is so exciting.

- JavaScript lets you play music and create amazing visual effects. For example, you can fly through an interactive music video created by HelloEnjoy for Ellie Goulding's song "Lights" (<http://lights.helloenjoy.com/>), as shown in Figure 1-1.
- JavaScript lets you build tools for others to make their own art. Patatap (<http://www.patatap.com/>) is a kind of virtual "drum machine" that creates all kinds of cool noises—and cool animations to go along with them—as shown in Figure 1-2.



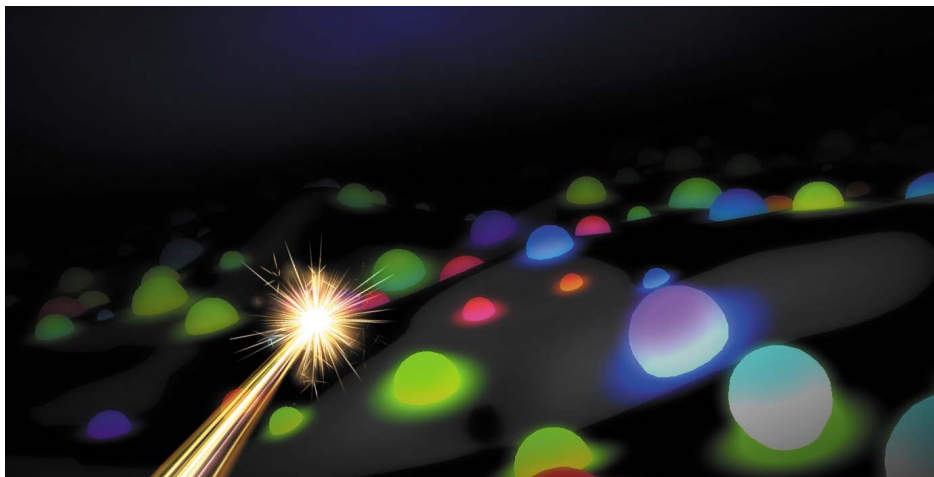


Figure 1-1: You control the flashing cursor in HelloEnjoy's "Lights" music video.

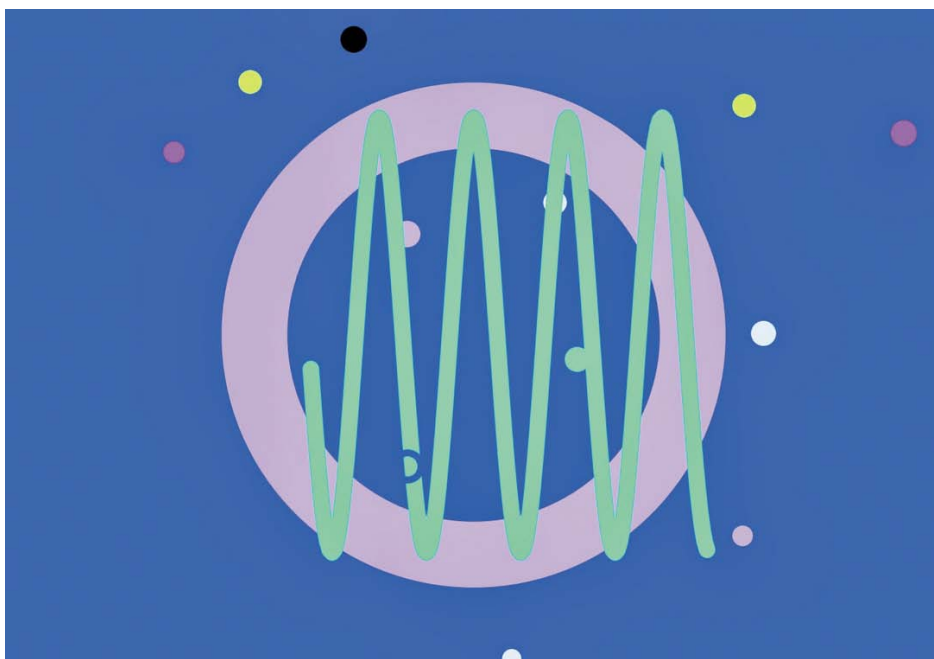


Figure 1-2: When you visit Patatap, try pressing a bunch of keys to make different noises!

- JavaScript lets you play fun games. *CubeSlam* (<https://www.cubeslam.com/>) is a 3D re-creation of the classic game Pong, which looks a little like air hockey. You can play against one of your friends or a computer-generated bear, as shown in Figure 1-3.



Figure 1-3: The CubeSlam game is programmed entirely in JavaScript!

WHY LEARN JAVASCRIPT?

JavaScript isn't the only programming language out there—in fact, there are literally hundreds of programming languages. But there are many reasons to learn JavaScript. For one, it's a lot easier (and more fun) to learn than many other programming languages. But perhaps best of all, in order to write and run JavaScript programs, all you need is a web browser like Internet Explorer, Mozilla Firefox, or Google Chrome. Every web browser comes with a JavaScript *interpreter* that understands how to read JavaScript programs.

Once you've written a JavaScript program, you can send people a link to it, and they can run it in a web browser on their computer, too! (See "Sharing Your Code Using JSFiddle" on page 297.)

WRITING SOME JAVASCRIPT

Let's write a bit of simple JavaScript in Google Chrome (<http://www.google.com/chrome/>). Install Chrome on your computer (if it's not already installed), and then open it and type **about:blank** in the address bar. Now press ENTER and you'll see a blank page, like the one in Figure 1-4.

We'll begin by coding in Chrome's JavaScript console, which is a secret way programmers can test out short JavaScript programs. On Microsoft Windows or Linux, hold down the CTRL and SHIFT keys and press J. On Mac OS, hold down the COMMAND and OPTION keys and press J.

If you've done everything correctly, you should see a blank web page and, beneath that, a blinking cursor (|) next to a right angle bracket (>), as shown in Figure 1-4. That's where you'll write JavaScript!

NOTE

The Chrome console will color your code text; for example, the text you input will be blue, and output will be colored based on its type. In this book, we'll use similar colors for our code text wherever we're using the console.

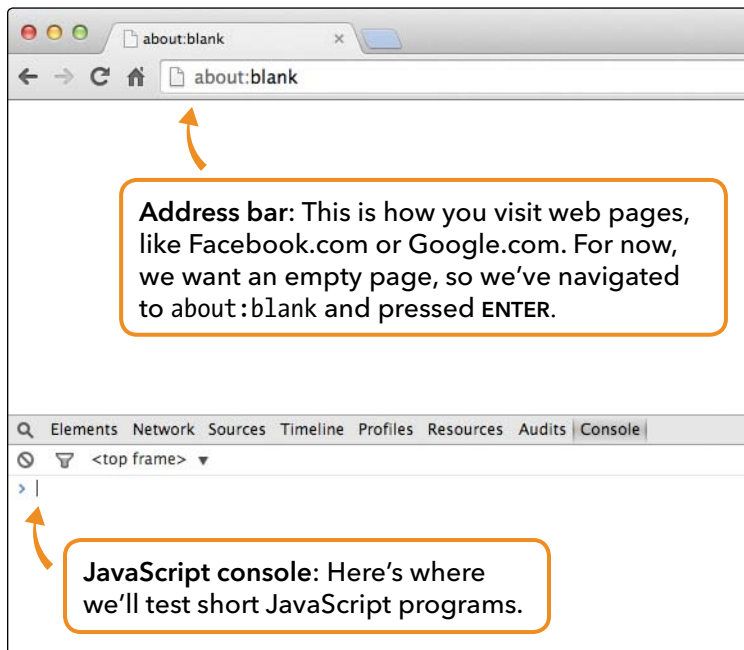


Figure 1-4: Google Chrome's JavaScript console

When you enter code at the cursor and press ENTER, JavaScript should run, or *execute*, your code and display the result (if any) on the next line. For example, type this into the console:

```
3 + 4;
```

Now press ENTER. JavaScript should output the answer (7) to this simple bit of addition on the following line:

```
3 + 4;  
7
```

Well, that's easy enough. But isn't JavaScript more than a glorified calculator? Let's try something else.

THE STRUCTURE OF A JAVASCRIPT PROGRAM

Let's create something a bit sillier—a JavaScript program to print a series of cat faces that look like this:

```
=^.^=
```

Unlike our addition program, this JavaScript program will take up multiple lines. To type the program into the console, you'll have to add new lines by pressing SHIFT-ENTER at the end of each line. (If you just press ENTER, Chrome will try to execute what you've written, and the program won't work as expected. I warned you that computers were dumb!)

Type this into your browser console:

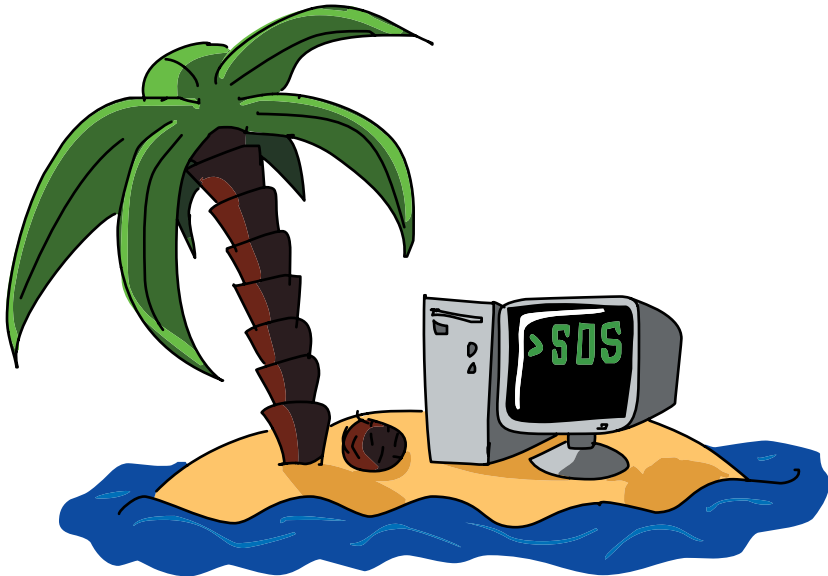
```
// Draw as many cats as you want!  
var drawCats = function (howManyTimes) {  
  for (var i = 0; i < howManyTimes; i++) {  
    console.log(i + " =^.^=");  
  }  
};  
  
drawCats(10); // You can put any number here instead of 10.
```



At the very end, press ENTER instead of SHIFT-ENTER. When you do that, you should see the following output:

```
0 =^.^=  
1 =^.^=  
2 =^.^=  
3 =^.^=  
4 =^.^=  
5 =^.^=  
6 =^.^=  
7 =^.^=  
8 =^.^=  
9 =^.^=
```

If you made any typos, your output might look very different or you might get an error. That's what I mean when I say computers are dumb—even a simple piece of code must be perfect for a computer to understand what you want it to do!



I won't go through exactly how this code *works* for now (we'll return to this program in Chapter 8), but let's look at some of the features of this program and of JavaScript programs in general.

SYNTAX

Our program includes lots of symbols, including parentheses `()`, semicolons `;`, curly brackets `{}`, plus signs `+`, and a few words that might seem mysterious at first (like `var` and `console.log`). These are all part of JavaScript's *syntax*—that is, JavaScript's rules for how to combine symbols and words to create working programs.

When you're learning a new programming language, one of the trickiest parts is getting used to the rules for how to write different kinds of instructions to the computer. When you're first starting out, it's easy to forget when to include parentheses, or to mix up the order in which you need to include certain values. But as you practice, you'll start to get the hang of it.

In this book, we'll go slow and steady, introducing new syntax little by little so that you can build increasingly powerful programs.

COMMENTS

The first line in our `cats` program is this:

```
// Draw as many cats as you want!
```

This is called a *comment*. Programmers use comments to make it easier for other programmers to read and understand their code. The computer ignores comments completely. Comments in JavaScript start with two forward slashes (`//`). Everything following the slashes (on the same line) is ignored by the JavaScript interpreter, so the comments don't have any effect on how a program is executed—they are just there to provide a description.

In the code in this book, you'll see comments that describe what's happening in the code. As you write your own code, add your own comments. Then when you look at your code later, your comments will remind you how the code works and what's happening in each step.

There's another code comment on the last line of our program. Remember, everything after that `//` isn't run by the computer!

```
drawCats(10); // You can put any number here instead of 10.
```

Code comments can be on their own line, or they can come after your code. If you put the `//` at the front, like this:

```
// drawCats(10);
```

... nothing will happen! Chrome sees the whole line as a comment, even if it's JavaScript.

Once you start reading JavaScript code out in the wild world, you'll also see comments that look like this:

```
/*  
Draw as many cats  
as you want!  
*/
```

This is a different style of commenting, which is typically used for comments that are longer than one line. But it does the same thing: everything between the `/*` and the `*/` is a comment that the computer won't run.

WHAT YOU LEARNED

In this chapter, you learned a bit about what JavaScript is and what it can be used for. You also learned how to run JavaScript using the Google Chrome browser and tried out a sample program. All of the code examples in this book, unless I say otherwise, can (and should!) be used in Chrome's JavaScript console. Don't just read the code—try typing things out! It's the only way to learn to program.

In the next chapter, you'll start learning the fundamentals of JavaScript, beginning with the three basic types of information you can work with: numbers, strings, and Booleans.



