

# Intro to Web Development A Guided Tutorial

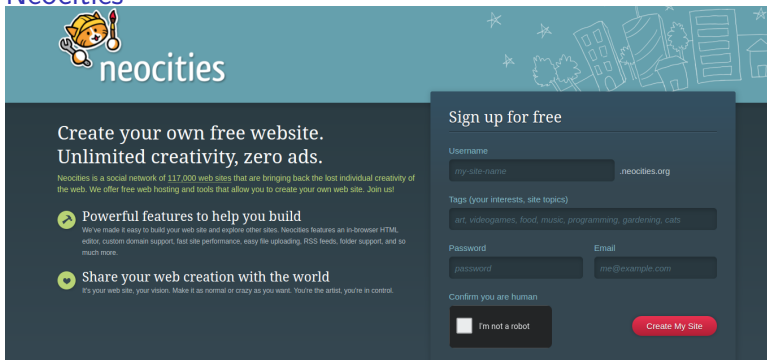
Updates as of: August 11, 2018

## What we're covering

- ▶ Review of HTML
- ▶ Basic JavaScript
- ▶ Examples of the Document Object Model (DOM)
- ▶ A hands-on example of building a todo-list page

# What we'll be using

## Neocities



The screenshot shows the Neocities homepage. At the top left is the Neocities logo, which features a cartoon cat wearing a yellow construction hat and holding a paintbrush. To the right of the logo is the word "neocities" in a white, lowercase, sans-serif font. The background of the header is a teal color with faint white line art of buildings and stars. Below the header, the main content area has a dark teal background. On the left, there is a white text block that says "Create your own free website. Unlimited creativity, zero ads." followed by a paragraph of smaller text. Below this are two green circular icons with white symbols, each followed by a heading and a paragraph of text. On the right, there is a dark teal box with white text that says "Sign up for free". Inside this box are four input fields: "Username" (with a placeholder "my-site-name" and ".neocities.org" to the right), "Tags (your interests, site topics)" (with a placeholder "art, videogames, food, music, programming, gardening, cats"), "Password" (with a placeholder "password"), and "Email" (with a placeholder "me@example.com"). Below these fields is a section titled "Confirm you are human" with a checkbox labeled "I'm not a robot" and a red button labeled "Create My Site".

**neocities**

Create your own free website.  
Unlimited creativity, zero ads.

Neocities is a social network of 117,000 web sites that are bringing back the lost individual creativity of the web. We offer free web hosting and tools that allow you to create your own web site. Join us!

**Powerful features to help you build**  
We've made it easy to build your web site and explore other sites. Neocities features an in-browser HTML editor, custom domain support, fast site performance, easy file uploading, RSS feeds, folder support, and so much more.

**Share your web creation with the world**  
It's your web site, your vision. Make it as normal or crazy as you want. You're the artist, you're in control.

**Sign up for free**

Username  .neocities.org

Tags (your interests, site topics)

Password  Email

Confirm you are human

☐ I'm not a robot [Create My Site](#)

# Intro to Web Development A Guided Tutorial

## └ Introduction

### └ What we'll be using

What we'll be using



Neocities is a free hosting service that allows you to write all the code for your website. It's a pretty solid product that I highly recommend. There's a password provided for an in-class account that we'll be sharing, but if you want to continue working on your own you can sign up for your own account.

# HTML & JavaScript

What are they

**HTML** **H**yper**T**ext **M**arkup **L**anguage

**JavaScript** A programming language

# Intro to Web Development A Guided Tutorial

## └ Introduction

## └ HTML & JavaScript

HTML & JavaScript

What are they

HTML HyperText Markup Language

JavaScript A programming language

HTML is the like the bones, the architecture, of a webpage. It includes the basic *content* of the webpage: there's a button here, a paragraph there, a link here. HTML by itself leads to very plain looking pages, like the ones we had in the 90s. You need CSS, which we aren't covering explicitly in this class, to make things look pretty.

JavaScript is a *programming language* that gives the web browser instructions on what to do in the web page.

## The DOM

# Document Object Model

## └ Introduction

## └ The DOM

**Document Object Model**

This is a technical term for “the way that your JavaScript code can affect the web page”. The web page itself is the “document” and “object” references a kind of data inside the JavaScript language. By data we mean things like numbers or lists or pairs: information that the program needs to do its job. So the “document object model” means that the browser provides a way for your JavaScript code to understand the web page as data it can manipulate.



## Our Example

1. heck
2. dog
3. pup
4. existential dread

 Add your todo items

# The first step

1. On neocities, make a new file
2. Call it "YOURINITIALStodo.html"
3. Type in exactly what you see on the right and save the file
4. View the file

```
<html>
```

```
<ol>
```

```
</ol>
```

```
<input>
```

```
<button>Add!</button>
```

```
</html>
```

# Your first touch of JavaScript

## Change your todo.html file

```
<html>
<head>
  <script src="YOURINITIALStodo.js"></script>
</head>
<body>
  <ol>
</ol>

  <input>
  <button>Add!</button>
</body>
</html>
```

## YOURINITIALStodo.js

```
window.onload = function () {
  console.log("Is this working?");
}
```

# Make the button do something

## Change your todo.html file

```
<html>
  <head>
    <script src="YOURINITIALStodo.js"></script>
  </head>
  <body>
    <ol id="list">

      <input id="input">
      <button id="add">Add!</button>

    </ol>
  </body>
</html>
```

## Change your todo.js file

```
window.onload = function () {
  var inputElement = document.getElementById("input");
  var todoList = document.getElementById("list");
  var addButton = document.getElementById("add");

  addButton.addEventListener("click", function () {
    console.log("Is this working?");
  });
}
```

# Adding addition

## Change your todo.js file

```
window.onload = function () {  
  var inputElement = document.getElementById("input");  
  var todoList = document.getElementById("list");  
  var addButton = document.getElementById("add");  
  
  addButton.addEventListener("click", function () {  
    console.log("Is this working?");  
    var itemText = document.createTextNode(inputElement.value);  
    var newItem = document.createElement("li");  
    newItem.appendChild(itemText);  
    todoList.appendChild(newItem);  
    inputElement.value = "";  
  });  
}
```

# Delete buttons and <div>s

## Change your todo.js file

```
window.onload = function () {  
  var inputElement = document.getElementById("input");  
  var todoList = document.getElementById("list");  
  var addButton = document.getElementById("add");  
  
  addButton.addEventListener("click", function () {  
    console.log("Is this working?");  
    var itemText = document.createTextNode(inputElement.value + " ");  
    var newItem = document.createElement("li");  
    var divvy = document.createElement("div");  
  
    var deleteBut = document.createElement("button");  
    var deleteText = document.createTextNode("X");  
  
    deleteBut.appendChild(deleteText);  
  
    divvy.appendChild(itemText);  
    divvy.appendChild(deleteBut);  
  
    newItem.appendChild(divvy);  
    todoList.appendChild(newItem);  
    inputElement.value = "";  
  });  
}
```

# Adding IDs to the items

## Change your todo.js file

```
window.onload = function () {  
  var inputElement = document.getElementById("input");  
  var todoList = document.getElementById("list");  
  var addButton = document.getElementById("add");  
  
  var listIndex = 0;  
  
  addButton.addEventListener("click", function () {  
    console.log("Is this working?");  
    var itemText = document.createTextNode(inputElement.value + " ");  
    var newItem = document.createElement("li");  
    var divvy = document.createElement("div");  
    var deleteBut = document.createElement("button");  
    var deleteText = document.createTextNode("X");  
    newItem.setAttribute("id", "item"+listIndex);  
    deleteBut.appendChild(deleteText);  
    divvy.appendChild(itemText);  
    divvy.appendChild(deleteBut);  
    newItem.appendChild(divvy);  
    todoList.appendChild(newItem);  
    inputElement.value = "";  
  });  
}
```

# Making the delete button work

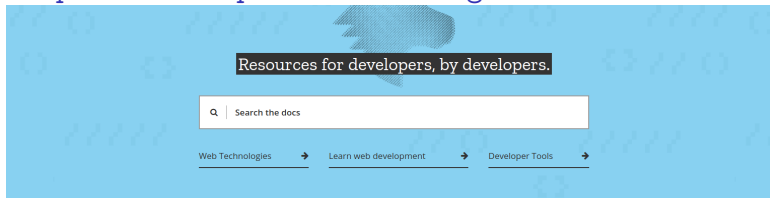
```
window.onload = function () {  
  var inputElement = document.getElementById("input");  
  var todoList = document.getElementById("list");  
  var addButton = document.getElementById("add");  
  
  var listIndex = 0;  
  
  addButton.addEventListener("click", function () {  
    console.log("Is this working?");  
    listIndex++  
    var theIndex = listIndex;  
    var itemText = document.createTextNode(inputElement.value + " ");  
    var newItem = document.createElement("li");  
    var divvy = document.createElement("div");  
    var deleteBut = document.createElement("button");  
    var deleteText = document.createTextNode("X");  
    newItem.setAttribute("id", "item"+listIndex);  
    deleteBut.appendChild(deleteText);  
    deleteBut.addEventListener("click", function () {  
      var theItem = document.getElementById("item"+theIndex);  
      todoList.removeChild(theItem);  
    });  
    divvy.appendChild(itemText);  
    divvy.appendChild(deleteBut);  
    newItem.appendChild(divvy);  
    todoList.appendChild(newItem);  
    inputElement.value = "";  
  });  
}
```



# Where do you go from here?

Mozilla Developer Network

<https://developer.mozilla.org/en-US/>



Test in Firefox and over 800 browser/OS combos with Mozilla and Sauce Labs

Test on Sauce Labs



## Learn the best of web development

Get the latest and greatest from MDN delivered straight to your inbox.

you@example.com

Sign up now



# Intro to Web Development A Guided Tutorial

## └ Introduction

### └ Where do you go from here?

Where do you go from here?

Mozilla Developer Network  
<https://developer.mozilla.org/en-US/>



The Mozilla Developer Network is a great place to continue learning about web development. They have a lot of tutorials and reference guides that are all freely available. Go ahead and navigate to this page and take a few minutes to look around before we go onto the next exercise.

## Conclusion

Question? Comments?