

Introduction to JavaScript

Contents

- [1 About this course](#)
- [2 Table of Contents](#)
 - [2.1 Session 1: Getting Started with the web](#)
 - [2.2 Session 2: Introduction to HTML](#)
 - [2.3 Session 3: Introduction to CSS](#)
 - [2.4 Session 4: JavaScript First Steps](#)
 - [2.5 Session 5: JavaScript Building Blocks](#)
 - [2.6 Session 6: Introducing JavaScript Objects](#)
 - [2.7 Session 7: Client-side Web APIs](#)
 - [2.8 Share this:](#)
 - [2.9 Like this:](#)

About this course

Want to understand the language of web? Eager to build your own website? This introductory course will bring you to the world of web. After step-by-step learning, you will lay a foundation for adding dynamic behavior, storing information, handling requests and responses on a website, etc.

JavaScript is arguably more difficult to learn than related technologies such as HTML and CSS. Before attempting to learn JavaScript, you are strongly advised to get familiar with these two technologies first. The detailed materials on HTML and CSS will also be given in our course.

Table of Contents

Session 1: Getting Started with the web

This is a concise series introducing you to the practicalities of web development. You'll set up the tools you need to construct a simple webpage and publish your own simple code.

- [Installing basic software](#)
 - In this section, we show you step-by-step how to install just the software you need to begin some basic web development
- [What will your website look like?](#)
 - We outline a simple method you can follow to plan out your site's content and design
- [Dealing with files](#)
 - This section explains how to set up a sensible file structure for your website and what issues you should be aware of
- [HTML basics](#)
 - Here you can get familiar with HTML

- [CSS basics](#)
 - This section takes you through what you need to get started with CSS
- [JavaScript basics](#)
 - This section gives you an idea of what is possible with this exciting language, and how to get started
- [Publishing your sample code](#)
 - This section describes how to get your simple sample code online with minimum effort
- [How the web works](#)
 - This section outlines what happens when you view a webpage on your computer

Session 2: Introduction to HTML

- [Getting started with HTML](#)
 - This section covers the absolute basics of HTML, to get you started
- [What's in the head? Metadata in HTML](#)
 - The head of an HTML document is the part that *is not* displayed in the web browser when the page is loaded. This section will show you the secret of the head of an HTML document
- [HTML text fundamentals](#)
 - This section digs into how to use HTML to break a block of text up into a structure of headings and paragraphs, add emphasis to words, create lists, and many more
- [Creating hyperlinks](#)
 - This section shows the syntax required to make a link, and discusses link best practices
- [Advanced text formatting](#)
 - This section will introduce less well-known elements than the elements in [HTML text fundamentals](#), but still useful to know about
- [Document and website structure](#)
 - This section looks into how to plan a basic website structure, and write the HTML to represent this structure
- [Debugging HTML](#)
 - This section will introduce you to some tools that can help you when something is going wrong, and you can't work out where the error in the code is

Session 3: Introduction to CSS

- [How CSS works](#)
 - In this section we start off with a theoretical grounding
- [CSS syntax](#)
 - Let's dive into CSS syntax in a lot more detail
- [Selectors](#)
 - We are going to show what selector types are available and how they work
- [CSS values and units](#)

- In this guide, we look at more common values like length, colour and simple functions, as well as exploring less common units like degrees, and even unitless numerical values
- [Cascade and inheritance](#)
 - This section covers two systems which resolve situations where you have selector conflicts and elements nested inside other elements in enough detail
- [The box model](#)
 - This section introduce the foundation of layout on the Web
- [Debugging CSS](#)
 - This section provides the tools that can help you find errors in your CSS code

Session 4: JavaScript First Steps

- [What is JavaScript?](#)
 - This section will ensure you are comfortable with JavaScript's purpose
- [A first splash into JavaScript](#)
 - Here you'll build up a simple "Guess the number" game, step by step
- [What went wrong? Troubleshooting JavaScript](#)
 - This section aims to save you from tearing your hair out over such problems by providing you with some simple tips on how to find and fix errors in JavaScript programs
- [Storing the information you need — Variables](#)
 - We will get down to the real basics, looking at how to work with most basic building blocks of JavaScript — Variables
- [Basic math in JavaScript — numbers and operators](#)
 - We will discuss how we can combine operators and other features to successfully manipulate numbers to do our bidding
- [Handling text — strings in JavaScript](#)
 - We will look at all the common things that you really ought to know about strings when learning JavaScript
- [Useful string methods](#)
 - We will start thinking about what useful operations we can do on strings with built-in methods
- [Arrays](#)
 - We look at why this is useful, then explore detailed skills about arrays

Session 5: JavaScript Building Blocks

- [Making decisions in your code — conditionals](#)
 - We will explore how conditional structures work in JavaScript
- [Looping code](#)
 - We will look at loop structures in JavaScript.
- [Functions — reusable blocks of code](#)
 - We will explore fundamental concepts behind functions such as basic syntax, how to invoke and define functions, scope, and parameters.
- [Build your own function](#)

- Here you'll get some practice with building up your own custom function
- [Function return values](#)
 - The last essential concept you must know about a function is return values
- [Introduction to events](#)
 - We will discuss some important concepts surrounding events, and look at how they work in browsers

Session 6: Introducing JavaScript Objects

- [Object basics](#)
 - We will look at fundamental JavaScript object syntax, and revisit some JavaScript features we've already looked at earlier
- [Object-oriented JavaScript for beginners](#)
 - This section presents a basic view of object-oriented programming (OOP) theory, then explores how JavaScript emulates object classes via constructor functions, and how to create object instances
- [Object prototypes](#)
 - We explore that difference, explain how prototype chains work, and look at how the prototype property can be used to add methods to existing constructors
- [Inheritance in JavaScript](#)
 - In this section you will learn how to create “child” object classes (constructors) that inherit features from their “parent” classes
- [Working with JSON data](#)
 - In this section we give you all you need to work with JSON using JavaScript, including parsing the JSON so you can access data items within it and writing your own JSON
- [Object building practice](#)
 - In this section we dive into a practical exercise, giving you some more practice in building custom JavaScript objects

Session 7: Client-side Web APIs

- [Introduction to web APIs](#)
 - We will start by looking at APIs from a high level
- [Manipulating documents](#)
 - We will look at how to use the DOM in detail, along with some other interesting APIs that can alter your environment in interesting ways
- [Fetching data from the server](#)
 - We will explain the concept, and look at technologies that make it possible, such as [XMLHttpRequest](#) and the [Fetch API](#).
- [Third party APIs](#)
 - We will look at the difference between browser APIs and 3rd party APIs and shows some typical uses of the latter
- [Drawing graphics](#)
 - This section provides an introduction to the Canvas API, and further resources to allow you to learn more

- [Video and audio APIs](#)
 - This section shows you how to do common tasks such as creating custom playback controls
- [Client-side storage](#)
 - This section explains the very basics of how these work

If the course is of your interest, then simply register Hackathon and get started!