SCRIPTING OVERVIEW

View online: https://www.construct.net/en/make-games/manuals/construct-3/scripting/overview

Construct supports writing **JavaScript** or **TypeScript** code in the place of actions and events, as well as in separate JavaScript or TypeScript files. This is a great way to learn to code, as well as taking advantage of the full power of modern industry-standard programming languages in your projects.

JavaScript is not Java

Note **Java** is a completely different programming language to **JavaScript**. This is a common point of confusion. Try not to get the two mixed up. For example if you search the web for help on JavaScript, be sure to specifically search for *JavaScript* and not *Java*, otherwise you will get results for the wrong programming language!

JavaScript vs. TypeScript

TypeScript is essentially an extension of JavaScript that adds static type features. Whereas JavaScript is a dynamic language that allows variables and properties to hold any data and change at any time, TypeScript uses type annotations to more strictly validate code. To illustrate the difference, here is a sample JavaScript function:

```
function add(a, b)
{
	return a + b;
}
```

This function is intended to add two numbers, but can actually be called with strings, objects, or anything else - and it may then produce unintended results. The same function in TypeScript looks like this:

```
function add(a: number, b: number)
{
	return a + b;
}
```

The only difference is the addition of `: number` in the parameters, which specifies they must be numbers. Then TypeScript will enforce this restriction, helping to avoid any mistakes.

Browsers can only directly run JavaScript, so TypeScript code is converted (or "compiled") to JavaScript essentially just by deleting the type annotations. This illustrates how closely related the languages are - TypeScript is essentially adding a layer of additional checks on top of JavaScript.

Some of the advantages and disadvantages of JavaScript vs. TypeScript are:

- **Ease of learning**: TypeScript may take a little longer to learn as you have to also learn about type annotations.
- **Coding accuracy**: TypeScript's type checks can catch lots of kinds of errors for you before you even try to run your project. JavaScript on the other hand allows many kinds of incorrect code to execute, and you may only be able to find out the problem from either the browser console or using a debugger.
- **Coding flexibility**: JavaScript allows writing a wider range of code styles, allowing things that TypeScript would not normally allow due to its type checking (but as noted previously that also includes incorrect code).
- **Autocomplete**: TypeScript allows providing an exact autocomplete feature. As JavaScript is a dynamic language it is not possible to always accurately identify which properties and methods are available, and so often the autocomplete list will include guesses which are not actually valid, or miss out possible entries.
- Other developer tools: TypeScript also allows other tools like **Go to definition** and **Find references** to work accurately thanks to its type system, whereas much like with autocomplete in JavaScript these may resort to guesses or miss out some results.

If you are new to coding and wondering which to learn, we would suggest to consider starting with TypeScript. Despite the fact there is a little more to learn, its extra checks mean it catches lots of kinds of beginner mistakes, and the precise autocomplete helps you more quickly identify what kind of code you can write.

Terminology

JavaScript is often shortened to *JS* and normally uses files with the extension *.js*. TypeScript is often shortened to *TS* and normally uses files with the extension *.ts*. The terms *script* and *code* refer to the same thing: the programming code that you or someone else has written. Similarly *scripting*, *coding* and *programming* are used interchangeably, and all refer to the process of writing code in a programming language. In the context of Construct these terms refer to JavaScript or TypeScript code, as those are the two languages it supports.

Quick start guide

If you already know JavaScript, check out the Construct for JavaScript developers quick start guide for a summary of what you might need to know when working in Construct. If you already know TypeScript, you can also check out the Construct for TypeScript developers quick start guide.

If you'd also prefer to use an external code editor like VS Code, see Using an external editor.

Learning JavaScript

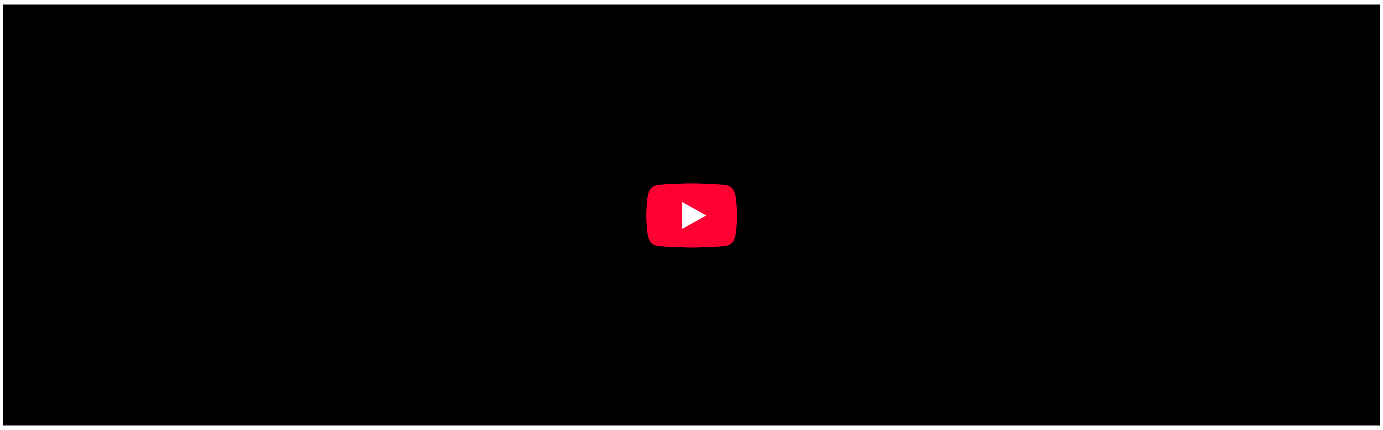
If you're interested in learning JavaScript from scratch using Construct, see our 13-part tutorial series Learn JavaScript in Construct.

JavaScript is one of the most popular languages in the world, and is widely used in the technology industry, especially in web development. As a result there are many more materials available across the web for teaching the JavaScript language. The tutorial series linked to above includes lots of links towards the end where you can continue learning more about JavaScript.

This manual section focuses on the unique details of using JavaScript within Construct. We recommend the MDN web docs as a good reference to use for the JavaScript language itself. This manual links to it when referring to specific parts of the JavaScript programming language so you can learn more about it.

Introductory video

For a video introduction to using JavaScript in Construct, see the video below.



Subscribe to Construct videos now

Learning TypeScript

If you're interested in learning TypeScript from scratch using Construct, see our 14-part tutorial series Learn TypeScript in Construct.

As with JavaScript, TypeScript is one of the most popular languages in the world, and is widely used in the technology industry, especially in web development. The tutorial series linked to above also includes lots of links towards the end where you can continue learning more about TypeScript.

The official TypeScript website provides lots of useful guides and information, including:

- TypeScript for the New Programmer
- TypeScript for JavaScript Programmers
- The TypeScript Handbook

See also the guide Using TypeScript in Construct.

Examples

In the Example Browser you can filter for JavaScript or TypeScript examples under the *Coding* header. You'll find a variety of example projects making use of JavaScript/TypeScript coding in Construct, including many examples that come with both JavaScript and TypeScript variants. These cover everything from simple beginner examples to sophisticated fully-coded games.

The reference section of the manual also includes some code samples in places, with snippets demonstrating how to use specific features. These will usually need to be pasted in to a project with appropriately named objects and behaviors to work correctly, or otherwise edited as necessary for use in your own projects.