

USING IMPORT MAPS

View online: <https://www.construct.net/en/make-games/manuals/construct-3/scripting/guides/using-import-maps>

Construct allows using **import maps** to customize how you write `import` statements.

Normally you must import a script by its file path, such as:

```
// Standard path-based import
import * as MyModule from "./mymodule.js";
```

However with an import map you can write something like this instead:

```
// Standard path-based import
import * as MyModule from "mymodule";
```

The import map then determines what path to use for "mymodule".

Adding an import map

Import maps are defined as JSON files. To add an import map to your project, follow these steps.

- 1 In the Project Bar, right-click on *Files* and select **New▶ JSON**
- 2 Name the file something like *importMap.json*.
- 3 Click the file in the Project Bar to show its properties in the Properties Bar.
- 4 Change the *Purpose* property to *Import map*.

For the contents of the import map, use some JSON data like the following. Refer to the MDN article linked above for full details on the format.

```
{
  "imports": {
    "mymodule": "./mymodule-v1.js"
  }
}
```

Now if you write this in a script file:

```
// Standard path-based import
import * as MyModule from "mymodule";
```

...the import map means that importing "mymodule" will load the path "./mymodule-v1.js".

Why to use import maps

You don't have to use import maps, but they can be a handy way to organize your imports. To illustrate this, using the example above, suppose you import "mymodule" in lots of places. You can then change that from importing "./mymodule-v1.js" to importing "./mymodule-v2.js" merely by changing the entry in the import map - you don't have to go through every place it's imported and update it. So import maps can help manage complex sets of imports more easily.

Some third-party libraries like three.js also rely on import maps, and so supporting them makes it easier to use these libraries. You can find an example of this in the [Integrated 3D engine example](#).

Limitations

As of September 2024, browsers don't support using import maps in web workers. Therefore if you use import maps, your project will run in DOM mode and not be able to use worker mode.

Currently Construct only supports the `"imports"` field in an import map. The `"scopes"` and `"integrity"` fields are supported by browsers, but are not yet supported in exported projects, as these need special handling for cases like minified exports.