

Module usage

To import three as module the script tag must be of type module.

```
<script type="module">
```

The three framework uses a folder structure.

If you take over the complete system, all import statements are correct. If you want to import three.js ([three.module.js](#)) and a control ([Orbitcontrols.js](#)), the following results.

See <https://discourse.threejs.org/t/module-import-usage/11574/18>

```
- vendor
  - three
    - build
      three.module.js
    - examples
      - jsm
        - controls
          Orbitcontrols.js
```

The collection of basic examples requires only a few modules besides three.module.js.

Therefore it makes sense to put these modules in a common folder.

You can take the current revision of three into account. You simply add to it, and you have about ten three files for one year.

The import

```
import * as THREE from "https://threejs.org/build/three.module.js";
import { OrbitControls } from "https://threejs.org/examples/jsm/controls/OrbitControls.js";
```

then becomes

```
import * as THREE from "../jsm/three.module.112.js";
import { OrbitControls } from "../jsm/OrbitControls.js";
```

if you put the modules in your own folder `jsm`. One has a very simple structure.

```
└─ BasicExample
└─ jsm
```

Now you have to adjust the import path in the modules.

But for the few modules this is more clear than creating the complete folder structure.

In module OrbitControls.js

```
10 import {
11     EventDispatcher,
12     MOUSE,
13     Quaternion,
14     Spherical,
15     TOUCH,
16     Vector2,
17     Vector3
18 } from "../../build/three.module.js";
```

turns into

```
10 import {
11     EventDispatcher,
12     MOUSE,
13     Quaternion,
14     Spherical,
15     TOUCH,
16     Vector2,
17     Vector3
18 } from "../jsm/three.module.112.js";
```

The first use was in 2020 in the first example (at the bottom).

<https://hofk.de/main/discourse.threejs/2020/displacementMap/displacementMap.html>

That's all!