

## Diferenças na importação de JS

### Transcrição

Por último, vale a pena notar na importação do **renderer.js** dentro do **index.html**:

```
<script>require('./js/renderer')</script>
```

Essa importação é diferente da que costumamos fazer habitualmente. Por exemplo:

```
<script src="js/renderer.js"></script>
```

Será que as duas importações funcionam do mesmo jeito?

### Entendendo a diferença das importações

As duas importações funcionam, se importamos estilo Node.js, vale o caminho relativo ao arquivo em questão. Por exemplo, no **index.html**:

```
<script>require('./js/renderer')</script>
```

E no **renderer.js**:

```
const timer = require('./timer');
```

O ponto significa que estamos partindo do diretório onde o arquivo está. No caso do **index.html**, dentro do seu diretório, entramos na pasta **js** e lá encontramos o **renderer.js**. E no caso do **renderer.js**, dentro do seu diretório encontramos o **timer.js**.

Agora vejamos o jeito "tradicional" de importação:

```
<script src="js/renderer.js"></script>
```

Nesse caso, **vale o diretório onde a página HTML está**. Então, no **renderer.js**, na hora de importar, temos que imaginar que estamos partindo do arquivo HTML. Logo, a importação deve ficar assim:

```
const timer = require('../js/timer');
```

Do diretório do **index.html**, entramos na pasta **js** e lá encontramos o **timer.js**.

No nosso projeto, vamos continuar utilizando as importações no estilo Node.js, até para facilitá-las.

