

Fundamentos de la Web

Bloque II: Tecnologías de cliente web

Tema 7.3 – Módulos en JavaScript

EcmaScript Modules

- Cuando el código **JS es largo** no es manejable usar un único fichero
- A lo largo del tiempo se han desarrollado varias tecnologías para separar el código en varios ficheros (**CommonJS, AMD...**)
- En **ES6 (2015)** se definió el sistema de módulos estándar para aplicaciones JavaScript llamado **EcmaScript Modules (ES Modules, ESM)**

EcmaScript Modules

- Cada **fichero** es un **módulo**
- Si un módulo A quiere usar variables, funciones, clases de un módulo B:
- El módulo B tiene que **exportar** lo que quiera hacer público

moduleB.js

```
export function hola() { ... }
```

- El módulo A tiene que **importar** lo que quiera usar

moduleA.js

```
import { hola } from './moduleB.js'
```

Módulos

Ejecutar código de un módulo

- El código del módulo se ejecuta cuando se importa

Ejemplo1

hello.js

```
console.log('Hello world');
```

app.js

```
import './hello.js';
```

index.html

```
<html>
  <head>
    <script type="module" src="app.js">
    </script>
  </head>
  <body>
  </body>
</html>
```

Exportar función con nombre

fiz.js

```
export function fiz(){  
  console.log('fiz!');  
}
```

app.js

```
import { fiz } from './fiz.js';  
  
fiz();
```

Ejemplo2

app.js

```
import * as module from './fiz.js';  
  
module.fiz();
```

Ejemplo3

Exportar una única función anónima

- Se le da nombre al importarla

Ejemplo4

bar.js

```
export default function(){  
  console.log('bar!');  
}
```

app.js

```
import bar from './bar.js';  
bar();
```

Exportar varios elementos

- Se pueden exportar varios elementos y de diferentes tipos (funciones, objetos, valores, clases...)

Ejemplo5

utils.js

```
export var obj = {  
  prop1: 3,  
  prop2: 'value'  
}  
  
export function log(msg){  
  console.log(msg);  
}  
  
export class Date {  
  constructor(date){  
    this.date = date;  
  }  
}
```

app.js

```
import { log, obj, Date } from './utils.js';  
  
log('message');  
  
console.log(obj);  
  
var date = new Date('01-12-19');
```

Exportar varios elementos

- Se pueden exportar varios elementos y de diferentes tipos (funciones, objetos, valores, clases...)

Ejemplo6

utils.js

```
export var obj = {  
  prop1: 3,  
  prop2: 'value'  
}  
  
export function log(msg){  
  console.log(msg);  
}  
  
export class Date {  
  constructor(date){  
    this.date = date;  
  }  
}
```

app.js

```
import * as utils from './utils.js';  
  
utils.log('message');  
  
console.log(utils.obj);  
  
var date = new utils.Date('01-12-19');
```


Ejercicio 1

- Se proporciona un único script que quita ceros y ordena un array de números
- Se debe dividir el script en 3 módulos:
 - **app.js**: Principal
 - **ordena.js**: Función ordena exportada con nombre.
 - **quitaCeros.js**: Función quitaCeros exportada por defecto

