

Los módulos (Node.js modules)

“

Un módulo es **un bloque de código reutilizable**, cuya existencia o no, no altera el comportamiento de otros bloques de código. A partir de eso, Node.js propone fragmentar nuestro código en pequeños módulos, en donde **cada uno tendrá una funcionalidad específica para alcanzar un objetivo definido.**

”



Tipos de módulos

Existen tres tipos de módulos:

- Los módulos **nativos**: ya vienen instalados cuando creamos un proyecto de Node.js.
- Los módulos **de terceros**: podemos instalarlos usando algo llamado npm que vamos a conocer más adelante.
- Los módulos **creados**: aquellos que escribimos nosotros.



Por convención, el nombre de la variable que almacene el módulo que estamos requiriendo suele recibir **el mismo nombre del módulo**, o una abreviatura.

Módulos creados

Para requerir un módulo creado por nosotros, primero hay que crear un archivo con extensión **.js** y, dentro del mismo, escribir el script que necesitamos.

Una vez definido nuestro código, tenemos que dejarlo accesible para poder importarlo dentro de nuestra aplicación. Para eso, hay que hacer uso del objeto nativo **module** y de su propiedad **.exports**. A este le asignaremos el nombre de la variable que contenga la información que queremos exponer.

{código}

```
let series = ['Friends', 'Mr. Robot',  
              'Breaking Bad'];
```

```
module.exports = series;
```

Definimos un **array** de series, en donde en cada posición hay una serie.

{código}

```
let series = ['Friends', 'Mr. Robot',  
'Breaking Bad'];
```

```
module.exports = series;
```

Hacemos uso del objeto **module** y su propiedad **exports**, y le asignamos la variable que queremos exportar, en este caso, series.

Tener en cuenta que siempre que queramos exportar módulos de un script, tendremos que escribir esta línea al final del mismo.

Módulos creados

Una vez que exportamos nuestro módulo, vamos al archivo en donde lo queremos importar y usamos la función **require()**.

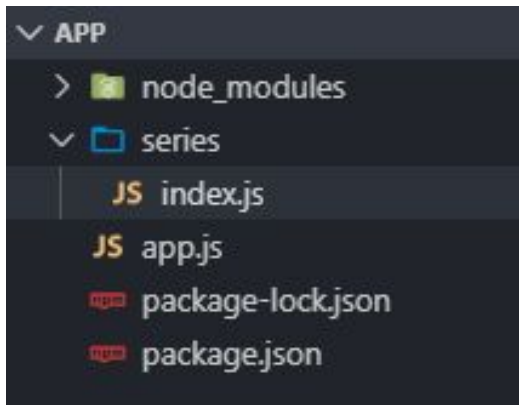
En este caso, le pasamos como argumento la ruta hacia el script donde se encuentra el módulo que queremos requerir. Para eso, usamos el **./**. De esta forma, le estamos indicando a Node.js que el camino para llegar a ese módulo empieza desde donde estamos parados (app.js), hasta el nombre del archivo que le pasemos.



Como buena práctica, se suelen almacenar los módulos que creamos dentro de una carpeta **con el mismo nombre del módulo que estamos por crear.**

Módulos creados

Cuando nos referimos a archivos de JavaScript, **no hace falta escribir la extensión.**



```
JS  const series = require('./series/index');
```


¿Cómo requerir un módulo?

Para requerir un módulo hace falta situarse dentro del archivo en el que queremos incorporarlo y hacer uso de la función nativa de Node.js: **require()**. Esta recibe como parámetro un string que será el nombre del módulo.

Esta función devuelve un objeto literal, por lo tanto, es importante guardar la ejecución en una variable para poder acceder, a través del dot notation, a todas las propiedades y funcionalidades del módulo.

```
JS  let modulo = require('nombreDelModulo');
```

“

Para poder ver **todo** lo que trae consigo un módulo podemos hacer un **console.log()** de la **variable** en la que lo almacenamos.



”

DigitalHouse >
Coding School