

## Publicación de paquetes (práctica)

Tiempo estimado: 20min

El objeto de esta práctica es afianzar, reforzar y consolidar los conocimientos teóricos presentados en la lección.

Al finalizarla, el estudiante:

- Habrá creado una cuenta de usuario en el repositorio oficial de **NPM**.
- Habrá publicado un paquete de prueba.
- Habrá suprimido las publicaciones de un paquete.

### Objetivos

El objetivo de la práctica es mostrar cómo publicar paquetes en el repositorio de **NPM**. Para ello, crearemos una cuenta de usuario y publicaremos un paquete de prueba. Este paquete consistirá en una pequeña aplicación calculadora, como la que hemos visto en las lecciones anteriores.

### Creación de la cuenta de usuario

Para comenzar, crearemos una cuenta de usuario en el repositorio de **NPM** mediante la cual publicar nuestros propios paquetes:

1. Abrir el navegador.
2. Ir a [npmjs.org](https://npmjs.org).
3. Hacer clic en el enlace **sign up**:



4. Rellenar el formulario de alta:

Create a profile *or*

Log in

Name

Public Email

Username

<https://www.npmjs.com/~username>

Password

☒ Sign up for the **npm Weekly**

☐ I agree to the **End User License Agreement** and the **Privacy Policy**.

Your email address will show on your profile page, but npm will never share or sell it.

Create an Account

5. Aceptar los términos de licencia.
6. Hacer clic en el botón **Create an Account** para crear la cuenta.

### Apertura de sesión con el repositorio de NPM

Una vez creada la cuenta, hay que abrir sesión con el servidor **NPM** para poder publicar paquetes en él.

1. Abrir una consola.
2. Conectar al repositorio:  

```
$ npm login
```

Rellene los datos que le solicita.
3. Consultar la cuenta con la que tenemos abierta la sesión:  

```
$ npm whoami
```

### Creación del paquete a publicar

Ahora, crearemos el paquete a publicar:

1. Ir a la consola.
2. Crear el directorio del paquete e ir a él.  
En nuestro caso, usaremos el paquete `calcul-app`. Pero sólo un usuario podrá publicar este paquete. Cada estudiante debe elegir un nombre que no esté cogido ya. Para consultarlo, puede ir a [npmjs.org](https://npmjs.org) y hacer una búsqueda.
3. Crear el directorio del proyecto mediante el generador de **Justo**:

```
$ justo -g node
```

Esta vez, por favor, tómese su tiempo para responder detenidamente a las preguntas que le haga el generador. Responda como considere mejor, teniendo en cuenta que:

- A la pregunta del tipo de paquete, seleccionar **app**.
- A la pregunta sobre el comando de aplicación, introducir **calcul**.
- A la pregunta sobre el usuario **NPM** bajo el cual publicar el paquete, indique su usuario.

4. Echar un vistazo al archivo **package.json**.

5. Echar un vistazo al archivo **Justo.js** que contiene las tareas automatizadas del proyecto.

Busque aquella que tiene el nombre **publish**. Mediante esta tarea se automatiza el proceso de publicación. Observe que el nombre de la cuenta bajo la cual se debe publicar el paquete es el del estudiante. Si la sesión abierta con el repositorio no coincide con este nombre, la tarea no realizará la publicación.

6. Instalar dependencias:

```
$ npm install
```

7. Mostrar las tareas automatizadas del catálogo del proyecto:

```
$ justo -c
```

8. Crear el archivo **lib/calcul.js**:

```
/**
 * Suma los valores pasados como argumentos.
 *
 * @param ...args:number[] Los valores a sumar.
 */
export function sum(...args) {
  var res = 0;
  for (let arg of args) res += arg;
  return res;
}

/**
 * Resta los valores pasados como argumentos.
 *
 * @param ...args:number[] Los valores a restar.
 */
export function sub(...args) {
  var res = 0;
  for (let arg of args) res -= arg;
  return res;
}
```

9. Editar el archivo **index.js** y añadir la API del paquete cuando se use como biblioteca:

```
export {sum, sub} from "../lib/calcul";
```

10. Guardar cambios.

11. Modificar el archivo **bin/calcul.js**:

```
#!/usr/bin/env node

"use strict";

const cmd = process.argv[2];
const vals = process.argv.slice(3);
var res = 0;

// (1) op
if (cmd == "sum") {
  for (let val of vals) res += Number(val);
} else if (cmd == "sub") {
  for (let val of vals) res -= Number(val);
} else {
  console.error(`Operación no válida: ${cmd}.`);
  process.exit(1);
}
```

- ```
//(2) mostramos resultado
console.log(res);
```
12. Guardar cambios.
  13. Ir a la consola.
  14. Generar el paquete final:  

```
$ justo build
```
  15. Instalar el paquete globalmente:  

```
$ npm install -g --production dist/es5/nodejs/calcul-app
```
  16. Comprobar la instalación del paquete:  

```
$ npm ls -g calcul-app
```
  17. Realizar un cálculo de prueba:  

```
$ calcul sum 1 3 5 7 9
25
$
```
  18. Desinstalar el paquete:  

```
$ npm uninstall -g calcul-app
```

## Publicación del paquete

---

A continuación, vamos a publicar el paquete:

1. Publicar el paquete.  
Esto puede hacerlo de dos formas, mediante el comando `npm publish` directamente o mediante la tarea automatizada `publish`. Se recomienda usar la tarea del catálogo.  
Vamos a hacerlo mediante el comando `npm`:  

```
$ npm publish dist/es5/nodejs/calcul-app
```

  
Si hace modificaciones, puede publicar la siguiente mediante la tarea automatizada `publish` que tiene en cuenta el nombre del usuario a la hora de publicar:  

```
$ justo publish
```

  
Si trabaja en varios proyectos y usa cuentas de usuario distintas, la automatización le ayudará a evitar errores de cuenta.
2. Abrir el navegador.
3. Ir a [npmjs.org](https://npmjs.org).
4. Buscar su paquete.  
La página de inicio del paquete muestra el contenido del archivo `README.md`.
5. Ir a la consola.
6. Instalar el paquete desde el repositorio:  

```
$ npm install -g calcul-app
```
7. Probar el paquete:  

```
$ calcul sum 1 3 5 7 9
```

## Supresión de publicación de paquete

---

Finalmente, vamos a ver cómo suprimir el paquete:

1. Ir a la consola.
2. Suprimir la publicación del paquete:  

```
$ npm unpublish calcul-app
```

  
Mediante este comando, se suprime todas las publicaciones del paquete. `npm` sólo lo permite si explícitamente se proporciona la opción `--force`. Es por seguridad.
3. Suprimir la publicación del paquete:

```
$ npm unpublish --force calcul-app
```