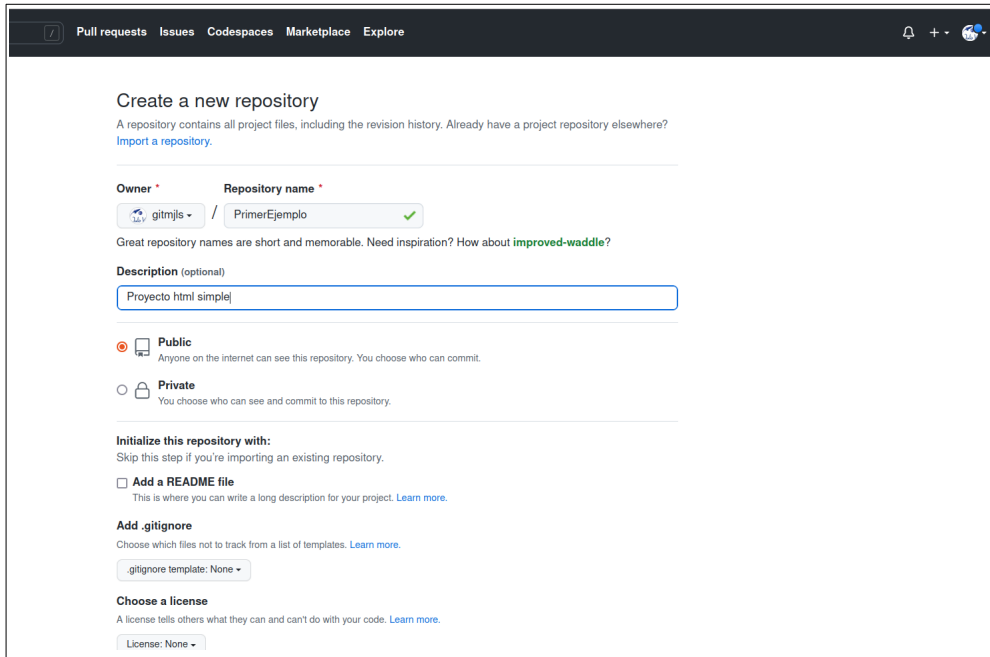


Práctica 1: Subir un proyecto a un repositorio en github


1.- Crear un repositorio en github para albergar el proyecto



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?
[Import a repository.](#)

Owner ^{*} Repository name ^{*}

 gitmjls / PrimerEjemplo ✓

Great repository names are short and memorable. Need inspiration? How about [improved-waddle](#)?

Description (optional)

Proyecto html simple

☒ **Public**
Anyone on the internet can see this repository. You choose who can commit.

☐ **Private**
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☐ **Add a README file**
This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore
Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: None

Choose a license
A license tells others what they can and can't do with your code. [Learn more.](#)

License: None

2.- Instalar git en el equipo local:

```
sudo apt-get update
```

```
sudo add-apt-repository ppa:git-core/ppa  
sudo apt-get update  
sudo apt-get install git
```

```
git --version
```

Cada cuenta tiene su token de seguridad

Personal Account- Settings - Developer settings--personal access tokens- tokens
generate new token- generate new token classic
no expiration – selecciono todo.

3.- Ubicarse a través de la consola en la raíz del proyecto.

```
git init
```

```
git add .
```

```
git remote add origin https://github.com/gitmjls/PrimerEjemplo.git  
(URL del repositorio)
```

```
git commit -a (se abre el editor nano para introducir comentario)
```

(introducir comentario, grabar y salir)

4. Sube el proyecto a la nube:

git push origin master

5. Hago cambios en local y subo los cambios a la nube.

- Realizo los cambios en local en el terminal desde la carpeta raiz del proyecto.
- git add .
- git commit -a (se abre el editor nano para introducir comentario)
- git push origin master

Subir un proyecto eclipse a un repositorio en github

0.- Vamos a crear un proyecto maven web en Eclipse eclipse-jee-2022-12-R-linux-gtk-x86_64

Video explicativo: <https://www.youtube.com/watch?v=UOwC2hmv2Us>

- org-apache-maven-archetypes 1.4 de all Catalogos (no seleccionar run)
- Añadimos dependencias

```
<dependency>
  <groupId>javax.servlet</groupId>
  <artifactId>javax.servlet-api</artifactId>
  <version>4.0.1</version>
  <scope>provided</scope>
</dependency>

<properties>
  <project.build.sourceEncoding>UTF-
8</project.build.sourceEncoding>
  <maven.compiler.source>11</maven.compiler.source>
  <maven.compiler.target>11</maven.compiler.target>
</properties>
```

- En preferences del proyecto Web Module 4.0 (para Tomcat 9) y java 11
- En la raiz del proyecto- click derecho – Build path- y seleccionamos Maven Dependencies- aplicar

1.- Crear un repositorio en github para albergar el proyecto

3.- Ubicarse a través de la consola en la raiz del proyecto.

git init

git add .

git remote add origin <https://github.com/gitmjls/SegundoEjercicio.git> (URL del repositorio)

git commit -a (se abre el editor nano para introducir comentario)

(introducir comentario, grabar y salir)

git push origin master

4.- Al abrir el proyecto en eclipse se observará como aparece en cada fichero y carpeta del proyecto un pequeño icono con un cilindro amarillo. Eso nos indica que nuestro proyecto está gestionado por git.

5.- Cuando hagamos algún cambio observaremos que nos aparece en el fichero una flecha '>'. Para que se actualicen los cambios en el repositorio deberemos hacer click en botón derecho del proyecto e ir a Team - Commit

Introducimos un comentario explicando los cambios realizados y se pulsa Commit and Push para actualizar los cambios en el repositorio de GitHub.