Metodología de trabajo con IntelliJ, Git y GitHub

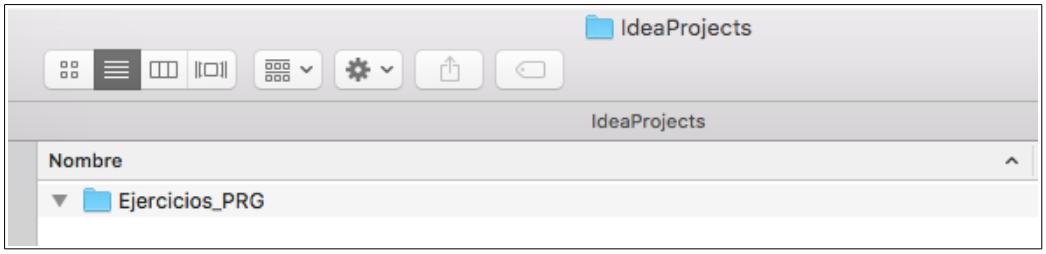
IMPORTANTE: En este apartado vamos a ver como estructurar nuestros proyectos de la asignatura de PRG, se puede extraespolar a otras asignaturas, pero crear un directorio diferente por cada asignatura.

IMPORTANTE: Debemos tener instalado Git en nuestro equipo.

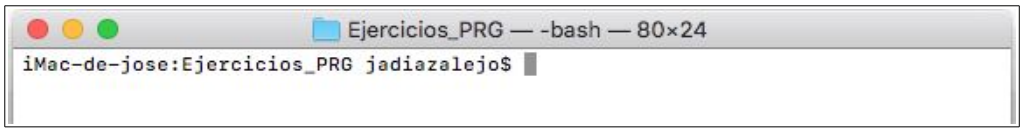
IMPORTANTE: Debemos tener creada una cuenta de GitHub.

COMENZAMOS (Este proceso inicial solo hay que realizarlo una vez)

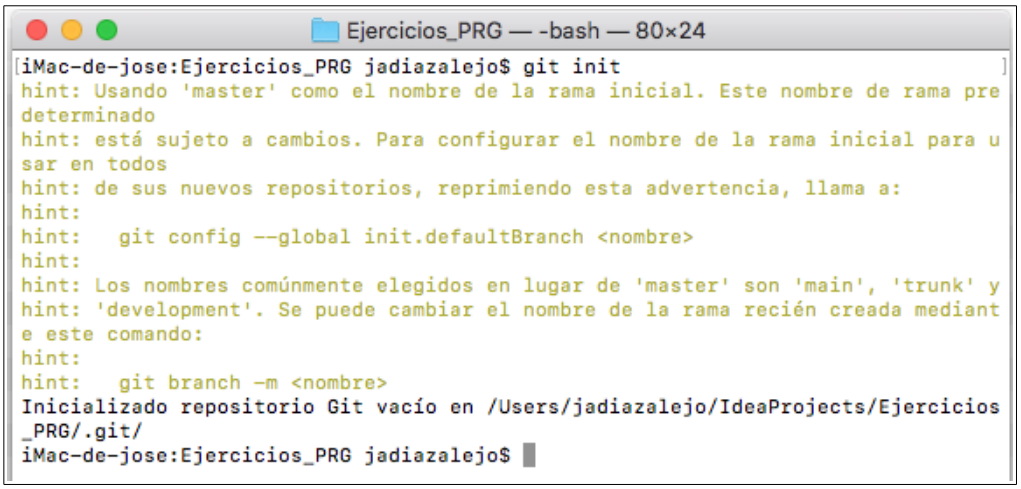
1. Crear un directorio en vuestro sistema de archivos, por ejemplo **Ejercicios_PRG**. Yo la he creado dentro de la carpeta **IdeaProjects**.



2. Abrimos un terminal y nos posicionamos en la carpeta creada en el punto anterior.



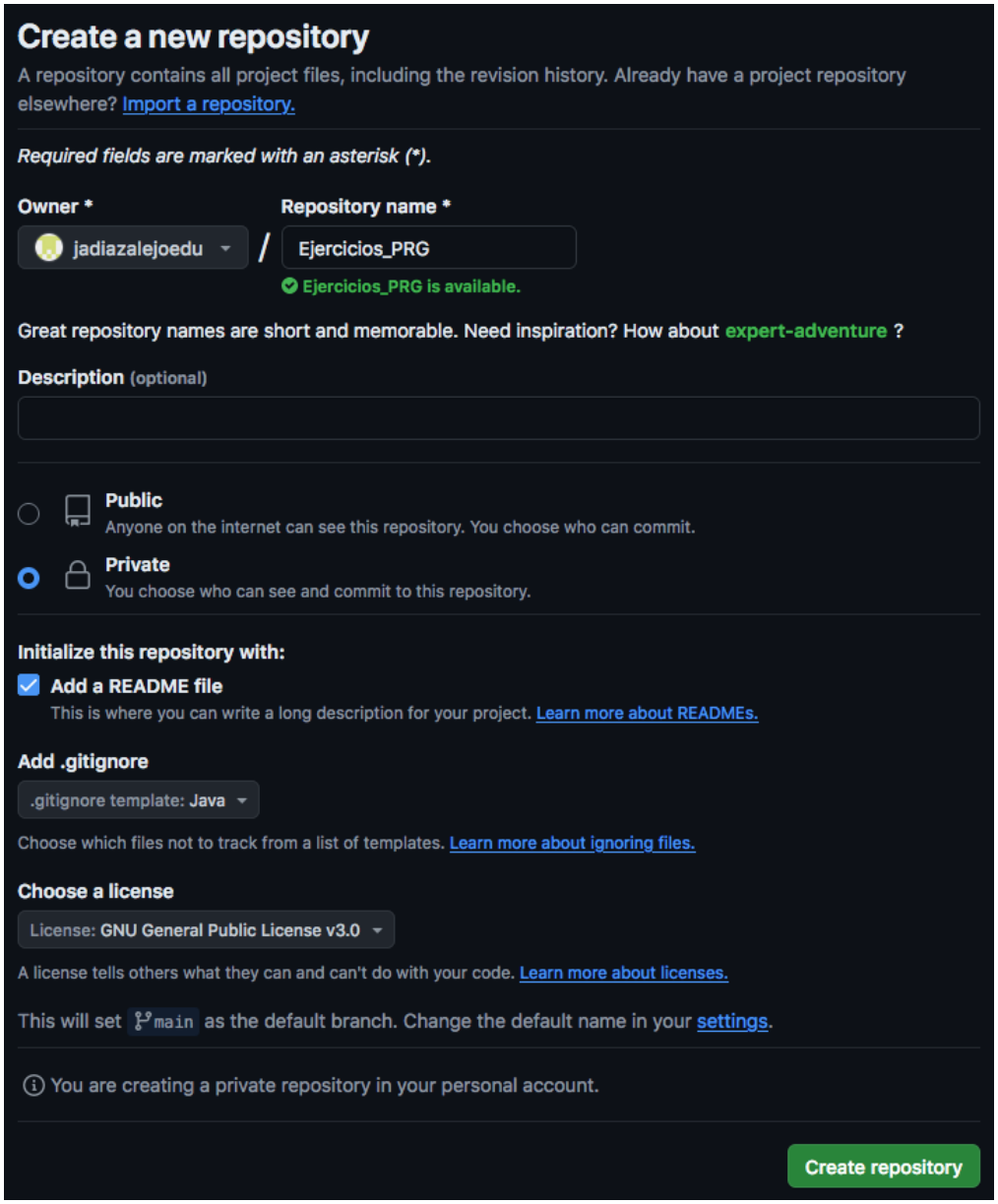
3. Iniciamos el repositorio local con la instrucción: ***git init***



4. Abrimos nuestra cuenta en GitHub.

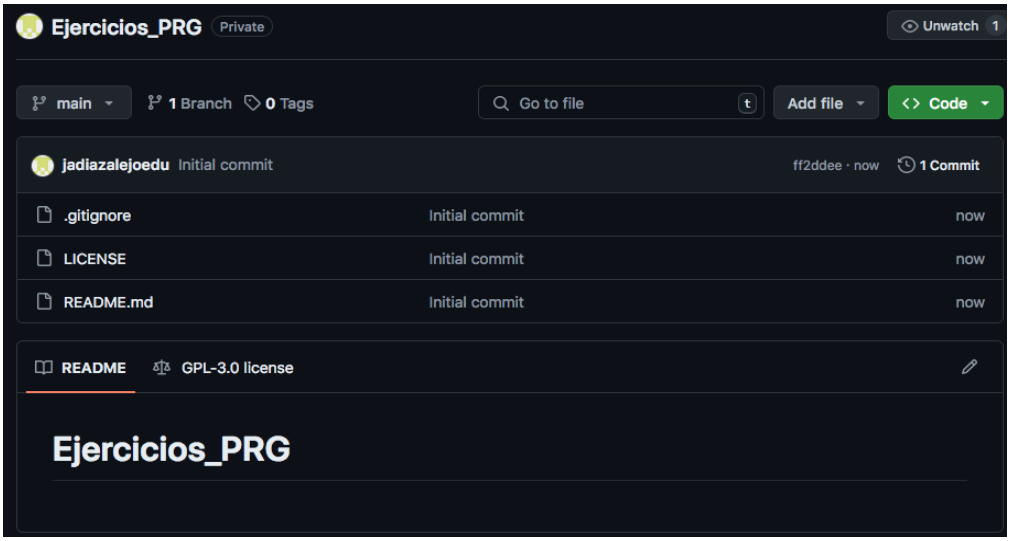
5. **Creamos un repositorio remoto** para guardar todos nuestros ejercicios de la asignatura de programación.

    - **Nombre**, por ejemplo: **Ejercicios_PRG**
    - **Public o Private: Private** (por el momento)
    - Añadimos **Readme file**, para poder hacer una descripción del repositorio.
    - Añadimos **.gitignore** de Java para que se ignoren determinados ficheros
    - Licencia, la que queráis. **GNU General Public Licence v3.0** es una buena opción.



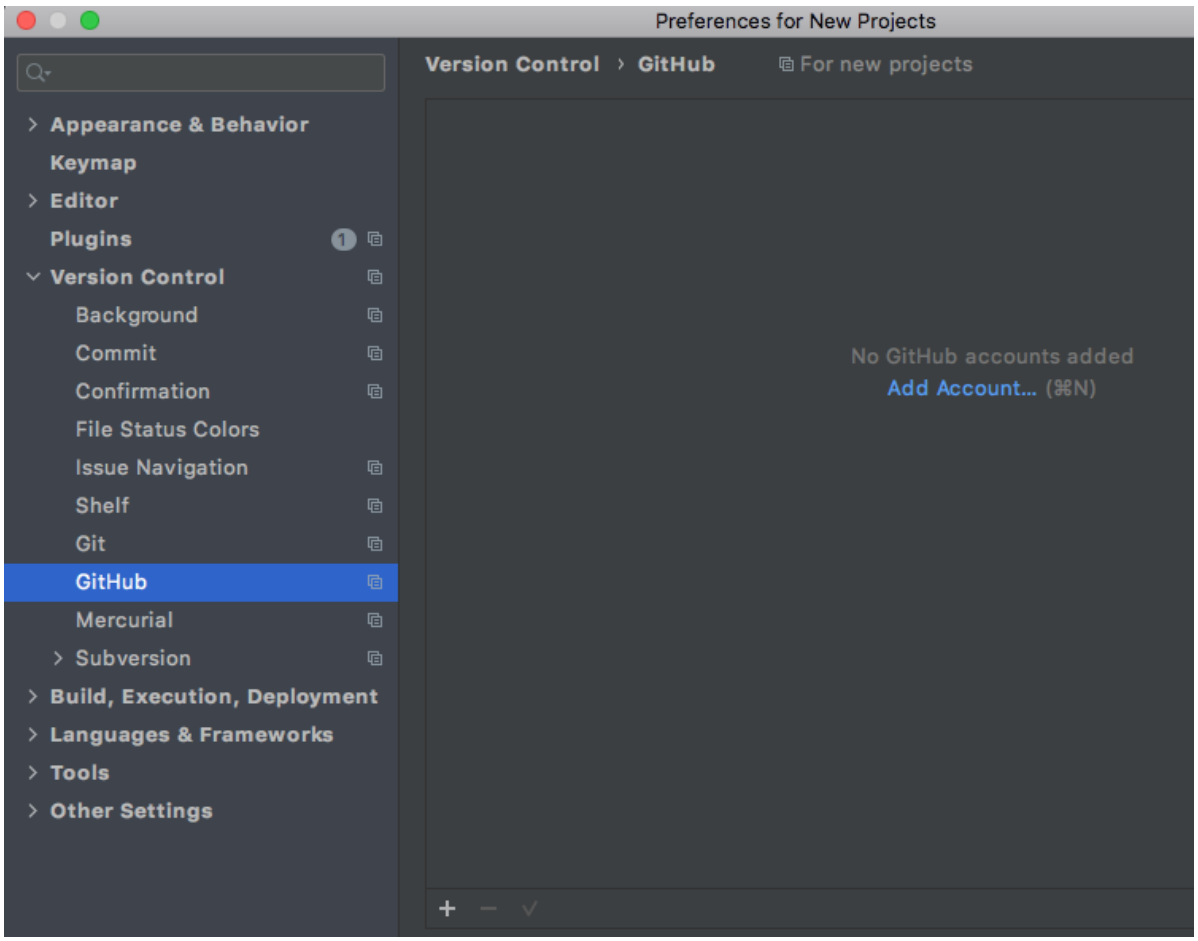
6. Hacemos click en el botón **Create repository.** Para crear un repositorio con los parámetros indicados.

7. Y obtenemos nuestro nuevo repositorio.

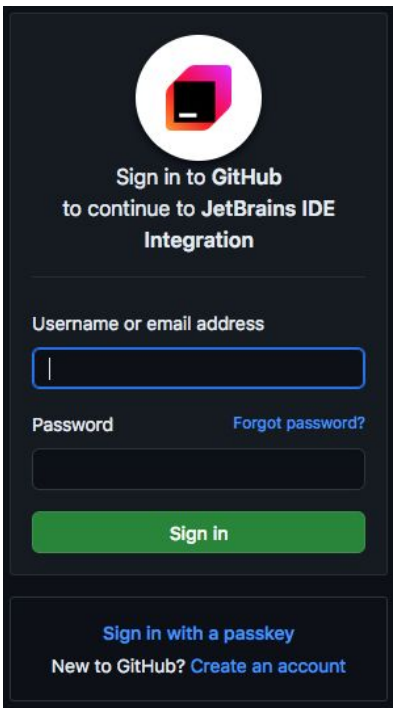


8. Ahora nos vamos a nuestro IDE, IntelliJ IDEA

   - En las **preferencias del IDE → Version Control → GitHub**
   - **Add Account o +**



   - Autorizamos a JetBrains en GitHub, si no estamos validados en GitHub nos pedirá usuario y contraseña.

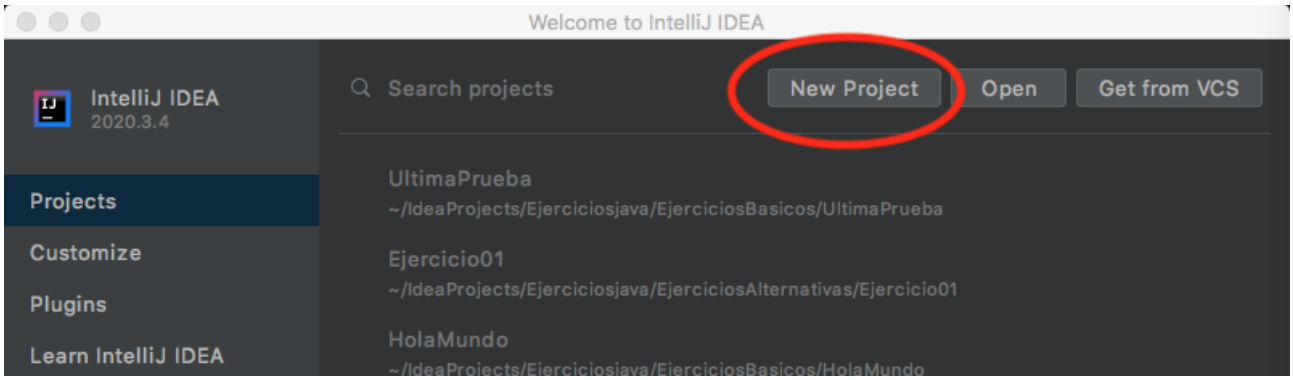


- Una vez estemos autorizados tendremos nuestra cuenta de GitHub vinculada con nuestro IDE IntelliJ IDEA CE

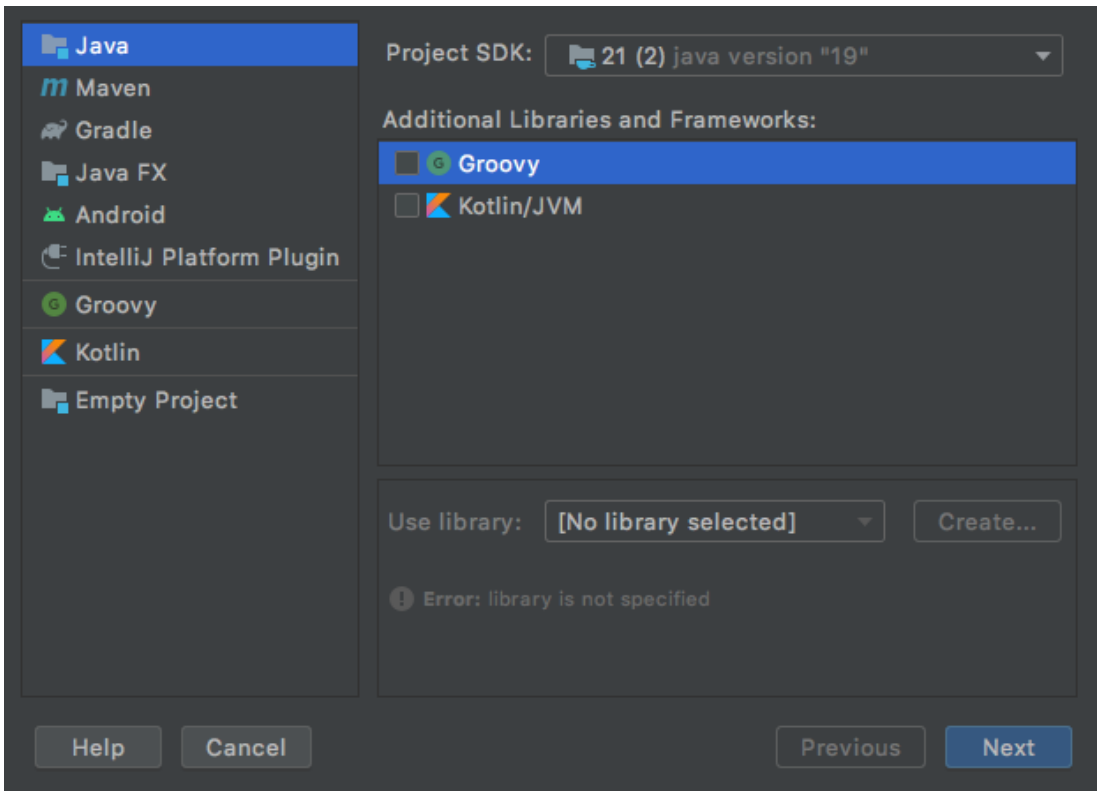
IMPORTANTE: Todos los pasos anteriores solo se deben hacer una vez. A partir de este momento vamos a crear nuestros proyectos, **uno por ejercicio**, y a guardarlos en la carpeta que hemos creado, se pueden guardar a su vez en carpetas, por tipo de ejercicios, simples, alternativas, bucles, vectores,…

Desde IntelliJ IDEA CE (Recordatorio de como crear un proyecto)

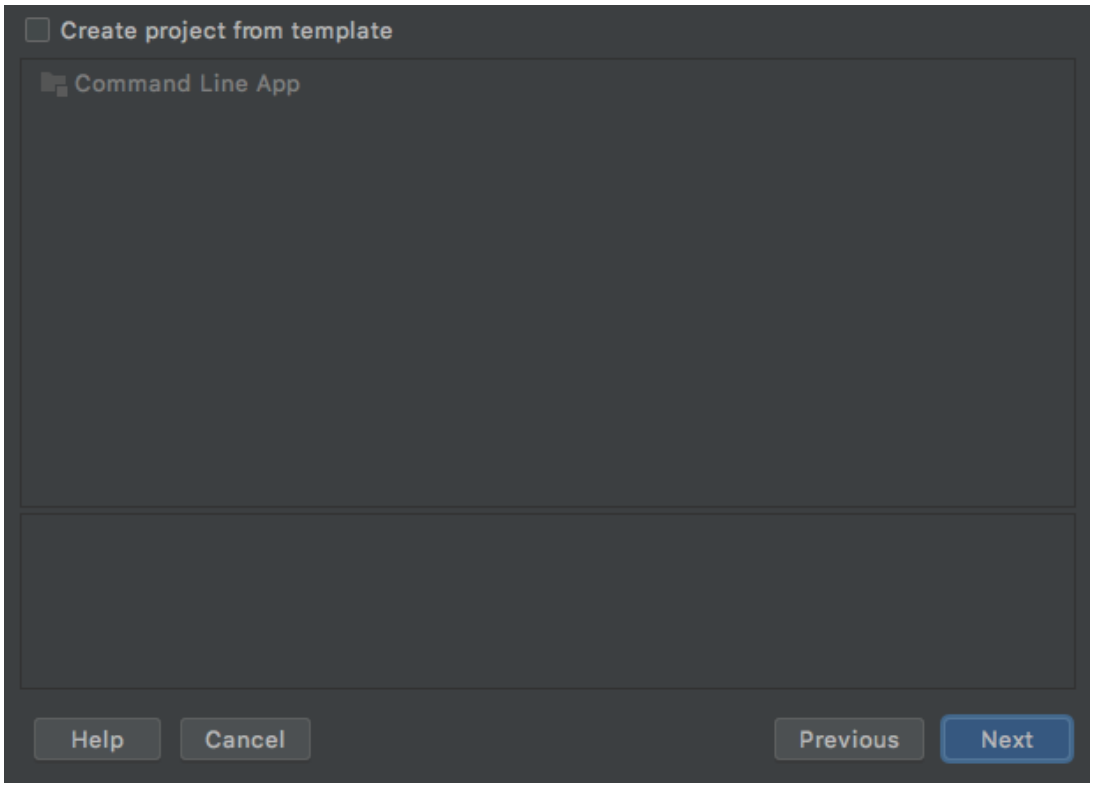
1.- Abrimos IntelliJ y pulsamos sobre "**new project**" o **File → New →Project**



2.- Seleccionamos de entre las opciones de la izquierda **Java**, selecionamos el Project SDK: **19, 21, la que tengamos instalada, pero 19 o superior** y el botón **Next**



3. -Seleccionamos el check "**Create project from template**" para que nos cree el método main() y **Next** de nuevo.

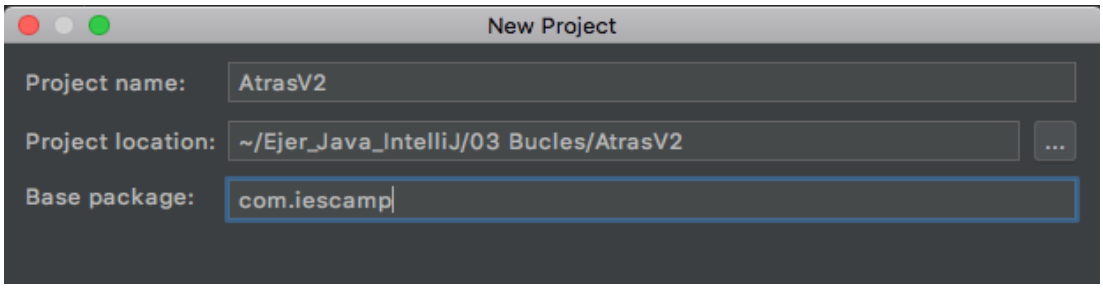


4.- Le damos **nombre a nuestro proyecto** (HolaMundo, Saludo, CuentaAtras, etc), recuerda que debe comenzar en mayúscula, no debe haber espacios en blanco, no es NADA recomendable utilizar acentos, y si hay más de una palabra, también comienzan en mayúscula.

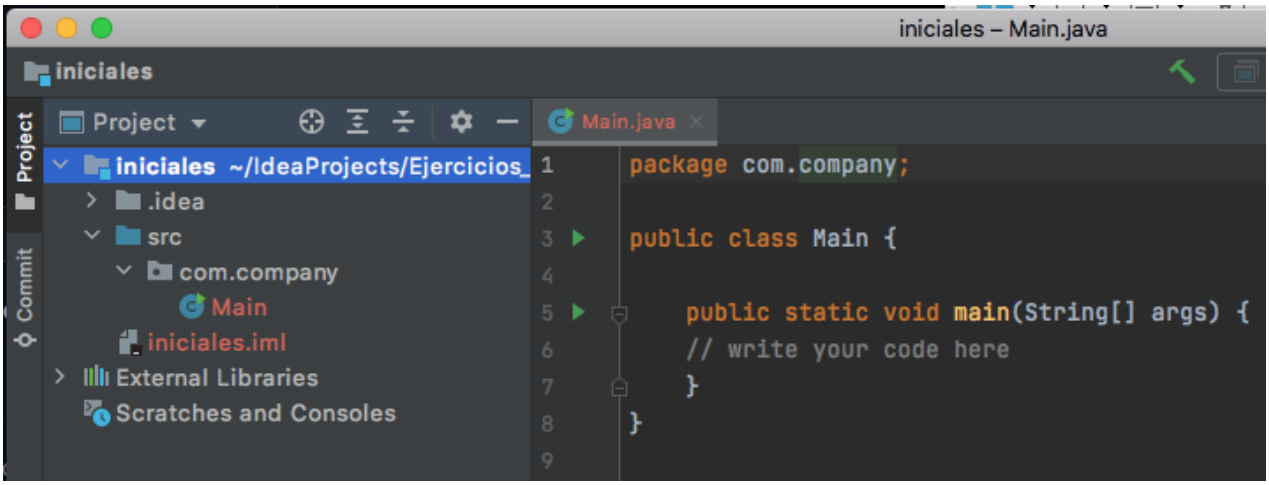
Seleccionamos el directorio donde se guardará el proyecto (si es necesario, para posteriores proyectos de la misma carpeta no será necesario), vamos a crear una carpeta dentro de Ejercicios_PRG (nuestro repositorio local) llamada **iniciales**.

IMPORTANTE: en este paso se puede cambiar el nombre del proyecto, revisar !

Modificamos el paquete “Base package” a com.iescamp



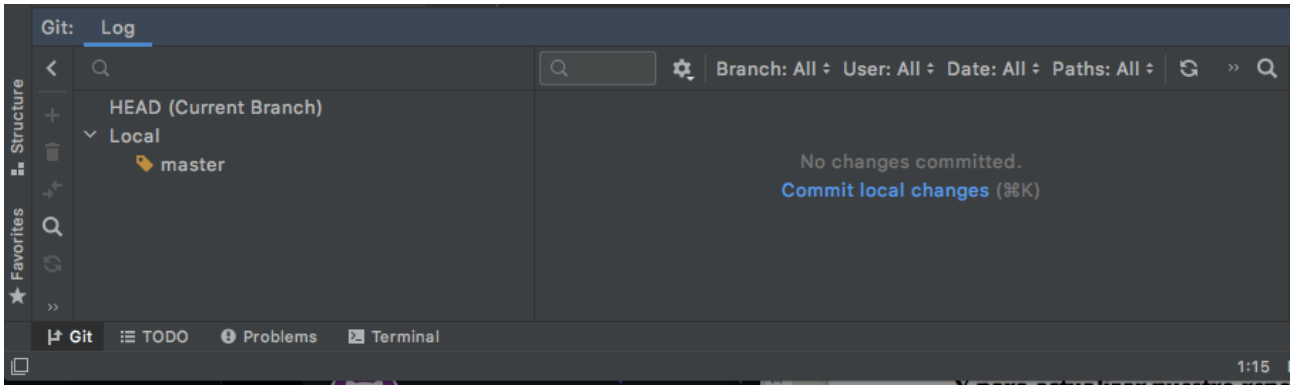
Pulsamos **Finish** y se crea el proyecto, con la estructura y carpetas correspondientes. Podemos abrir el proyecto en otra ventana o reemplazar la que tenemos abierta.



Ahora escribiriamos nuestro código, lo depuramos, lo probaríamos, etc.

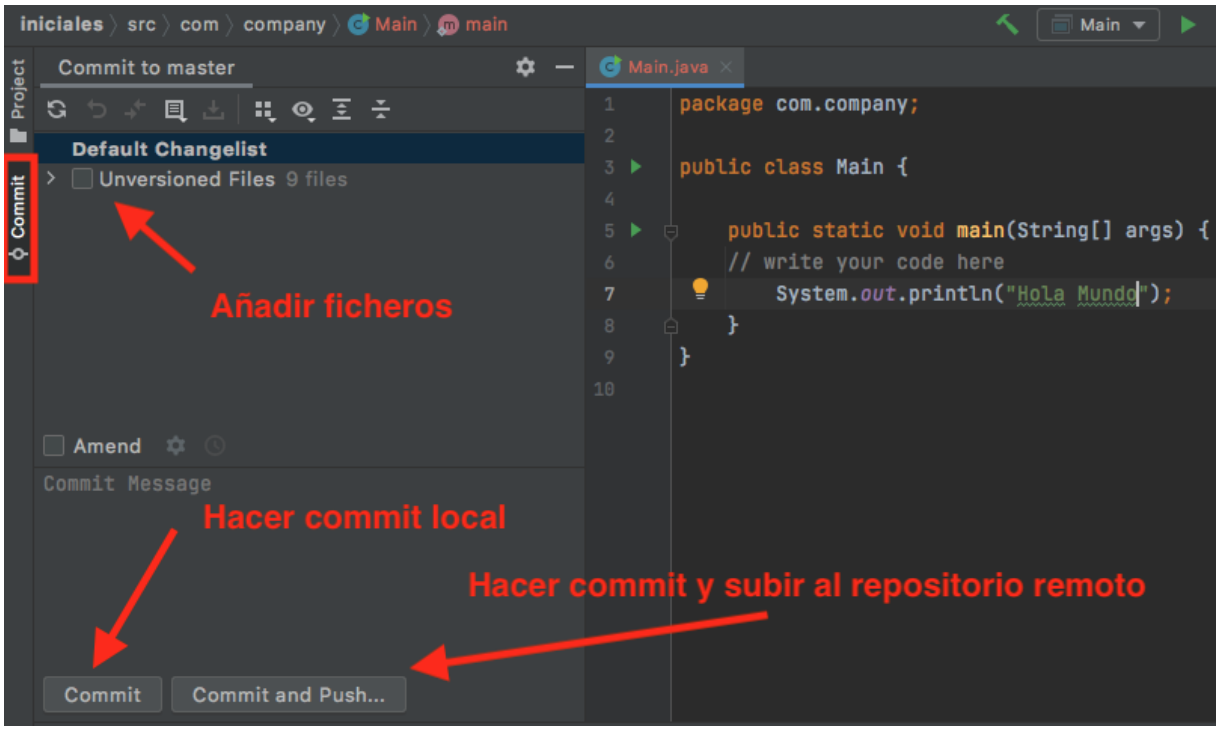
Para actualizar nuestro repositorio local harámos un git add y un git commit

Desde IntelliJ, abajo a la izquierda podemos ver la información del repositorio:



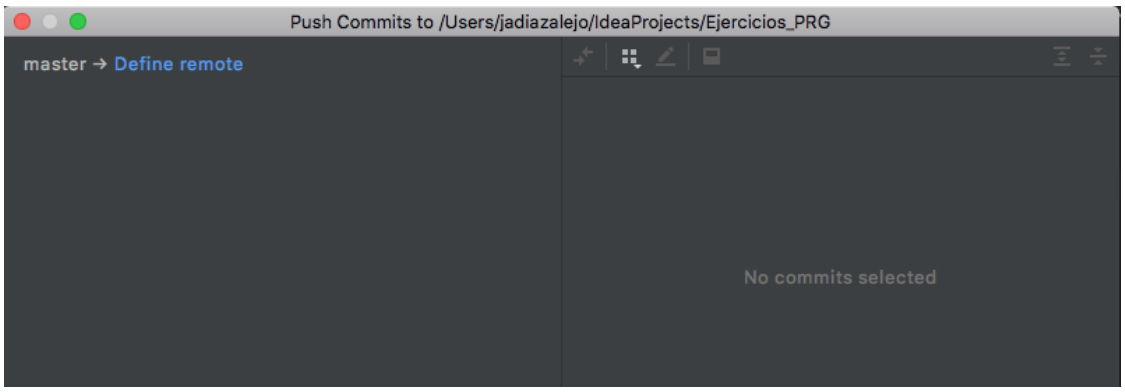
Y desde la opción **Commit** de la izquierda podemos comenzar el proceso de actualización, se puede hacer la actualización local y luego la remota o se puede hacer a la vez la actualización local y el repositorio remoto:

- Paso 1: añadiremos los ficheros a actualizar.
- Paso 2: Escribir un mensaje que nos indique en que punto del desarrollo nos encontramos.
- Paso 3: Pulsar el botón Commit and Push

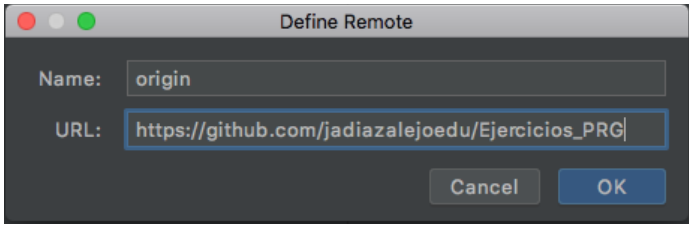


Si hacemos solo la actualización del repositorio local “Commit”, se actualizarán todos los ficheros seleccionados en nuestro repositorio local y podremos ver en la ventana de git el commit realizado.

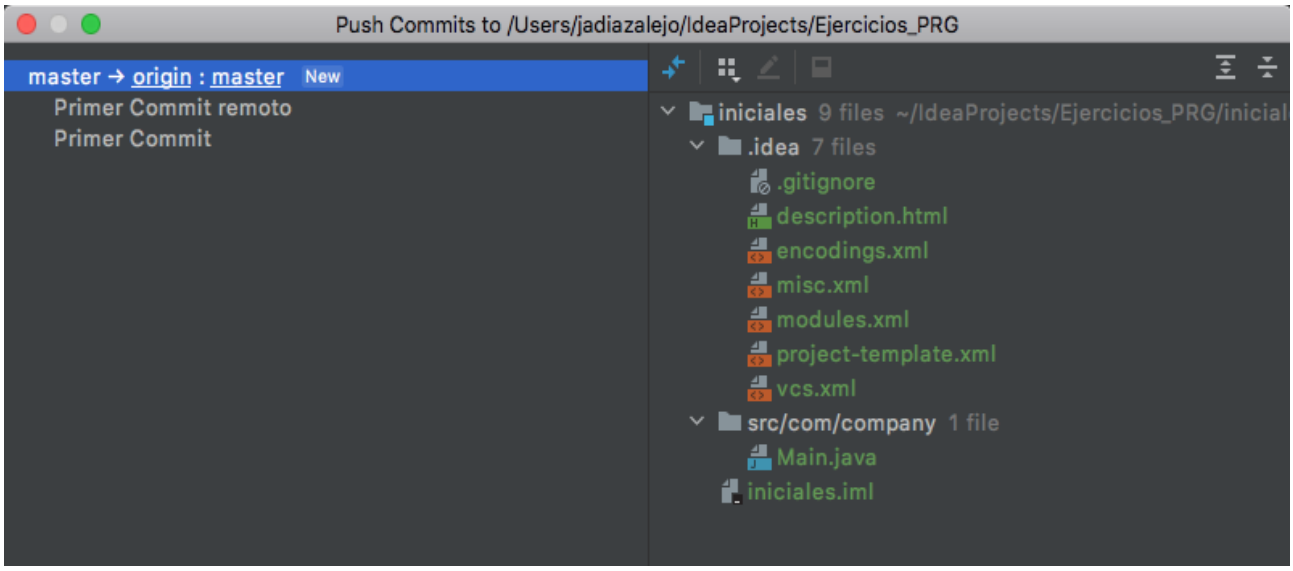
Al hacer el Commit en local y en remoto a la vez, después de pulsar el botón **Commit and Push** nos pedirá que definamos el repositorio remoto (si no lo hemos hecho ya):



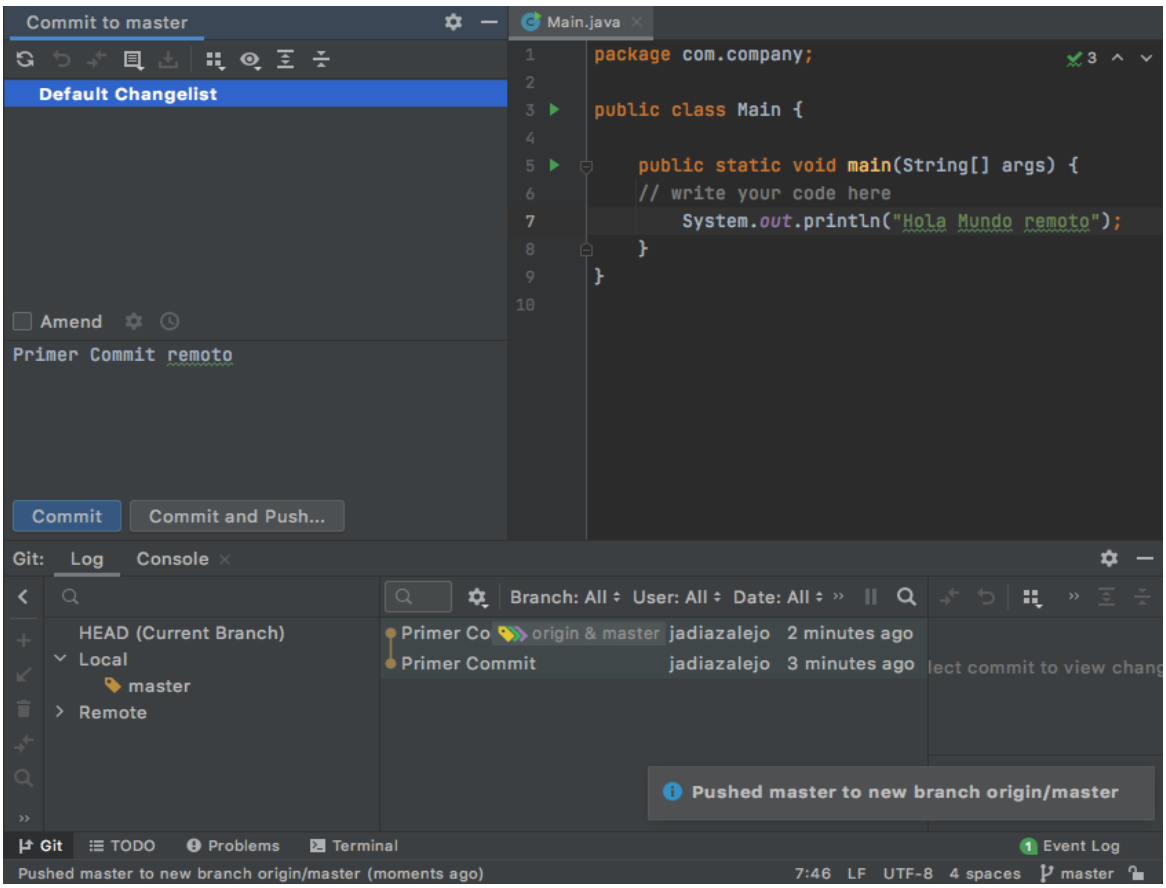
Nos pedirá la URL del repositorio en GitHub (podemos ir a GitHub ver el repositorio y copiar su URL)



Hará la conexión y nos ofrecerá hacer el Push de los ficheros seleccionados:

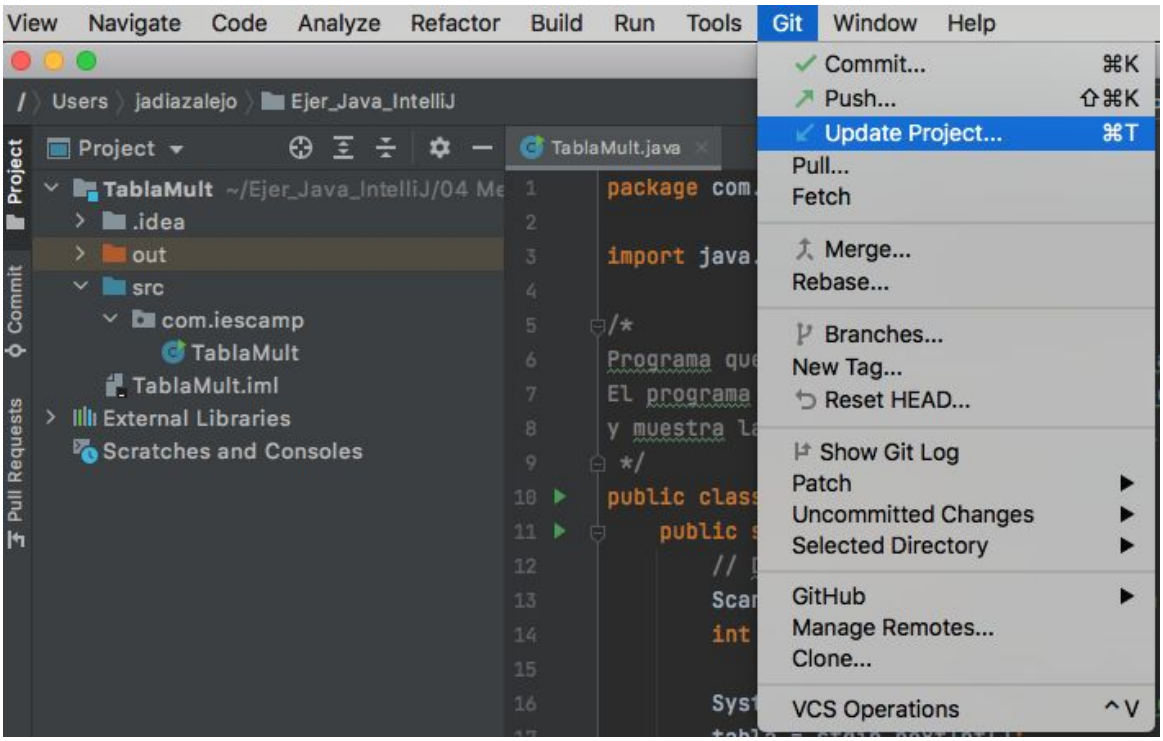


En la parte inferior de IntelliJ podrás ver el estado de los repositorios, tanto local como remoto



MUY MUY IMPORTANTE: SI TRABAJAMOS EN ORDENADORES DISTINTOS Y EL MISMO REPOSITORIO REMOTO, TENDREMOS QUE:

SIEMPRE que comencemos a trabajar debemos actualizar nuestro repositorio local, para ello hacemos **Git → Update Project**



Y SIEMPRE que acabemos de trabajar debemos actualizar nuestro repositorio remoto, para ello podemos hacer un **Commit and Push** o simplemente hacemos **Git → Push**