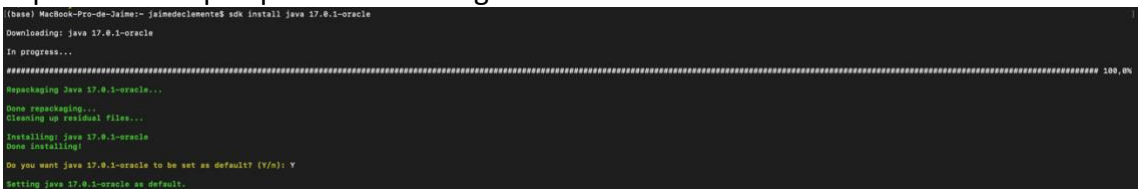
Jaime de Clemente Fernández-Picazo
3ºB GITT+BA 30 de enero de 2022

PRÁCTICA 1: PLATAFORMA GITHUB Y GIT

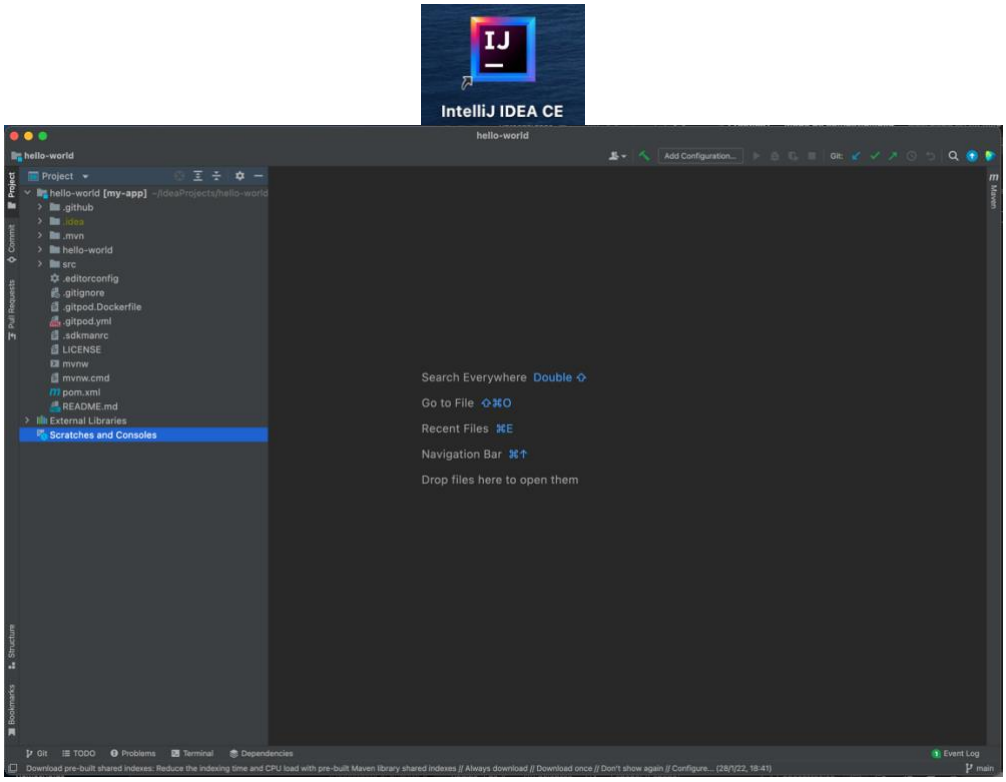
Para empezar, descargamos SDKMan, que será el gestor de programas que utilizaremos para descargar después el resto de programas que necesitamos:

Hecho esto, pasamos a descargar la versión actual de Java a través de SDK. Le especificamos que queremos descargar la versión 17.0.1 de Oracle:



Ahora, descargamos Maven también a través de SDK:

La descarga de IntelliJIDEA se hace desde su propia web, descargando la versión para MAC. El resultado es:



Una vez descargados todos los programas necesarios, pasamos al desarrollo de la práctica como tal, que consiste en familiarizarse con el entorno git y el uso de GitHub. Lo primero que haremos es clonar el repositorio hello-world que hemos creado para esta práctica:

```
(base) MacBook-Pro-de-Jaime:hello-world jaimedeclemente$ git clone https://github.com/gitt-3-pat/hello-world
Cloning into 'hello-world'...
remote: Enumerating objects: 38, done.
remote: Counting objects: 100% (38/38), done.
remote: Compressing objects: 100% (23/23), done.
remote: Total 38 (delta 1), reused 31 (delta 0), pack-reused 0
Receiving objects: 100% (38/38), 58.97 KiB | 1006.00 KiB/s, done.
Resolving deltas: 100% (1/1), done.
(base) MacBook-Pro-de-Jaime:hello-world jaimedeclemente$
```

Ahora, podemos ver el estado del repositorio utilizando el comando ‘git status’. Para este ejemplo, he creado un nuevo fichero, ‘PruebaJaime.txt’, que he dejado dentro del espacio del repositorio. Por ello, al hacer el comando antes comentado obtenemos:

```
(base) MacBook-Pro-de-Jaime:hello-world jaimedeclemente$ git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        PruebaJaime.txt

nothing added to commit but untracked files present (use "git add" to track)
(base) MacBook-Pro-de-Jaime:hello-world jaimedeclemente$
```

Como puede verse, nos indica que estamos al día con la rama ‘main’ pero que tenemos un fichero que no está correctamente añadido. El propio git nos indica que deberíamos utilizar el comando ‘git add’ para añadir el fichero al repositorio. Lo hacemos:

```
(base) MacBook-Pro-de-Jaime:hello-world jaimedeclemente$ git add .
(base) MacBook-Pro-de-Jaime:hello-world jaimedeclemente$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   PruebaJaime.txt

(base) MacBook-Pro-de-Jaime:hello-world jaimedeclemente$
```

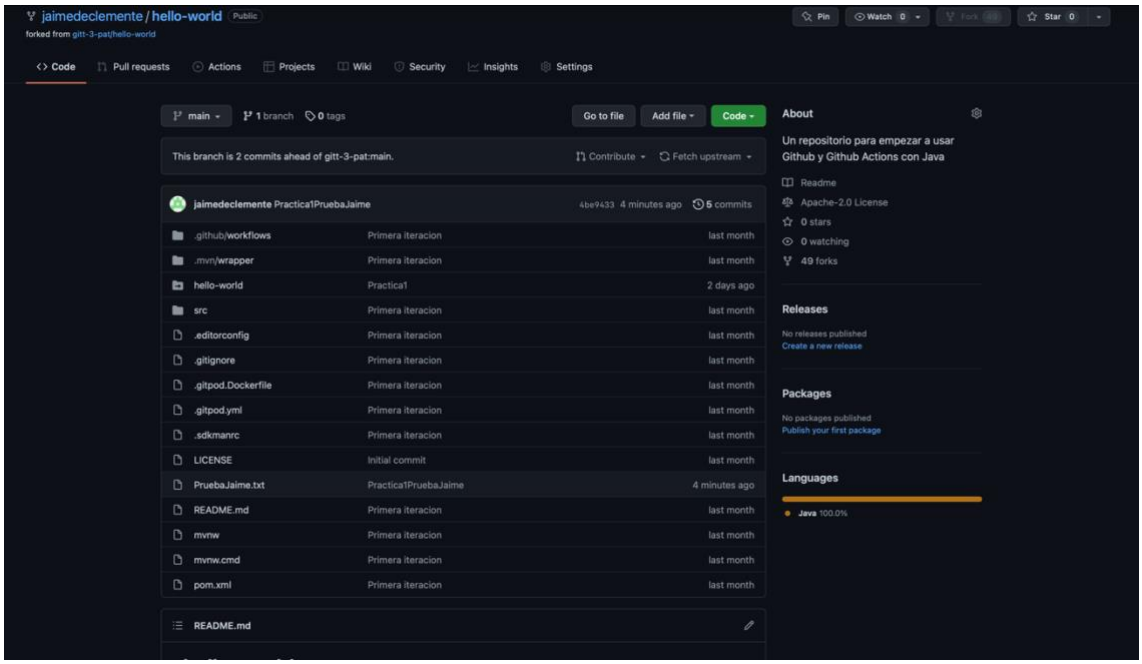
Ahora, ya tenemos el fichero añadido al repositorio. Sin embargo, nos quedan un par de pasos para hacer que éste sea accesible al público. Primero, hacemos un ‘git commit’ para subir el archivo creado al repositorio local:

```
(base) MacBook-Pro-de-Jaime:hello-world jaimedeclemente$ git commit -m "Practica1PruebaJaime"
[main 4be9433] Practica1PruebaJaime
 1 file changed, 3 insertions(+)
 create mode 100644 PruebaJaime.txt
(base) MacBook-Pro-de-Jaime:hello-world jaimedeclemente$
```

Una vez que ya tenemos el archivo añadido al repositorio local, utilizamos el comando ‘git push’ para subirlo al repositorio público:

```
(base) MacBook-Pro-de-Jaime:hello-world jaimedeclemente$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 401 bytes | 401.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/jaimedeclemente/hello-world.git
   68ffab4..4be9433  main -> main
(base) MacBook-Pro-de-Jaime:hello-world jaimedeclemente$
```

Gracias a estos comandos, ahora podríamos ver nuestro fichero creado en el portal de GitHub:



Ahora que ya trabajamos con ficheros en git, probaremos a utilizar ramas. Haremos esto a través del comando ‘git checkout’. En primer lugar, crearemos una rama (feature/1) y nos meteremos en ella:

```
(base) MacBook-Pro-de-Jaime:hello-world jaimedeclemente$ git checkout -b feature/1
Switched to a new branch 'feature/1'
(base) MacBook-Pro-de-Jaime:hello-world jaimedeclemente$
```

Ahora, nos saldremos de la rama en la que estábamos para volver a la principal:

```
(base) MacBook-Pro-de-Jaime:hello-world jaimedeclemente$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
(base) MacBook-Pro-de-Jaime:hello-world jaimedeclemente$
```