

PRÁCTICA 1: PLATAFORMA GITHUB Y GIT



Emilio Kenji Hernández Kuramata

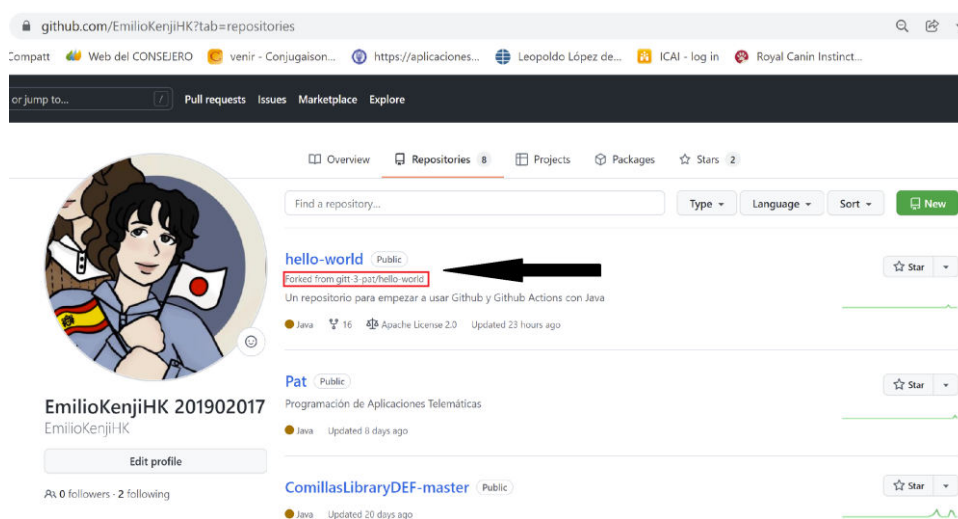
Objetivo de la Práctica

Tener unas nociones de como usar Github como plataforma de desarrollo y Git como sistema de control de versiones de código fuente de Software.

Prerrequisitos

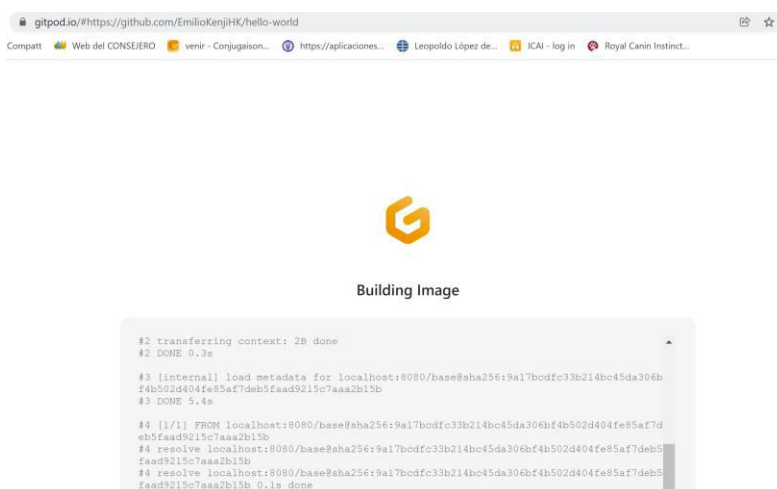
El alumno tendrá que tener cuenta en Github y con su cuenta de usuario, hará un “fork” sobre el repositorio:

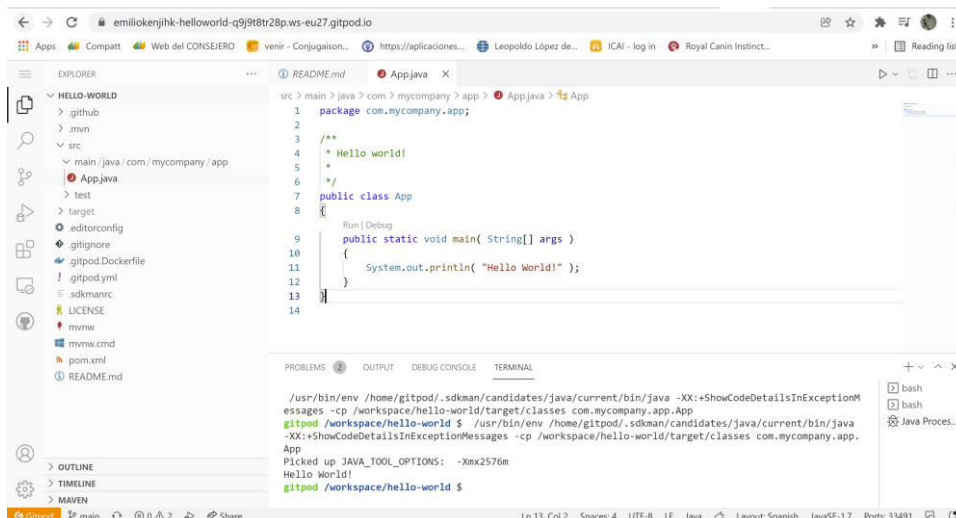
<https://github.com/gitt-3-pat/hello-world>



Una vez el alumno tiene el repositorio en su cuenta, es necesario abrir un entorno Gitpod de la siguiente manera:

gitpod.io/#https://github.com/ALUMNO-GITT-PAT/hello-world





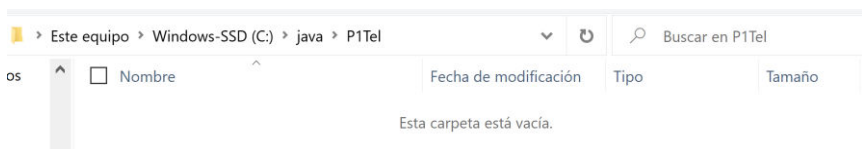
Desarrollo de la Práctica

Los comandos a probar son:

1) git clone <https://github.com/EmilioKenjiHK/hello-world>

Se utiliza este comando para descargar (o clonar) un repositorio (en este caso, el mío propio) y poder modificar y usar directamente los ficheros dentro sin tener que perder tiempo conectándolos al Github.

Primero se debe crear un archivo nuevo para guardar el repositorio:



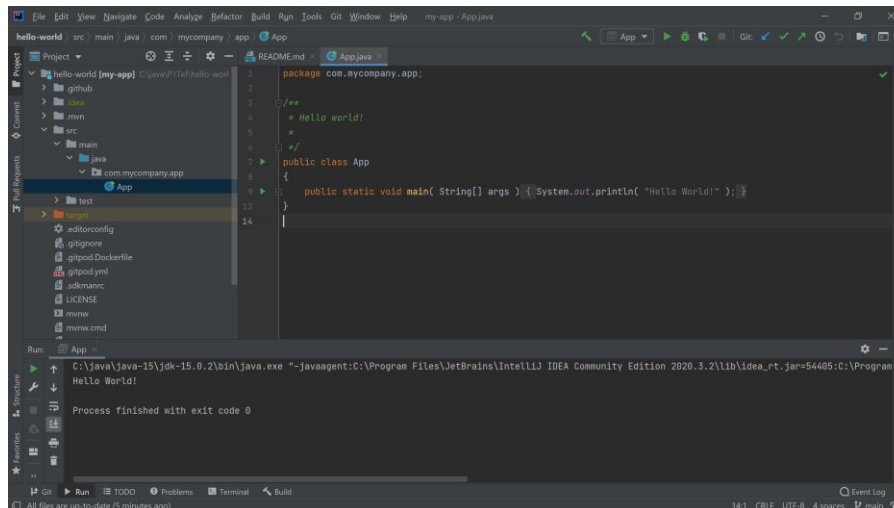
Ejecutando el comando donde es necesario:

```
C:\Users\Emili>cd /
C:\>cd java/P1Tel
C:\java\P1Tel>git clone https://github.com/EmilioKenjiHK/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 0R
Receiving objects: 100% (38/38), 58.97 KiB | 2.68 MiB/s, done.
Resolving deltas: 100% (1/1), done.
C:\java\P1Tel>
```

Se comprueba que el repositorio está efectivamente en el archivo:

Nombre	Fecha de modificación	Tipo	Tamaño
.github	20/01/2022 13:40	Carpeta de archivos	
.mvn	20/01/2022 13:40	Carpeta de archivos	
src	20/01/2022 13:40	Carpeta de archivos	
.editorconfig	20/01/2022 13:40	Archivo EDITORC...	1 KB
.gitignore	20/01/2022 13:40	Documento de tex...	1 KB
.gitpod.Dockerfile	20/01/2022 13:40	Archivo DOCKERFI...	1 KB
.gitpod.yml	20/01/2022 13:40	Archivo YML	1 KB
.sdkmanrc	20/01/2022 13:40	Archivo SDKMAN...	1 KB
LICENSE	20/01/2022 13:40	Archivo	12 KB
mvnw	20/01/2022 13:40	Archivo	11 KB
mvnw	20/01/2022 13:40	Script de comand...	7 KB
pom	20/01/2022 13:40	Documento XML	3 KB
README	20/01/2022 13:40	Archivo MD	1 KB

Podemos comprobar que se puede abrir y usar en IntelliJ:



2) git status

Sirve para comprobar el estado de la carpeta del proyecto y la zona del “stage” (será muy necesario para poder ejecutar el siguiente comando).

```
C:\java\P1Tel>cd hello-world

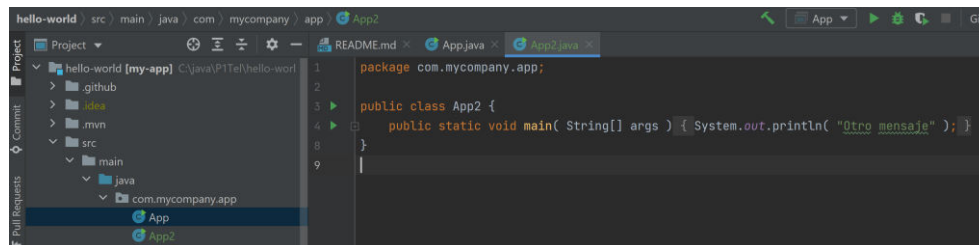
C:\java\P1Tel\hello-world>git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
```

Se puede observar que todo está actualizado y que no tenemos ninguna actualización preparada para enviar a Github.

3) git add .

ES el comando que va a mover al índice las modificaciones que esté realizado en el proyecto. Para poder ver este comando en acción, se va a crear un nuevo fichero "App2.java", en el cual actualizaremos al Github:



Ejecutando el comando:

```
C:\java\P1Tel\hello-world>git add .
warning: LF will be replaced by CRLF in src/main/java/com/mycompany/app/App2.java.
The file will have its original line endings in your working directory

C:\java\P1Tel\hello-world>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:   src/main/java/com/mycompany/app/App2.java

C:\java\P1Tel\hello-world>
```

Podemos ver en status que hay un cambio que debe actualizar al repositorio de Github.

4) git commit + git push

Commit se utiliza para confirmar una instantánea del directorio y Push sirve para cargar el contenido del proyecto (en IntelliJ) a un repositorio remoto (el repositorio en mi Github para este caso).

```
C:\java\P1Tel\hello-world>git commit -m "Prueba commit"
[main 0376844] Prueba commit
1 file changed, 8 insertions(+)
create mode 100644 src/main/java/com/mycompany/app/App2.java

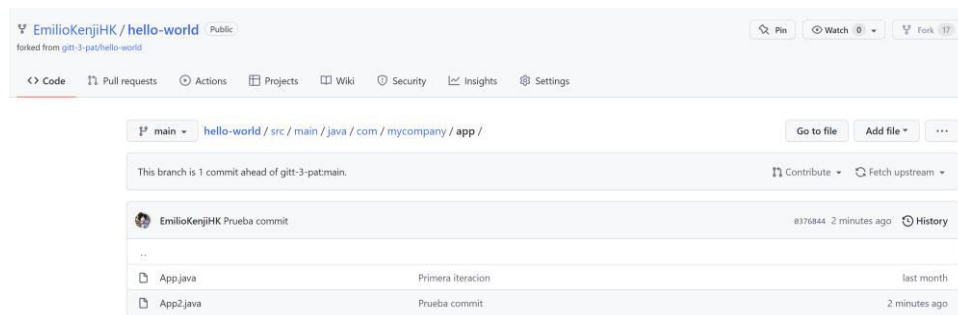
C:\java\P1Tel\hello-world>git push
info: please complete authentication in your browser...
Enumerating objects: 16, done.
Counting objects: 100% (16/16), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (9/9), 708 bytes | 236.00 KiB/s, done.
Total 9 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/EmilioKenjiHK/hello-world
  48fe276..0376844  main -> main

C:\java\P1Tel\hello-world>git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

C:\java\P1Tel\hello-world>
```

Se comprueba en Github si se ha actualizado el repositorio con el mensaje del “commit” puesto:

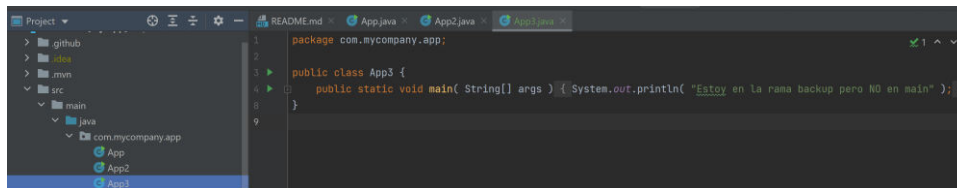


5) git checkout

Este comando sirve para poder cambiar entre que rama se está observando el repositorio. Ahora mismo, se está mirando por defecto la rama “main”. Para poder ver este comando, tenemos que crear una nueva rama:

```
C:\java\P1Tel\hello-world>git checkout -b rama0
Switched to a new branch 'rama0'
```

En la rama “rama0”, voy a crear un nuevo fichero java que se encontrará SÓLO en esa rama:



Una vez que se ha actualizado todo en el repositorio Github, se puede comprobar que este es el caso:

```
C:\java\P1Tel\hello-world>git checkout main
Already on 'main'
Your branch is up to date with 'origin/main'.

C:\java\P1Tel\hello-world>dir src\main\java\com\mycompany\app
El volumen de la unidad C es Windows-SSD
El número de serie del volumen es: 86A8-51BA

Directorio de C:\java\P1Tel\hello-world\src\main\java\com\mycompany\app

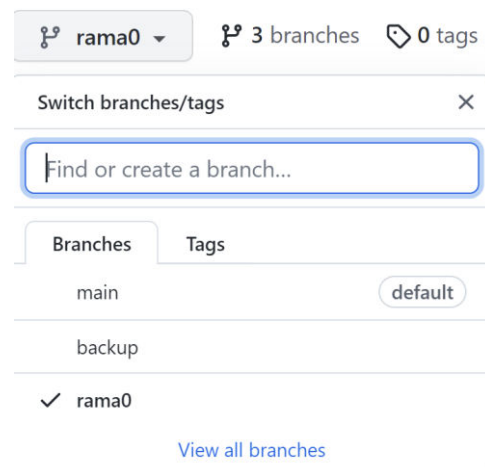
20/01/2022 17:05 <DIR>      ..
20/01/2022 17:05 <DIR>      .
20/01/2022 13:40          193 App.java
20/01/2022 15:01          153 App2.java
                2 archivos          346 bytes
                2 dirs 859.320.066.048 bytes libres

C:\java\P1Tel\hello-world>git checkout rama0
Switched to branch 'rama0'

C:\java\P1Tel\hello-world>dir src\main\java\com\mycompany\app
El volumen de la unidad C es Windows-SSD
El número de serie del volumen es: 86A8-51BA

Directorio de C:\java\P1Tel\hello-world\src\main\java\com\mycompany\app

20/01/2022 17:06 <DIR>      ..
20/01/2022 17:06 <DIR>      .
20/01/2022 13:40          193 App.java
20/01/2022 15:01          153 App2.java
20/01/2022 17:06          188 App3.java
                3 archivos          534 bytes
                2 dirs 859.270.414.336 bytes libres
```



rama0 hello-world / src / main / java / com / mycompany / app /

This branch is 2 commits ahead of gitt-3-patmain.

EmilioKenjiHK Probando Checkout @78caea 15 minutes ago History

App.java	Primera iteracion	last month
App2.java	Prueba commit	2 hours ago
App3.java	Probando Checkout	15 minutes ago

main hello-world / src / main / java / com / mycompany / app /

This branch is 3 commits ahead of gitt-3-patmain.

EmilioKenjiHK Prueba commit 6676cfe 4 minutes ago History

App.java	Primera iteracion	last month
App2.java	Prueba commit	4 minutes ago

Entorno de desarrollo JDK

Instala en el equipo:

- Java 17
- Maven
- IntelliJ

```
C:\java\P1Tel\hello-world>java -version
java version "17.0.1" 2021-10-19 LTS
Java(TM) SE Runtime Environment (build 17.0.1+12-LTS-39)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.1+12-LTS-39, mixed mode, sharing)

C:\java\P1Tel\hello-world>mvn -version
Apache Maven 3.8.4 (9b656c72d54e5bacbed989b64718c159fe39b537)
Maven home: C:\apache-maven-3.8.4
Java version: 17.0.1, vendor: Oracle Corporation, runtime: C:\Program Files\Java\jdk-17.0.1
Default locale: es_ES, platform encoding: Cp1252
OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"
```



IntelliJ IDEA Community Edition 2020.... 15/03/2021 9:05

Acceso directo

2 KB

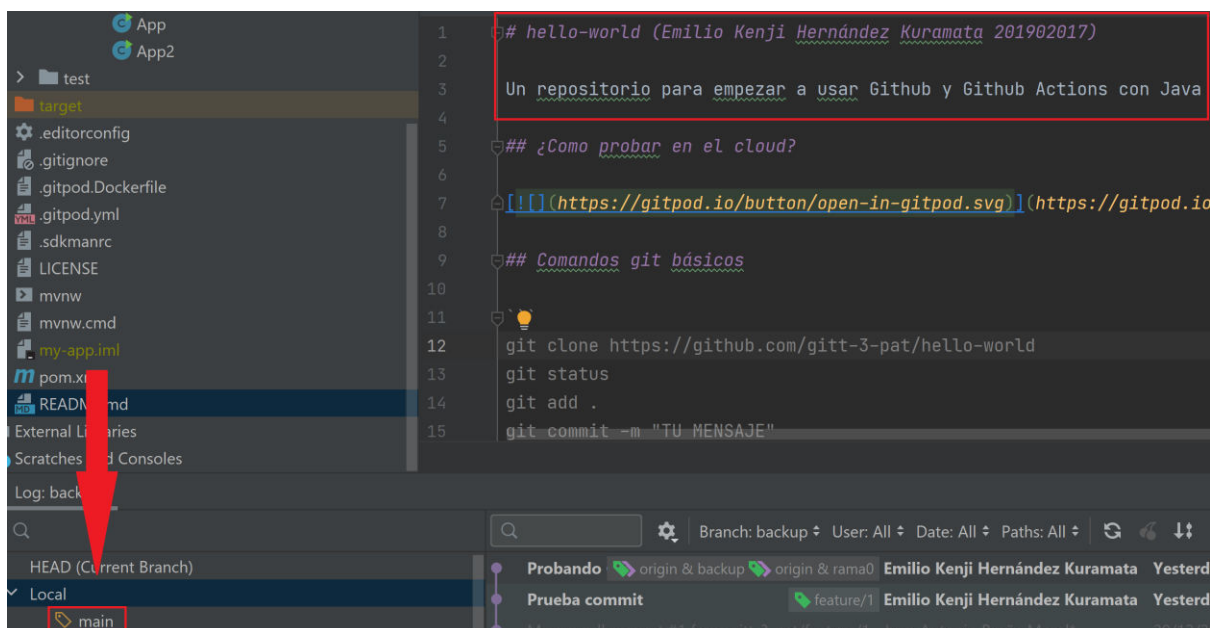
Extra: Comandos Útiles no Dadas

Aprovechando que hay muchos comandos en Git, vamos a demostrar el uso de algunos que podrían ser útiles para prácticas futuras:

1) git pull

Pull sirve para cargar el contenido de un repositorio remoto al proyecto local (en IntelliJ).

Por ejemplo, voy a modificar el READ ME.md directamente de GitHub:

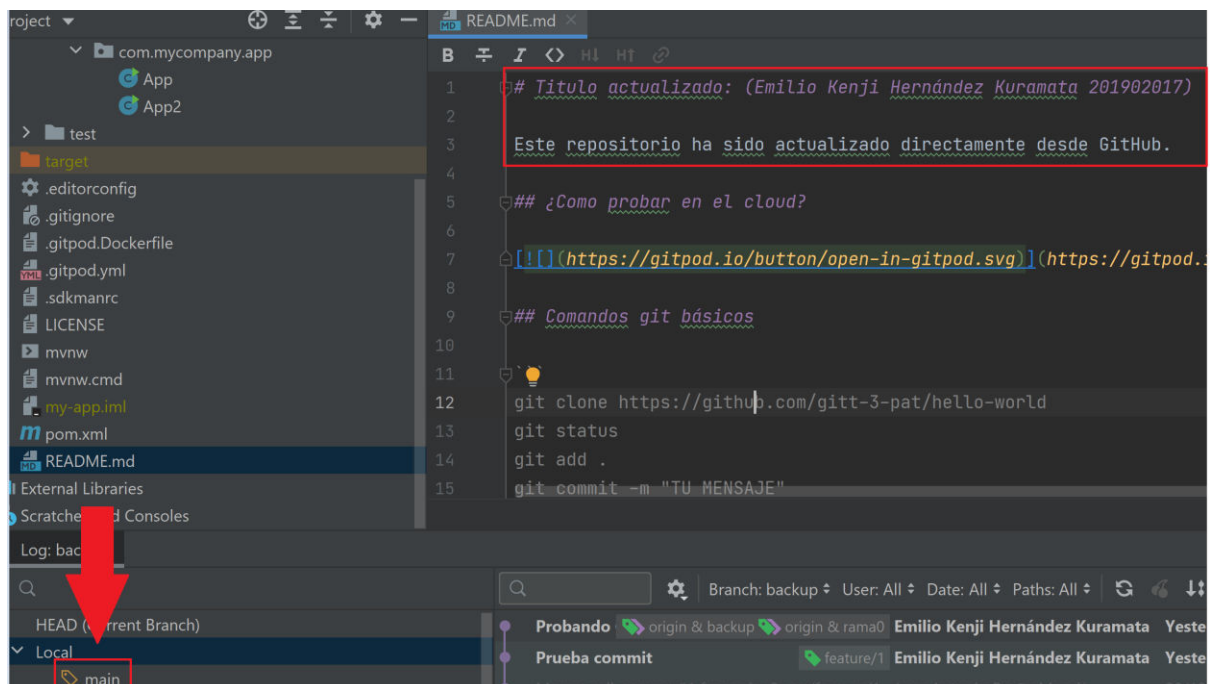


Esta es la versión inicial en IntelliJ (se observa que estamos en la misma rama que la versión actualizada del Github: main).

Una vez ejecutado el comando:

```
C:\java\P1Tel\hello-world>git pull
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 751 bytes | 39.00 KiB/s, done.
From https://github.com/EmilioKenjiHK/hello-world
   30cc694..4ce9b71  main       -> origin/main
Updating 30cc694..4ce9b71
Fast-forward
 README.md | 4 ++--
 1 file changed, 2 insertions(+), 2 deletions(-)

C:\java\P1Tel\hello-world>
```



Se observa que se ha actualizado correctamente el fichero en IntelliJ.

2) git log

Este comando es útil si se quiere observar la historial de todos los “commits” ejecutados de un proyecto:

```
C:\java\P1Tel\hello-world>git log
commit 4ce9b7195abff76996f0b99d3612fd5a81ee9fd0 (HEAD -> main, origin/main, origin/HEAD)
Author: EmilioKenjiHK 201902017 <91135161+EmilioKenjiHK@users.noreply.github.com>
Date:   Fri Jan 21 19:05:59 2022 +0100

    Update README.md

commit 30cc694f6ec47380434e200865f0af6d5433e9db
Author: EmilioKenjiHK 201902017 <91135161+EmilioKenjiHK@users.noreply.github.com>
Date:   Fri Jan 21 18:01:34 2022 +0100

    Update README.md

commit f4060df496746076fedf06998ed0327c6a37827e
Merge: 6676cff 0376844
Author: Emilio Kenji Hernández Kuramata <emilio.her.kun@gmail.com>
Date:   Thu Jan 20 17:10:07 2022 +0100

    Merge remote-tracking branch 'origin/main'

commit 6676cff7aef669da769f3dc80f9f75c1f36e13a5
Author: Emilio Kenji Hernández Kuramata <emilio.her.kun@gmail.com>
Date:   Thu Jan 20 15:15:18 2022 +0100

    Prueba commit

commit 0376844eac32d919784a01a2aaf1bc2ec3e880b (feature/1)
Author: Emilio Kenji Hernández Kuramata <emilio.her.kun@gmail.com>
Date:   Thu Jan 20 15:15:18 2022 +0100

    Prueba commit

commit 48fe2767a1dec83ae423cca0c0f8c92204ba6f65
Merge: 5038239 5b68377
Author: Juan Antonio Breña Moral <bren@juanantonio.info>
Date:   Mon Dec 20 19:26:31 2021 +0100

    Merge pull request #1 from gitt-3-pat/feature/1

    Primera iteracion
```

```
commit 5b683771defff11ef7f0086ef25c48a3cb95deed
Author: Juan Antonio Breña Moral <bren@juanantonio.info>
Date:   Mon Dec 20 19:25:31 2021 +0100

    Primera iteracion

commit 5038239e93d2da05a94667e8f9e66a16b5353076
Author: Juan Antonio Breña Moral <bren@juanantonio.info>
Date:   Mon Dec 20 19:14:40 2021 +0100

    Initial commit

C:\java\P1Tel\hello-world>
```

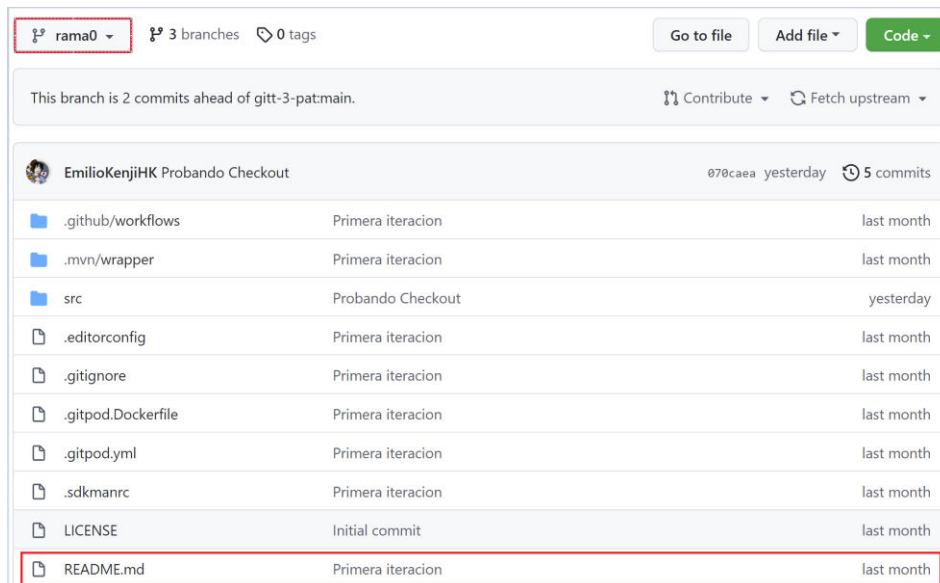
Si ejecutas “-- stat”, se puede observar cuantos bytes se han cambiado en ciertos ficheros:

```
.editorconfig | 14 ++
.github/workflows/build.yml | 16 ++
.gitignore | 26 +--
.gitpod.Dockerfile | 14 ++
.gitpod.yml | 9 +
.mvn/wrapper/MavenWrapperDownloader.java | 117 ++++++++
.mvn/wrapper/maven-wrapper.jar | Bin 0 -> 50710 bytes
.mvn/wrapper/maven-wrapper.properties | 2 +
.sdkmanrc | 1 +
README.md | 22 ++
mvnw | 310 ++++++++
mvnw.cmd | 182 ++++++++
pom.xml | 75 ++++++
src/main/java/com/mycompany/app/App.java | 13 ++
src/test/java/com/mycompany/app/AppTest.java | 20 ++
15 files changed, 798 insertions(+), 23 deletions(-)
```

3) git rm [archivo]

Este comando sirve para eliminar un fichero de la rama que se encuentra el usuario. Nota que, cuando se ejecuta, prepara un “commit” para ser actualizado al Github mediante “git push”.

Para este ejemplo, borraré el READ ME.md de la rama “rama0”:



```
C:\java\P1Tel\hello-world>git checkout rama0
Switched to branch 'rama0'
Your branch is up to date with 'origin/rama0'.

C:\java\P1Tel\hello-world>git rm README.md
rm 'README.md'

C:\java\P1Tel\hello-world>git status
On branch rama0
Your branch is up to date with 'origin/rama0'.

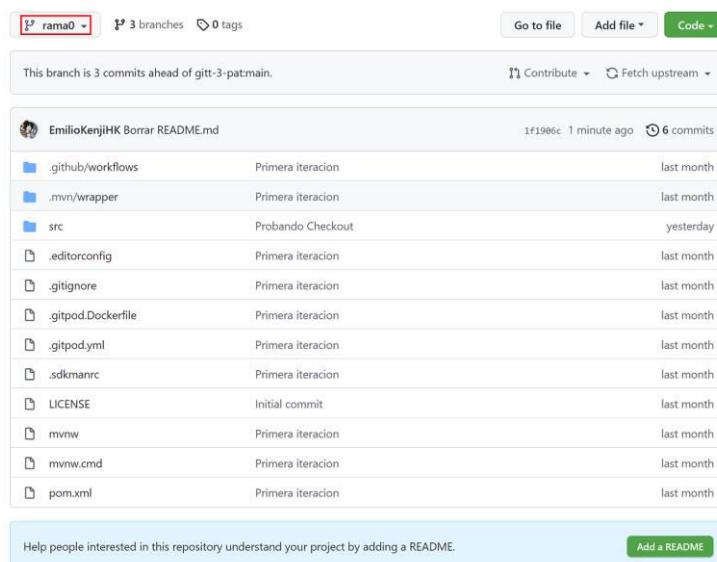
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    deleted:    README.md
```

```
C:\java\P1Tel\hello-world>git commit -m "Borrar README.md"
[rama0 1f1906c] Borrar README.md
1 file changed, 24 deletions(-)
delete mode 100644 README.md

C:\java\P1Tel\hello-world>git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 259 bytes | 259.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/EmilioKenjiHK/hello-world
070caea..1f1906c rama0 -> rama0

C:\java\P1Tel\hello-world>
```

Ahora vamos a ver que este archivo ya no está en la rama “rama0”:

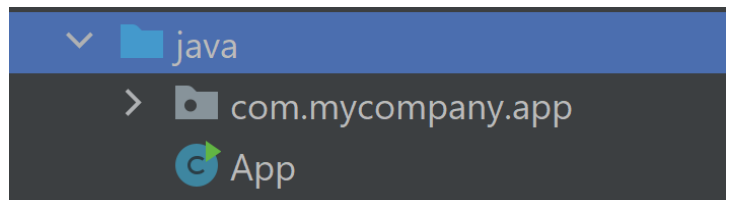
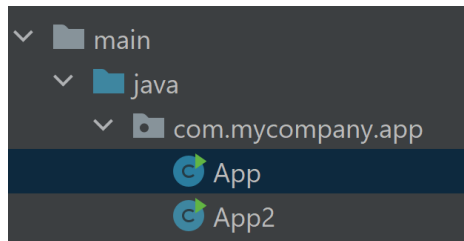


4) git mv [dirección existente] [nueva dirección]

Este comando sirve para mover un fichero. Funciona de manera parecida al “git rm”.

Como ejemplo, voy a sacar el App.java fuera del archivo “com/mycompany/app” de la rama “main”.

Es importante escribir la dirección entera:



```
C:\java\P1Tel\hello-world>git mv C:\java\P1Tel\hello-world\src\main\java\com\mycompany\app\App.java C:\java\P1Tel\hello-world\src\main\java\App.java

C:\java\P1Tel\hello-world>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)
        renamed:    src/main/java/com/mycompany/app/App.java -> src/main/java/App.java

C:\java\P1Tel\hello-world>git commit -m "Fichero movido"
[main 7242f5c] Fichero movido
 1 file changed, 0 insertions(+), 0 deletions(-)
 rename src/main/java/{com/mycompany/app => }/App.java (100%)

C:\java\P1Tel\hello-world>git push
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (8/8), 590 bytes | 196.00 KiB/s, done.
Total 8 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/EmilioKenjiHK/hello-world
   4ce9b71..7242f5c  main -> main

C:\java\P1Tel\hello-world>
```

main

hello-world / src / main / java /

Go to file


Add file

...

This branch is 6 commits ahead of gitt-3-pat:main.

Contribute

Fetch upstream



 EmilioKenjiHK

Fichero movido

7242f5c 9 minutes ago

History

..

 com/mycompany/app	Fichero movido	9 minutes ago
 App.java	Fichero movido	9 minutes ago