

# PRÁCTICA 1-Instalación de Herramientas

**Repositorio:** <https://github.com/202006359/202006359-GITT-PAT-practica-1>

## Verificación Docker

```
C:\>docker

Usage:  docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

Options:
  --config string      Location of client config files (default
                        "C:\\Users\\Alfredo\\.docker")
  -c, --context string  Name of the context to use to connect to the
                        daemon (overrides DOCKER_HOST env var and
                        default context set with "docker context use")
  -D, --debug           Enable debug mode
  -H, --host list       Daemon socket(s) to connect to
  -l, --log-level string Set the logging level
                        ("debug"|"info"|"warn"|"error"|"fatal")
                        (default "info")
  --tls                Use TLS; implied by --tlsverify
  --tlscacert string    Trust certs signed only by this CA (default
                        "C:\\Users\\Alfredo\\.docker\\ca.pem")
  --tlscert string      Path to TLS certificate file (default
                        "C:\\Users\\Alfredo\\.docker\\cert.pem")
  --tlskey string       Path to TLS key file (default
                        "C:\\Users\\Alfredo\\.docker\\key.pem")
  --tlsverify           Use TLS and verify the remote
  -v, --version         Print version information and quit

Management Commands:
  builder      Manage builds
  buildx*      Docker Buildx (Docker Inc., v0.10.0)
```

## Verificación JAVA

```
C:\>java -version
java version "17.0.5" 2022-10-18 LTS
Java(TM) SE Runtime Environment (build 17.0.5+9-LTS-191)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.5+9-LTS-191, mixed mode, sharing)
```

## Verificación MAVEN

```
C:\>mvn -version
Apache Maven 3.8.7 (b89d5959fcde851dcb1c8946a785a163f14e1e29)
Maven home: C:\PAT\apache-maven-3.8.7
Java version: 17.0.5, vendor: Oracle Corporation, runtime: C:\Program Files\Java\jdk-17.0.5
Default locale: es_ES, platform encoding: Cp1252
OS name: "windows 11", version: "10.0", arch: "amd64", family: "windows"
```

## Verificación GIT

```
C:\>git
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
        [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
        [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
        [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
        [--super-prefix=<path>] [--config-env=<name>=<envvar>]
        <command> [<args>]

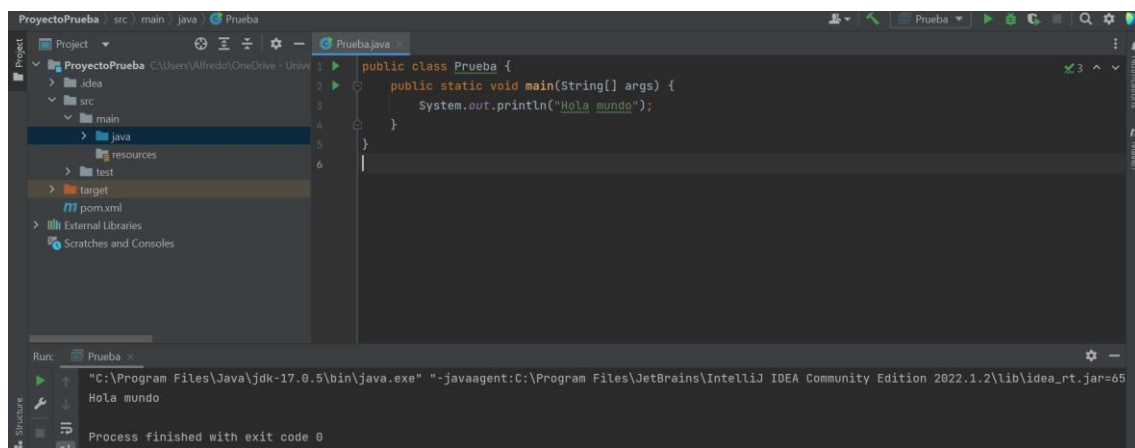
These are common Git commands used in various situations:

start a working area (see also: git help tutorial)
    clone             Clone a repository into a new directory
    init              Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)
    add               Add file contents to the index
    mv                Move or rename a file, a directory, or a symlink
    restore            Restore working tree files
    rm                Remove files from the working tree and from the index
    sparse-checkout    Initialize and modify the sparse-checkout

examine the history and state (see also: git help revisions)
    bisect            Use binary search to find the commit that introduced a bug
    diff              Show changes between commits, commit and working tree, etc
    grep              Print lines matching a pattern
    log               Show commit logs
```

## Verificación INTELij



## Explicación Comandos GIT

**git clone:** Este comando sirve para clonar un repositorio subido a Git y tenerlo en local en el ordenador. De esta forma podemos trabajar con él desde el PC.

```
C:\>git clone https://github.com/202006359/202006359-GITT-PAT-practica-1
Cloning into '202006359-GITT-PAT-practica-1'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (5/5), 4.84 KiB | 826.00 KiB/s, done.
```

**git status:** Este comando muestra el estado del repositorio Git, puede ser muy útil para saber si estamos trabajando con los últimos cambios y para poder ver si tenemos que hacer algún commit. Asimismo, también nos sirve para poder ver a que rama estamos apuntando; en el ejemplo, apuntamos a la rama “main”.

```
C:\202006359-GITT-PAT-practica-1>git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
```

**git add:** Este comando prepara un fichero que tengamos en local en nuestro

proyecto, es decir, añade un cambio del directorio de trabajo en el entorno de ensayo, en la siguiente confirmación que hagamos con el commit, ya se nos guardará en la capa de Staging, pero antes no. Para probarlo, he añadido un fichero .txt de prueba.

```
C:\202006359-GITT-PAT-practica-1>git status
On branch develop
Your branch is up to date with 'origin/develop'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   Prueba-ADD.txt

no changes added to commit (use "git add" and/or "git commit -a")

C:\202006359-GITT-PAT-practica-1>git add Prueba-ADD.txt

C:\202006359-GITT-PAT-practica-1>git status
On branch develop
Your branch is up to date with 'origin/develop'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   Prueba-ADD.txt
```

**git commit:** Este comando te actualiza los cambios más recientes junto con una pequeña descripción o comentario que lo acompaña, subiéndolo a la capa de Staging.

```
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch main
# Your branch is up to date with 'origin/main'.
#
Changes to be committed:
  new file:   Prueba-ADD.txt

```

C:/202006359-GITT-PAT-practica-1/.git/COMMIT\_EDITMSG [unix] (12:23 19/01/2023) 1,0-1 All

"C:/202006359-GITT-PAT-practica-1/.git/COMMIT\_EDITMSG" [unix] 10L, 263B

Subo un fichero .txt de prueba

```
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
[main b8ebad5] Subo un fichero .txt de prueba
1 file changed, 1 insertion(+)
 create mode 100644 Prueba-ADD.txt
# Changes to be committed:
C:\202006359-GITT-PAT-practica-1>t
```

**git push:** Con este comando, cargamos el archivo que hemos hecho commit y que ya está en la capa de Staging, y lo subimos al repositorio remoto en la nube de GitHub. La segunda imagen inferior, muestra como el fichero Prueba-ADD.txt ha sido subido al repositorio remoto correctamente.

```
C:\202006359-GITT-PAT-practica-1>git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 303 bytes | 303.00 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/202006359/202006359-GITT-PAT-practica-1
b8ebad5..f8dba27  main -> main
```

202006359 Subo una modificación del fichero .txt			f8dba27 1 minute ago	🕒 3 commits
📄 .gitignore	Initial commit			yesterday
📄 LICENSE	Initial commit			yesterday
📄 Prueba-ADD.txt	Subo una modificación del fichero .txt			1 minute ago
📄 README.md	Initial commit			yesterday

**git checkout:** Este comando tiene varias funciones: recuperar archivos, moverse entre ramas, volver hacia atrás con antiguos commits... En la imagen previa, hacemos uso de este comando para movernos a la rama develop.

```
C:\202006359-GITT-PAT-practica-1>git checkout develop
Switched to branch 'develop'
```

## Comandos Extra

**git branch new\_branch + git branch:** Con el primer comando, añadimos una nueva rama a nuestro repositorio. Con el segundo comando, listamos todas las ramas del repositorio y vemos a cual estamos apuntando. En el ejemplo inferior, creamos una rama nueva de develop. Posteriormente, vemos que se ha creado correctamente y que aún estamos apuntando a la rama main.

```
C:\202006359-GITT-PAT-practica-1>git branch develop
```

```
C:\202006359-GITT-PAT-practica-1>git branch
  develop
* main
```

**git push --set-upstream origin develop:** Una vez tenemos la rama develop creada, debemos hacer un push para subirla al repositorio remoto.

```
C:\202006359-GITT-PAT-practica-1>git push --set-upstream origin develop
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:   https://github.com/202006359/202006359-GITT-PAT-practica-1/pull/new/develop
remote:
To https://github.com/202006359/202006359-GITT-PAT-practica-1
 * [new branch]      develop -> develop
Branch 'develop' set up to track remote branch 'develop' from 'origin'.
```

**git pull:** Este comando se utiliza cuando queremos descargar en nuestro repositorio local la última versión que este subida al repositorio remoto de Git. Muy útil para trabajar con los cambios más recientes que hayan hecho otros desarrolladores.

```

C:\202006359-GITT-PAT-practica-1>git pull
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 668 bytes | 133.00 KiB/s, done.
From https://github.com/202006359/202006359-GITT-PAT-practica-1
   f8dba27..f73a39f  main       -> origin/main
Updating f8dba27..f73a39f
Fast-forward
 Prueba-ADD.txt | 1 +
 1 file changed, 1 insertion(+)

```

**git log:** Este comando lo utilizamos para ver el historial de commits en nuestro proyecto.

```

commit f8dba27990e3509f4c8c035f76c884894e929e9a (HEAD -> develop, origin/develop)
Author: 202006359 <202006359@alu.comillas.edu>
Date: Thu Jan 19 12:39:15 2023 +0100

    Subo una modificación del fichero .txt

commit b8ebad53ce6fa7e8c6ab92355826e7078e63cb66
Author: 202006359 <202006359@alu.comillas.edu>
Date: Thu Jan 19 12:26:45 2023 +0100

    Subo un fichero .txt de prueba

commit 52191e327bda7cbec100c2d795ac59fb40cb5907
Author: 202006359 <113789409+202006359@users.noreply.github.com>
Date: Tue Jan 17 17:30:03 2023 +0100

    Initial commit

```