



Módulo 8 - Ejercicio práctico de GIT

1. Configurar Git definiendo el nombre del usuario y el correo electrónico en forma global.

```
> git config --global user.name "Your-Full-Name"  
> git config --global user.email "your-email-address"  
> git config --list
```

2. Crear un repositorio nuevo con el nombre mundosE y mostrar su contenido.

```
> mkdir mundosE  
> cd mundosE  
> git init  
> ls -la
```

3. Crear un fichero indice.txt con el siguiente contenido:

Capítulo 1: Introducción a Git
Capítulo 2: Flujo de trabajo básico
Capítulo 3: Repositorios remotos

```
> cat > indice.txt  
Capítulo 1: Introducción a Git  
Capítulo 2: Flujo de trabajo básico  
Capítulo 3: Repositorios remotos  
Ctrl+D
```

4. Comprobar de nuevo el estado del repositorio.

```
> git status
```

5. Añadir el fichero a la staging zone y ver el estado del repositorio

```
> git add indice.txt  
> git status
```

6. Realizar un commit de los últimos cambios con el mensaje “Se añadió el archivo indice.txt” y ver el estado del repositorio.

```
> git commit -m "Se añadió el archivo indice.txt."  
> git status
```

7. Cambiar el fichero indice.txt para que contenga lo siguiente:

Capítulo 1: Introducción a Git
Capítulo 2: Flujo de trabajo básico
Capítulo 3: Gestión de ramas
Capítulo 4: Repositorios remotos

```
> cat > indice.txt
Capítulo 1: Introducción a Git
Capítulo 2: Flujo de trabajo básico
Capítulo 3: Gestión de ramas
Capítulo 4: Repositorios remotos
Ctrl+D
```

8. Mostrar los cambios con respecto a la última versión guardada en el repositorio.

```
> git diff
```

9. Hacer un commit de los cambios con el mensaje "Se agregó capítulo 3 - gestión de ramas".

```
> git add indice.txt
> git commit -m "Se agregó capítulo 3 - gestión de ramas"
```

10. Mostrar el historial de cambios del repositorio.

```
> git log
```

11. Crear el archivo .gitignore y agregarlo al repositorio

- Crear el archivo .gitignore en la raíz del repositorio
- hacer el commit de los cambio con el mensaje "Se agrego el archivo .gitignore"
- mostrar el historial de cambios del repositorio

```
> touch .gitignore
> git add .
> git commit -m "Se agrego el archivo .gitignore"
> git log
```

12. Agregar el siguiente contenido en el archivo .gitignore y guardar los cambios en el repositorio


```
> cat > .gitignore
*.DS_Store
.tmp
.idea/workspace.xml
/out
.class
Ctrl+D
> git commit -am "Se modifiko el archivo .gitignore"
> git status
```

13. Crear un repositorio público llamado "mundosE" en GitHub

1. En la página de inicio de GitHub, lleve a cabo alguna de las siguientes operaciones:
 - En Your repositories, seleccionar New repository.
 - En la barra de navegación, seleccione Create new (+) y luego elegir **New repository**.
2. En la página Create a new repository, haga lo siguiente:
 - En el cuadro Repository name (Nombre del repositorio), indique **mundosE**
 - Seleccione Public.
 - Desactive la casilla Initialize this repository with a README.
 - Elija Create repository.



Create a new repository


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

Owner *  ruben-nieva ▾ / **Repository name *** mundosE ✓

Great repository names are mundosE is available.. Need inspiration? How about **vigilant-disco**?

Description (optional)



☒  **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.

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

☐ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer.

Add .gitignore: None ▾ Add a license: None ▾ ⓘ

Para una descripción más detallada se puede seguir la siguiente guía:

<https://docs.github.com/es/github/getting-started-with-github/create-a-repo>

14. Configure el repositorio local para que apunte al repositorio remoto en Github

```
> git remote add origin git@github.com:your-github-account/mundosE.git
> git push -u origin master
> git status
> git remote -v
```

15. Acceder a GitHub y comprobar que se han subido los cambios mostrando el historial de versiones.

16. Crear un nuevo archivo en el repositorio y guardarlo en el repo local y remoto.

- a. Añadir el fichero alumnos.txt que contenga el nombre del alumno y su correo electrónico.
- b. Añadir los cambios(git add)
- c. Hacer un commit con el mensaje "Se agregó un listado de alumnos"
- d. Subir los cambios al repositorio remoto.

```
> echo "nombre email@mail.com" >> alumnos.txt
> git add .
> git commit -m "Se agregó un listado de alumnos"
> git status
> git push origin master.
> git status
```

17. Acceder a GitHub y comprobar que se han subido los cambios

18. Acceder a Github y modificar el archivo alumnos.txt agregando una o más líneas y guardar los cambios

mundosE / alumno.txt Cancel

<> Edit file Preview changes Spaces 2 No wrap

```
1 alumno1 alumno1@mail.com
2 alumno2 alumno2@mail.com
3 alumno3 alumno3@mail.com
```

Commit changes

Update alumno.txt

nuevos alumnos agregados a la lista.

☒ Commit directly to the master branch.

☐ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Commit changes Cancel

19. Bajar los cambios del repositorio remoto al repositorio local y verificar que estan los ultimos cambios introducidos desde GitHub

```
> git pull origin master
> git status
> cat alumnos.txt
```