

## TRABAJO PRÁCTICO 2 - GIT - GITHUB

### 1. ¿Qué es GitHub?

GitHub es una plataforma de desarrollo colaborativo basada en la web que utiliza Git para el control de versiones. Permite a los desarrolladores alojar y revisar código, gestionar proyectos y colaborar con otros. En esencia, es un lugar donde puedes almacenar tus proyectos de código, trabajar en ellos con otros y realizar un seguimiento de los cambios.

### 2. ¿Cómo crear un repositorio en GitHub?

Para crear un repositorio en GitHub:

Inicia sesión en tu cuenta de GitHub.

Haz clic en el botón "New" (Nuevo) en la parte superior izquierda de la página.

Asigna un nombre a tu repositorio.

Elige si quieres que sea público o privado.

Puedes añadir un archivo README, un archivo .gitignore o una licencia.

Haz clic en "Create repository" (Crear repositorio).

### 3. ¿Cómo crear una rama en Git?

Para crear una rama en Git, utiliza el siguiente comando en la línea de comandos:

```
git branch <nombre-de-la-rama>
```

### 4. ¿Cómo cambiar a una rama en Git?

Para cambiar a una rama existente en Git, utiliza el siguiente comando:

```
git checkout <nombre-de-la-rama>
```

Para crear una rama y cambiar a ella al mismo tiempo se puede usar el comando:

```
git checkout -b <nombre-de-la-rama>
```

## 5. ¿Cómo fusionar ramas en Git?

Para fusionar una rama en otra en Git, utiliza el siguiente comando:

```
git merge <nombre-de-la-rama-a-fusionar>
```

Debes estar en la rama donde quieres incorporar los cambios de la otra rama.

## 6. ¿Cómo crear un commit en Git?

Para crear un commit en Git, sigue estos pasos:

Añade los archivos que quieres incluir en el commit con `git add <archivo>`.

Crea el commit con `git commit -m "Mensaje del commit"`.

## 7. ¿Cómo enviar un commit a GitHub?

Para enviar un commit a GitHub (un repositorio remoto), primero debes añadir el repositorio remoto a tu repositorio local y luego usar el comando push:

```
git push origin <nombre-de-la-rama>
```

## 8. ¿Qué es un repositorio remoto?

Un repositorio remoto es una versión de tu repositorio que se aloja en un servidor en línea, como GitHub. Permite a los desarrolladores colaborar y compartir cambios en el código.

## 9. ¿Cómo agregar un repositorio remoto a Git?

Para agregar un repositorio remoto a Git, utiliza el siguiente comando:

```
git remote add origin <URL-del-repositorio-remoto>
```

## 10. ¿Cómo empujar cambios a un repositorio remoto?

Para empujar cambios a un repositorio remoto, utiliza el siguiente comando:

```
git push origin <nombre-de-la-rama>
```

## 11. ¿Cómo tirar de cambios de un repositorio remoto?

Para obtener los cambios más recientes de un repositorio remoto, utiliza el siguiente comando:

```
git pull origin <nombre-de-la-rama>
```

## 12. ¿Qué es un fork de repositorio?

Un fork es una copia de un repositorio que haces en tu propia cuenta de GitHub. Te permite modificar el código original sin afectar al repositorio original.

## 13. ¿Cómo crear un fork de un repositorio?

Para crear un fork de un repositorio:

Ir al repositorio que quieres hacer fork en GitHub.

Haz clic en el botón "Fork" (Bifurcar) en la parte superior derecha de la página.

GitHub creará una copia del repositorio en tu cuenta.

## 14. ¿Cómo enviar una solicitud de extracción (pull request) a un repositorio?

1. **Crea una bifurcación (fork) del repositorio:** Si no tienes permisos de escritura en el repositorio original, crea una bifurcación en tu cuenta de GitHub.
2. **Clona el repositorio bifurcado:** Clona el repositorio bifurcado a tu máquina local.
3. **Crea una rama para tus cambios:** Crea una nueva rama para tus cambios con `git checkout -b <nombre-de-la-rama>`.
4. **Realiza tus cambios y haz commits:** Realiza los cambios necesarios y crea commits con mensajes descriptivos.
5. **Envía los cambios a tu repositorio bifurcado:** Envía los cambios a tu repositorio bifurcado en GitHub con `git push origin <nombre-de-la-rama>`.
6. **Crea una solicitud de extracción:** Ve a tu repositorio bifurcado en GitHub y haz clic en "New pull request" (Nueva solicitud de extracción).
7. **Compara las ramas y crea la solicitud:** Asegúrate de que la rama base sea la rama correcta del repositorio original y la rama compare sea tu rama con los cambios. Luego, haz clic en "Create pull request" (Crear solicitud de extracción).

## 15. ¿Cómo aceptar una solicitud de extracción?

1. **Revisa la solicitud de extracción:** Revisa los cambios propuestos en la solicitud de extracción.
2. **Deja comentarios (opcional):** Deja comentarios para dar retroalimentación al autor de la solicitud.
3. **Aprueba la solicitud (si es necesario):** Si eres el propietario del repositorio o tienes permisos de escritura, aprueba la solicitud.
4. **Fusiona la solicitud:** Haz clic en "Merge pull request" (Fusionar solicitud de extracción) para fusionar los cambios en la rama base.
5. **Elimina la rama (opcional):** Una vez fusionada la solicitud, puedes eliminar la rama.

## 16. ¿Qué es una etiqueta en Git?

Una etiqueta (tag) en Git es una instantánea de un punto específico en el historial del repositorio. Se utiliza para marcar versiones importantes del código, como lanzamientos (releases).

## 17. ¿Cómo crear una etiqueta en Git?

Para crear una etiqueta en Git, utiliza el siguiente comando:

```
git tag <nombre-de-la-etiqueta> (etiqueta ligera)
```

```
git tag -a <nombre-de-la-etiqueta> -m "Mensaje de la etiqueta" (etiqueta anotada)
```

## 18. ¿Cómo enviar una etiqueta a GitHub?

Para enviar una etiqueta a GitHub, utiliza el siguiente comando:

```
git push origin <nombre-de-la-etiqueta>
```

Para enviar todas las etiquetas usar. `git push origin --tags`

## 19. ¿Qué es un historial de Git?

El historial de Git es un registro de todos los commits realizados en un repositorio. Muestra quién hizo los cambios, cuándo y qué cambios se realizaron.

## 20. ¿Cómo ver el historial de Git?

Para ver el historial de Git, utiliza el siguiente comando:

```
git log
```

## 21. ¿Cómo buscar en el historial de Git?

Para buscar en el historial de Git, puedes usar varias opciones con el comando `git log`, como:

```
git log --grep="<patrón-de-búsqueda>" (busca en los mensajes de commit)
```

```
git log <archivo> (muestra los commits que modificaron un archivo específico)
```

## 22. ¿Cómo borrar el historial de Git?

Borrar el historial de Git puede ser peligroso y generalmente no se recomienda. Si realmente necesitas hacerlo, puedes usar `git reset --hard <commit-deseado>`, pero ten en cuenta que esto eliminará todos los commits posteriores.

## 23. ¿Qué es un repositorio privado en GitHub?

Un repositorio privado en GitHub es un repositorio que solo es accesible para las personas a las que les concedes permiso.

## 24. ¿Cómo crear un repositorio privado en GitHub?

Al crear un nuevo repositorio en GitHub, selecciona la opción "Private" (Privado).

## 25. ¿Cómo invitar a alguien a un repositorio privado en GitHub?

1. **Ve a la configuración del repositorio:** Ve a la página del repositorio privado en GitHub y haz clic en "Settings" (Configuración).
2. **Ve a "Collaborators" (Colaboradores):** En la barra lateral, haz clic en "Collaborators" (Colaboradores).
3. **Invita a colaboradores:** Escribe el nombre de usuario de la persona que quieres invitar y haz clic en "Add collaborator" (Añadir colaborador).

## 26. ¿Qué es un repositorio público en GitHub?

Un repositorio público en GitHub es un repositorio que es accesible para cualquier persona en Internet.

## 27. ¿Cómo crear un repositorio público en GitHub?

Al crear un nuevo repositorio en GitHub, selecciona la opción "Public" (Público).

## 28. ¿Cómo compartir un repositorio público en GitHub?

Puedes compartir un repositorio público compartiendo su URL. Cualquier persona puede ver, clonar y bifurcar un repositorio público.

Por ejemplo el link de mi repositorio es:

<https://github.com/gabox2002/UTN-TUPaD-P1>