

Trabajo practico Git y GitHub

1. ¿Qué es GitHub?

GitHub es una plataforma de alojamiento de código fuente que utiliza el sistema de control de versiones Git. Permite a los desarrolladores almacenar, gestionar y colaborar en proyectos de software de manera remota, brindando funcionalidades como control de versiones, colaboración en equipo, y revisión de código.

2. ¿Cómo crear un repositorio en GitHub?

- Inicia sesión en GitHub.
- Haz clic en el botón verde **"New"** en tu página de repositorios o en el menú de tu perfil.
- Rellena el formulario con el nombre del repositorio, una descripción opcional, y si deseas que sea público o privado.
- Haz clic en **"Createrepository"**.

3. ¿Cómo crear una rama en Git?

Para crear una rama en Git, puedes usar el siguiente comando:

```
gitbranchnombre_de_rama
```

Esto creará una nueva rama llamada nombre_de_rama.

4. ¿Cómo cambiar a una rama en Git?

Para cambiar a una rama existente, usa el comando:

```
gitcheckoutnombre_de_rama
```

O, con versiones más recientes de Git, puedes usar:

```
gitswitchnombre_de_rama
```

5. ¿Cómo fusionar ramas en Git?

Para fusionar una rama en tu rama actual, usa el siguiente comando:

```
gitmergenombre_de_rama
```

Esto fusionará los cambios de nombre_de_rama a la rama en la que te encuentres actualmente.

6. ¿Cómo crear un commit en Git?

Primero, asegúrate de haber hecho cambios en tus archivos. Luego, para crear un commit, utiliza estos comandos:

`git add .`

`git commit -m "Mensaje descriptivo del commit"`

7. ¿Cómo enviar un commit a GitHub?

Después de haber realizado un commit localmente, envíalo a GitHub con:

`git push origin nombre_de_rama`

8. ¿Qué es un repositorio remoto?

Un repositorio remoto es una copia de tu repositorio Git almacenada en un servidor externo, como GitHub, GitLab o Bitbucket. Esto permite que varias personas colaboren y compartan cambios fácilmente.

9. ¿Cómo agregar un repositorio remoto a Git?

Para agregar un repositorio remoto, utiliza el siguiente comando:

`git remote add origin URL_del_repositorio`

Donde `URL_del_repositorio` es la dirección del repositorio en GitHub o en otro servicio.

10. ¿Cómo empujar cambios a un repositorio remoto?

Para enviar tus cambios locales a un repositorio remoto, utiliza:

`git push origin nombre_de_rama`

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

Para obtener cambios del repositorio remoto, utiliza:

`git pull origin nombre_de_rama`

12. ¿Qué es un fork de repositorio?

Un **fork** es una copia de un repositorio que te permite realizar cambios de manera independiente sin afectar el repositorio original. Es comúnmente usado para contribuir a proyectos de código abierto.

13. ¿Cómo crear un fork de un repositorio?

- Ve al repositorio en GitHub.
- Haz clic en el botón **"Fork"** en la esquina superior derecha.
- Esto creará una copia del repositorio en tu cuenta de GitHub.

14. ¿Cómo enviar una solicitud de extracción (pullrequest) a un repositorio?

- Después de hacer cambios en tu fork y empujarlos, ve a la página del repositorio original.
- GitHub te sugerirá abrir un **PullRequest**.

- Haz clic en "**New PullRequest**" y elige tu rama y los cambios que deseas enviar.

15. ¿Cómo aceptar una solicitud de extracción?

Como propietario de un repositorio, puedes aceptar un pullrequest haciendo clic en "**MergePullRequest**" y luego en "**ConfirmMerge**".

16. ¿Qué es una etiqueta en Git?

Una **etiqueta** (tag) en Git es una marca utilizada para señalar puntos específicos en el historial de un proyecto, generalmente usada para marcar versiones (releases).

17. ¿Cómo crear una etiqueta en Git?

Para crear una etiqueta en Git, usa:

```
git tag -a nombre_etiqueta -m "Mensaje de la etiqueta"
```

18. ¿Cómo enviar una etiqueta a GitHub?

Para enviar una etiqueta a GitHub, utiliza:

```
git push origin nombre_etiqueta
```

19. ¿Qué es un historial de Git?

El historial de Git es un registro de todos los cambios que se han hecho en el repositorio, incluyendo los commits realizados, quién los hizo, y cuándo.

20. ¿Cómo ver el historial de Git?

Para ver el historial de commits, usa el comando:

```
git log
```

21. ¿Cómo buscar en el historial de Git?

Puedes buscar en el historial utilizando:

```
git log --grep="texto_a_buscar"
```

22. ¿Cómo borrar el historial de Git?

Borrar el historial no es una operación sencilla, pero puedes eliminar un commit reciente utilizando:

```
git reset --hard HEAD~n
```

Donde n es el número de commits a retroceder.

23. ¿Qué es un repositorio privado en GitHub?

Un **repositorio privado** es aquel que solo puede ser accedido por los usuarios que han sido invitados a él. No es visible para el público en general.

24. ¿Cómo crear un repositorio privado en GitHub?

Al crear un nuevo repositorio en GitHub, selecciona la opción "**Private**" durante la configuración.

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

- Ve a la página de tu repositorio.
- Haz clic en "**Settings**" > "**Manage access**".
- Invita a usuarios por su nombre de usuario o dirección de correo electrónico.

26. ¿Qué es un repositorio público en GitHub?

Un **repositorio público** es aquel que cualquiera puede ver y clonar, es accesible para todo el mundo.

27. ¿Cómo crear un repositorio público en GitHub?

Al crear un nuevo repositorio en GitHub, selecciona la opción "**Public**" durante la configuración.

28. ¿Cómo compartir un repositorio público en GitHub?

Simplemente comparte la URL de tu repositorio público, la cual se encuentra en la barra de direcciones de tu navegador, por ejemplo:

<https://github.com/usuario/repositorio>.