

Trabajo Practico N ° 3 GIT

Actividades:

Contestar las siguientes preguntas utilizando las guías y documentación proporcionada (Desarrollar las respuestas) :

1- ¿Qué es GitHub?

Github es una plataforma que esta alojada en la nube que sirve para alojar y gestionar tus proyectos en GIT, es e decir GIT no es Github sino que ambos se complementan ya que hay plataformas distintas a Github, como lo es por ejemplo Gitlab.

2- ¿Cómo crear un repositorio en GitHub?

Para crear un repositorio en Github, lo puedes hacer distintas maneras, la mas usada es usar un correo electrónico generar la cuenta, y una vez generada la cuenta puedes crear desde la misma pagina web un nuevo repositorio, o bajar la aplicación de escritorio y crearla desde allí.

3- ¿Cómo crear una rama en Git?

Para crear una rama en Git se usa la terminal sea de Windows o de Git y se usa el comando ***git branch*** seguido del nombre que le queremos dar.

4- ¿Cómo cambiar a una rama en Git?

Para cambiar una rama en Git se usa el comando en la terminal de Windows o de Git llamado, ***git checkout*** y nombre de la rama a la que queremos cambiar.

5- ¿Cómo fusionar ramas en Git?

Para fusionar dos ramas distintas en Git, primero tendremos que ubicarnos en nuestra rama principal con el comando ***git checkout*** y luego agregamos el comando ***git merge*** y el nombre con la que queremos fusionar.

6- ¿Cómo crear un commit en Git?

Para generar un commit que es la funcion de guardar un cambio primero se usa el comando ***git add*** . Si el cambio se quiere guardar en la rama que estamos parados y luego se usa el comando ***git commit -m "comentario del usuario"***.

7- ¿Cómo enviar un commit a GitHub?

Para enviar u commit a Github primero hay que estar logeado a Github o sino tendra que logearse por la misma consola de Git en el momento que se indique, luego se usa el comando ***git add*** . Luego ***git commit -m "comentario del usuario"***, después se agrega el comando ***git push -u*** origin (puede ser otro pero por convención se usa origin) y luego el nombre de la rama, quedaria así ***git push -u*** origin master, si no estas logeado te va pedir que este logeado para autorizar la operación.

8- ¿Qué es un repositorio remoto?

Es un lugar en la nube donde se guarda nuestros repositorios de trabajo que están de manera local, estos puede ser públicos o privados, permitiéndonos compartir y trabajar con otras personas alrededor del mundo de una manera mas fácil y segura.

9- ¿Cómo agregar un repositorio remoto a Git?

Para agregar un repositorio remoto a Git, puedes usar el comando ***git remote add*** origin url (del repositorio) en el terminal.

10- ¿Cómo empujar cambios a un repositorio remoto?

Para empujar cambios a un repositorio remoto, por primera vez usamos el comando ***git push -u origin master*** y despues de esto se puede usar el comando ***git push*** solamente.

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

12- ¿Qué es un fork de repositorio?

Un fork de repositorio es la manera de obtener otro ajeno al mio sin clonarlo localmente sino tenerlo aparte en mi repositorio remoto y poder trabajar en el sin afectar al creador original del repositorio.

13- ¿Cómo crear un fork de un repositorio?

Para crear un fork de un repositorio en GitHub se necesita iniciar se sesion luego ir al repositorio que se quiere copiar y dar al click en el boton de fork se va abrir otra ventana y es cuestion de seguir los pasos y se copia en tu repositorio de GitHub.

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

Para solicitar un pull request es necesario antes haber hecho commit y a haber hecho push hacia al repositorio remoto, una vez ahí aparecerá el cartel **compare&pull request**, al hacer click te llevara a otra pantalla donde te dira si quieres hacer el pull en la rama de tu repositorio o de la original del fork, hay espacio de escritura donde comentamos los cambios que hemos hecho, despues de esto hacemos click en el boton **creat pull request** y con ello finalizamos.

15- ¿Cómo aceptar una solicitud de extracción?

Para aceptar la solicitud del pull resquest es necesario ir a la seccion pull resquest, despues hacer click en el boton merge pull request, agregar un comentario y confirmar.

16- ¿Qué es un etiqueta en Git?

Una etiqueta git es una manera de marcar, los puntos importantes en el historial de un repositorio.

17 - ¿Cómo crear una etiqueta en Git?

Existen dos clases de tipos de etiquetas en Git, las ligeras y las anotadas las que mas se usan son las etiquetas anotadas estas, se puede crear de la siguiente manera, se usa el comando ***git tag -a*** luego la version v1.4 -m y luego el nombre o el comentario que le queremos dar 'my version 1.4', la -m es para que se especifique el comentario que se quiere agregar.

18- ¿Cómo enviar una etiqueta a GitHub?

Para enviar etiquetas a Github, se puede hacer de dos maneras uno a través del comando ***git push origin [etiqueta]*** (nombre de la etiqueta), que envia de forma individual y la segunda es para enviar varias etiquetas al mismo tiempo ***git push origin -tags***.

19- ¿Qué es un historial de Git?

Un historial de git es un registro detallado de las modificaciones, realizados en los proyecto a lo largo del tiempo.

20- ¿Cómo ver el historial de Git?

Para ver el historial de git se puede usar el comando ***git log***, que lista desde lo mas reciente hasta los primeros cambios

21- ¿Cómo buscar en el historial de Git?

Se puede usar varios comandos para buscar de manera mas especificas los cambios hechos en el proyecto a continuación una imagen con los comandos mas usados, todas deben empezar con el comando git log.

Tabla 3. Opciones para limitar la salida de git log

Opción	Descripción
<code>-(n)</code>	Muestra solamente las últimas n confirmaciones
<code>--since, --after</code>	Muestra aquellas confirmaciones hechas después de la fecha especificada.
<code>--until, --before</code>	Muestra aquellas confirmaciones hechas antes de la fecha especificada.
<code>--author</code>	Muestra sólo aquellas confirmaciones cuyo autor coincide con la cadena especificada.
<code>--committer</code>	Muestra sólo aquellas confirmaciones cuyo confirmador coincide con la cadena especificada.
<code>-S</code>	Muestra sólo aquellas confirmaciones que añaden o eliminan código que corresponda con la cadena especificada.

22- ¿Cómo borrar el historial de Git?

El comando git reset permite deshacer cambios en un repositorio de Git. Sirve para volver a una confirmación específica, descartando todas las confirmaciones posteriores. Esto modifica los punteros de referencia de las confirmaciones y también cambia el estado de los árboles de Git: el historial de confirmaciones (HEAD), el índice del entorno de ensayo y el directorio de trabajo.

23- ¿Qué es un repositorio privado en GitHub?

Es un repositorio que solo lo puedes ver tu o las personas a las que le brindes acceso.

24- ¿Cómo crear un repositorio privado en GitHub?

Se puede crear en el botón nuevo repositorio y desde sección se puede elegir si el repositorio va ser publico o privado.

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

Se Puede enviar una invitación para colaborar directamente en el repositorio a alguien en GitHub, o a la dirección de correo electrónico de la persona, siempre y cuando esta cuente con una cuenta GitHub.

26- ¿Qué es un repositorio público en GitHub?

Un repositorio publico es repositorio en el cual esta abierto al publico general, donde los permisos estan abiertos para se copiados y usado sin ninguna restricción. Y no necesariamente se necesita una cuenta de GitHub para poder verla.

27- ¿Cómo crear un repositorio público en GitHub?

Se puede crear en el botón nuevo repositorio y desde sección se puede elegir si el repositorio va ser publico o privado.

28- ¿Cómo compartir un repositorio público en GitHub?

Se Puede enviar una invitación para colaborar directamente en el repositorio a alguien en GitHub, o a la dirección de correo electrónico de la persona, siempre y cuando esta cuente con una cuenta GitHub, y como es publico tambien se puede compartir el link directamente.