

1. Contestar las preguntas

a) ¿Qué es GitHub?

GitHub es una plataforma en la nube donde uno puede guardar, compartir y trabajar conjuntamente con otros/as para codear. Sirve también como portfolio y red social donde aportar a proyectos colaborativos, ver el trabajo de otros e incluso sugerir aportes.

b) ¿Cómo crear un repositorio de GitHub?

Para crear un repositorio de GitHub se puede optar por tres caminos principalmente, el más conocido y usual es dirigirse hasta la página web, ingresar a la pestaña de your repositories y darle a new. Una vez seleccionado un nombre disponible y configurado el repo se oprime el botón para crear y tarea finalizada.

La segunda opción es crearlo desde gh, la CLI(Command Line Interface) de GitHub, para lo cual primero hay que instalar y configurar dicha herramienta. Una vez instalada, las credenciales cargadas y la CLI configurada, se puede crear el repositorio desde la consola local con el comando:

```
gh repo create {nombre del repo} {flags correspondientes (--public, --private, otras)}
```

La tercera es crearlo utilizando la API de GitHub, también desde la terminal. Para utilizar este método se deberá crear primero un token, que luego se colocará en la request POST que se hace para crear dicho repo.

La documentación de la api puede encontrarse en:

<https://docs.github.com/es/rest?apiVersion=2022-11-28>

c) ¿Cómo crear una rama en Git?

Para crear una rama se debe utilizar el comando:

```
git branch {nombre de la nueva rama}
```

Para crear y cambiar a la rama creada en el mismo comando se utiliza:

```
git branch -b {nombre de la nueva rama}
```

d) ¿Cómo cambiar a una rama en Git?

Para cambiar a otra rama se utiliza el comando:

```
git checkout {nombre de la rama}
```

e) ¿Cómo fusionar ramas en Git?

Para fusionar ramas debe cambiarse a la rama a la que se fusionarán las otras ramas, por ejemplo main. Una vez en esa rama – para el ejemplo se utilizaría git checkout main – se utiliza el comando:

```
git merge {nombre de la rama}
```

f) ¿Cómo crear un commit en Git?

Para crear un commit primero debería haberse guardado algo con el comando `git add {elemento a guardar}` o `git add .` (si se desea guardar todo). Una vez realizado el paso anterior se puede crear el commit utilizando el comando:

```
git commit -m "{Nombre del commit}"
```

g) ¿Cómo enviar un commit a GitHub?

Para enviar-subir un commit a GitHub, si es el primer push, se utiliza el comando:

```
git push -u origin {nombre de la rama a pushear}
```

Si ya se ha subido algo previamente y estoy en la misma rama, puede utilizarse simplemente el comando:

```
git push
```

h,i) ¿Qué es un repositorio remoto? Y ¿Cómo agregar un repositorio remoto a Git?

Es una versión de tu repositorio almacenada en la nube, en este caso de GitHub. Para agregar un repositorio remoto se utiliza el comando:

```
git remote add origin {URL}
```

j) ¿Cómo empujar cambios a un repositorio remoto?

Para pushear cambios a un repositorio remoto debería antes haberse hecho al menos un commit que pushear. Una vez cumplido ese requisito, se utiliza, si es el primer push, el comando:

```
git push -u origin {nombre de la rama a pushear}
```

Si ya se ha subido algo previamente y estoy en la misma rama, puede utilizarse simplemente el comando:

```
git push
```

k) ¿Cómo tirar de cambios de un repositorio remoto?

Pullear cambios implica actualizar el repo local con los cambios más recientes del repo remoto. En el proceso tienen lugar dos subprocesos:

Fetch y merge, se descargan los cambios y se mergean con la rama local.

Es muy importante conocer y saber utilizar este comando ya que es fundamental a la hora de trabajar colaborativamente en una misma rama de un proyecto. Si no actualizamos el repo podemos pisar el trabajo realizado y obstaculizar a futuro las fusiones.

Para pullear cambios de un repositorio se utiliza el comando:

```
git pull origin {nombre de la rama a pullear}
```

l,m) ¿Qué es un fork de repositorio? ¿Cómo crear un fork de un repositorio?

Básicamente es una forma de crear una copia de un repositorio en otra propia cuenta, lo que permite hacer cambios sin afectar el original. Para hacer un fork debemos dirigirnos a la página del repositorio que se desea forkear, y presionar el botón fork en la parte superior derecha. Es pertinente mencionar que para realizar dicho procedimiento se debe estar logeado con la cuenta en la que se copiará el repositorio.

n,o) ¿Cómo enviar una solicitud de extracción (pull request) a un repositorio? Y ¿Cómo aceptar una solicitud de extracción?

Se deben subir los cambios a GitHub y luego ir al repositorio en la web, dirigirse a la pestaña pull request, seleccionar la rama, escribir la descripción y crear la pull request.

Se puede aceptar directamente desde la misma pestaña pero en el repositorio original oprimiendo Merge pull request. Sin embargo una mejor práctica es descargar la rama y probarla en local, y luego decidir como mergear los aportes. Las tres maneras de hacerlo son merge simple, squash and merge y rebase.

p,q) ¿Qué es un etiqueta en Git? ¿Cómo crear una etiqueta en Git?

Una etiqueta es similar a un marcador, es una referencia a un estado, una versión importante en un repositorio. Para crearla se utiliza el comando git tag, pudiendo variar el tipo de la misma (ligera – no se agrega ninguna flag, anotada – flag -a para indicarla y -m para el mensaje).

r) ¿Cómo enviar una etiqueta a GitHub?

Para subir una sola etiqueta se utiliza el comando:

```
git push origin {nombre de la etiqueta}
```

Para subir todas las etiquetas se utiliza el comando:

```
Git push --tags
```

Para eliminar una etiqueta se utiliza el comando:

```
git push --delete origin {nombre de la etiqueta}
```

s,t,u) ¿Qué es un historial de Git? ¿Cómo ver el historial de Git? ¿Cómo buscar en el historial de Git? Y ¿Cómo borrar el historial de Git?

El historial de git es el registro de todos los commits de un repositorio. Para ver el mismo se utiliza el comando:

```
git log
```

Este puede utilizar flags para utilizar diversos parámetros de búsqueda (fecha, rango de fechas, n° ultimos commits, etc).

Buscar n últimos commits:

```
git log -n {número de commits a mostrar}
```

Rango de fechas:

```
git log --since="{fecha}" --until="{fecha}"
```

Log específico:

```
git log --grep= "{cadena de búsqueda}"
```

Autor:

```
git log --author = "{nombre del/a autor/a}"
```

Para eliminar todo el historial de git se podría reestablecer el repositorio a su estado original eliminando la carpeta oculta de git ubicada en la carpeta raíz del repositorio o con el comando:

```
rm -rf .git
```

Para eliminar un commit específico se puede utilizar el código:

```
git rebase -i o git rebase -i {has del commit desde el que se quiere modificar o HEAD para utilizar un número específico de commits}
```

v,w,x) ¿Qué es un repositorio privado en GitHub? ¿Cómo crear un repositorio privado en GitHub? ¿Cómo invitar a alguien a un repositorio privado en GitHub?

Es un repositorio en el cual solo puede ser visto o accedido por quienes tienen el permiso del propietario para hacerlo. El proceso para crear un repositorio privado es el mismo que el que se realiza para crear un repositorio, de hecho es menester para hacerlo seleccionar la opción privado o público en la web o colocar `private:true` o `false` en la API. En caso de hacerlo por la CLI de GitHub se debe colocar la flag `--private`, también al momento de crear el repositorio. Para invitar a alguien a un repositorio privado se debe enviar la invitación al mail de dicha persona, para hacerlo debemos dirigirnos al apartado Settings o configuración, manage acces y agregar un colaborador.

y,z,za) ¿Qué es un repositorio público en GitHub? ¿Cómo crear un repositorio público en GitHub? ¿Cómo compartir un repositorio público en GitHub?

Un repositorio público es aquel que puede ser visto y utilizado por cualquiera que ingrese por el link del mismo o la página del perfil del propietario. Al igual que se dijo en el caso anterior, se debe colocar en el caso de la API `private:false`, en el caso de la web seleccionar la opción Público, y en el caso de la CLI `--public`. Para compartirlo simplemente es necesario enviar el link del repositorio.