

Programación I

Trabajo práctico N° 2 - Git y Github

- 1) GitHub es una plataforma en línea propiedad de Microsoft donde los desarrolladores pueden alojar código como también compartir y contribuir en proyectos de otras personas (si los dueños de estos repositorios así lo permitieran). Usa el sistema de versionado GIT.
- 2) Para crear un repositorio en GitHub se debe crear un usuario en la plataforma y luego presionar en el botón "Nuevo". Allí se pedirán datos como el nombre del repositorio, su descripción y si es público o privado, entre otros.
- 3) En la línea de comandos de GIT, para crear una nueva rama ejecutamos **git checkout -b nombre-rama**
- 4) Para cambiar a una rama en GIT, ejecutamos **git checkout nombre-rama**
- 5) Para fusionar dos ramas en GIT, ejecutamos el comando **git checkout nombre-rama** para posicionarnos en la rama donde queremos recibir los cambios. Luego, ejecutamos **git merge nombre-rama-2** donde **nombre-rama-2** es la rama a la cual queremos fusionar.
- 6) Para crear un commit en GIT, luego de hacer nuestros cambios ejecutamos **git add** agregando un punto para incluir todos los cambios en el directorio (podemos especificar el nombre del archivo si queremos sólo agregar algunos cambios). Luego ejecutamos **git commit -m "descripcion del commit"**.
- 7) Para enviar un commit a GitHub ejecutamos el comando **git push nombre-remoto nombre-rama**
- 8) Un repositorio remoto es un repositorio que está alojado en un lugar distinto a nuestra computadora. Puede ser en un servidor con su propio GIT corriendo, o en GitLab, Github o soluciones similares.
- 9) Para agregar un repositorio remoto ejecutamos el comando **git remote add nombre-remoto url-remoto**.
- 10) Para empujar cambios a un repositorio remoto ejecutamos el comando **git push nombre-remoto nombre-rama**.
- 11) Para tirar de cambios de un repositorio remoto ejecutamos el comando **git pull**. Podemos especificar la rama agregando el nombre del remoto y el nombre de la rama.
- 12) Un fork de un repositorio es una copia del mismo, donde se puede trabajar independientemente y luego, mediante un pull request, solicitar que nuestros cambios se agreguen al repositorio original.
- 13) Crear un fork es una acción propia de GitHub. Se debe ingresar al repositorio del que se desea crear un fork y clicar en el botón "Fork".
- 14) Para enviar una solicitud de extracción, ingresamos a nuestro fork y en la pestaña de contribuir podremos elegir la opción de crear un pull request.
- 15) Para aceptar un pull request, en la solapa de pull requests en nuestro repositorio, elegimos la solicitud en cuestión. Allí podremos revisar los cambios que impactarán y aceptarla.
- 16) Una etiqueta en Git es una manera de marcar un punto en particular del historial del repositorio, normalmente para señalar su importancia (por ejemplo, delimitar el fin de la compatibilidad del programa con alguna librería, etc.)
- 17) Para crear una etiqueta en git ejecutamos el comando **git tag -a nombre.tag -m "descripcion del tag"**.

- 18) Para enviar un tag a nuestro repositorio remoto (GitHub) ejecutamos **git push nombre-remoto nombre.tag**.
- 19) El historial de git es un listado histórico de commits en un repositorio, ordenados por fecha que contiene el ID del commit, su autor, fecha y descripción.
- 20) El historial de git puede verse ejecutando **git log**.
 - a) Borrar el historial de git no es recomendable ni hay un modo directo de hacerlo. Puede hacerse un reset hard para volver a un punto del historial y eliminar todo lo de adelante. Lo más sencillo es crear un nuevo repositorio con el código existente.
- 21) Un repositorio privado en GitHub es aquel que no es públicamente accesible, el propietario puede invitar colaboradores y establecer permisos distintos a cada uno.
- 22) Para crear un repositorio privado los pasos son similares a cuando creamos un repositorio nuevo en el punto 2), solo que en el formulario inicial de creación debe seleccionarse la opción de “privado” en lugar de “público”.
- 23) Para invitar a alguien a un repositorio privado cliqueamos en la pestaña de configuración de nuestro repositorio. Luego, en la sección de acceso cliqueamos la opción de colaboradores y por último el botón de agregar personas.
- 24) Un repositorio público en GitHub es aquel en el cuál las personas pueden inspeccionar el código, las distintas ramas, hacer forks y proponer pull requests.
- 25) Para crear un repositorio público en GitHub se deben seguir los pasos en el punto 2.
- 26) Para compartir un repositorio público alcanza con compartir su URL de GitHub. También se puede descargar en zip.

Actividad 2

<https://github.com/martinetoledo/TUP-TP-GITHUB>

Actividad 3

<https://github.com/martinetoledo/conflict-exercise-tup>