

shoniisrael / Platzi-Tests Public

[Code](#) [Issues](#) 5 [Pull requests](#) 5 [Actions](#) [Projects](#) [Security](#) ...

master ▾

...

Platzi-Tests / git.md



JOSANT Update git.md ...

[History](#)

6 contributors



151 lines (83 sloc) | 5.87 KB

...

Curso Profesional de Git y GitHub

1. Los cambios a un archivo que están en “staging” están en:

Un área temporal en memoria antes de llegar al repositorio.

2. ¿Puedo crear ramas en GitHub que no tenga en mi repositorio local?

Sí, tú decides si traer esas ramas a tu repositorio local.

3. En un commit, los mensajes del commit son:

Importantes y obligatorios.

4. ¿La terminal ideal para usar Git en Windows es?

Git Bash o una terminal derivada de Linux

5. Con checkout puedo:

Ver todos los archivos de mi proyecto como los dejé en otras ramas.

6. ¿Puedes conectar tu repositorio local con más de un repositorio remoto?

Verdadero

7. Es mejor aprender a manejar Git con la terminal antes de hacerlo con herramientas visuales como Gitk porque:

Debemos aprender Git con sus comandos de la terminal para resolver problemas o conflictos más avanzados. Gitk funciona bien, pero no nos permite realizar operaciones tan complejas.

8. Un repositorio remoto y un repositorio local:

Tienen ramas similares dependiendo de cuáles haya enviado entre uno y otro.

9. El dueño de un repositorio al ver un pull request puede:

Editarlo, aceptarlo, darle merge, comentarlo para pedir cambios.

10. ¿Cuándo debería hacer un merge?

Cuando quiero fusionar los cambios de una rama con otra.

11. En un repositorio público en GitHub, ¿qué puede hacer los colaboradores?

Hacer cambios al repositorio, hacer push/pull, crear ramas, etc.

12. ¿Qué es un branch o una rama en Git?

Un repositorio aparte del master donde puedes trabajar en paralelo.

13. Si usas:

```
git config --global alias.platzi "shortlog"
```

Si usas: `git config --global alias.platzi "shortlog"` ¿Cómo invocar ese comando?

```
git platzi
```

14. ¿Con amend puedo?

Corregir los mensajes de un commit que hice mal sin que quede en la historia del repositorio.

15. ¿Cómo instalas Gitk?

En Windows y Mac podemos instalarlo desde gitk.com/download. En Linux debemos buscarlo con el instalador de paquetes de nuestra distribución.

16. Estás en la rama master y quieres mandar los últimos cambios que guardaste con git stash a la rama retomando-los-cambios (la rama no ha sido creada aún). ¿Cuál de las siguientes soluciones es incorrecta?

`git stash apply retomando-los-cambios`

17. Cuando hago un fork de un proyecto en GitHub, ¿lo que logro es?

Copiar un repositorio público a mis repositorios en GitHub, con todas sus ramas e historia anterior.

18. Git funciona para muchos tipos de archivo, ¿Qué tal funciona Git con archivos binarios?

Los puede agregar y versionar pero es mejor usar Git con texto plano.

19. ¿Cuál es la diferencia entre git rm y git reset HEAD?

`git rm` saca los archivos del repositorio y (opcionalmente) del disco duro.
`git reset head` saca los archivos de Staging, pero no del disco duro.

20. Cuando usas git stash, los cambios que “guardas” temporalmente se guardan en:

Memoria temporal, volviendo al estado del último commit.

21. ¿Qué crean los tags en Git?

Versiones descargables y puntos únicos en una rama de un repositorio.

22. Si en el .gitignore agrego esto: images/*.js. ¿Cuál de los siguientes archivos sería ignorado?

images/jquery.js

23. ¿Para qué sirve git grep?

Para encontrar las veces que hemos usado una palabra en los archivos del repositorio.

24. ¿GitHub es?

Un sistema online de manejo de repositorios de Git.

25. ¿Cuándo deberías usar cherry-pick?

Cuando quiero los cambios de un commit pasado sin dañar la historia de la rama.

26. ¿Las llaves públicas son?

Fáciles de compartir y sus mensajes imposibles de descifrar.

27. ¿Para qué sirve GitHub Pages?

Es un servicio de GitHub que nos permite publicar nuestros repositorios en internet (por ejemplo, nombre. github.io o nombre.github.io/proyecto).

28. ¿Cómo llaman GitHub y GitLab a los "merges" (propuestas de cambios desde otra rama o repositorio)?

GitHub los llama Pull Request y GitLab Merge Request.

29. El Jefe Freddy despidió a Anita y no piensa volver a contratarla. ¿Qué debe hacer Anita para seguir haciendo contribuciones a los proyectos públicos de Platzi?

Crear un Fork del proyecto y enviar Pull Request al proyecto original.

30. ¿Qué guarda Git?

Los cambios de los archivos de un proyecto.

31. Para transmitir cambios seguros entre tu repositorio local y GitHub, ¿qué se recomienda?

Usar una llave SSH.

32. ¿El HEAD en una rama es?

El apuntador al estado actual del repositorio basado en la rama en la que estoy trabajando.

33. Se crean dos archivos cuando creas una llave pública y privada para SSH, ¿cuál es la extensión de la llave privada?

Ninguna extensión

34. ¿Qué puedes hacer con gitk?

Ver en una interfaz visual las ramas y el log de tu repositorio.

35. Un pull request es:

Un cambio sugerido a un repositorio que el dueño del repositorio puede autorizar y hacer merge a la rama que elija.

36. Si ya hiciste el desafío de "Hazme un pull request," ¿dentro de cuál etiqueta tenías que agregar el cambio?

`<div id="post">`

37. Por defecto, GitHub usa la rama main como la rama principal. ¿Podemos cambiar la rama principal?

Verdadero

38. Para que dos personas trabajen en paralelo sobre el mismo archivo se recomienda:

Una rama independiente por cada persona y sus cambios que luego con verificación se hace merge a master.

39. Cuando hay un conflicto entre archivos lo mejor es:

Analizar los conflictos tal como los reportó Git y elegir los cambios finales, luego commit.

Este proyecto sigue la especificación de [todos los contribuyentes](#) . ¡Contribuciones de cualquier tipo son bienvenidas!