

Práctica Git

1. Crear una carpeta nueva: `mkdir practicagit` y entrar (`cd practicagit`)
2. Inicializar la carpeta como un repositorio: `git init`
3. Escribir: `dir /AH` y observar el resultado
4. Crear un archivo: `echo abc > uno.txt`
5. Ver los contenidos del archivo: `type uno.txt`
6. Crear otro archivo nuevo: `echo def > dos.txt`
7. Ver los contenidos del archivo: `type dos.txt`
8. Ejecutar: `git status`
9. Ejecutar: `git add uno.txt` & `git status`
10. Ejecutar: `git add .` & `git status`
11. Ejecutar: `git rm --cached dos.txt` & `git status`
12. Ejecutar: `git add dos.txt` & `git status`
13. Ejecutar: `git commit -m "Primer commit"` & `git status`
14. Ejecutar `git log`
15. Ejecutar: `echo xyz >> dos.txt` & `git status`
16. Ejecutar: `git diff`
17. Ejecutar: `git checkout dos.txt` & `git status`
18. Ejecutar: `type dos.txt`
19. Ejecutar: `echo xyz >> dos.txt` & `git status`
20. Ejecutar: `git add dos.txt` & `git status`
21. Ejecutar: `git commit -m "Segundo commit"` & `git status` & `git log`
22. Ejecutar: `git log` & `git log -1`
23. Crear una cuenta en github.com
24. Crear un repositorio vacío llamada: `practicagit`, y opcionalmente una descripción. Nada más
25. Tras pinchar en crear repositorio, leer la página que aparece, especialmente las instrucciones para enganchar un repo local a éste:
 - a. **...or push an existing repository from the command line**
 - b. `git remote add origin`
`https://github.com/iconojdiazch/practicagit.git`
`git push -u origin master`

26. Nos vamos al repositorio local, y desde la línea de comandos tecleamos la primera orden
27. Ejecutar: `git remote -v`
28. Ejecutamos la segunda orden, el push
29. Nos vamos al repositorio remoto y refrescamos la página Web
30. Pinchamos en "Add a README"
31. Pinchamos en "Commit new file"
32. Volvemos a la línea de comandos
33. Ejecutar: `git pull & dir & git status`
34. Ejecutar: `git tag v1.0 & git tag`
35. Ejecutar: `git show v1.0`
36. Las etiquetas no se suben al servidor por defecto. Para hacerlo, debemos ejecutar: `git push origin v1.0` o bien `git push --tags` para enviar todas
37. Ejecutar: `git push origin v1.0`
38. Ir a la Web y refrescar la página
39. Pinchar en Branch: master y, en el desplegable que aparece, en Tags
40. Ejecutar: `git branch nuevarama & git branch`
41. Ejecutar `git branch -v`
42. Ejecutar: `git log --oneline --decorate`
43. Ejecutar: `git checkout nuevarama`
44. Ejecutar de nuevo las órdenes 41 y 42
45. Ejecutar: `echo ghi > tres.txt`
46. Ejecutar: `git add .`
47. Ejecutar: `git commit -m "Añadido tres.txt a nuevarama"`
48. Ejecutar `git push origin nuevarama`
49. Ir a la página web, refrescarla, pinchar en Branch:master y elegir nuevarama. En ella aparece al nuevo archivo "tres.txt", el cual no está presente en la rama master
50. Ejecutar: `git checkout master`
51. Ejecutar: `git merge nuevarama`
52. Ejecutar: `git status`
53. Ejecutar: `git push`

54. Ir a la página web, refrescarla y ver los contenidos de la rama master. Observar que "nuevarama" sigue existiendo
55. Tras fusionar las dos ramas y subir los cambios al repo remoto, si queremos podemos eliminar la rama remota.
56. Ejecutar: `git push origin --delete nuevarama`
57. Ir a la página Web y refrescar. La rama "nuevarama" ha desaparecido
58. También podemos borrarla del repositorio local. Comprobar que aún existe: `git branch -v`
59. Ejecutar: `git branch -d nuevarama`
60. Ejecutar: `git branch & git status`