

# Test de Comandos GIT



## Comandos GIT 1

¿Para qué sirve un control de versiones?

- Sirve como sistema de copias de seguridad de archivos
- Sirve para almacenar código
- **Sirve para llevar un historial de los cambios de cualquier carpeta de archivos**
- Todos son correctas

## Comandos GIT 2

De los siguientes elementos de la estructura de un repositorio, indica qué elementos estructurales contiene un repositorio:

- **Repositorio, módulo, revisión, etiqueta, rama**
- Repositorio, directorio, revisión, etiqueta, rama
- Repositorio, módulo, revisión, ficheros, rama
- Repositorio, revisión, etiqueta, rama

## Comandos GIT 3

En un repositorio, los ficheros pueden pasar por tres estados, ¿cuáles de ellos son?

[Múltiple]

- **Committed**
- **Modified**
- **Staged**
- Prepared

## Comandos GIT 4

¿Qué es el directorio de GIT?

- **Donde se almacenan los metadatos y la base de datos de objetos para tu proyecto. Es lo que se copia cuando clonas un repositorio desde otra computadora.**
- Es una copia de una versión del proyecto. Estos archivos se sacan de la base de datos comprimida en el directorio de Git, y se colocan en disco para que los puedas usar/modificar.
- Es un archivo en tu directorio de Git que almacena información de lo que se añade a tu próxima confirmación. También se denomina índice ("index").

## Comandos GIT 5

¿Qué es el directorio de trabajo de GIT?

- Donde se almacenan los metadatos y la base de datos de objetos para tu proyecto. Es lo que se copia cuando clonas un repositorio desde otra computadora.
- **Es una copia de una versión del proyecto. Estos archivos se sacan de la base de datos comprimida en el directorio de Git, y se colocan en disco para que los puedas usar/modificar.**
- Es un archivo en tu directorio de Git que almacena información de lo que se añade a tu próxima confirmación. También se denomina índice ("index").

## Comandos GIT 6

¿Qué es el área de preparación?

- Donde se almacenan los metadatos y la base de datos de objetos para tu proyecto. Es lo que se copia cuando clonas un repositorio desde otra computadora.
- Es una copia de una versión del proyecto. Estos archivos se sacan de la base de datos comprimida en el directorio de Git, y se colocan en disco para que los puedas usar/modificar.
- **Es un archivo en tu directorio de Git que almacena información de lo que se añade a tu próxima confirmación. También se denomina índice (“index”).**

## Comandos GIT 7

¿Por qué estados y en qué orden pasa un fichero hasta que es confirmado?

- **Working Directory, modified, staged, committed**
- Modified, staged, committed , working Directory
- Working Directory, staged, modified, committed
- Working Directory, modified, committed, staged

## Comandos GIT 8

¿Cual es el comando para ver la configuración global de GIT?

- git config --global --list
- **git config --list**
- git config --global --vars
- git config --global --user

## Comandos GIT 9

¿Con qué comandos inicializamos un repositorio?

- **git init**
- git start
- git clone
- git clone --bare

## Comandos GIT 10

¿Cómo comprobamos el estado de los ficheros en un repositorio?

- git log
- **git status**
- git diff
- git diff --staged

## Comandos GIT 11

¿Cómo se añaden los cambios de un fichero llamado “file.txt” al staging area? [Múltiple]

- git log file.txt
- **git add .**
- **git add file.txt**
- git add status file.txt

### Comandos GIT 12

¿Cómo puedes confirmar los cambios en un fichero modificado? [Múltiple]

- **git add . ; git commit -m "Mensaje"**
- git commit -m "Mensaje"
- git add .
- **git commit -a -m "Mensaje"**

### Comandos GIT 13

¿Cómo se pueden ver los cambios realizados en un repositorio y la diferencia entre los ficheros actuales y las versiones anteriores?

- git log
- **git log --p**
- git log --diff HEAD~1
- git log --diff

### Comandos GIT 14

¿Con qué comandos se puede crear una rama? [Múltiple]

- **git checkout -b branch\_name**
- **git branch branch\_name**
- git create branch\_name
- git checkout branch\_name

### Comandos GIT 15

¿Cómo borramos una rama de un repositorio local?

- git branch -D branch\_name
- **git branch -d branch\_name**
- git branch --delete branch\_name
- git remove branch\_name

### Comandos GIT 16

¿Cómo puedes exportar el contenido de un repositorio GIT a partir del cual pueda clonar otro repositorio y mantener toda la historia?

- git clone --bare mirepo.git mirepo
- **git clone --bare mirepo mirepo.git**
- git clone mirepo.git mirepo
- git clone mirepo mirepo.git

### Comandos GIT 17

¿Cómo puedes reescribir el mensaje de un commit?

- **git commit --amend**
- **git commit --amend -m "Nuevo mensaje"**
- git commit --rename -m "Nuevo mensaje"
- git reset HEAD~1 --soft

### Comandos GIT 18

Escribe el comando que te permite eliminar el último commit sin perder los cambios

**git reset HEAD~1 --soft**

### Comandos GIT 19

Escribe el comando que te permite eliminar el último commit sin dejar registro de cambios  
**git reset HEAD~1 --hard**

### Comandos GIT 20

¿Cómo puedes llevar un fichero del área de staged al área de modificado?

- **git reset file.txt**
- **git restore file.txt**
- **git checkout file.txt**
- **git reset file.txt --staged**

### Comandos GIT 21

Escribe el comando que te permite ver la diferencia entre el commit al que apunta HEAD y dos commits previos.

**git diff HEAD HEAD~2**

### Comandos GIT 22

Se puede enmendar un commit sin escribir un mensaje en la ejecución del comando.

- **Verdadero**
- **Falso**

### Comandos GIT 23

Ponemos por situación que hemos realizado un "git add ." y hemos añadido tres ficheros al stage area pero ahora queremos realizar varios commits diferentes para cada fichero. ¿Cómo lo haremos?

- **git commit file1.txt -m "Commit file1" ; git commit file2.txt file3.txt -m "Commit files"**
- **git commit -m "Commit files"**
- **git commit file1.txt file2.txt file3.txt -m "Commit files"**
- **git add . ; git commit -m "Commit files"**

### Comandos GIT 24

¿Con qué opción de GIT puedo simular un commit para ver qué se incluiría en él sin que sea persistente?

- **git commit --dry-run -m "Test commit"**
- **git commit --preview -m "Test commit"**
- **git commit --staged -m "Test commit"**
- **git commit --review -m "Test commit"**

### Comandos GIT 25

Cuando clonamos un repositorio, éste puede contener ramas. Esas ramas se referencian a las del repositorio origen. ¿Cómo puedo seguir una rama de un repositorio clonado en mi repositorio local?

- **git checkout -t origin/nombre\_de\_la\_rama**
- **git branch -f origin/nombre\_de\_la\_rama**
- **git checkout -b origin/nombre\_de\_la\_rama**
- **git branch -t origin/nombre\_de\_la\_rama**

### Comandos GIT 26

¿Cómo puedo crear una rama local asociada a la misma rama remota y con el mismo nombre y al mismo tiempo cambiarme a ella?

- **git branch -u origin/nombre\_de\_la\_rama**
- **git branch -t origin/nombre\_de\_la\_rama**
- **git checkout -b -t origin/nombre\_de\_la\_rama**
- **git checkout -b -u origin/nombre\_de\_la\_rama**

### Comandos GIT 27

¿Cómo puedo borrar una rama local?

- **git branch -d la\_rama**
- **git checkout -d la\_rama**
- **git branch -r la\_rama**
- **git checkout -r la\_rama**

### Comandos GIT 28

¿De qué forma puedo eliminar una rama remota en un solo comando?

- **git branch :rama**
- **git push origin :rama**
- **git branch -D rama**
- **git checkout -D rama**