

Ejercicios de comandos 1

Comandos básicos

Objetivos:

- Manejo de comandos básicos.
- Creación de un repositorio local y exploración de su estructura.
- Consulta y modificación de parámetros de configuración.

Pre-requisitos:

- Trabajar con un usuario de Linux llamado *alumno*.
- Tener instalado Git.

Comandos Git utilizados:

- `git init`
- `git status`
- `git log`
- `git show`
- `git config`
- `git var`
- `git gui`
- `git help`
- `git remote`

Comandos Linux utilizados:

- `cd`
- `ls -la`
- `less`
- `rm`

- 1) Posicionarse en el directorio de repositorios locales.
`cd /home/alumno/Documentos/girRepos`

- 2) Mostrar la ayuda simplificada en línea del comando `git init`.
`git init -h`

Mostrar la ayuda detallada del mismo comando.
`git init --help`

- 3) Crear un repositorio llamado `repo_uno`.
`git init repo_uno`

Comprobar:

- o Que se ha creado un directorio llamado como el repositorio. Entrar en él.
`cd repo_uno`
- o Que ese directorio contiene un directorio oculto llamado `.git` (metadatos del repositorio). Listar su contenido.
`ls -la .git`

- 4) Comprobar el estado del repositorio.
`git status`
- 5) Comprobar el listado de los *commits* ya realizados (ninguno).
`git log`
Se informa que el valor por defecto del archivo HEAD es incorrecto.
`fatal: bad default revision 'HEAD'`
- 6) Comprobar el contenido del archivo HEAD.
`less .git/HEAD`
Contiene:
`ref: refs/heads/master`
Si se busca esa ubicación, se verá que no existe (aún).
`ls .git/refs/heads/master`
- 7) Sabiendo ya que producirá un error, comprobar la información sobre el último *commit* realizado (sigue sin haber ninguno).
`git show`
Como sucedió con `git log`, se informa que el valor por defecto del archivo HEAD es incorrecto.
`fatal: bad default revision 'HEAD'`
- 8) Listar todos los parámetros de configuración.
`git config -l`
- 9) Establecer parámetros de configuración para el usuario: su nombre y su correo.
`git config --global user.name <el_de_GitHub>`
`git config --global user.email <el_de_GitHub>`
Comprobar los parámetros globales (del usuario) definidos.
`git config --global -l`
- 10) Comprobar la existencia y el **contenido del archivo de configuración global** de Git, `.gitconfig`, propio de cada usuario y ubicado en el directorio personal del usuario. Si existen parámetros de configuración globales definidos, el archivo existirá. Sino, no existirá.
`less /home/alumno/.gitconfig`
- 11) Listar los **parámetros de configuración de sistema** (comunes a todos los usuarios).
`git config --system -l`
Si existen parámetros del sistema definidos, comprobar el **contenido del archivo de configuración de sistema** de Git. Si no existen parámetros definidos, el archivo no existirá.
`less /etc/gitconfig`
- 12) Listar los **parámetros de configuración locales** (específicos de un repositorio).
`git config --local -l`
Comprobar el contenido del **archivo de configuración local** de Git.
`less .git/config`

13) Comprobar las variables lógicas.

```
git var -l
```

Comprobar que son un super-conjunto de los parámetros.

14) Arranque de la herramienta gráfica integrada (no existe en todas las versiones de Git).

```
git gui
```

15) Muestra de la ayuda en línea.

```
git help
```

Aquí se ven los comandos más usados. Para ver un listado completo.

```
git help -a
```

16) Comprobar los repositorios remotos vinculados (no debe haber ninguno).

```
git remote
```

17) Para eliminar el repositorio local, basta con eliminar la carpeta .git.

```
rm -fr .git
```