Cursos de GitHup

Temas: posh-git

**Configuración inicial**

git config --global user.name "Charles Arboleda"

git config --global user.email charles@gmail.com

Indicamos a git que VS Code es nuestro editor de texto

git config --global core.editor "code --wait"

**Como ver nuestra configuración global**

git config --global -e

**Configuración de nivel**

core.autocrlf = Se usa para cambiar la configuración de la manera en que se inserta un espacio o tabulación en windows o en linux

git confg --global core.autocrlf true

sí estamos en Linux cambiamos true por input

**Para saber todas las configuraciones de git usamos el comando**

git config -h

muestra todos los directorios ocultos

ls -a

**Flujo de trabajo en git**

|  |  |  |  |
| --- | --- | --- | --- |
| **Computador** | **Stage** | **Commit** | **Server** |
| Archivo local | Tenemos archivos | Pasamos archivos a archivos de commit | Los cambios que incluimos quedaran aquí |
| Archivo a eliminar | Git add | Hacemos commit | Eliminamos archivo del repositorio |

**Agrega todos los archivos al Stage**

git add .

**Para agregar un commit**

git commit -m "inicio de comentarios"

**Eliminar archivos**

rm nombre del archivo

**Elimina un archivo y nos ahorra un paso**

git rm nombre del archivo

**Para restaurar un archivo**

git restore --staged <file>

**como cambiar el nombre de un archivo**

mv pruebagit.txt archivo,txt

**Si queremos agregar a Staged nuestro cambio**

git mv pruebagit.txt archivo,txt

Como hacer que un archivo No sea tenido en cuenta por git

Creamos un archivo .gitignore y en este archivo escribimos el

Nombre del archivo o carpeta que queremos ignorar

**Para ver los archivos de foma diferente luego de modificados**

gif diff

**Para ver solo los cambios que queremos realizar**

git diff –staged

**muestra los cambios realizados los commit y su número identificador**

git log

git log –oneline --🡪 Versión mejorada del comando

**Para saber en que rama estamos trabajando**

git Branch

**Como crear una rama**

git checkouk -b *nombre de la rama*

Algunas empresas utilizan esta nomenclatura para nombrar la rama de desarrollo

**git checkout -b *features/nombre-de-la funcionalidad***

Otras empresas colocan el numero de tiket que se quiere reparar

**git checkout -b *ss-4515***

Para unir dos ramas ejemplo rama master con ramab

Estando parado en la rama que vamos a unir escribimos el comando

git merge nombre-de-la rama para nuestro ejemplo

git merge ramab

**Como subir nuestro código a la nube utilizando github**

Abrimos github creamos un nuevo repositorio “nombre de la carpeta que tenemos en loca”

Y copiamos la línea git remote add origin https://github.com/charles-phyton/Python\_Diario.git

Luego copiamos la línea git push -u origin main

En mi caso la rama principal se llama master

Si queremos subir una rama pero sin hacer merge debemos

Cambiar de rama

git checkout -b nombredelarama**ramab**

**git push -u origin ramab**