26/08/2016

Cd // entramos al directorio donde trabajaremos

Ls  // Nos permite ver los archivos que contiene los directorios

Git clone [http://git........](http://git......../) // copiamos los archivos online a raíz

Git log // vemos todos los cambios y comentarios que se han hecho

Git diff documento1 documento2// compara 2 documentos y muestra cambios

git checkout documento // regresar a versión anterior

git commit/ cambio que se hizo y que esté disponible

Git push/ Sube cambios online

git pull / Baja cambios online a raíz

Usaremos tabuladores espacios con \ es continuación

Codificación UTF-8

Cabecera

#!/usr/bin/env python 

# -\*- coding: utf-8 -\*-

CameCase

\_\_funciones de inicio\_\_

Self: para todas las funciones

03/09/2016

git checkout HEAD// Regresa a la version HEAD branch

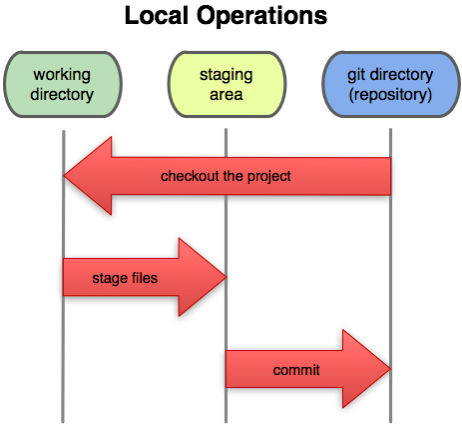
git checkout master// se utiliza cuando el archivo tiene varios branch

git init//es para crear nuevo repositorio

Git status// estado de nuestra carpeta git

Git add/ agregar seguimiento a archivos dentro del git

Git commit /se agrega commit



Al hacer push se va a Staging área

Crear commit

Git commit

Git diff

Git diff --staged //Muestra la diferencia con los cambios en stagin area

Git reset –hard Head//regresa y borra branches

09/09/2016

Dirección del proyecto en consola: python manage.py runserver

Arranca el servidor

python manage.py createsuperuser

Crea un super usuarios

Para aceder a mysql

mysql -u root -p

python manage.py  makemigration  // Crea la migración a partir de los modelos

python manage.py  migrate  // Aplica la migración

python manage.py starapp basededatos modulouno

Crea módulo de aplicación

Startapp modulouno

Inicia el modulo

23/09/2016

Branch: punto en el que se desvia el repositorio

Se hace cuando se va resolviendo un problema muy grande

Merge: convina el ultimo repositorio del master y el branch

Master 0—0—0—0—0

git checkout master // Regresa a la rama master

git checkout branch // Regresa a la rama branch

git branch ulises // crea el branch

Se veran graficamente las ramas

git log --graph --oneline master ulises

Iniciar python

cd /home/ulises/proyectos/git/prueba   
python manage.py runserver 

Cambios en la base de datos

Python manage makemigrations

python manage.py migrate

python manage.py createsuperuser   
   
paquete-> modulo que hace algo

Se crea modulo

python manage.py startapp nombredemodulo

A los modelos robustos los controladores, y las vistas lo más sencillo.

30/09/2016

Python magae.py nombre del modulo

Git ignore (ignarar archivos que ya estaba siguiendo)

Git branch nombredelbranch crea rama branch

Git merge Master y otrobranch Une las 2 ramas en una sola

git branch -d nombredelbranch  borra branch.

El usuario que realizo el ultimo commit fue: zeack