



Universidad Tecnológica Nacional

COMANDOS DE GIT

Y TERMINAL DE GITBASH

Grupo CopyPaste





Comandos básicos en terminal

Aquellas herramientas que utilizamos en nuestra computadora de manera local



Espacio de trabajo

Donde generamos los archivos en el directorio de manera local



Área de preparación

Se prepara el trabajo realizado que se quiera subir, agregándolos a esta área



Repositorio

Guarda los archivos y el trabajo realizado de manera permanente y cronológica en el directorio de GIT

Comandos básicos en terminal

•**mkdir + nombre**

aquí tipeamos el comando mkdir+el nombre de la carpeta que queremos crear

•**pwd**

Nos indica en que carpeta nos encontramos

•**cd .. / cd - / cd +nombre**

Con este comando ingresamos a una carpeta indicando su nombre. Si utilizamos "-" volvemos al ultimo lugar que ingresamos y con ".." salimos de esa carpeta hacia la anterior

•**clear**

limpia la pantalla de la terminal

•**ls / ls -l**

lista las carpetas del directorio que nos encontramos "ls -l" brinda más información sobre el contenido de cada carpeta

•**Git Config --local User.name "Nombre y Apellido"**

Configuramos nuestro repositorio de manera Local con el nombre de usuario

•**Git Config --local user.email"ejemplo@mail.com"**

Seguimos con la configuración de nuestro repositorio indicando nuestro correo electrónico de manera local

•**Git rm**

Eliminamos un elemento/carpeta

•**Git config --list**

Muestra la lista de configuraciones predeterminadas de GIT

Espacio de trabajo

•Git init

Iniciamos un repositorio

•Git add

Agrega los cambios y los lleva al área de preparación y luego ser confirmados

•Git commit

Confirma los cambios añadidos al área de trabajo (staging area) se guarda la versión en el repositorio con un mensaje describiendo cambios

•Git status

Se muestra el estado de los archivos que han cambiado

•Git diff

Visualizamos las modificaciones realizadas

•Git ignore

Contiene archivos y carpetas que deben ser ignorados por GIT



Área de preparación



•Git branch

Vemos las ramas del repositorio

•Git merge

Genera la fusión de las ramas creadas

•Git show

Muestra información del commit específico solo cuando utilizo la etiqueta con comentario.

•Git checkout

Viajamos en la línea cronológica de tiempo por los diferentes commit de nuestro proyecto

•Git tag

Muestra la lista de las versiones del proyecto que creamos

•Git log

Visualizamos el historial de nuestros commit

•Git stash

Generamos un espacio temporal en el cual podemos guardar progresos o modificaciones de nuestro proyecto

•Git switch

Se utiliza para cambiar de una rama a otra