CHEATSHEET

Comandos más utilizados en clase

**GIT**

**git init** -> Sirve para inicializar un repositorio.

**git config --list** -> Nos permite visualizar la configuración de git. Además, los siguientes comandos a continuación sirven para establecer el usuario y el email para todos los repositorios que iniciemos:

**git config --global user.name** "nombre de usuario"

**git config --global user.email** email@email.com

**git config --global credential.helper store** -> Nos sirve para configurar un token como contraseña global

**git config --global credential.github.com.username <token>** -> Una vez creado el token personal, lo ponemos dentro de <token> y donde dice username ponemos nuestro usuario de github.

**git status** -> Nos permite visualizar el estado de los archivos dentro de su repositorio

**git add .** -> Permite agregar al repositorio todos los archivos

**git add nombre\_archivo** -> Permite agregar al repositosio un archivo específico

**git commit -m "mensaje"** -> Podemos poner un mensaje al commit y es nuestro punto de guardado en nuestro historial del proyecto

**git -M main** -> permite cambiar a la rama main

**git log** ->Muestra historial de nuestros commits

**git log --oneline** -> Muestras los commits realizados en una sola línea.

**git rm archivo** -> Borra un archivo

**git rm -r** ->Elimina el directorio y los archivos

**git push -u origin main** -Sube los archivos locales al repositorio remoto

**git remote -v** -> Permite visualizar conexión con repositorio remoto

**git remote add origin (nombre o url del repositorio remoto**) -> Enlaza el repositorio local con el remoto

**git remote rm nombre\_git** -> desvincula un repositorio remoto

**git pull origin main** ->Baja los cambios o actualizaciones del repositorio remoto al local

**git fetch** -> obtiene los cambios, pero no los aplica

**git clone ruta**-> clona un repositorio

**git branch nombre** -> Crear una nueva rama

**git checkout nombre** ->entrar a la rama

**git checkout main** -> volver a rama principal

**git pull origin nombreBranch** ->Saca los archivos de la branch existente

**git push origin nuevaBranch\_nombre** -> Crea branch remota con el mismo nombre

**git merge newbranchversion** ->Fusiona una o más ramas con otra rama activa y

crea automáticamente un nuevo commit si no hay conflictos

**GIT BASH**

**ls -a** -> Ver los elementos y también los elementos ocultos

**ls -a -R** -> Permite ver las carpetas y archivos de las capetas, así como elementos ocultos

**clear** ->limpia la consola

**cd nombrecarpeta** -> Entra a la carpeta seleccionada

**cd ..** -> Sale de la carpeta

**mkdir nombre** ->Crea carpeta

**touch nombre** ->Crea un archivo

**rm nombre** ->Borra un archivo

**rm -r nombre** -> Borra directorio y los arhivos que contiene

**rmdir** -> Elimina carpeta vacía

**rm -rf nombre** -> Borra directorio y todos los archivos en forma forzada

**rm -rf. git** ->Borra el repositorio git

**pwd** -> Muestra la carpeta actual donde se está en ese momento

**mv (ruta/archivo1) (ruta/archivo2)**-> Renombra archivos

**mv (ruta/carpeta1) (ruta/carpeta2)**-> Mueve el contenido de la carpeta 1 a carpeta 2

**cp (ruta/archivo1) (ruta/archivo2)** -> Copia archivo o carpeta

**"" (Comillas)** -> Nos permite utilizar terminas con espacios o con varias palabras

**find -name / find -size**: Busca archivos o carpetas por nombre o por tamaño

**nano** -> Nos permite agregar texto a un documento o archivo