Instalación:

* Descarga: **git-scm.com/downloads**
* Consultar la version: **git –version**

Configuración de GIT desde la línea de comando Git Bash (Bourne-Again Shell)

* Configurar git con nuestro nombre: **git config --global user.name “Martin Cano Segura”**
* Configurar correo electrónico para git: **git config --global** [**mecano\_mcs@hotmail.com**](mailto:mecano_mcs@hotmail.com)
* Configurar el editor a usar en git: **git config --global code.editor“code --wait"**
* Ver el archivo de configuración global: **git config --global –e**  el archivo de configuración global se abrirá dentro del editor Visual Studio Code, la línea de comando esperara hasta que se cierre el archivo de configuración.
* Configuración para el salto de línea (Windows= true, Linux/Mac=input): **git config --global core.autocrlf true**
* Para ver más sobre configuraciones: **git config –h**

Comandos Unix/Linux:

* Listar todos los archivos y carpetas dentro del directorio: **ls**
* Ruta del directorio/carpeta actual: **pwd**
* Moverse a una carpeta dentro del directorio actual: **cd <nombreCarpeta>** si escribes cd + primera letra del directorio + tab, autocompletara el nombre de la carpeta.
* Subir un nivel hacia arriba en las carpetas: **cd ..**
* Crear una carpeta: **mkdir <nombreCarpeta>**
* Mostrar carpetas y archivos ocultos dentro del directorio/carpeta: **ls –a**
* Eliminar un archivo: **rm <nombreArchivo.ext>**
* Cambiar nombre de un archivo (renombrar): **mv <fileOrigen> <fileDestino>**
* Mostrar el contenido de un archivo: **cat <file>**
* Inicializar un nuevo repositorio local para gestionar los archivos y carpetas del directorio: **git init**  esto creara la carpeta oculta .git
* Abrir el VSC desde el GitBash en la carpeta actual: **code .**
* Ver el estado actual de los archivos y carpetas en el repositorio local de git: **git status** o **git status –s** para ver el seguimiento de los archivos (M rojo=modificado, M verde=En el stage, ? rojo= Sin seguimiento, A verde= Agregado al stage
* Agregar un archivo al repositorio local (stage): **git add <nombreArchivo>**  o todos los archivos **git add .** o solo los que cumplan con una Expresión Regular **git add .txt** o **git add <arch1.txt> <arch2.txt> …**
* Enviar los archivos al repositorio local (stage) 🡪 (repositorio): **git commit –m “<comentario>”** o solo **git commit** esto abrirá el VSC con el archivo COMMIT\_EDITMSG con indicaciones de las modificaciones hechas, al inicio de este archivo se puede agregar el comentario
* Eliminar un archivo del stage: **git rm <nombreArchivo.ext>** para posteriormente hacer el commit
* Eliminar o Revertir un cambio de uno o más archivos en el stage: **git restore -- stage <file>**
* Restaurar/Descartar los cambios (eliminación del stage) para recuperar archivos: **git restore <file>**
* Renombrar un archivo dentro del stage: **git mv <fileOrigen> <fileDestino>**
* Resumen de cambios de un archivo: **git diff**
* Resumen de cambios en el stage: **git diff --staged**
* Revisar el historial de todo el repositorio con los id de los commit ejecutados: **git log --oneline**
* Revisar la rama en la que se está trabajando: **git branch**
* Crear una rama nueva: **git checkout –b <nombre-rama>**  al crear la rama se estará en dicha rama
* Cambiarse a otra rama: **git checkout <nombre\_rama>**
* Enviar los cambios de una rama\_origen a otra rama\_destino (mezcla/merge), debiendo estar situados en la rama\_destino: **git merge <rama\_origen>** ejemplo: (estando en la rama main) git merge ramab

Subir el código fuente a un repositorio de GitHub:

* Crear un repositorio en GitHub: ir al sitio y crearlo
* Agregar el repositorio remoto en GitHub: **git add origin <url del repositorio>** ejemplo: **git add origin https://github.com/MartinCanoSegura/ExamenGpoSalinas.git**
* Subir cambios a la rama respecto a la rama local en que se está trabajando: git push –u origin main donde main es la rama si la rama no existe en el remoto será creada esto se indica con los parámetros –u origin. Esto pedirá el usuario de git y la contraseña (no es la de registro en GitHub) sino es necesario generar un tocken desde el mismo GitHub (signed in as ‘icono de la foto – esquina superior derecha’) /Settings/<>Developer settings/Personal Access tocken/Generate new token/note=<un string>, seleccionar que expire en 30 días, checkear repo, clic al botón Generate token, lo copiamos y lo pegamos en la ventana donde requiere la contraseña para subir los cambios.
* En los posteriores cambios al código local solo se agrega el código **git add.** al stage se hace commit **git commit –m “comentario”** y se subirá al repositorio remoto con **git push**
* Si se está trabajando en otra rama localmente y se quieren subir los cambios a GitHub en la misma rama que la local: por ejemplo si localmente estamos en la rama ramaC hacemos: **git push –u origin ramaC** esto creara la ramaC remota del repositorio remoto de GitHub.

**git reset**

Revierte lo hecho por (git add .), saca/elimina del stage los archivos

**git checkout --.**

Recupera los archivos (borrados o modificados) desde el último commit

**git log**

Se obtiene todo el listado de los commit

**git commit --amed**

Modifica el mensaje del ultimo commit hecho (tecla i para incertar / modificar el mensaje, (para terminar de editar el mensaje teclas: esc,:,w,q,!)), (q para salir de editar comando)

**git checkout -b <nombre-rama>**

Crear una nueva rama tomando todo lo hecho desde el último commit

**git branch**

Muestra todas las ramas incluyendo la master y la creada, a la rama creada le aplican todos los comandos anteriores

**git checkout <nombre-rama>**

Cambiarse a la rama indicada

**git merge <nombre-rama>**

Fusionar la rama con la rama master 'Estando en la rama master', (teclear: esc,:,w,q,!)

**git branch -d <nombre-rama>**

Eliminar una rama

Para enviar los archivos a un repositorio remoto 'github' se crea el repositorio remoto, se ejecutan los comandos indicados en git:

a) **git remote add origin http://github.com/klerith/youtube-git.git**

b) **git branch -M main**

c) **git push -u origin main**

Pedirá las credenciales de git

**git push**

Envía los archivos del repositorio local al repositorio remoto en este caso github

**git commit -am <mensaje>**

Para ejecutar los comandos (git add . y git commit) de forma simultanea