



CREA UN ARCHIVO GITIGNORE

Propósito del documento

Este documento tiene como propósito el ser una guía práctica y sencilla para que usuarios principiantes en GitHub aprendan a crear un archivo .gitignore desde la terminal de VSCode.

Escenarios en donde se aplica

Este documento se aplica para los usuarios que ya hayan clonado el repositorio de la clase, ya hayan creado una branch y finalmente ya tengan una carpeta principal en la que suben sus trabajos, todo esto para que ahora puedan crear un archivo .gitignore desde la terminal de VSCode.

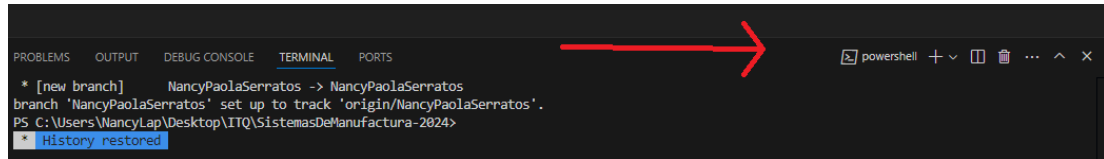
Definiciones y acrónimos

GitHub	Es una plataforma en línea que permite a los usuarios almacenar, compartir y trabajar en proyectos de software.
VScode	Es un editor de código para programadores gratuito, de código abierto y multiplataforma.
Git	Git es un sistema de control de versiones. Permite a los desarrolladores controlar y administrar los cambios en su código fuente a lo largo del tiempo.
Repositorio de GitHub	Es un espacio de almacenamiento virtual donde se puede guardar el código, los archivos y el historial de revisiones de un proyecto.
Rama (branch)	División del estado del código, que se encarga de desarrollar un nuevo mecanismo para su evolución.
Git bash	Es una aplicación para Windows que permite a los usuarios utilizar las funciones de Git a través de una línea de comandos
Archivo .gitignore	El archivo . gitignore, es un archivo de texto que le dice a Git qué archivos o carpetas ignorar en un proyecto.

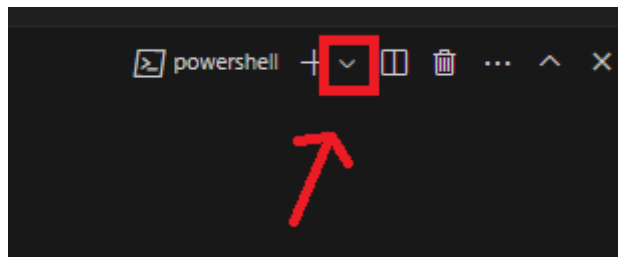


Nota antes de iniciar: Es recomendable realizar el procedimiento desde git bash para que sea más fácil saber que estás trabajando desde tu rama.

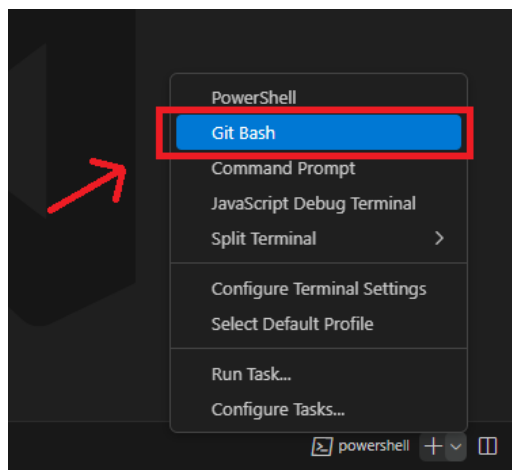
Para entrar al apartado de git bash en VScode te diriges al lado derecho de la pantalla de inicio de VScode, cerca de los apartados de terminal y ports.



Das clic al siguiente botón.



Se desplegarán una serie de opciones y se selecciona el apartado de Git bash



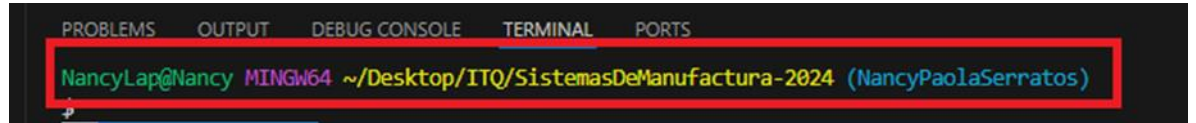
Después de seleccionar git bash se abrirá el siguiente formato y puedes empezar a trabajar desde git bash, ya que trabajar desde ahí te permitirá tener más claridad desde que rama estas trabajando y no arriesgarse a que accidentalmente estes trabajando desde la rama de main.





INICIO DEL PROCEDIMIENTO LA CREACION DEL ARCHIVO GITIGNORE

1. Tienes que asegurarte de estar en tu Branch antes de crear cualquier nuevo archivo o carpeta. Si estás trabajando desde git bash, el nombre de tu rama aparecerá al final de la dirección de la terminal.

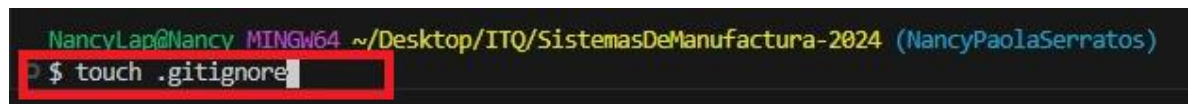


```
NancyLap@Nancy MINGW64 ~/Desktop/ITQ/SistemasDeManufectura-2024 (NancyPaolaSerratos)
```

(Si no aparece el apartado de la **terminal** en VScode, presionar las teclas Ctrl+ñ, ambas al mismo tiempo para que aparezca el apartado).

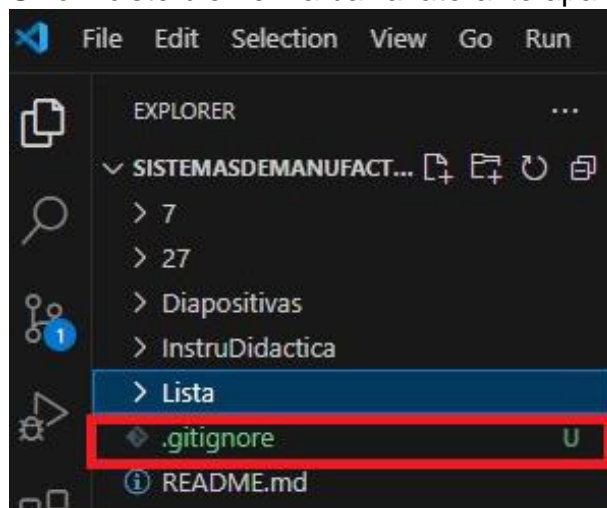
Nota: Al final tiene que estar escrito el nombre de tu Branch, ya que si aparece main quiere decir que estás trabajando en la rama incorrecta y corres el riesgo de que te bajen calificación por modificar archivos desde main.

2. Crea un archivo. gitignore, para crear el archivo escribe en la línea de comandos **touch .gitignore** y presiona enter.



```
NancyLap@Nancy MINGW64 ~/Desktop/ITQ/SistemasDeManufectura-2024 (NancyPaolaSerratos)
$ touch .gitignore
```

Si lo hiciste bien en la barra lateral te aparecerá el archivo **.gitignore**



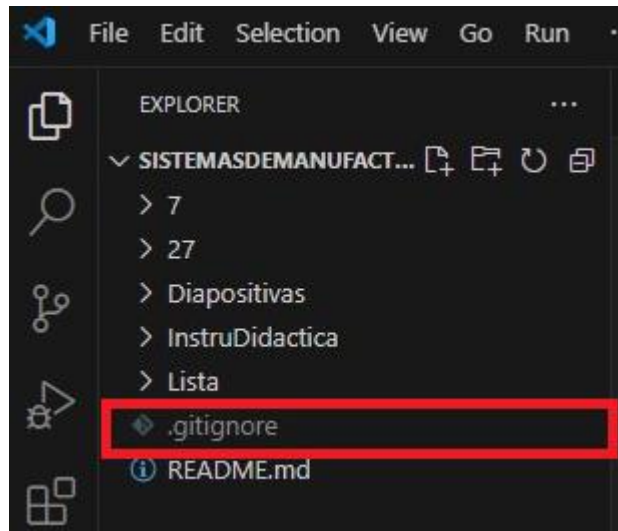


3. Da doble clic en el archivo **.gitignore** y escribe lo siguiente:

```
◆ .gitignore
1 # Ignora todos los nuevos archivos que se van a crear
2 /*
3 # Registra todos los cambios de mi carpeta e ignora la de mis compañeros
4 !/27
5 #Si alguien crea un nuevo archivo que no sea yo ignoralo
6 /27/*
7 # Excluir todo excepto el directorio 27/img
8 !/27/img
```

Nota: El número que aparece en el ejemplo vas a reemplazarlo por tu número de lista.

El estatus de tus archivos cambiara ha guardado y ya no aparecerá la letra **U**.



4. Escribe **git status** para obtener información de en que estado estas y que es lo que te sugiere git

```
NancyLap@Nancy MINGW64 ~/Desktop/ITQ/SistemasDeManufactura-2024 (NancyPaolaSerratos)
$ git status
On branch NancyPaolaSerratos
```

Al dar enter te desplegara la información:

```
NancyLap@Nancy MINGW64 ~/Desktop/ITQ/SistemasDeManufactura-2024 (NancyPaolaSerratos)
$ git status
On branch NancyPaolaSerratos
Your branch is up to date with 'origin/NancyPaolaSerratos'.

nothing to commit, working tree clean
```



5. A partir de este paso puedes usar el SOP de cómo crear directorios u archivos para subirlos en tu carpeta.
Si por error modificas otro archivo **git** no te permitirá subir los archivos a GitHub.