

1.Instalar GIT en nuestro equipo

```
C:\github\Ecommerce>git --version  
git version 2.51.2.windows.1
```

2. Acceder a la carpeta de proyecto de las clases anteriores, mediante la línea de comandos de la terminal e inicializar GIT

```
C:\github>git clone https://github.com/SoundEmpire25/Ecommerce.git  
Cloning into 'Ecommerce'...  
remote: Enumerating objects: 308, done.  
remote: Counting objects: 100% (15/15), done.  
remote: Compressing objects: 100% (10/10), done.  
remote: Total 308 (delta 5), reused 6 (delta 5), pack-reused 293 (from 1)  
Receiving objects: 100% (308/308), 1.27 MiB | 3.37 MiB/s, done.  
Resolving deltas: 100% (112/112), done.
```

```
C:\github>cd Ecommerce
```

```
C:\github\Ecommerce>dir
```

El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: C87E-76BC

Directorio de C:\github\Ecommerce

24-11-2025	20:44	<DIR>	.
24-11-2025	20:44	<DIR>	..
24-11-2025	20:44		15 .gitignore
24-11-2025	20:44	<DIR>	cypress
24-11-2025	20:44		191 cypress.config.js
24-11-2025	20:44	<DIR>	Diseño
24-11-2025	20:44	<DIR>	Documentación
24-11-2025	20:44		1.091 LICENSE
24-11-2025	20:44		78.470 package-lock.json
24-11-2025	20:44		314 package.json
24-11-2025	20:44	<DIR>	Prototipos
24-11-2025	20:44	<DIR>	Proyecto
24-11-2025	20:44	<DIR>	QA
24-11-2025	20:44		4.815 README.md
		6 archivos	84.896 bytes
		8 dirs	256.800.153.600 bytes libres

```
C:\github\Ecommerce>code .
```

3. Definir los siguientes comandos:

git --version: Muestra la versión de git instalada en el sistema.

```
Marcelo@KURURU MINGW64 /c/github/Ecommerce (backend/mcastilloc)
$ git --version
git version 2.51.2.windows.1
```

git init: Inicializa un nuevo repositorio Git en la carpeta actual, crea la carpeta oculta .git, donde Git guarda toda la información del repositorio.

```
Marcelo@KURURU MINGW64 /c/github/Ecommerce (backend/mcastilloc)
$ git init
Reinitialized existing Git repository in C:/github/Ecommerce/.git/
```

git add: Agrega archivos al área de preparación, es decir, Git los marca para incluirlos en el próximo commit, estos pueden ser de forma individual o usar el comodín (.) para agregar todos los archivos.

```
Marcelo@KURURU MINGW64 /c/github/Ecommerce (backend/mcastilloc)
$ git add .
```

git commit: Crea un commit, es decir, guarda una versión de los cambios añadidos al staging con git add, siempre va acompañado de un mensaje con el flag -m

```
Marcelo@KURURU MINGW64 /c/github/Ecommerce (backend/mcastilloc)
$ git commit -m "actualizaciones varias"
[backend/mcastilloc 0de263f] actualizaciones varias
1 file changed, 2 deletions(-)
```

git status: Muestra el estado del repositorio, indicando:

- cambios sin guardar
- archivos listos para commit
- archivos sin seguimiento
- en qué rama estás

```
Marcelo@KURURU MINGW64 /c/github/Ecommerce (backend/mcastilloc)
$ git status
On branch backend/mcastilloc
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   Proyecto/ASSETS/CSS/styles.css

no changes added to commit (use "git add" and/or "git commit -a")
```