

```
+++ date = '2026-02-19T22:06:10-08:00' draft = false title = 'Practica0: Manejo de Repositorios' +++
```

# Markdown

---

Es un lenguaje de marcado ligero que permite dar formato a texto plano de manera sencilla y legible.

Su ventaja principal es que es fácil de leer y escribir, sin necesidad de conocer HTML o herramientas complejas.

## **Utilizado para:**

- Documentación técnica
- Archivos README en proyectos
- Blogs y páginas web estáticas
- Apuntes y notas
- Publicaciones en GitHub, GitLab, foros técnicos, etc.

## Ejemplos de Sintaxis

---

---

**Esto es un encabezado H1**

---

Esto es un encabezado H2

Esto es un encabezado H3

**Esto es un encabezado H4°**

**Esto es un encabezado H5**

Esto es un texto en *italicas*

*Esto es un texto en italicas*

Esto es un texto en *italicas*

*Esto es un texto en italicas*

Esto es un texto en **negritas**

**Esto es un texto en negritas**

Esto es un texto en **negritas**

**Esto es un texto en negritas**

Este es un tecto que puede ser código este es un texto ~~tachado~~ texto ~~tachado~~

- Elemento 1
- Elemento 2
- Elemento 3
  - Elemento 3.1
  - Elemento 3.2
    - Elemento 3.2.1
- Elemento 4
  
- Elemento 1
- Elemento 2
- Elemento 3
  - Elemento 3.1
  - Elemento 3.2
    - Elemento 3.2.1
- Elemento 4
  

  1. Elemento 1
  2. Elemento 2
  3. Elemento 3
  4. Elemento 3.1
  5. Elemento 3.2
  6. Elemento 4

[Texto de enlace](#)

[UABC](#)





```
This is un code block  
This is the second line of the code block
```

```
printf("Hola mundo!")
```

Productos	Precio	Cantidad
Laptotp	3.3	2
Mouse	13.3	1

Esto es una nota

- Primera tarea
  - Segunda tarea
  - Tercera tarea
  - Cuarta tarea
- 
- 

@troymoreno  

## Git

Es un sistema de control de versiones que permite guardar, organizar y controlar los cambios realizados en los archivos de un proyecto. Facilita el trabajo en equipo, el seguimiento del historial de cambios y la recuperación de versiones anteriores.

## GitHub

Es una plataforma en línea que permite almacenar repositorios Git en la nube, compartir proyectos y colaborar con otros desarrolladores. Funciona como un respaldo y un punto central para el trabajo

colaborativo.

## ¿Cómo se utilizan?

Se trabaja de manera local con Git para guardar cambios y, posteriormente, se suben los archivos a GitHub para almacenarlos y compartirlos. El flujo básico consiste en modificar archivos, registrarlos con Git y sincronizarlos con GitHub.

## Comandos esenciales de Git

- **git init** → Inicializa un repositorio.
- **git status** → Muestra el estado de los archivos.
- **git add .** → Agrega archivos al área de preparación.
- **git commit -m "mensaje"** → Guarda los cambios.
- **git push** → Sube los cambios a GitHub.
- **git pull** → Descarga cambios desde GitHub.

## Hugo

---

Es un generador de sitios web estáticos muy rápido. Permite crear páginas web a partir de archivos Markdown.

### Crear un sitio estático Hugo

1. Instalar Hugo y ejecutar -> **hugo new site mi-sitio**
2. Añadir un tema: Descargar un tema en la carpeta **/themes**
3. Crear contenido: **hugo new posts/mi-primer-post.md**
4. Probar localmente: **hugo server -D**
5. Abrir el https en el navegador

## GitHub Actions

---

Es un sistema de automatización que permite ejecutar procesos automáticos, como:

- Compilar proyectos
- Ejecutar pruebas
- Publicar sitios web automáticamente

Se usa para automatizar el despliegue de un sitio Hugo en GitHub Pages.

### Enlace de portafolio en Github

### Enlace de página estática en Github Pages

