📚 Resumen de la Lecture M0L2: Terminal y Git

| **Concepto** | **Significado** |
| --- | --- |
| Terminal/Shell | Interfaz para interactuar con el sistema operativo mediante comandos. |
| Repositorio | Carpeta con control de versiones administrada por Git. |
| Staging area | Área intermedia donde defines qué cambios se confirmarán. |
| Commit | Registro de cambios confirmado en el historial del repositorio. |
| SHA | Identificador único de cada commit. |
| HEAD | Referencia al último commit en la rama actual. |
| Branch | Rama de desarrollo independiente que permite trabajar en funcionalidades sin alterar la principal. |
| Remote | Repositorio alojado en un servidor (GitHub, GitLab, etc.) conectado con tu repo local. |

**🔹 Comandos de terminal para navegar y manipular archivos**

| **Comando** | **Descripción** |
| --- | --- |
| pwd | Muestra la ruta actual. |
| ls | Lista archivos/carpetas. |
| cd carpeta | Entra a una carpeta. |
| cd .. | Sube un nivel. |
| mkdir nombre | Crea una carpeta. |
| touch archivo | Crea un archivo vacío. |
| rm archivo | Elimina un archivo. |
| rm -r carpeta | Elimina carpeta y contenido. |
| mv origen destino | Mueve o renombra archivos/carpetas. |
| cp origen destino | Copia archivos/carpetas. |

**🔹 Comandos básicos de Git**

| **Comando** | **Descripción** |
| --- | --- |
| git init | Inicializa un repositorio local. |
| git config --global user.name "Tu Nombre" | Configura tu nombre como autor de commits. |
| git config --global user.email "email" | Configura tu email como autor de commits. |
| git status | Muestra archivos modificados, nuevos o eliminados. |
| git add archivo | Agrega un archivo al staging area. |
| git add . | Agrega todos los archivos modificados al staging area. |
| git commit -m "mensaje" | Crea un commit con los cambios del staging. |
| git log | Lista historial de commits. |
| git log --oneline | Lista el historial en formato breve. |
| git diff | Muestra diferencias entre archivos modificados y último commit. |
| git branch | Lista las ramas locales. |
| git branch -M main | Renombra la rama actual como main. |
| git checkout -b rama | Crea y cambia a una nueva rama. |
| git remote add origin <url> | Conecta tu repo local a un repo remoto. |
| git push -u origin rama | Sube commits locales a la rama remota. |
| git pull | Actualiza tu rama local con los cambios remotos. |
| git clone <url> | Clona un repositorio remoto. |
| git restore archivo | Descarta cambios locales no confirmados. |
| git reset --soft HEAD~1 | Deshace el último commit, manteniendo los cambios en staging. |
| git revert HEAD | Crea un commit que revierte el último commit. |

**🔹 Flujo de trabajo aprendido**

1️⃣ Inicializar repositorio:

git init

2️⃣ Configurar usuario:

git config --global user.name "Tu Nombre"

git config --global user.email [tuemail@ejemplo.com](mailto:tuemail@ejemplo.com)

3️⃣ Crear archivos/carpeta y trabajar en ellos.

4️⃣ Ver estado:

git status

5️⃣ Agregar cambios:

git add archivo

# o para todos: git add .

6️⃣ Confirmar cambios:

git commit -m "mensaje descriptivo"

7️⃣ Revisar historial:

git log –oneline

8️⃣ Subir proyecto a GitHub:

git remote add origin <url>

git branch -M main

git push -u origin main

9️⃣ Colaborar en proyectos existentes:

git clone <url>

git checkout -b mi-feature

git add .

git commit -m "mensaje"

git push -u origin mi-feature

📚 Resumen de prácticas M0L3: HTML & Maquetado

**🔹 Conceptos clave**

| **Etiqueta** | **Qué hace** | **Cuándo usar** |
| --- | --- | --- |
| <!DOCTYPE html> | Define el documento como HTML5 | Siempre al inicio del archivo HTML |
| <html lang="es"> | Etiqueta raíz; lang define el idioma | Alrededor de todo el documento HTML |
| <head> | Contiene metadatos: título, codificación, viewport | Antes del <body> |
| <body> | Contenido visible de la página | Estructura el contenido principal |
| <header> | Encabezado de la página o sección | Títulos, logos, navegación |
| <nav> | Navegación principal | Menús, enlaces principales |
| <main> | Contenido principal único por página | Artículos, secciones clave |
| <section> | Agrupa contenido relacionado | Bloques temáticos como “servicios” |
| <footer> | Pie de página | Información de contacto, derechos |
| <ul> | Lista desordenada | Elementos sin orden jerárquico |
| <ol> | Lista ordenada | Elementos con orden importante |
| <li> | Ítem de lista | Dentro de <ul> o <ol> |
| <table> | Tabla | Datos tabulados o comparativos |
| <thead> | Encabezado de tabla | Títulos de columnas |
| <tbody> | Cuerpo de tabla | Datos principales de la tabla |
| <tr> | Fila de tabla | Agrupa celdas en una fila |
| <th> | Celda de encabezado | Títulos de columnas/filas |
| <figure> | Contenedor semántico de medios | Asociar imágenes con texto |
| <figcaption> | Leyenda de figure | Describir imágenes o diagramas |
| <form> | Formulario | Recolectar datos de usuario |
| <label> | Etiqueta para campos | Mejora accesibilidad |
| <input> | Campo de entrada | Textos, correos, números, etc. |
| <textarea> | Campo de texto largo | Mensajes, comentarios |
| <button> | Botón | Enviar formularios, acciones |

**🔹 Atributos y validaciones HTML**

* type="text | email | tel" ➔ define el tipo de dato esperado.
* required ➔ hace obligatorio el campo.
* pattern="regex" ➔ aplica validaciones específicas.
* minlength y maxlength ➔ controla longitud de textos.
* placeholder ➔ muestra un texto guía dentro del campo.
* title ➔ mensaje que se muestra si la validación falla.
* action ➔ URL donde se envían los datos del formulario.
* method="POST|GET" ➔ cómo se envían los datos.