

# Ejercicios para el Itinerario de Clase

## Ejercicios para el Itinerario de Clase (3 horas)

Terminal + Git

---



### 0. Introducción – Ejercicio rápido (5 min)

#### Ejercicio “¿Cuándo usarías la terminal?”

Da 3 ejemplos de tareas que podrían ser más rápidas con terminal que con interfaz gráfica.

(Ejemplo: crear 10 carpetas, mover archivos masivamente, clonar un proyecto).

---

### 1. Introducción a la Terminal y Comandos Unix (60 min)

#### 1.1. Qué es la terminal (5 min)

##### Ejercicio corto

Escribe la diferencia entre **GUI** y **CLI** con tus propias palabras.

Luego, abre la terminal de tu sistema operativo.

---

#### 1.2. Navegación básica (15 min)

##### Ejercicio guiado

1. Ejecuta `pwd` y escribe cuál es tu ruta actual.
2. Usa `ls` para listar tus archivos.

3. Usa `ls -a` y di qué archivos nuevos aparecen.

4. Cambia a tu carpeta de usuario con:

```
cd ~
```

5. Navega a tu carpeta de Descargas o Escritorio.

6. Limpia la pantalla con `clear`.

### Ejercicio reto

Desde donde estés, navega **sin usar Explorer o Finder** a la siguiente ruta:

`Documentos → proyectos → javascript → clase1`

(Crea las carpetas si no existen).

---

## 1.3. Manejo de archivos y carpetas (20 min)

### Ejercicio guiado: “Proyecto rápido”

1. Crear un directorio

```
mkdir proyecto
```

2. Entrar a él

```
cd proyecto
```

3. Crear tres archivos

```
touch index.js notas.txt config.json
```

4. Crear un subdirectorio

```
mkdir src
```

5. Mover `index.js` dentro de `src`

```
mv index.js src/
```

6. Copiar `notas.txt` con otro nombre

```
cp notas.txt notas_copia.txt
```

7. Borrar `config.json`

```
rm config.json
```

### Ejercicio reto

Crear las siguientes carpetas **con un solo comando**:

```
app  
app/controllers  
app/models  
app/views  
app/assets/images  
app/assets/css
```

Pista: usa `mkdir -p`.

## 1.4. Lectura y edición de archivos (20 min)

### Ejercicio guiado

1. Crear archivo de texto:

```
echo "Hola terminal" > saludo.txt
```

2. Mostrar su contenido con `cat saludo.txt`.

3. Mostrar solo las primeras 2 líneas de otro archivo:

```
head -n 2 archivo
```

4. Abrir un archivo en nano y escribir un texto.

5. Guardarlo (`Ctrl + O`) y salir (`Ctrl + X`).

6. Ver la historia de comandos con `history`.

7. Buscar en `man` qué significa el parámetro `I` de `ls`.

### Ejercicio reto

Buscar en `man` qué hace esta opción:

```
rm -i
```

Escribir su explicación con tus palabras.



## 2. Git desde la terminal (1h 45 min)

### 2.1. Qué es Git (10 min)



#### Ejercicio conceptual

Piensa en un ejemplo del mundo real donde te gustaría tener **historial de cambios** (versiones).

Escribe cómo Git resolvería ese problema.

### 2.2. Configuración inicial (15 min)



#### Ejercicio guiado

Ejecuta los siguientes comandos configurando tus datos reales:

```
git config --global user.name "Tu Nombre"  
git config --global user.email "tu@email.com"
```

Comprueba tu configuración:

```
git config --global --list
```



### Ejercicio reto

Busca la diferencia entre `HTTPS` y `SSH` y escríbelas en una frase simple.

## 2.3. Crear un repo y hacer el primer commit (20 min)

### Ejercicio guiado

1. Crear carpeta de proyecto:

```
mkdir mi_repo  
cd mi_repo
```

2. Inicializar repo

```
git init
```

3. Crear archivo `readme.txt`.

4. Ver estado del repo

```
git status
```

5. Agregar archivo al staging

```
git add readme.txt
```

6. Hacer commit

```
git commit -m "Primer commit"
```

### Ejercicio reto

Modifica el archivo y crea **2 commits más**, cada uno con un mensaje diferente.

## 2.4. Subir el repo a GitHub (15 min)

### Ejercicio guiado

1. Crear repo vacío en GitHub.
2. Conectar remoto:

```
git remote add origin <url>
```

3. Subir cambios:

```
git push -u origin main
```

### Ejercicio reto

Cambia el nombre de la rama `master` a `main` (si aplica).

## 2.5. Clonar y sincronizar (20 min)

### Ejercicio guiado

1. Clonar repo del instructor:

```
git clone <url>
```

2. Hacer un cambio y subirlo:

```
git add .
git commit -m "cambio"
git push
```

3. Traer cambios nuevos del instructor:

```
git pull
```

### Ejercicio colaborativo

En parejas:

- Uno crea y sube un archivo nuevo.
- El otro hace `git pull` y lo detecta.

## 2.6. Ramas y merge (20 min)

### Ejercicio guiado

1. Crear una rama:

```
git branch nueva
```

2. Cambiar a ella:

```
git checkout nueva
```

3. Modificar archivo.

4. Hacer commit en esta rama.

5. Volver a la rama principal y hacer merge:

```
git merge nueva
```

### Ejercicio reto: crear un conflicto

1. En `main`, modificar la línea 2 de un archivo.
2. En `nueva`, modificar la misma línea de forma diferente.
3. Hacer merge y ver el conflicto.
4. Resolverlo editando el archivo manualmente.

---

## 2.7. Comandos adicionales (10 min)

### Ejercicio guiado

Practica estos comandos dentro del repo:

```
git log  
git diff  
git stash  
git rm archivo  
git mv archivo1 archivo2
```

### Ejercicio reto

Usa `git stash` para guardar cambios, crear un commit limpio y luego aplicar los cambios guardados.