

💻 Git & GitHub Cheatsheet – Working with Remotes

Curso: Introduction to Git and GitHub

Módulo: Working with Remotes

Tema: Trabajando con repositorios remotos

[Introducción]

Un **repositorio remoto** es una versión del proyecto almacenada en línea (ej. GitHub) que permite colaboración y backup.

Conceptos clave:

- **origin** → nombre por defecto del repositorio remoto
- **remote** → alias para un repositorio remoto
- **push** → enviar cambios locales al remoto
- **pull** → traer cambios del remoto a local
- **fetch** → traer cambios del remoto sin fusionarlos

◆ Configurar y Ver Repositorios Remotos

```
git remote add origin https://github.com/usuario/repositorio.git  
git remote -v          # Listar remotos
```

- **origin** es el nombre estándar del remoto
- Puedes tener múltiples remotos con distintos nombres

◆ Traer Cambios del Remoto

```
git fetch origin      # Traer cambios sin fusionar  
git pull origin main # Traer y fusionar cambios
```

- **fetch** → solo descarga datos, no cambia tu rama local
- **pull** → descarga y fusiona automáticamente

◆ Enviar Cambios al Remoto

```
git push origin main    # Subir cambios a la rama principal  
git push -u origin feature1 # Subir nueva rama y establecer upstream
```

- `-u` → establece upstream, para poder hacer solo `git push` luego
-

◆ Trabajar con Ramas Remotas

```
git branch -a          # Ver ramas locales y remotas
git checkout -b rama feature1 origin/feature1 # Crear rama local a partir de
remoto
git push origin rama      # Subir rama local al remoto
```

- Mantener ramas sincronizadas previene conflictos
-

◆ Buenas Prácticas

- Sincronizar repositorio remoto antes de iniciar cambios
 - Evitar trabajar directamente en `main`
 - Usar nombres claros para ramas: `feature/login`, `bugfix/header`
 - Revisar estado con `git status` antes de `push` o `pull`
-

📎 Referencias oficiales

- 🌐 [GitHub Docs – About remote repositories](#)
- 🌐 [Pro Git Book – Working with Remotes](#)

```markdown