

Guía de Uso de Git

Este manual proporciona una introducción práctica a los comandos más comunes y útiles de Git.

i Nota: Para usar Git necesitas tenerlo instalado en tu sistema. Puedes descargarlo desde git-scm.com.

Tabla de Contenidos

- [Configuración Inicial](#)
- [Comandos Básicos](#)
- [Trabajo con Ramas](#)
- [Gestión de Forks y Pull Requests](#)
- [Resolución de Conflictos](#)
- [Otros Datos de Interés](#)

Configuración Inicial

Antes de comenzar, configura tu identidad en Git:

```
# Configurar nombre de usuario
$ git config --global user.name "Tu Nombre"

# Configurar correo electrónico
$ git config --global user.email "tuemail@example.com"
```

☒ **Tip:** Usa `git config --list` para verificar la configuración.

Comandos Básicos

Inicializar un Repositorio

```
# Crear un repositorio en un directorio local
$ git init
```

Clonar un Repositorio

```
# Descargar un repositorio existente
$ git clone <URL-del-repositorio>
```

Comprobar el Estado

```
# Ver cambios pendientes
$ git status
```

Añadir Cambios

```
# Añadir un archivo específico
$ git add archivo.txt

# Añadir todos los cambios
$ git add .
```

Confirmar Cambios

```
# Crear un commit con un mensaje descriptivo
$ git commit -m "Descripción del cambio"
```

Ver el Historial

```
# Mostrar todos los commits del repositorio
$ git log
```

Subir Cambios al Repositorio Remoto

```
# Subir cambios a la rama principal del remoto
$ git push origin main
```

Descargar Cambios del Repositorio Remoto

```
# Descargar la última versión del repositorio remoto
$ git pull origin main
```

Trabajo con Ramas

Crear una Nueva Rama

```
# Crear y moverse a una nueva rama
$ git checkout -b nombre-de-la-rama
```

Cambiar de Rama

```
# Cambiar a una rama existente
$ git checkout nombre-de-la-rama
```

Combinar Ramas (Merge)

```
# Combinar una rama en la rama actual
$ git merge nombre-de-la-rama
```

Eliminar una Rama

```
# Eliminar una rama local
$ git branch -d nombre-de-la-rama
```

Gestión de Forks y Pull Requests

Hacer un Fork

1. Ve al repositorio que deseas bifurcar en GitHub.
2. Haz clic en el botón **Fork** para crear una copia en tu cuenta.

Clonar tu Fork

```
# Clonar el fork en tu máquina local
$ git clone <URL-de-tu-fork>
```

Sincronizar Cambios con el Original

```
# Agregar el repositorio original como remoto "upstream"
$ git remote add upstream <URL-del-repositorio-original>

# Descargar cambios del upstream
$ git fetch upstream

# Combinar cambios del upstream en tu rama local
$ git merge upstream/main
```

Crear un Pull Request

1. Sube tus cambios a tu fork:

```
$ git push origin nombre-de-la-rama
```

2. Ve a la página de tu fork y haz clic en **New Pull Request**.
3. Describe tus cambios y envía la solicitud.

Resolución de Conflictos

Los conflictos ocurren cuando Git no puede combinar automáticamente los cambios. Sigue estos pasos para resolverlos:

1. Identifica los conflictos:

```
$ git status
```

2. Edita los archivos en conflicto. Busca marcas como estas:

```
<<<<<< HEAD
Tu cambio
=====
Cambio en el remoto
>>>>>> upstream/main
```

3. Elimina las marcas y combina los cambios manualmente.
4. Añade los cambios resueltos:

```
$ git add archivo.txt
```

5. Crea un nuevo commit:

```
$ git commit -m "Resolver conflictos en archivo.txt"
```

Otros Datos de Interés

Aquí tienes recursos adicionales para aprender y optimizar tu uso de Git:

- [Learn Git Branching](#): Plataforma interactiva para aprender sobre branching y otros conceptos fundamentales de Git.
- [GitHub Skills](#): Curso interactivo oficial de GitHub que cubre desde lo básico hasta temas avanzados.

- [Cheat Sheet de Git](#): Resumen con los comandos más utilizados para tener como referencia rápida.

💡 **Tip:** Dedica tiempo a explorar estos recursos para profundizar en el uso de Git y mejorar tus habilidades.
