

El archivo **.gitignore** es una herramienta que utiliza Git para definir qué archivos o carpetas no deben ser rastreados por el control de versiones. Es fundamental para evitar que archivos innecesarios, temporales o sensibles se suban a un repositorio, como archivos de configuración locales, archivos temporales o binarios generados automáticamente.

¿Cómo funciona el archivo .gitignore?

1. Ubicación del archivo:

El archivo .gitignore se coloca normalmente en la raíz del repositorio Git. Sin embargo, puedes tener múltiples archivos .gitignore en distintos subdirectorios si lo necesitas.

2. Sintaxis básica del archivo .gitignore:

El archivo .gitignore contiene una lista de patrones que le dicen a Git qué archivos o carpetas debe ignorar. Aquí algunos ejemplos de las reglas más comunes:

- Ignorar un archivo específico:

```
bash
```

```
archivo.txt
```

- Ignorar una carpeta completa:

```
bash
```

```
/carpeta/
```

- Ignorar todos los archivos con una extensión específica, como archivos .log:

```
bash
```

```
*.log
```

- Ignorar todos los archivos dentro de una carpeta, pero no la carpeta en sí:

```
bash
```

```
carpeta/*
```

- Ignorar archivos temporales generados por el sistema o un editor:
 - Ignorar archivos de macOS:

bash

.DS_Store

- Ignorar archivos de backup generados por editores de texto (ej. file~):

bash

*~

- Ignorar un archivo oculto como .env:

bash

.env

3. Comentarios:

Puedes añadir comentarios en el archivo .gitignore utilizando el carácter # al inicio de la línea:

bash

Ignorar archivos log

*.log

4. Excluir archivos previamente ignorados:

Si quieres que ciertos archivos no sean ignorados aunque coincidan con un patrón general en .gitignore, puedes usar ! para forzar la inclusión:

bash

Ignorar todos los archivos .log

*.log

Pero no ignorar log importante

!important.log

Ejemplos comunes de .gitignore

1. Proyectos Node.js (con archivos generados por npm):

bash

Dependencias de Node

node_modules/

Archivos de log

npm-debug.log*

2. **Proyectos en Python:**

bash

Ignorar archivos compilados de Python

__pycache__/

*.py[cod]

*.pyo

Ignorar entornos virtuales

venv/

.venv/

3. **Proyectos en Java:**

bash

Ignorar compilados de Java

*.class

Ignorar archivos generados por IDEs como Eclipse

.classpath

.project

4. **Proyectos en C/C++:**

bash

Ignorar binarios y objetos

*.out

*.o

*.exe

Ignorar carpetas de compilación

/build/

5. Ignorar archivos específicos del sistema operativo:

- o **macOS:**

bash

.DS_Store

- o **Windows:**

bash

Thumbs.db

- o **Linux:**

bash

.Trash-*

Buenas prácticas con .gitignore:

- **Añadirlo desde el principio:** Es recomendable configurar el archivo .gitignore antes de hacer tu primer commit para evitar agregar archivos innecesarios al repositorio.
- **Ignorar archivos sensibles:** No debes versionar archivos que contengan información sensible como contraseñas, claves API, o configuraciones locales. Por ejemplo, archivos como .env o config.json.
- **Mantenerlo actualizado:** A medida que tu proyecto crezca o cambie, asegúrate de actualizar el .gitignore para reflejar nuevos archivos generados o dependencias.

Ejemplo completo de .gitignore para un proyecto típico en Node.js

bash

Node.js

node_modules/

Logs

logs/

*.log

npm-debug.log*

Entorno de desarrollo

.env

Directorios de compilación

/dist/

/build/

Archivos generados por el sistema

.DS_Store

Thumbs.db

Ignorar binarios

*.exe

*.out

*.o

¿Cómo ignorar archivos ya rastreados?

Si ya añadiste un archivo al repositorio y luego lo incluyes en .gitignore, debes eliminarlo del control de versiones (sin borrarlo de tu sistema local) con este comando:

bash

git rm --cached archivo.txt

Esto elimina el archivo del repositorio, pero mantiene una copia local.

Verificar qué archivos están siendo ignorados

Puedes utilizar el comando `git check-ignore` para verificar si un archivo está siendo ignorado por `.gitignore`:

bash

```
git check-ignore -v archivo.txt
```

Esto te mostrará la línea exacta de `.gitignore` que está causando que el archivo sea ignorado.