

# Borrar archivos

## Borrar archivos duplicados

Digamos que en una carpeta hay varios archivos duplicados, en mi caso estos archivos fueron creados con "(1)" al final del nombre. Y como es comun, eso siga una secuencia.

Se puede usar el comando `find` combinado con `rm` para eliminar archivos que contengan "(1)" en su nombre. Aquí está el comando para eliminar esos archivos:

```
find /home/user/directorio-con-archivos -type f -name "*\ (1\)*" -exec rm {} \;
```

- `find /home/user/directorio-con-archivos`: Busca en el directorio especificado.
- `-type f`: Asegura que solo se encuentren archivos (no directorios).
- `-name "*\ (1\)*"`: Busca archivos que tengan "(1)" en su nombre. Los paréntesis se escapan usando `\`.
- `-exec rm {} \;`: Ejecuta el comando `rm` (eliminar) en cada archivo encontrado (`{}` representa el archivo encontrado).

## Borrar archivos excepto...

Para eliminar todo excepto el archivo ejemplo.txt en un directorio, puede usar el siguiente comando en cualquier sistema Linux:

```
find /$path/to/files/ ! -name 'ejemplo.txt' -type f -delete
```

### Explicación:

- `find`: comando usado para buscar archivos y directorios.
- `/$path/to/files/`: la ruta donde se aplica la búsqueda.
- `! -name 'ejemplo.txt'`: excluye el archivo llamado ejemplo.txt.
- `-type f`: solo afecta archivos (no directorios).
- `-delete`: elimina los elementos encontrados.

# Borrar archivos subdirectorios y su contenido

Para eliminar todo (archivos y carpetas) excepto gitlab.rb, puede usar:

```
find /etc/gitlab/ ! -name 'ejemplo.txt' ! -path /$path/to/files/ -delete
```

⚠ **Tenga mucho cuidado** al usar comandos con -delete. Siempre es recomendable hacer primero una prueba sin eliminar nada, para revisar qué se va a borrar:

```
find /$path/to/files/ ! -name 'ejemplo.txt' -type f
```

---

Revisión #2

Creado 6 septiembre 2024 12:21:50 por Greivin

Actualizado 23 abril 2025 15:16:56 por Greivin