

Setup Your LLC

Search To Start LLC Business. Start Your Business The Right Way.

Yahoo! Search

Open

Pro Git, el libro oficial de Git

1.5. Configurando Git por primera vez

Ahora que tienes Git en tu sistema, querrás hacer algunas cosas para personalizar tu entorno de Git. Deberías tener que hacer estas cosas sólo una vez; se mantendrán entre actualizaciones. También puedes cambiarlas en cualquier momento volviendo a ejecutar los comandos correspondientes.

Git trae una herramienta llamada `git config` que te permite obtener y establecer variables de configuración que controlan el aspecto y funcionamiento de Git. Estas variables pueden almacenarse en tres sitios distintos:

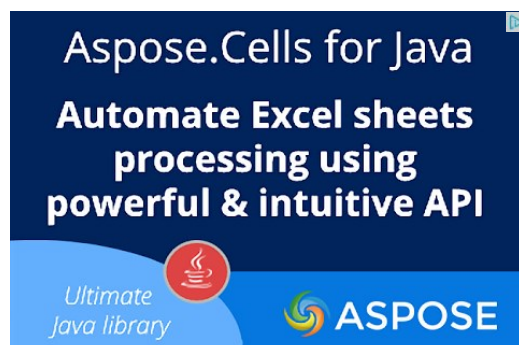
- Archivo `/etc/gitconfig`: Contiene valores para todos los usuarios del sistema y todos sus repositorios. Si pasas la opción `--system` a `git config`, lee y escribe específicamente en este archivo.
- Archivo `~/.gitconfig` file: Específico a tu usuario. Puedes hacer que Git lea y escriba específicamente en este archivo pasando la opción `--global`.
- Archivo config en el directorio de git (es decir, `.git/config`) del repositorio que estás utilizando actualmente: Específico a ese repositorio. Cada nivel sobrescribe los valores del nivel anterior, por lo que los valores de `.git/config` tienen preferencia sobre los de `/etc/gitconfig`.

En sistemas Windows, Git busca el archivo `.gitconfig` en el directorio `$HOME` (`C:\Documents and Settings\%USER%` para la mayoría de usuarios). También busca en el directorio `/etc/gitconfig`, aunque esta ruta es relativa a la raíz MSys, que es donde quiera que decidieses instalar Git en tu sistema Windows cuando ejecutaste el instalador.

1.5.1. Tu identidad

Lo primero que deberías hacer cuando instalas Git es establecer tu nombre de usuario y dirección de correo electrónico. Esto es importante porque las confirmaciones de cambios (*commits*) en Git usan esta información, y es introducida de manera inmutable en los commits que envías:

```
$ git config --global user.name "John Doe"
$ git config --global user.email johndoe@example.com
```



De nuevo, sólo necesitas hacer esto una vez si especificas la opción `--global`, ya que Git siempre usará esta información para todo lo que hagas en ese sistema. Si quieres sobrescribir esta información con otro nombre o dirección de correo para proyectos específicos, puedes ejecutar el comando sin la opción `--global` cuando estés en ese proyecto.

1.5.2. Tu editor



Indice de contenidos

Capítulo 1. Empezando

- 1.1. Acerca del control de versiones
- 1.2. Una breve historia de Git
- 1.3. Fundamentos de Git
- 1.4. Instalando Git
- 1.5. Configurando Git por primera vez**
- 1.6. Obteniendo ayuda
- 1.7. Resumen

2. Fundamentos de Git

3. Trabajando con ramas en Git

4. Git en un servidor

5. Git en entornos distribuidos

6. Las herramientas de Git

7. Personalizando Git

8. Git y otros sistemas

9. Funcionamiento interno de Git

Ahora que tu identidad está configurada, puedes elegir el editor de texto por defecto que se utilizará cuando Git necesite que introduzcas un mensaje. Si no indicas nada, Git usa el editor por defecto de tu sistema, que generalmente es Vi o Vim. Si quieres usar otro editor de texto, como Emacs, puedes hacer lo siguiente:

```
$ git config --global core.editor emacs
```

1.5.3. Tu herramienta de diferencias

Otra opción útil que puede que quieras configurar es la herramienta de diferencias por defecto, usada para resolver conflictos de unión (*merge*). Digamos que quieres usar `vimdiff`:

```
$ git config --global merge.tool vimdiff
```

Git acepta `kdiff3`, `tkdiff`, `meld`, `xxdiff`, `emerge`, `vimdiff`, `gvimdiff`, `ecmerge`, y `opendiff` como herramientas válidas. También puedes configurar la herramienta que tú quieras; véase el [Capítulo 7](#) para más información sobre cómo hacerlo.

1.5.4. Comprobando tu configuración

Si quieres comprobar tu configuración, puedes usar el comando `git config --list` para listar todas las propiedades que Git ha podido encontrar en ese punto:

```
$ git config --list
user.name=Scott Chacon
user.email=schacon@gmail.com
color.status=auto
color.branch=auto
color.interactive=auto
color.diff=auto
...
```

Puede que veas claves repetidas, porque Git lee la misma clave de distintos archivos (`/etc/gitconfig` y `~/.gitconfig`, por ejemplo). En ese caso, Git usa el último valor para cada clave única que ve.

También puedes comprobar qué valor cree Git que tiene una clave específica ejecutando `git config {clave}`:

```
$ git config user.name
Scott Chacon
```

Anterior

1.4. Instalando Git

Siguiente

1.6. Obteniendo ayuda