

# Configuração Inicial do Git

Agora que você tem o Git em seu sistema, você deve fazer algumas coisas para personalizar o ambiente Git. Você fará isso apenas uma vez por computador e o efeito se manterá após atualizações. Você também pode mudá-las em qualquer momento rodando esses comandos novamente.

O Git vem com uma ferramenta chamada `git config` que permite ver e atribuir variáveis de configuração que controlam todos os aspectos de como o Git aparece e opera. Estas variáveis podem ser armazenadas em três lugares diferentes:

1. `/etc/gitconfig`: válido para todos os usuários no sistema e todos os seus repositórios. Se você passar a opção `--system` para `git config`, ele lê e escreve neste arquivo.
2. `~/.gitconfig` ou `~/.config/git/config`: Somente para o seu usuário. Você pode fazer o Git ler e escrever neste arquivo passando a opção `--global`.
3. `config` no diretório Git (ou seja, `.git/config`) de qualquer repositório que você esteja usando: específico para este repositório.

Cada nível sobrescreve os valores no nível anterior, ou seja, valores em `.git/config` prevalecem sobre `/etc/gitconfig`.

No Windows, Git procura pelo arquivo `.gitconfig` no diretório `$HOME` (`C:\Users\%USER` para a maioria). Ele também procura por `/etc/gitconfig`, mesmo sendo relativo à raiz do sistema, que é onde quer que você tenha instalado Git no seu sistema.

## Sua Identidade

A primeira coisa que você deve fazer ao instalar Git é configurar seu nome de usuário e endereço de e-mail. Isto é importante porque cada *commit* usa esta informação, e ela é carimbada de forma imutável nos *commits* que você começa a criar:

```
$ git config --global user.name "Fulano de Tal"
$ git config --global user.email fulanodetal@exemplo.br
```

Reiterando, você precisará fazer isso somente uma vez se tiver usado a opção `--global`, porque então o Git usará esta informação para qualquer coisa que você fizer naquele sistema. Se você quiser substituir essa informação com nome diferente para um projeto específico, você pode rodar o comando sem a opção `--global` dentro daquele projeto.

Muitas ferramentas GUI o ajudarão com isso quando forem usadas pela primeira vez.

## Seu Editor

Agora que a sua identidade está configurada, você pode escolher o editor de texto padrão que será chamado quando Git precisar que você entre uma mensagem. Se não for configurado, o Git usará o editor padrão, que normalmente é o Vim. Se você quiser usar um editor de texto diferente, como o Emacs, você pode fazer o seguinte:

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

Vim e Emacs são editores de texto populares comumente usados por desenvolvedores em sistemas baseados em Unix como Linux e Max. Se você não for acostumado com estes editores ou estiver Warning em um sistema Windows, você precisará procurar por instruções de como configurar o seu editor preferido com Git. Se você não configurar o seu editor preferido e não sabe usar o Vim ou Emacs, é provável que você fique bastante confuso ao entrar neles.

## Testando Suas Configurações

Se você quiser testar as suas configurações, você pode usar o comando `git config --list` para listar todas as configurações que o Git conseguir encontrar naquele momento:

```
$ git config --list
user.name=Fulano de Tal
user.email=fulanodetal@exemplo.br
color.status=auto
color.branch=auto
color.interactive=auto
color.diff=auto
...
```

Pode ser que algumas palavras chave apareçam mais de uma vez, porque Git lê as mesmas chaves de arquivos diferentes (`/etc/gitconfig` e `~/.gitconfig`, por exemplo). Neste caso, Git usa o último valor para cada chave única que ele vê.

Você pode também testar o que Git tem em uma chave específica digitando `git config <key>`:

```
$ git config user.name
Fulano de Tal
```