Git - Environment Setup

Before you can use Git, you have to install and do some basic configuration changes. Below are the steps to install Git client on Ubuntu and Centos Linux.

Installation of Git Client

If you are using Debian base GNU/Linux distribution, then **apt-get** command will do the needful.

```
[ubuntu ~]$ sudo apt-get install git-core
[sudo] password for ubuntu:

[ubuntu ~]$ git --version
git version 1.8.1.2
```

And if you are using RPM based GNU/Linux distribution, then use **yum** command as given.

```
[CentOS ~]$
su -
Password:

[CentOS ~]# yum -y install git-core

[CentOS ~]# git --version
git version 1.7.1
```

Customize Git Environment

Git provides the git config tool, which allows you to set configuration variables. Git stores all global configurations in **.gitconfig** file, which is located in your home directory. To set these configuration values as global, add the **--global** option, and if you omit **--global** option, then your configurations are specific for the current Git repository.

You can also set up system wide configuration. Git stores these values in the **/etc/gitconfig** file, which contains the configuration for every user and repository on the system. To set these values, you must have the root rights and use the **--system** option.

When the above code is compiled and executed, it produces the following result −

Setting username

This information is used by Git for each commit.

```
[jerry@CentOS project]$ git config --global user.name "Jerry Mouse"
```

Setting email id

This information is used by Git for each commit.

```
[jerry@CentOS project]$ git config --global user.email "jerry@tutorialspoint.com"
```

Avoid merge commits for pulling

You pull the latest changes from a remote repository, and if these changes are divergent, then by default Git creates merge commits. We can avoid this via following settings.

```
jerry@CentOS project]$ git config --global branch.autosetuprebase always
```

Color highlighting

The following commands enable color highlighting for Git in the console.

```
[jerry@CentOS project]$ git config --global color.ui true

[jerry@CentOS project]$ git config --global color.status auto

[jerry@CentOS project]$ git config --global color.branch auto
```

Setting default editor

By default, Git uses the system default editor, which is taken from the VISUAL or EDITOR environment variable. We can configure a different one by using git config.

```
[jerry@CentOS project]$ git config --global core.editor vim
```

Setting default merge tool

Git does not provide a default merge tool for integrating conflicting changes into your working tree. We can set default merge tool by enabling following settings.

```
[jerry@CentOS project]$ git config --global merge.tool vimdiff
```

Listing Git settings

To verify your Git settings of the local repository, use **git config –list** command as given below.

```
[jerry@CentOS ~]$ git config --list
```

The above command will produce the following result.

```
user.name=Jerry Mouse
user.email=jerry@tutorialspoint.com
push.default=nothing
branch.autosetuprebase=always
color.ui=true
color.status=auto
color.branch=auto
core.editor=vim
merge.tool=vimdiff
```