

Git refresher

Important

Make sure that Git is installed on your machine. To verify, open a shell and type the command `git`. If this fails, follow the necessary steps to install Git.

Goal

Explain the essential steps to interact with version control tool `Git`.

Information



Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

- [Home of Git](#)
- [Installation of Git on Windows](#)
- [Documentation of Git](#)

Important commands:

| Description | Command |
|-----------------------------------|--|
| Start a Git repository | <code>git init</code> |
| Check the status of the repo | <code>git status</code> |
| Add all (changed) files to commit | <code>git add .</code> |
| Add a specific file to commit | <code>git add myfile.py</code> |
| Commit to Git | <code>git commit -m 'My commit message'</code> |
| Push to Git | <code>git push origin branch</code> |
| Pull from Git | <code>git pull origin branch</code> |
| Create branch | <code>git checkout -b my-branch</code> |
| Clone repository | <code>git clone myrepo.git</code> |

Example

The following example shows how to clone a repository, create a new branch, add a file to it and commit it back to the repository.

Clone the repository:

```
$ git clone https://github.com/bhuwanKarki/tutorials.git
```

Navigate into the folder:

```
$ cd tutorials
```

Check the file status:

```
tutorials(master) $ git status
```

Create a new branch:

```
tutorials (master) $ git checkout -b my-branch
```

Write some string to a text file:

```
tutorials(my-branch) $ echo "testfile" >> testfile.txt
```

Add the file to the repository:

```
tutorials (my-branch) $ git add testfile.txt
```

Check the file is added to the repo:

```
tutorials (my-branch) $ git status
```

Commit the file to Git:

```
tutorials (my-branch) $ git commit -m 'Added testfile'
```

Push the changes to the remote on BitBucket:

```
tutorials (my-branch) $ git push
```

Switch to the master branch:

```
tutorials(my-branch) $ git checkout master
```

Merge the branch we just created into master:

```
tutorials (master) $ git merge my-branch
```

Optional Git tools

There are also tools to work with Git using a GUI. Checkout the pages for installation instructions.

SourceTree



Sourcetree simplifies how you interact with your Git repositories so you can focus on coding. Visualize and manage your repositories through Sourcetree's simple Git GUI.

[Home of SourceTree](#)

SmartGit



SmartGit is a Git client with support for GitHub Pull Requests+Comments and SVN. It runs on Mac OS X, Windows and Linux.

[Home of SmartGit](#)

Recommended site to follow

<https://blog.teamtreehouse.com/git-for-designers-part-1>

<http://rogerdudler.github.io/git-guide/>

<https://wildyinaccurate.com/a-hackers-guide-to-git/#pushing> <https://alvinalexander.com/git/git-cheat-sheet-git-reference-commands/>