Git Cheat Sheet

|  |
| --- |
| **Concepts** |

Remember: $ git command --help

master : default development branch

origin : default upstream repository

|  |
| --- |
| **Config** |

Global Git configuration is stored in $HOME/.gitconfig

**$ git config --global --edit**

Open the global configuration file in a text editor for manual editing.

**$ git config --global user.name “Your Name”**

Set the name that will be attached to your commits and tags

**$ git config --global user.email “you@example.com”**

Set the email address that will be attached to your commits and tags

**$ git config --global color.ui auto**

Enable colorization of git output

|  |
| --- |
| **Basics** |

**Create from existing data**

cd ~/projects/myproject

git init

git add .

**Create from existing repo**

git clone ~/existing/repo ~/new/repo

git clone git://host.org/project.git

**$ git remote add <name> <url>**

Create a new connection to a remote repo. After adding a remote, you can use <name> as a shortcut for <url> in other commands.

**$ git add [file]**

Add a file to the staging area. Use . instead of full path to add all files from current directory

**$ git commit -m "<message>"**

Commit the staged snapshot, but instead of launching a text editor, use <message> as the commit message.

**$ git commit -a**

Commit all your local changes

**$ git commit -a --amend**

Fix the last commit, after editing broken files

**$ git push**

Push changes to origin

**$ git pull**

Pull latest changes from origin and merges changes

**$ git fetch**

Fetch latest changes from origin, but does not merge

**$ git status**

Files changed in working directory

**$ git diff**

Changes to tracked files

**$ git diff --staged**

Shows file differences between staging and the last file version

**$ git log**

History of changes

**$ git log -p $file $dir/ec/tory/**

History of changes for file with diffs

**$ get log --online --graph --decorate**

Overview of reference labels and history graph. One commit per line.

**$ git show [commit]**

Outputs metadata and content changes of the specific commit

|  |
| --- |
| **Advanced** |

**$ git rm [file]**

Deletes the file from the working directory and stages the deletion

**$ git mv [file-original] [file-renamed]**

Changes the file name and prepares it for commit

**$ git reset [file]**

Get file back from staging area to working directory

**$ git fetch --prune [remote]**

Remove remote refs that were removed from the remote repository

**$ git reset --hard**

Return to the last committed state (caution: cannot undo)