git config --global user.name "David"

git config --global user.email "david@gittesting"

git config --global color.ui true

git config --global color.status auto

git config --global color.branch auto

git config --global core.editor vim

git log --graph

git log –oneline

git hub public key adding and testing:

<https://modulesunraveled.com/very-basics-git/creating-account-github-and-setting-ssh-keys>

git undo

Ref: http://www.vogella.com/tutorials/Git/article.html

git commit –amend: will update and replace the most recent commit message with a new commit message.

* **switch** branches
  + **create** a new branch and switch to it
* **checkout** a file or directory as it were in another commit
  + **discard** all working copy changes since the last commit
  + **resolve** merge conflicts

git checkout – filename: **discard** all working copy changes since the last commit.

git revert: Using this command we can go back to previous commit.

(A->**B**->C) # Now let say we want to go back to previous commit B.

git revert –hard commit-id of B: # This will prompt for commit message for new commit.

Now this will become (A->**B**->C->**D**) and **D** is the new commit with exact snapshot of **B**.

git push –f origin master # This will publish the changes in the github.

git reset: Let us we want to delete the commit history from certain point of commit.

(A->B->C->D->E)

Now I want to go back to commit id B and remove other commit id’s (C, D, E).

git reset –hard commit-id of B: # This will reset back to commit id B and B will be the latest commit and C,D,E will be discarded.

git push –f origin master # This will publish the changes to github.

Note: If there is any merge commit it is not working check.