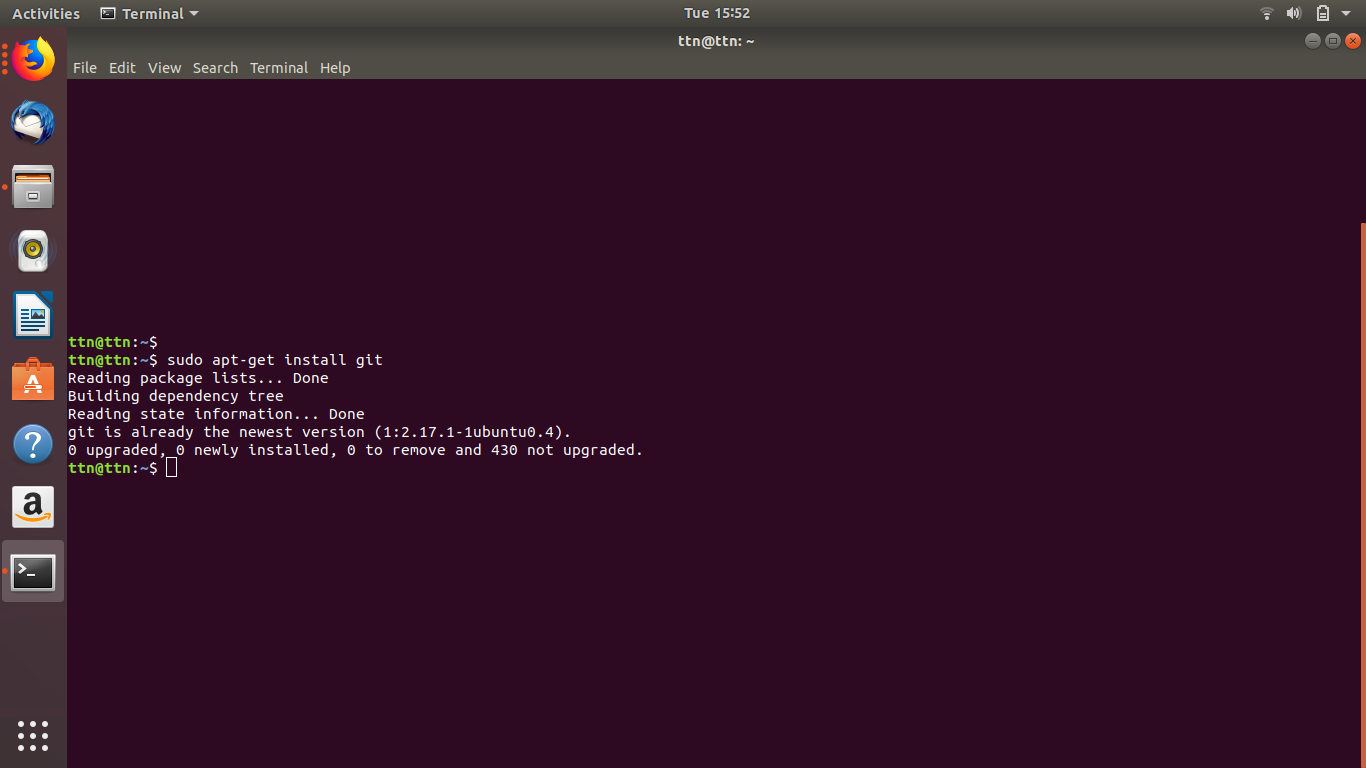
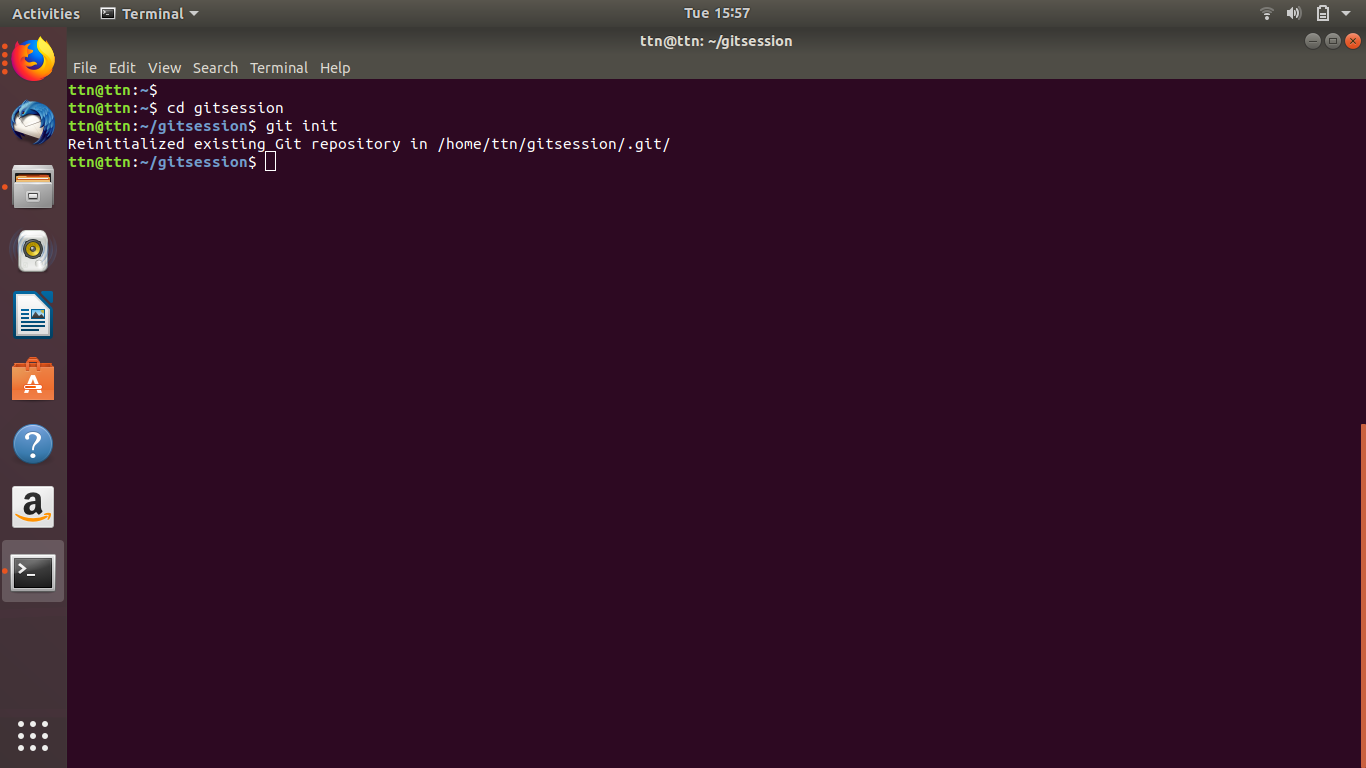
1.Git Setup<https://confluence.atlassian.com/bitbucket/set-up-git-744723531.html>

Cmd- sudo apt-get install git



2.Initialize a Git Repository

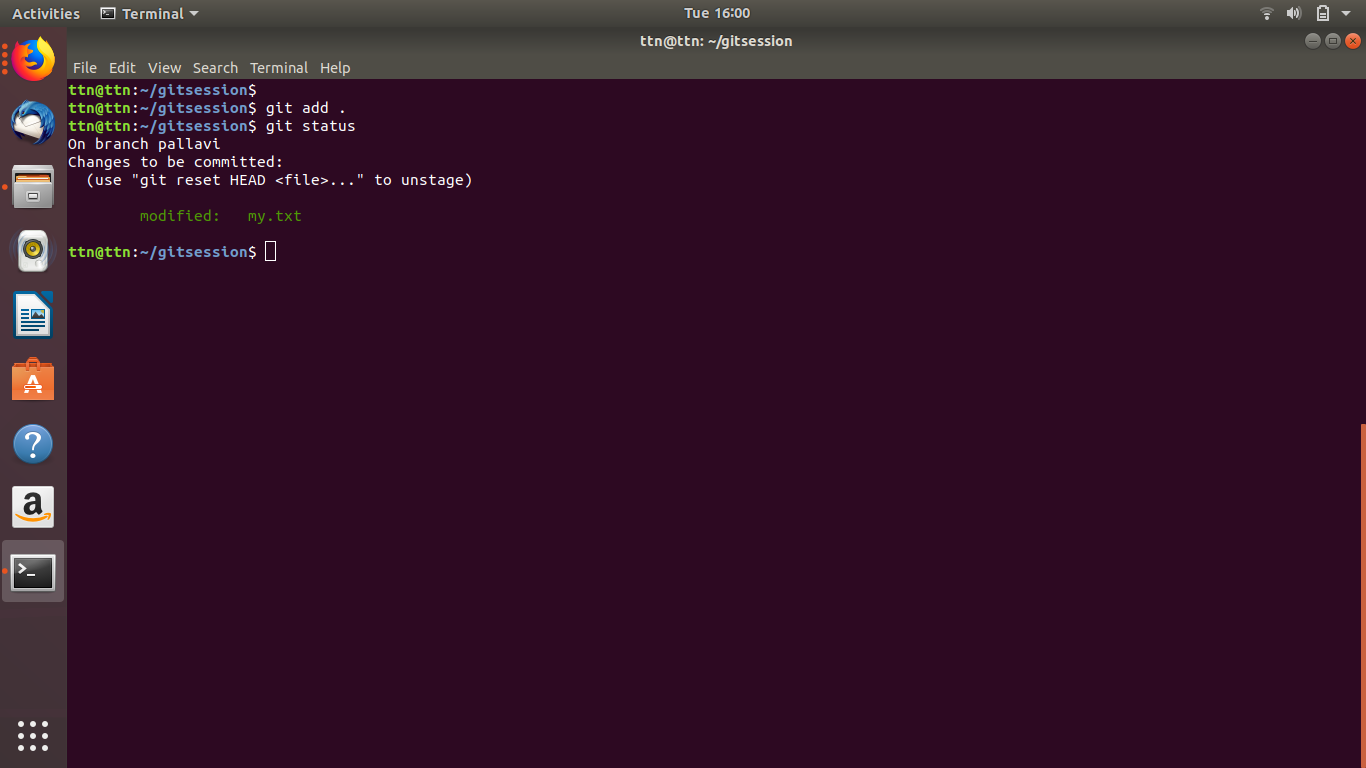
Cmd- git init



3.Add files to the repository

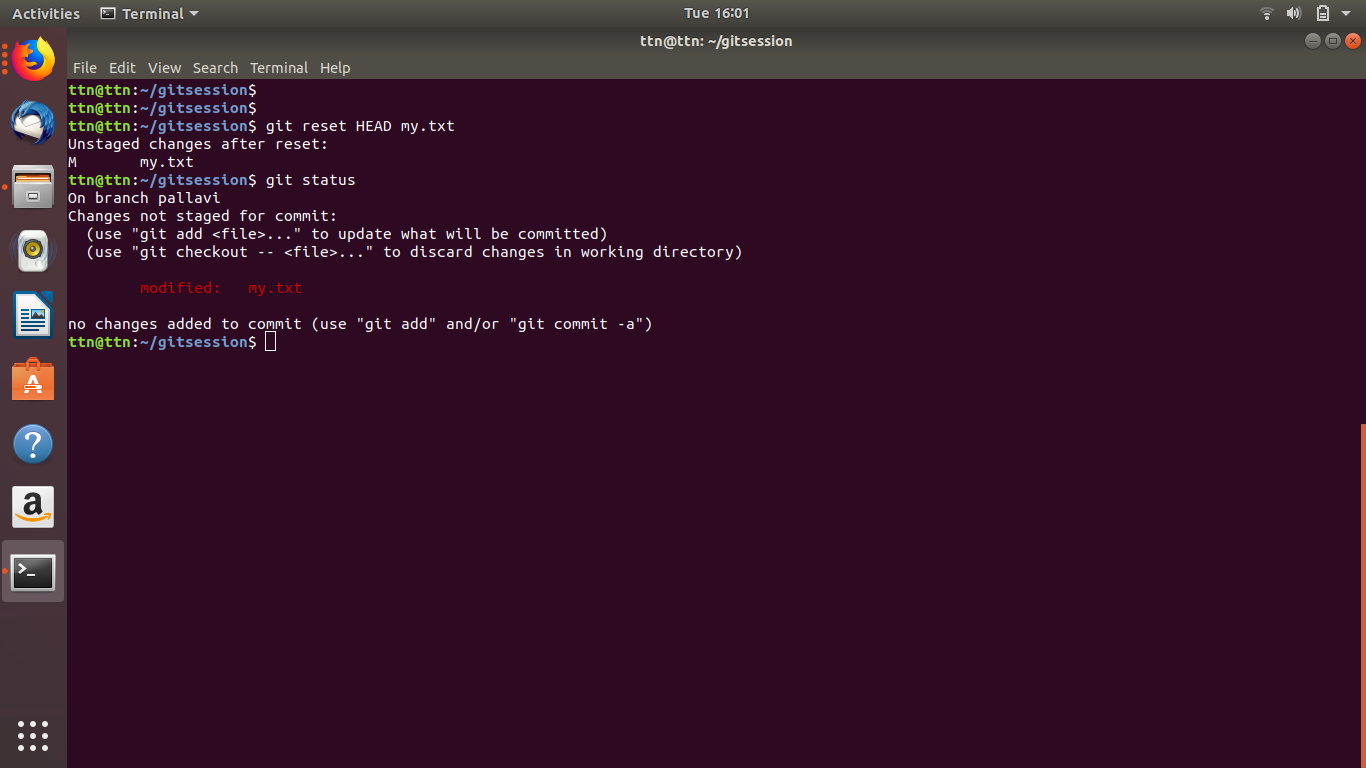
Cmd -vim my.txt

Git add .



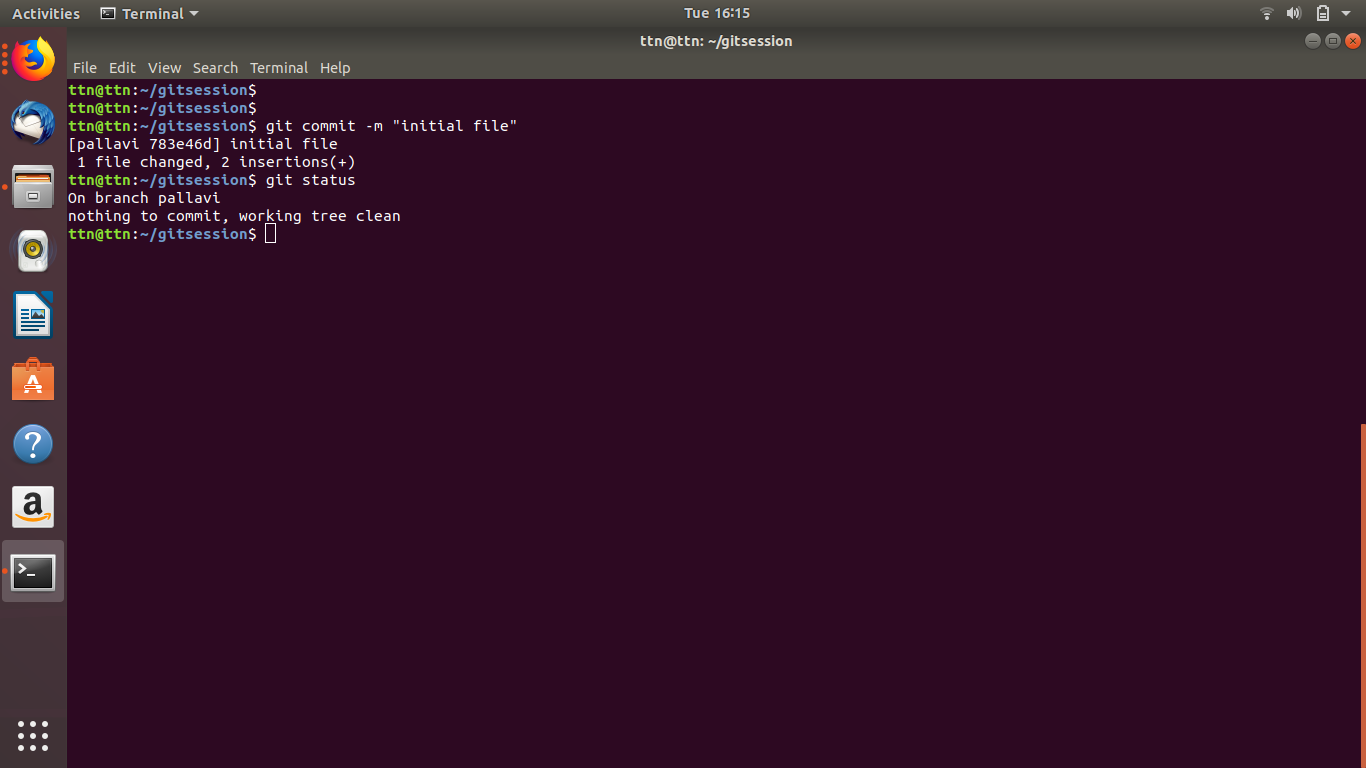
4.Unstage 1 file

Cmd - git reset HEAD my.txt



5.Commit the file

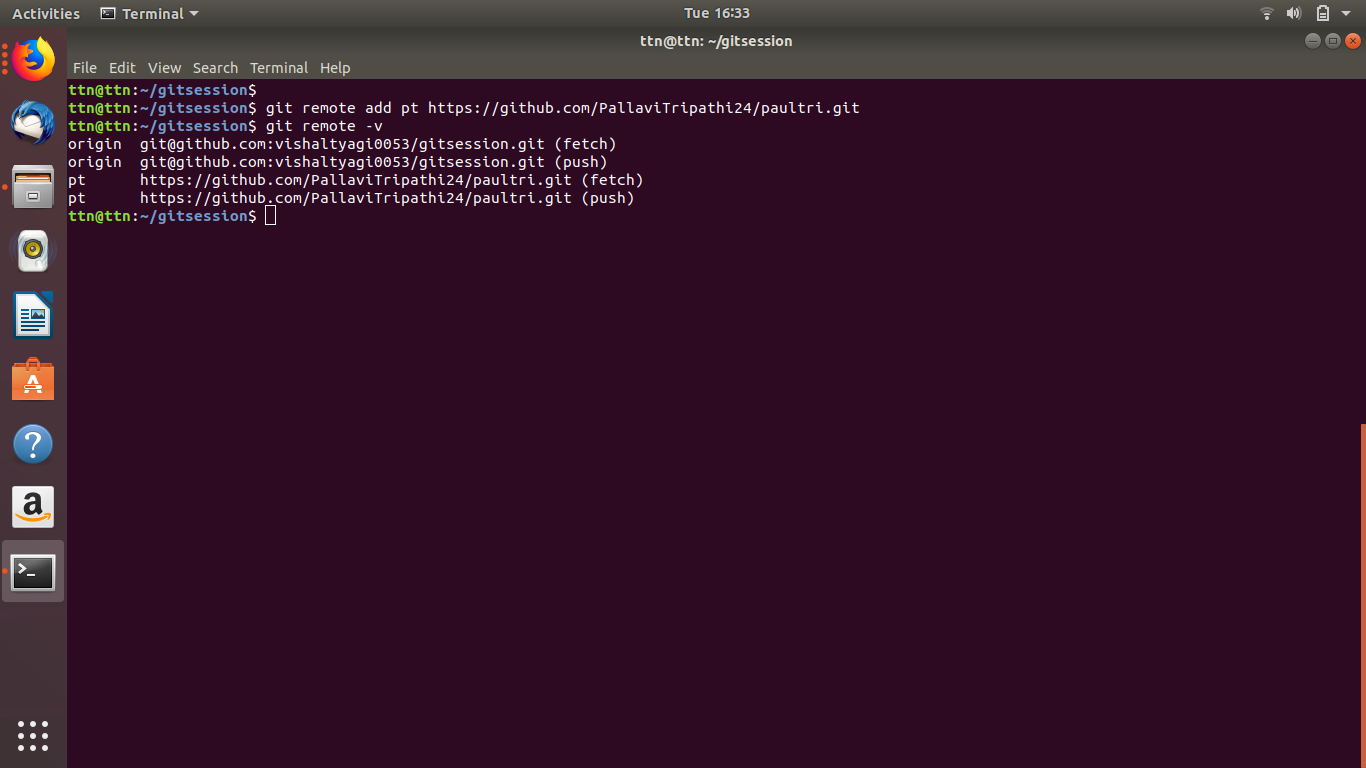
Cmd- git commit -m “first time commit”



6. Add a remote

Cmd- git remote add pt <url>

Git remote -v



7.Undo changes to a particular file

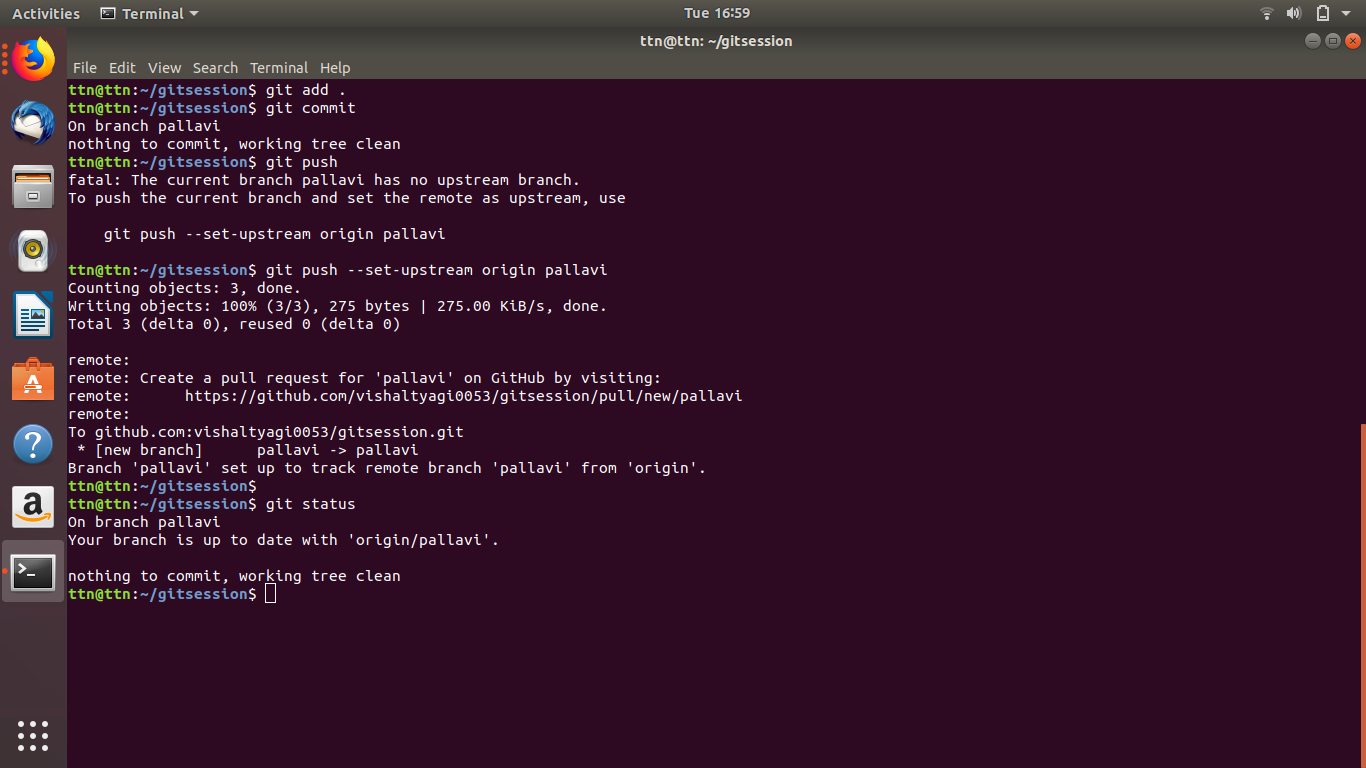
Cmd -

8.Push changes to Github

Cmd - git add .

git commit

Git push --set-upstream origin pallavi



9. Clone the repository

Cmd - git clone https://github.com/PallaviTripathi24/paultri.git

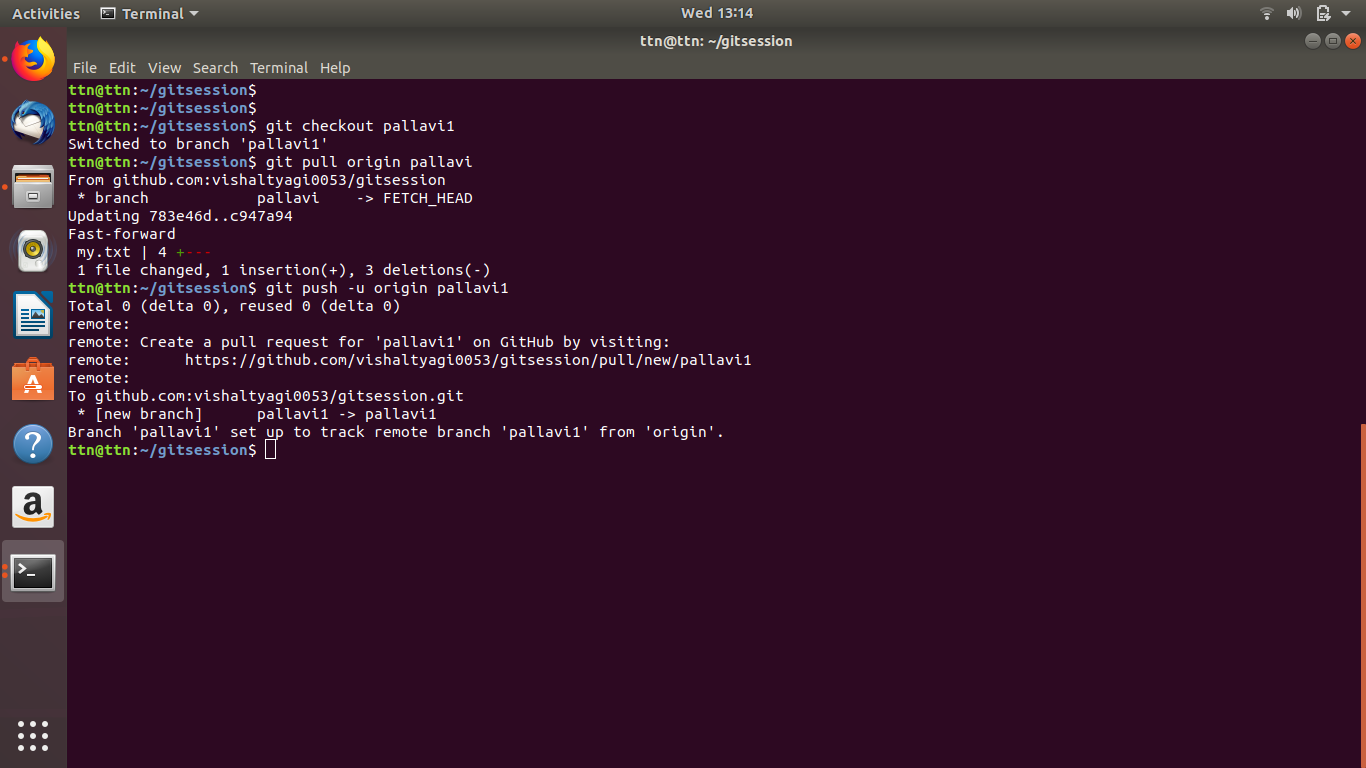
10.Add changes to one of the copies and pull the changes in the other.

Cmd- git checkout pallavi1

Git add .

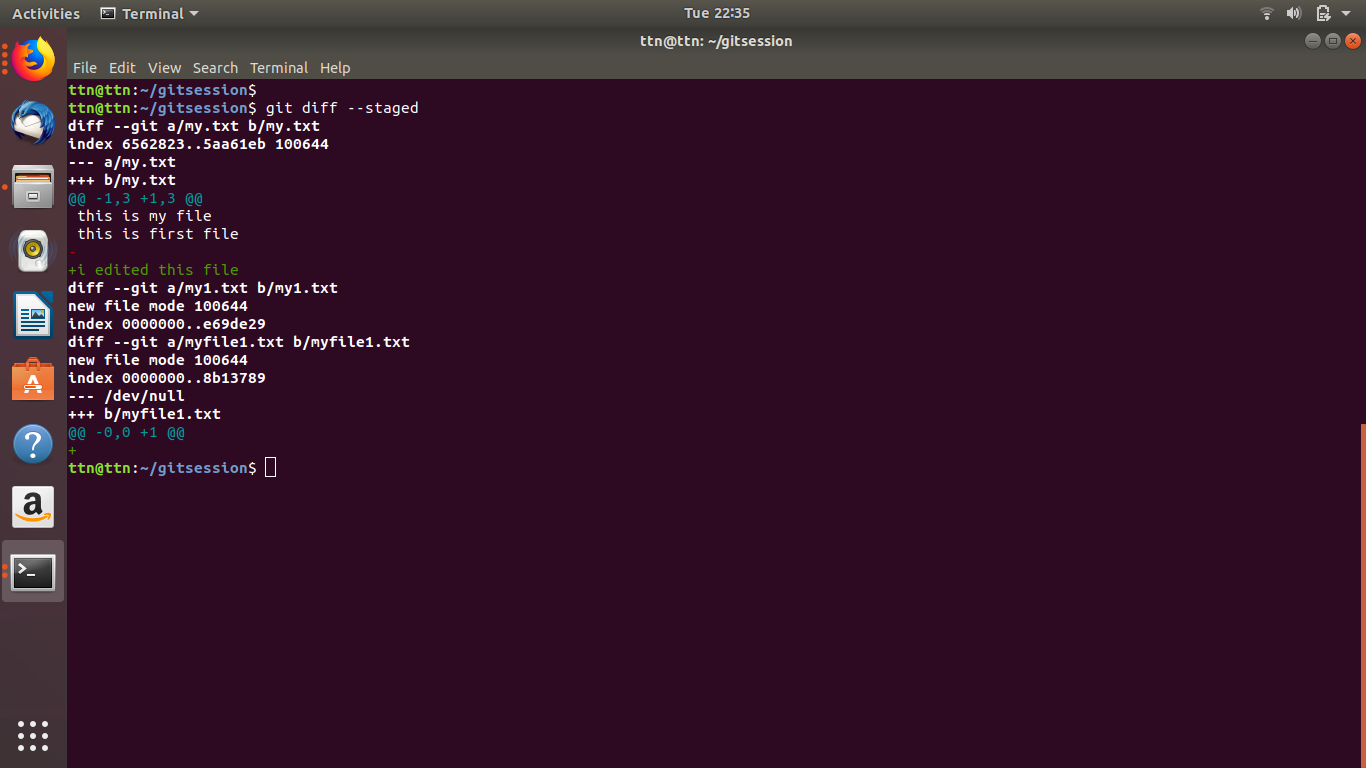
Git pull origin pallavi

Git push -u origin pallavi1



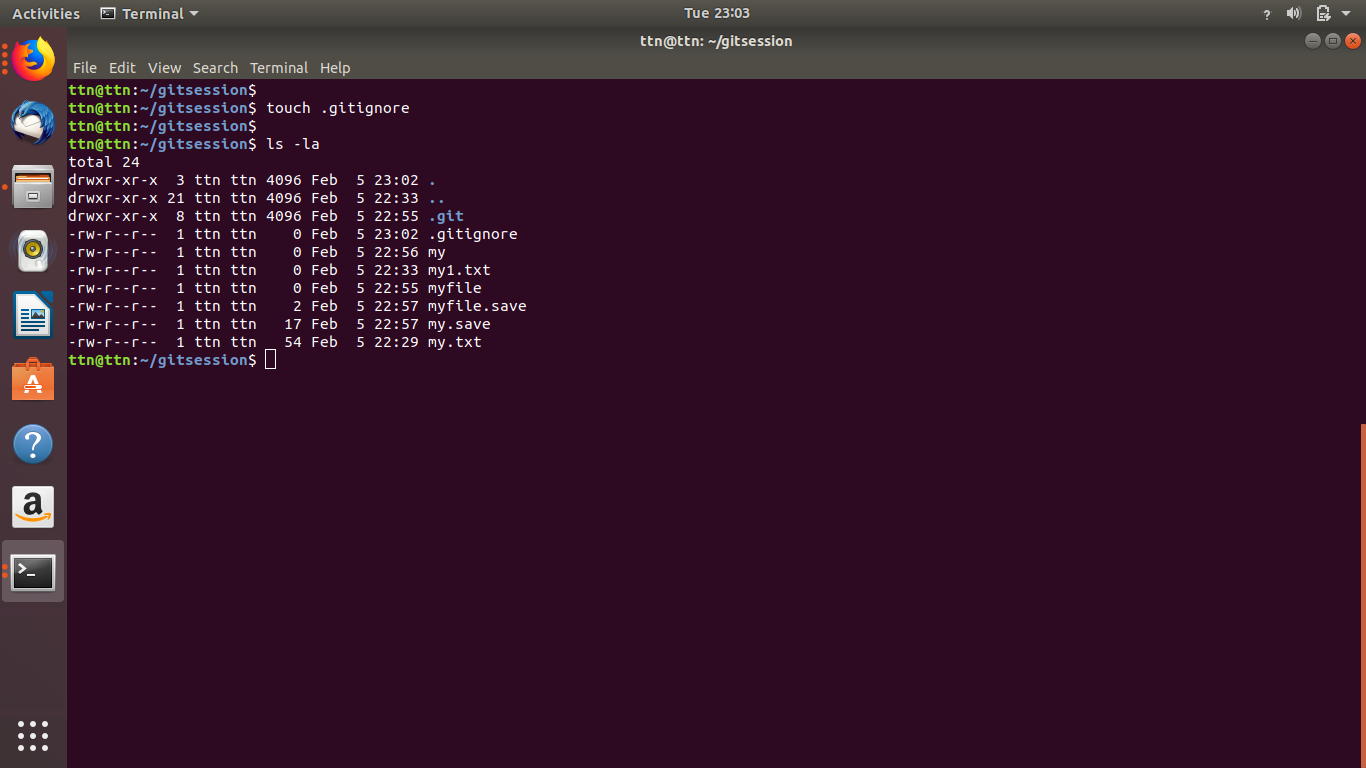
11.Check differences between a file and its staged version

Cmd - git diff --staged



12.Ignore a few files to be checked in

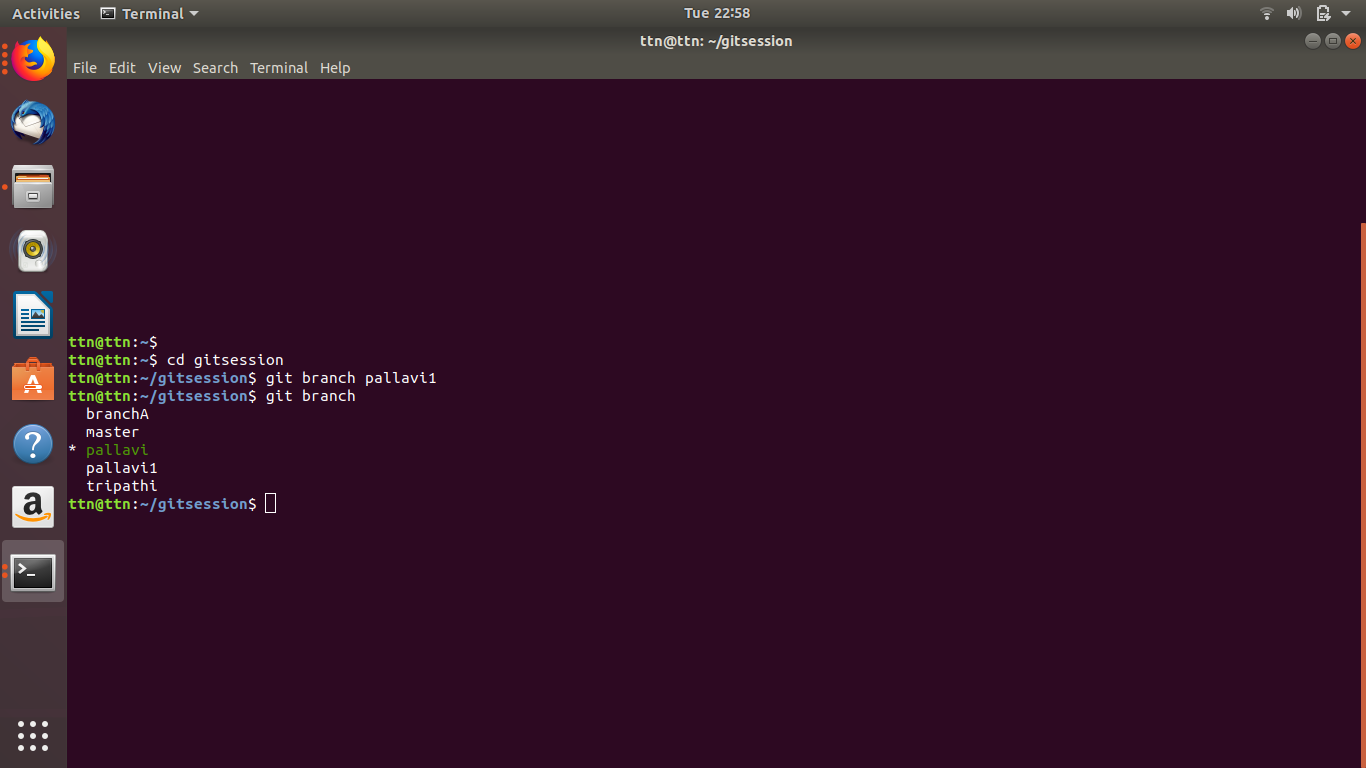
Cmd -touch .gitignore



13.Create a new branch.

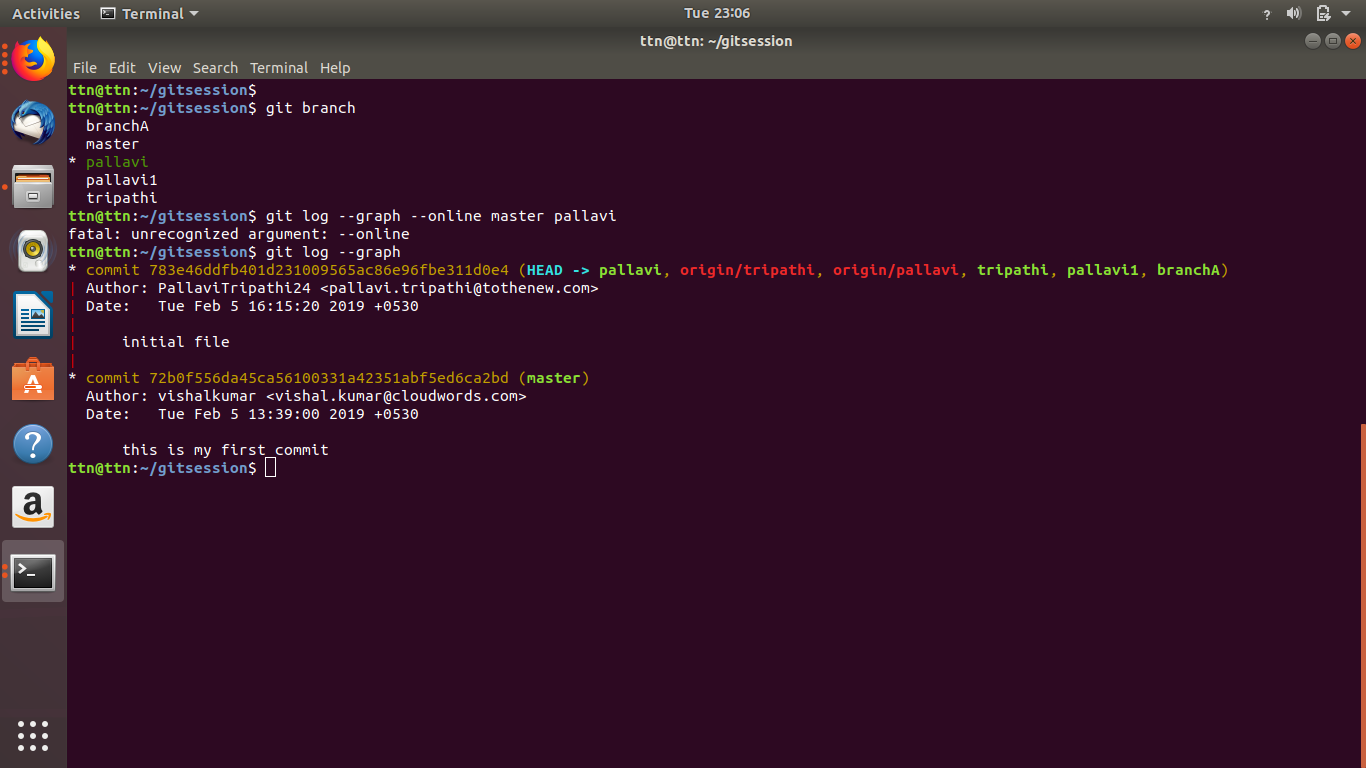
Cmd - git branch pallavi1

git branch pallavi



14.Diverge them with commits

Cmd - git log --graph



15.Edit the same file at the same line on both branches and commit

Cmd- git checkout pallavi

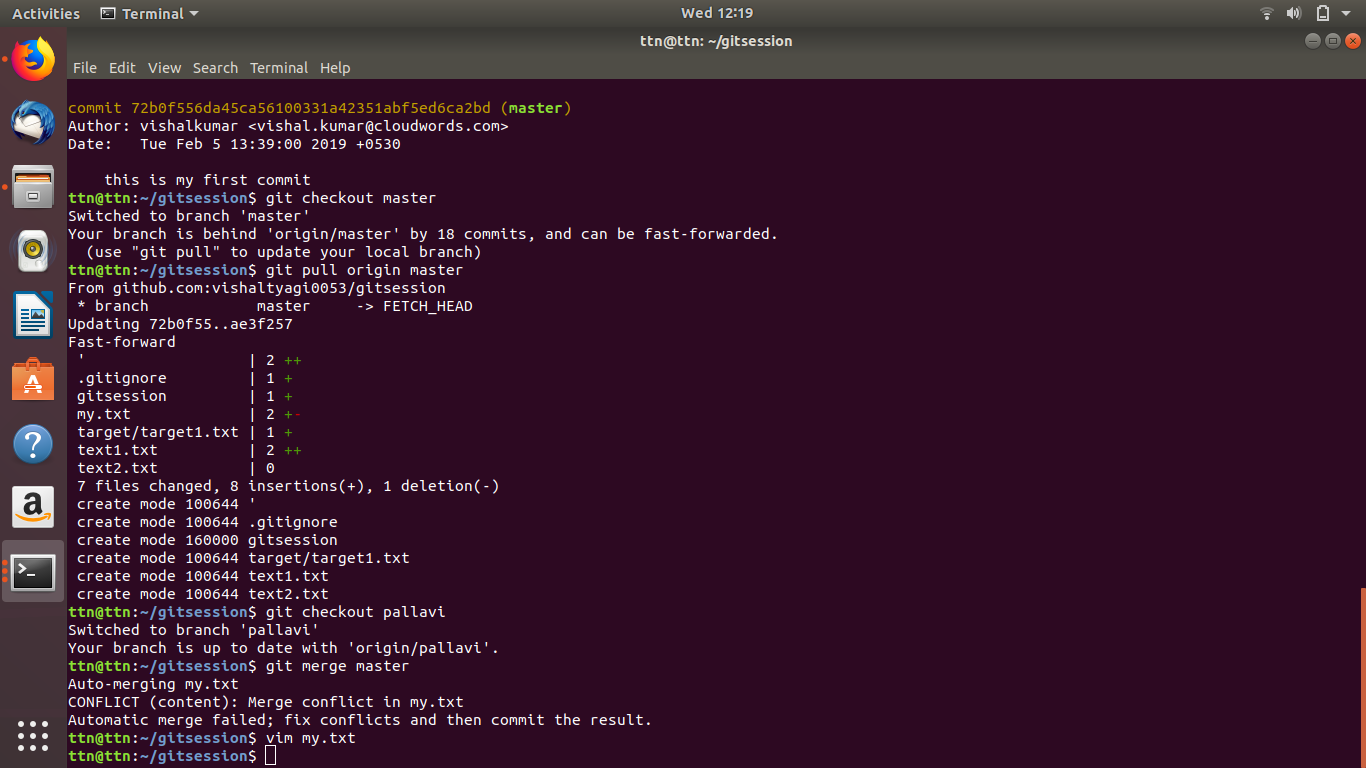
Vim my.txt

(there is change in third line)

Git commit -m “changed”

Git checkout master

Git pull origin master



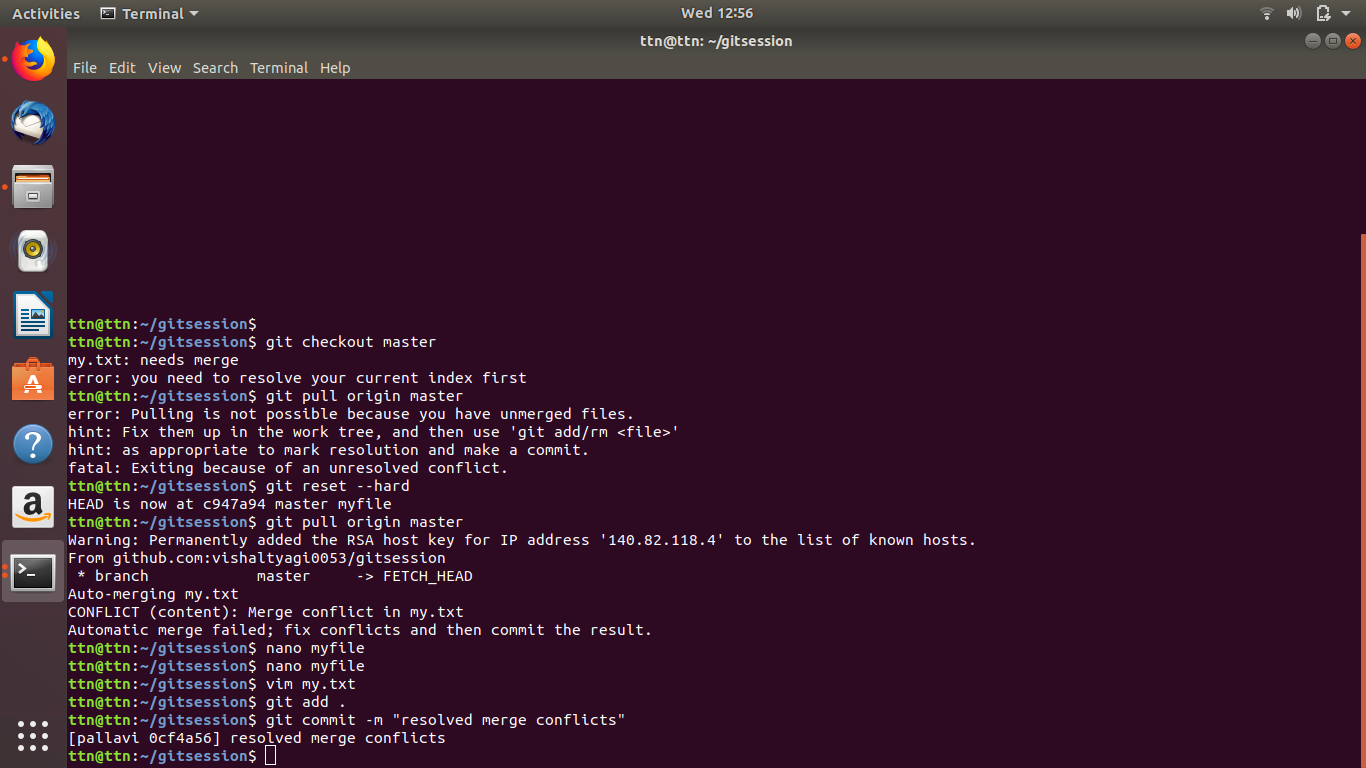
16. Try merging and resolve merge conflicts

Cmd - git merge master

To resolve conflict--deleted the conflicting lines

Git add .

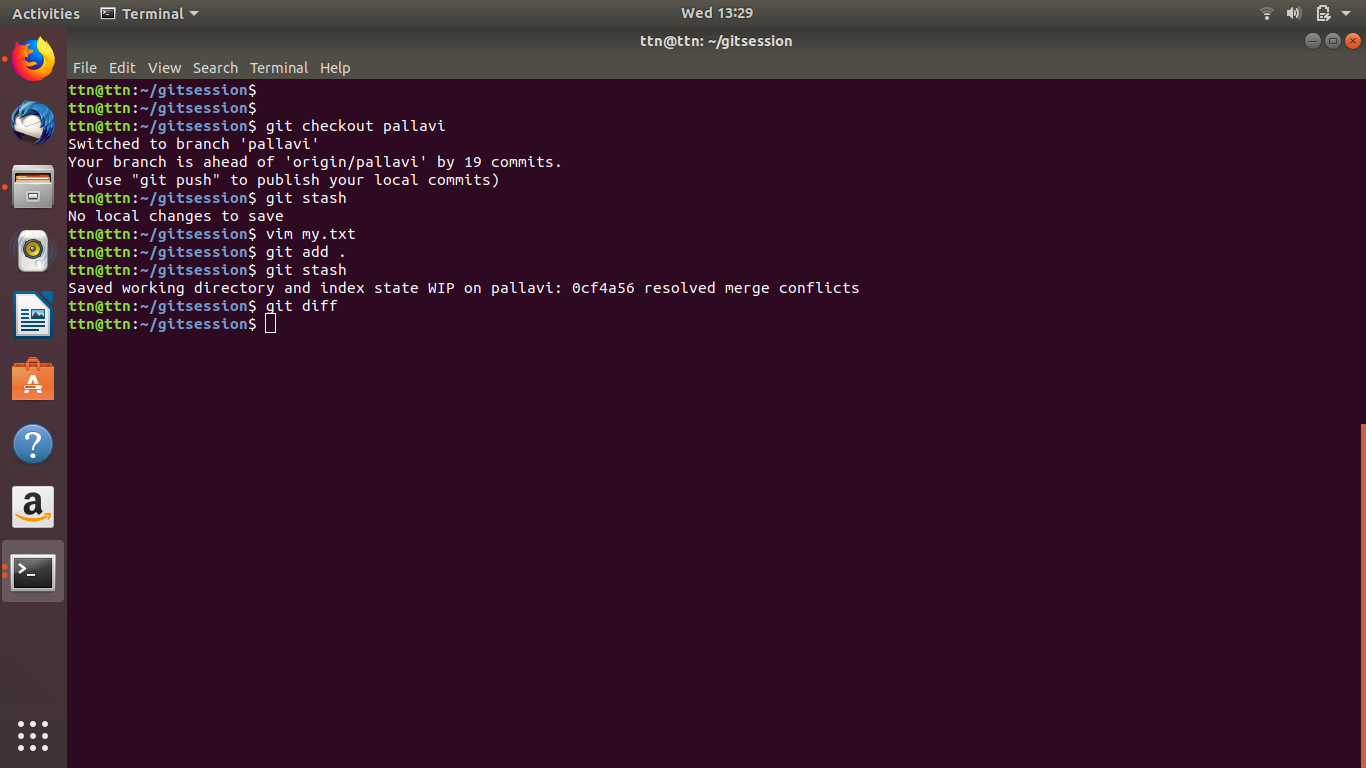
Git commit -m “resolve merge conflict”



17.Stash the changes and pop them

Cmd- git stash

Git diff



18.Add the following code to your .bashrc file : color\_prompt="yes"

parse\_git\_branch() {

git branch 2> /dev/null | sed -e '/^[^\*]/d' -e 's/\* \(.\*\)/(\1)/'

}

if [ "$color\_prompt" = yes ]; then

PS1='\u@\h\[\033[00m\]:\[\033[01;34m\]\W\[\033[01;31m\] $(parse\_git\_branch)\[\033[00m\]\$ '

else

PS1='\u@\h:\W $(parse\_git\_branch)\$ '

fi

unset color\_prompt force\_color\_prompt

Cmd -vim ~/.bashrc

