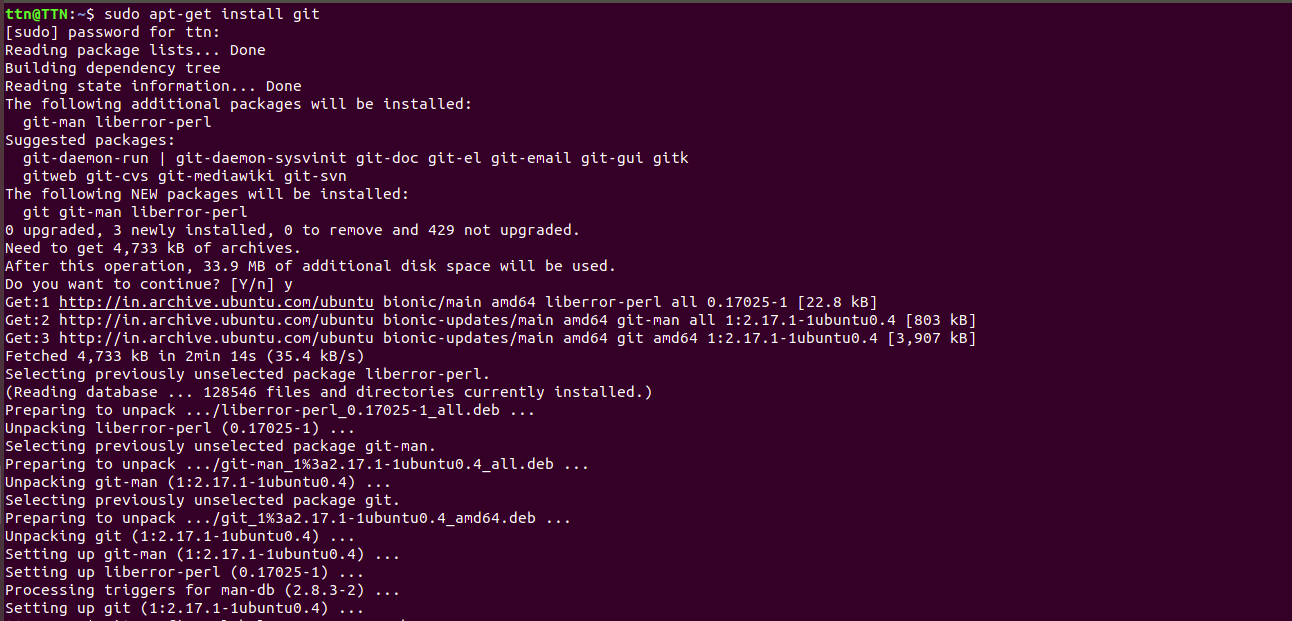
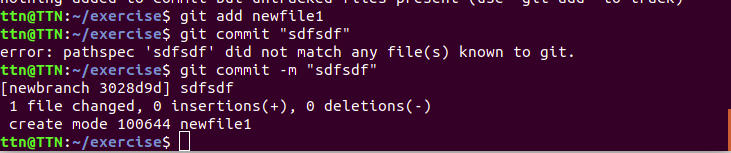
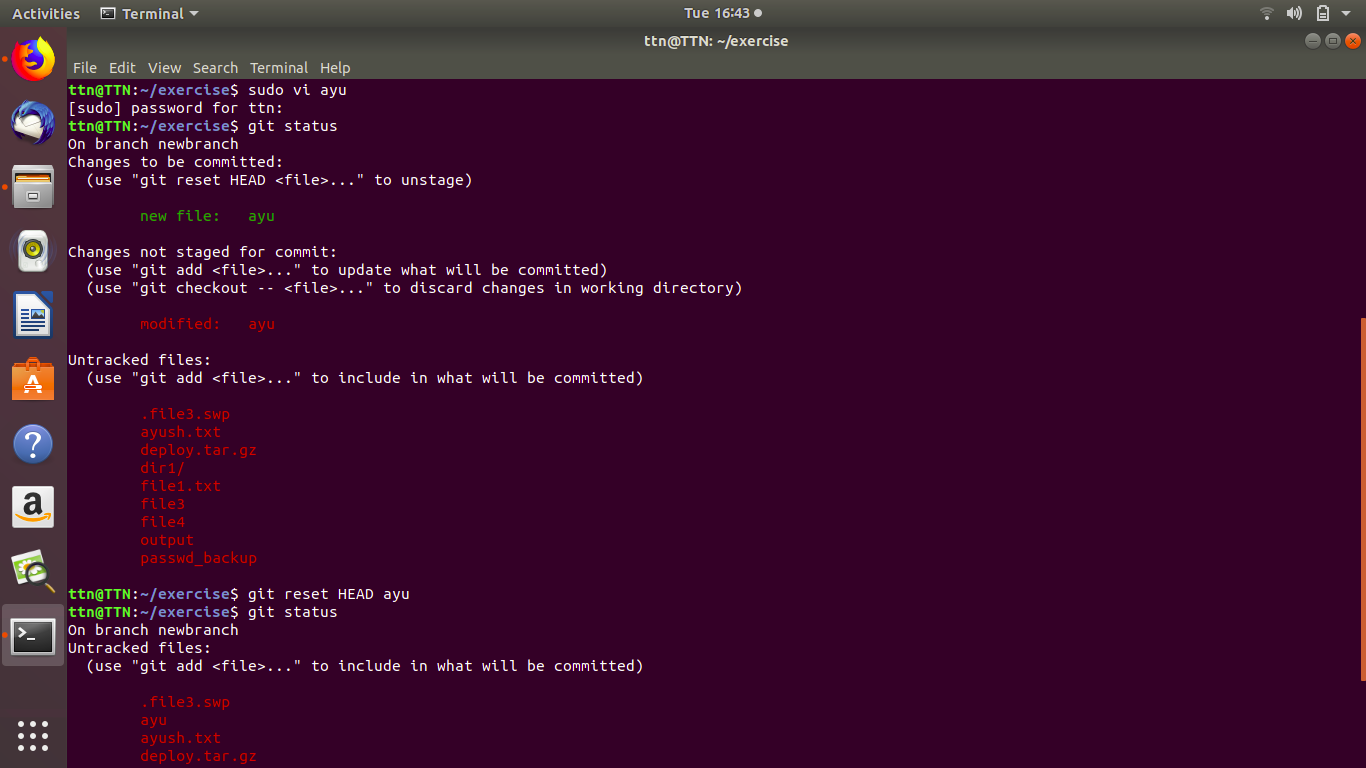
**ASSESSMENT-2**

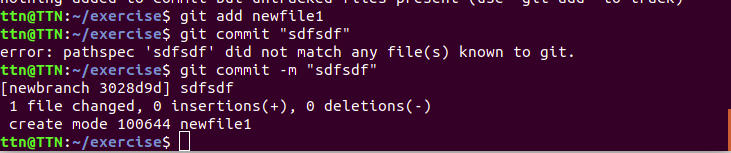
1. **Git Setup******
2. **Initialize a Git Repository**

3. **Add files to the repository**

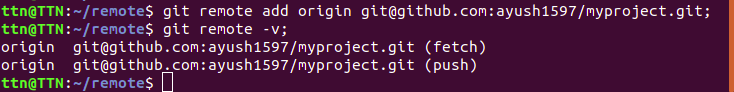
****

4. **Unstage 1 file**

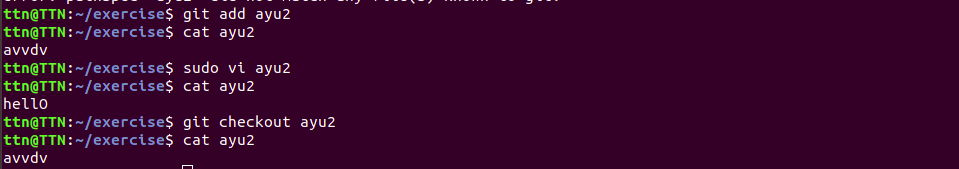
**5.Commit a file.**

****

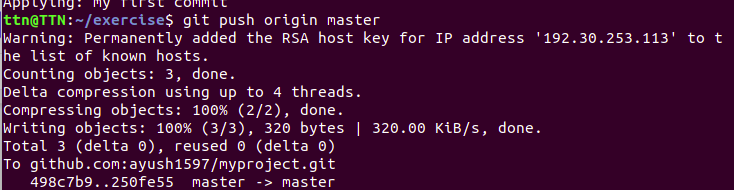
**6.Add a remote**

****

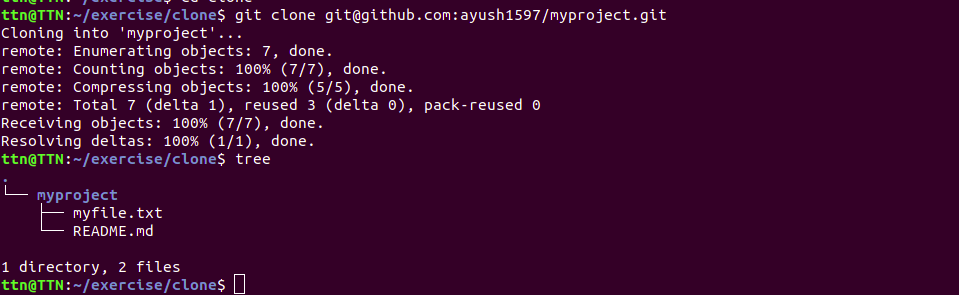
**7.Undo changes to a particular file**

****

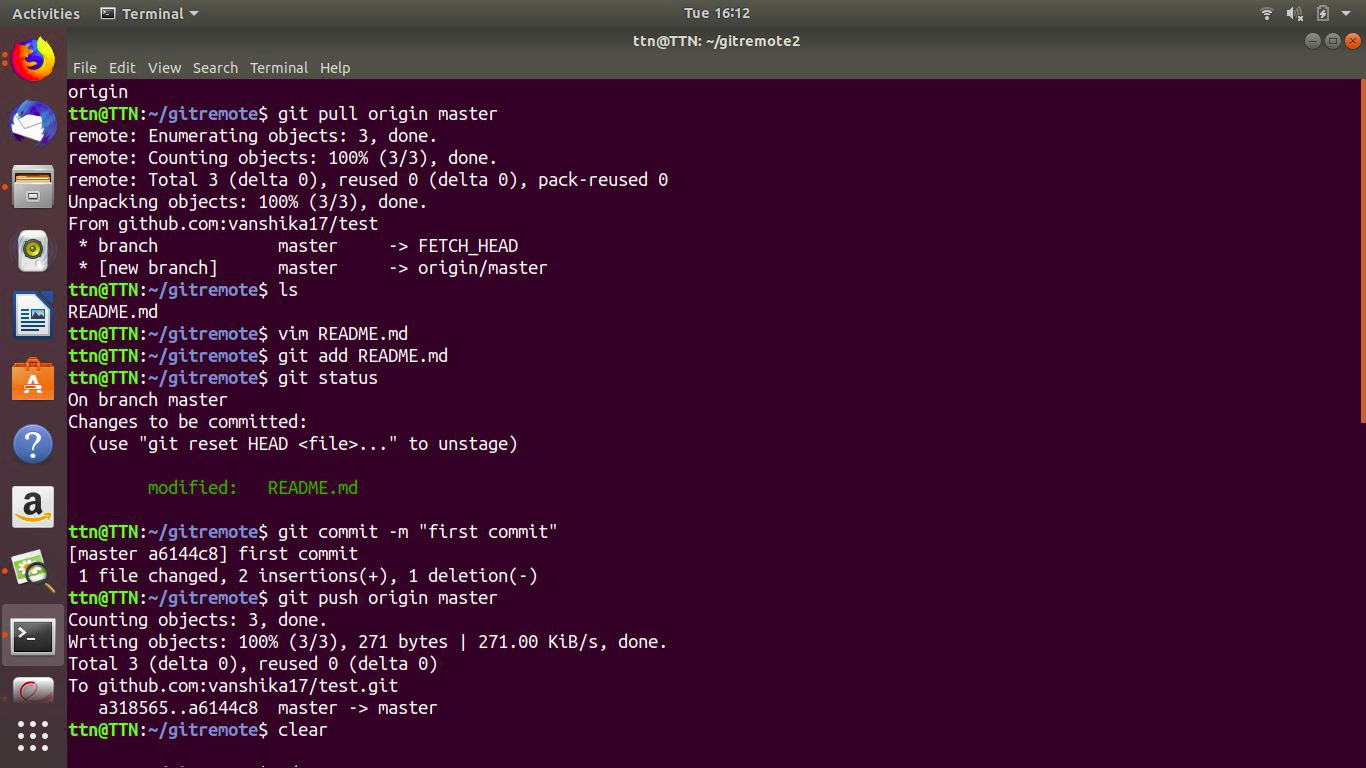
**8.push changes to github**

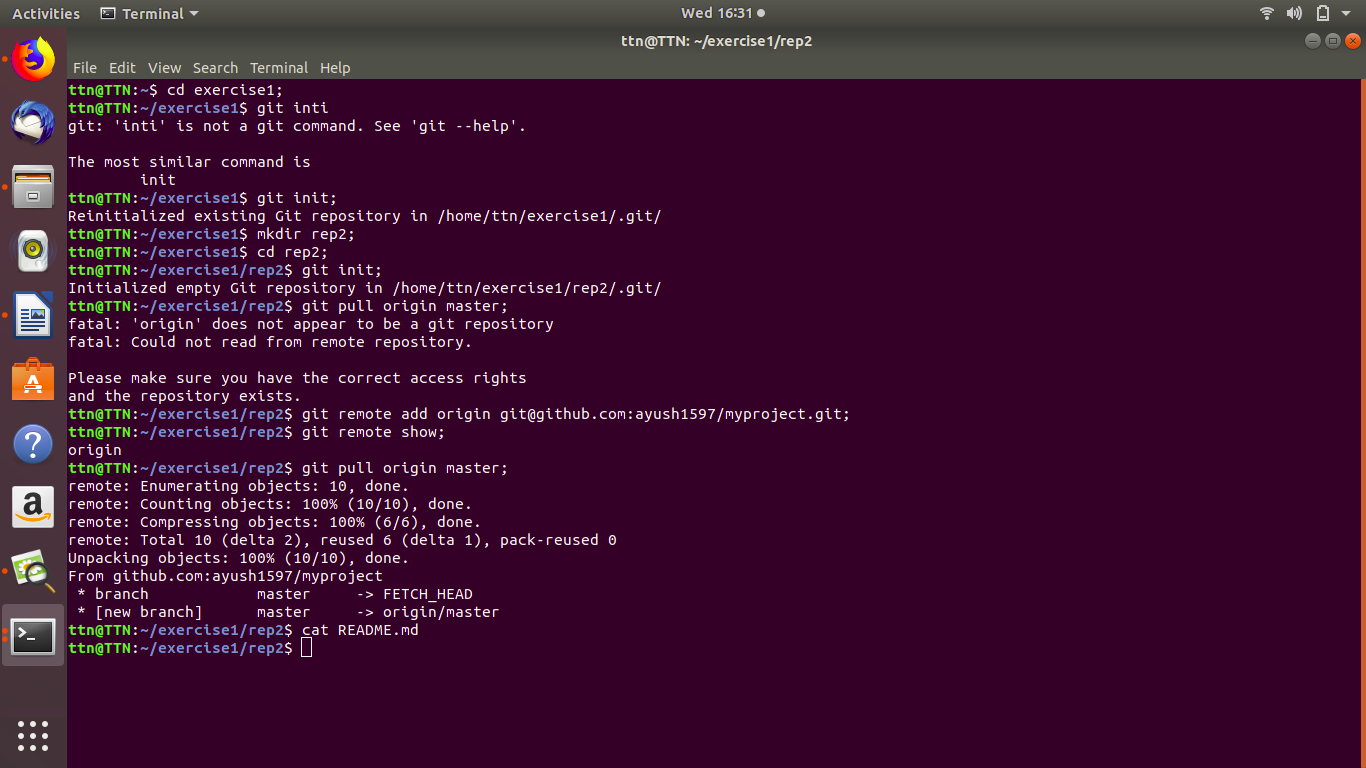
****

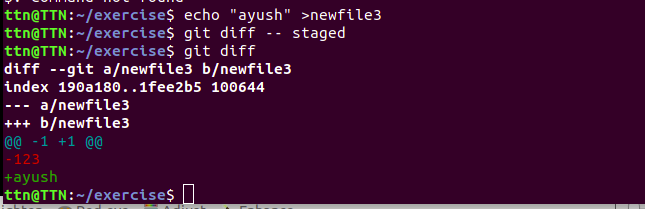
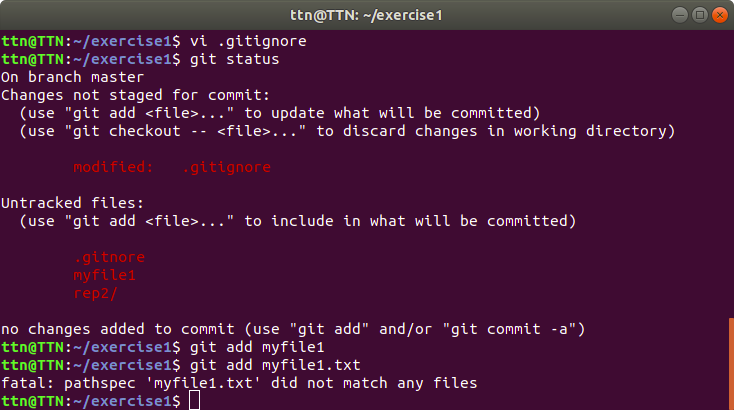
**9.Clone the repository**

****

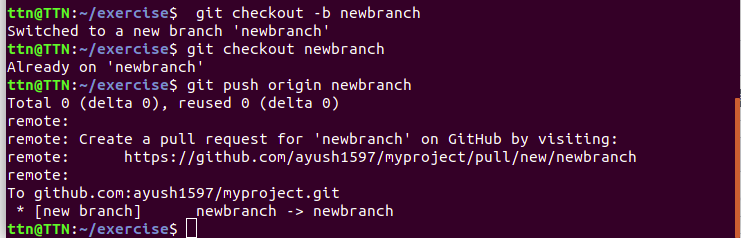
**10.Add changes to one of the copies and pull the changes in the other.**

****

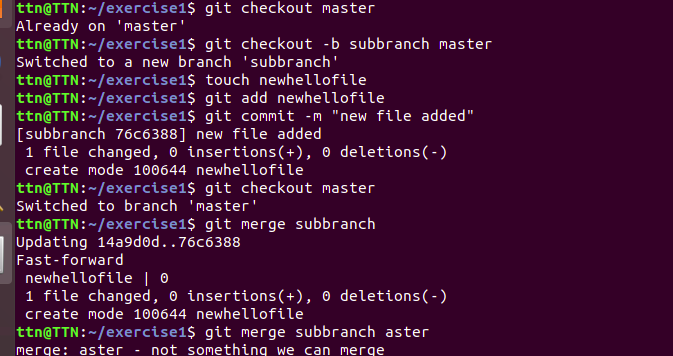
****

**11.Check differences between a file and its staged version12.Ignore a few files to be checked in**

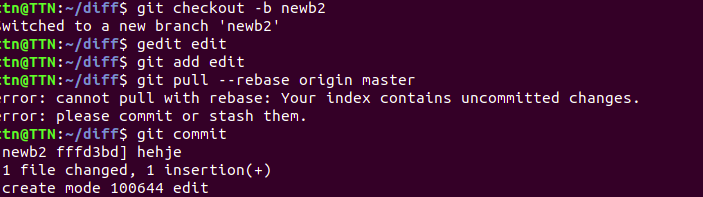
**13.create a new branch**

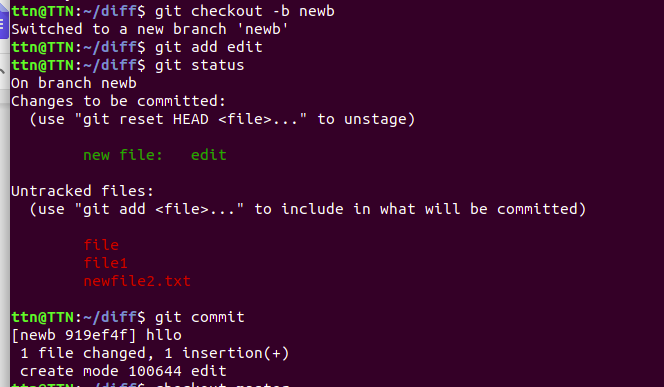


**14.Diverging them with commits**

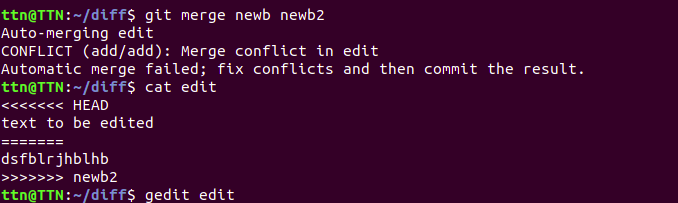
****

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

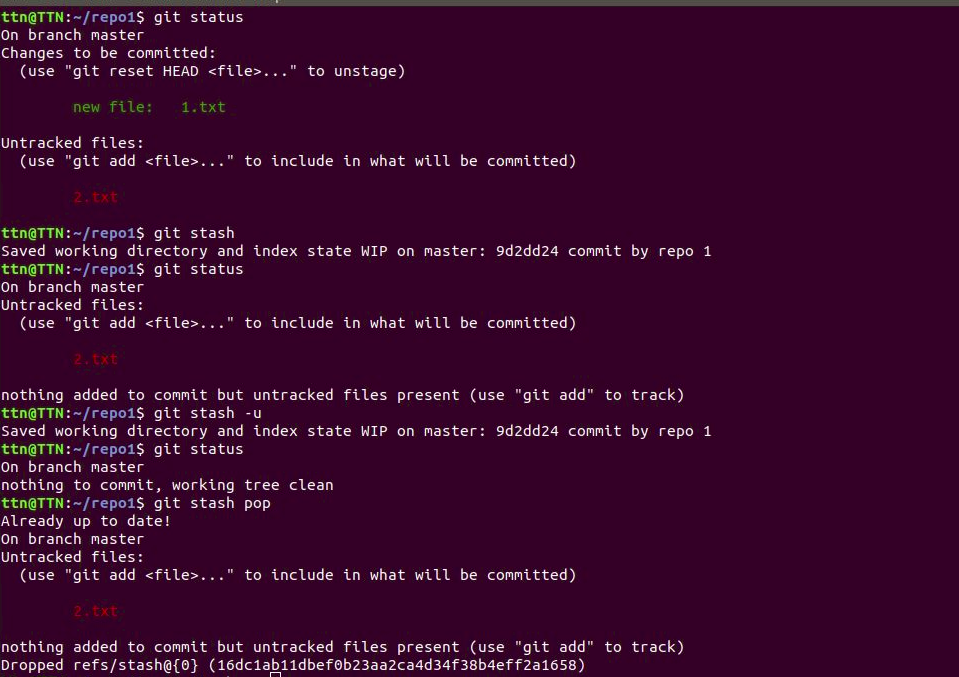
****

****

**16.Try merging and resolve merge conflicts**

****

**17.Stash the changes and pop them**



**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**

