Introduction to Version Control

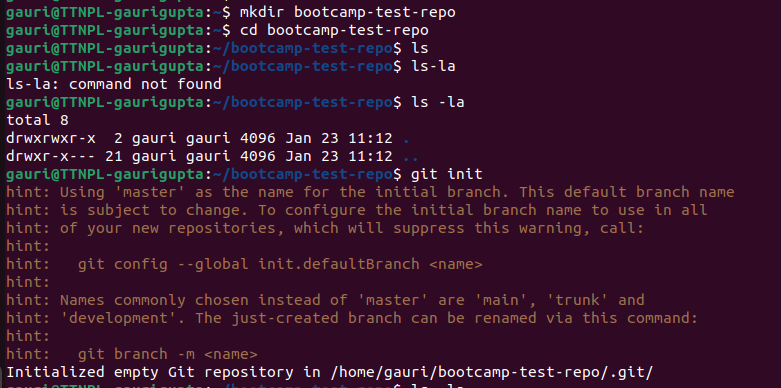
Question 1)

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



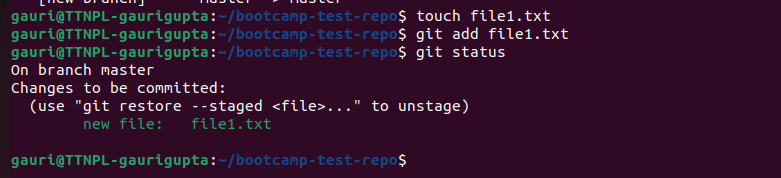
Question 2)

Initialize a Git Repository



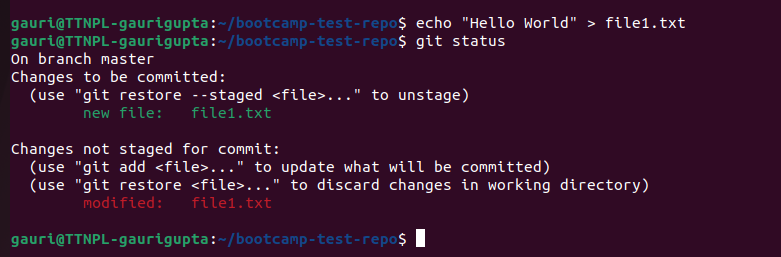
Question3)

Add files to the repository



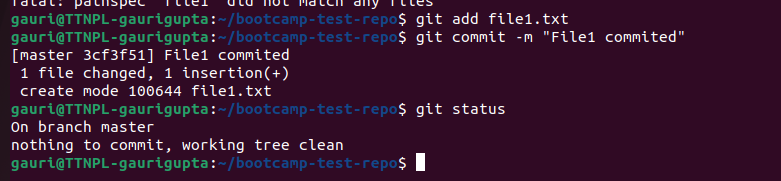
Question4)

Unstage 1 file



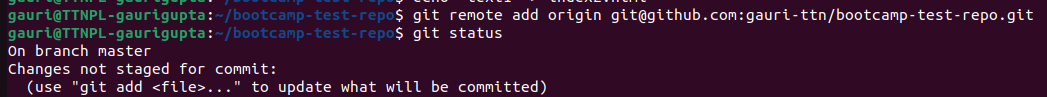
Question 5)

Commit the file



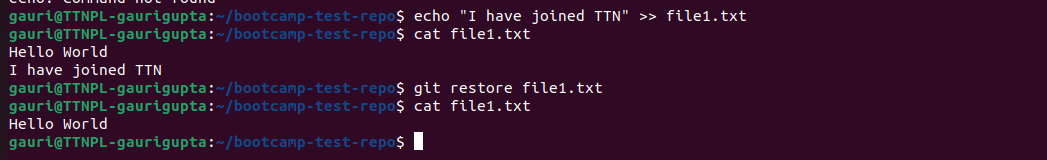
Question6)

Add a remote



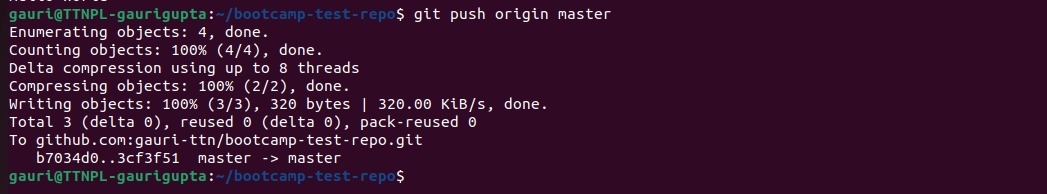
Question7)

Undo changes to a particular file



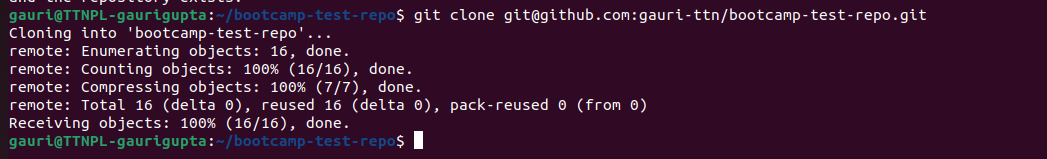
Question 8)

Push changes to Github



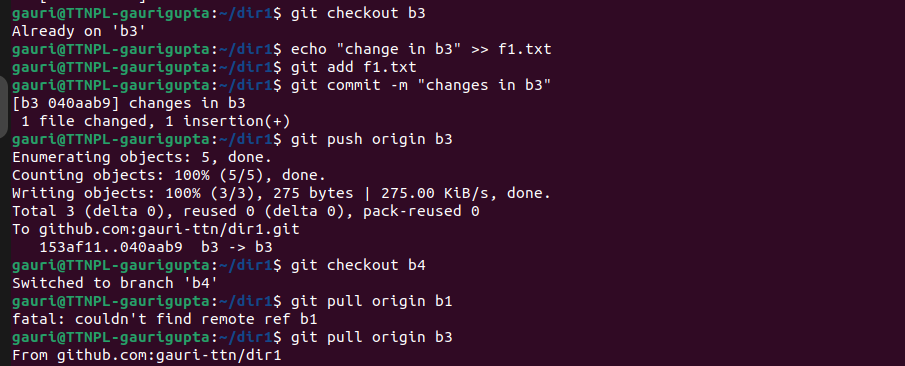
Question9)

Clone the repository



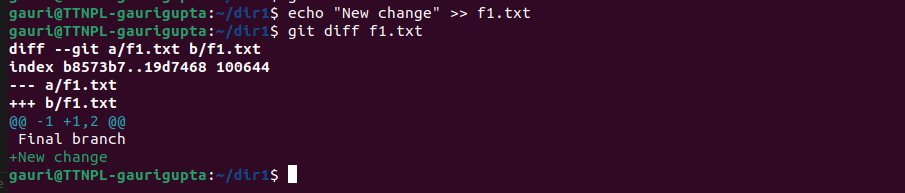
Question 10)

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



Question11)

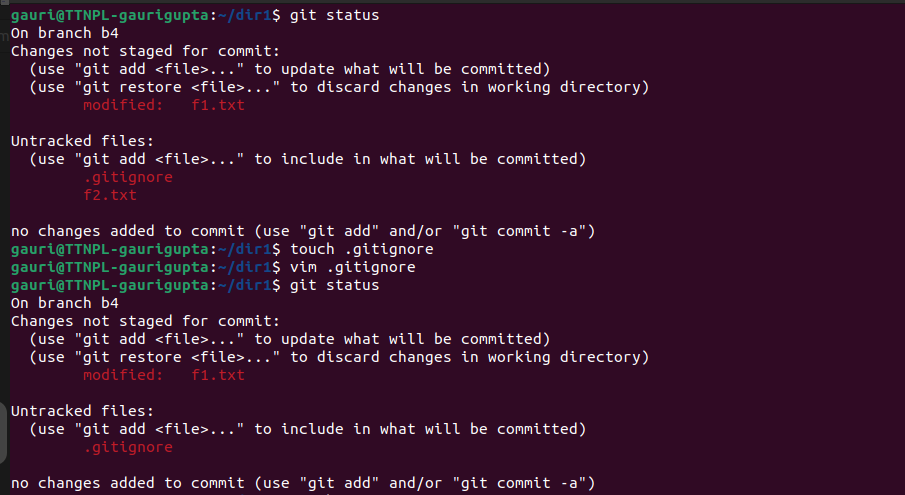
Check differences between a file and its staged version



Question12)

Ignore a few files to be checked in





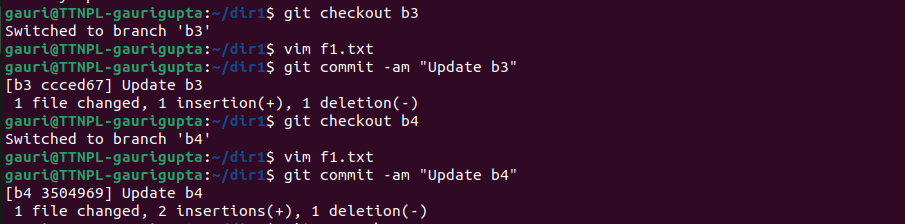
Question13)

Create a new branch



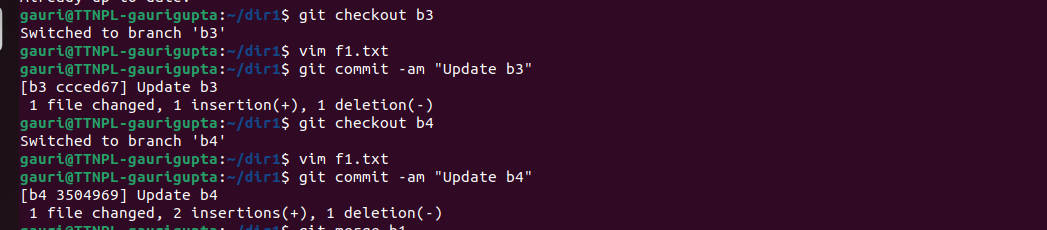
Question14)

Diverge them with commits



Question15)

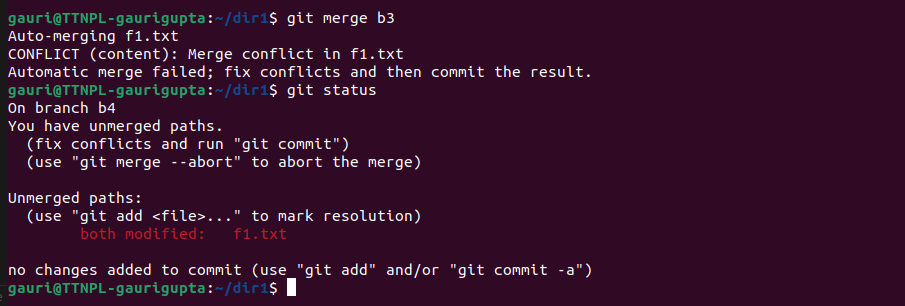
Edit the same file at the same line on both branches and commit



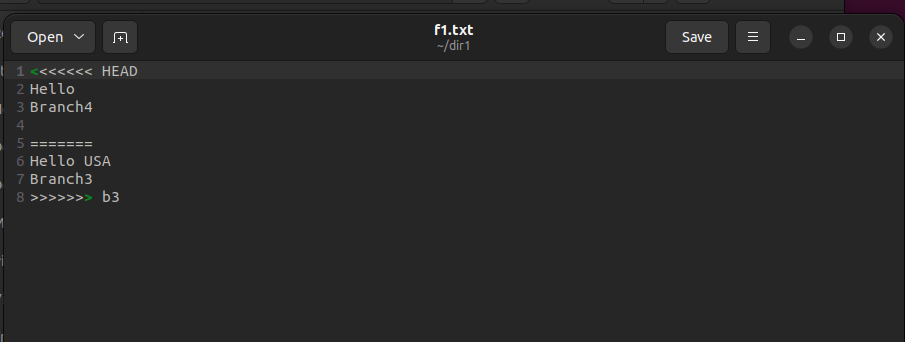
Question16)

Try merging and resolve merge conflicts

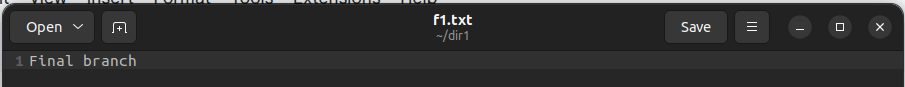
a) After merging , conflicts has rised.



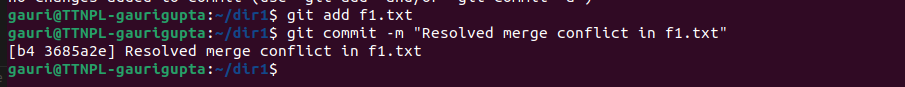
b)In f1.txt it is showing , about conflict.



c)To resolve the conflict , we selected the final line.

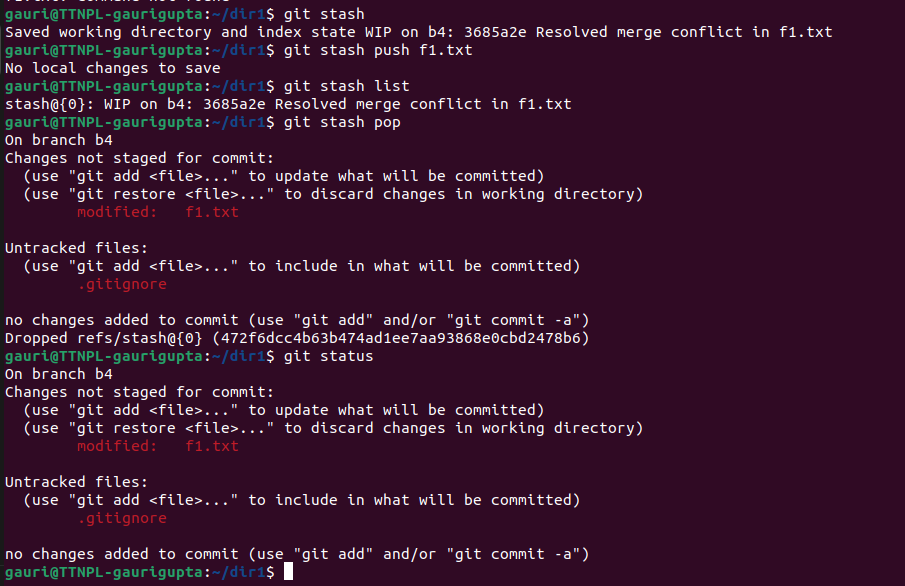


d) After resolving the conflict.



Question17)

Stash the changes and pop them



Question18)

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

