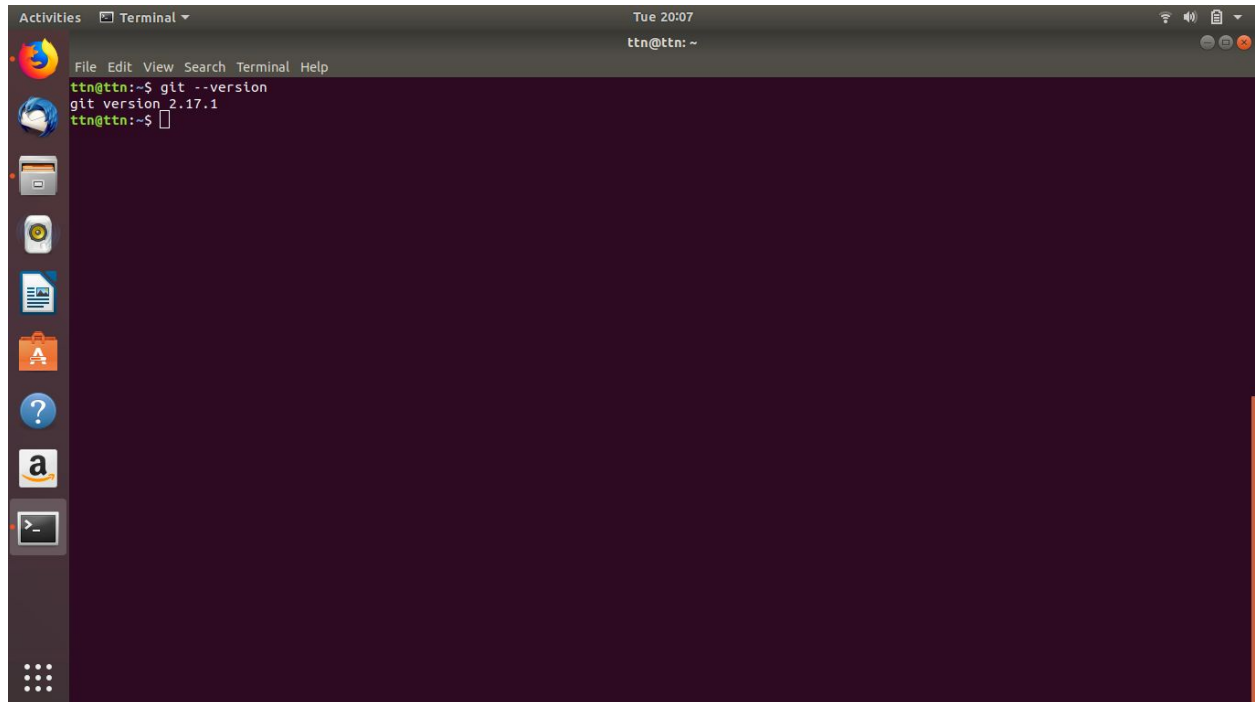


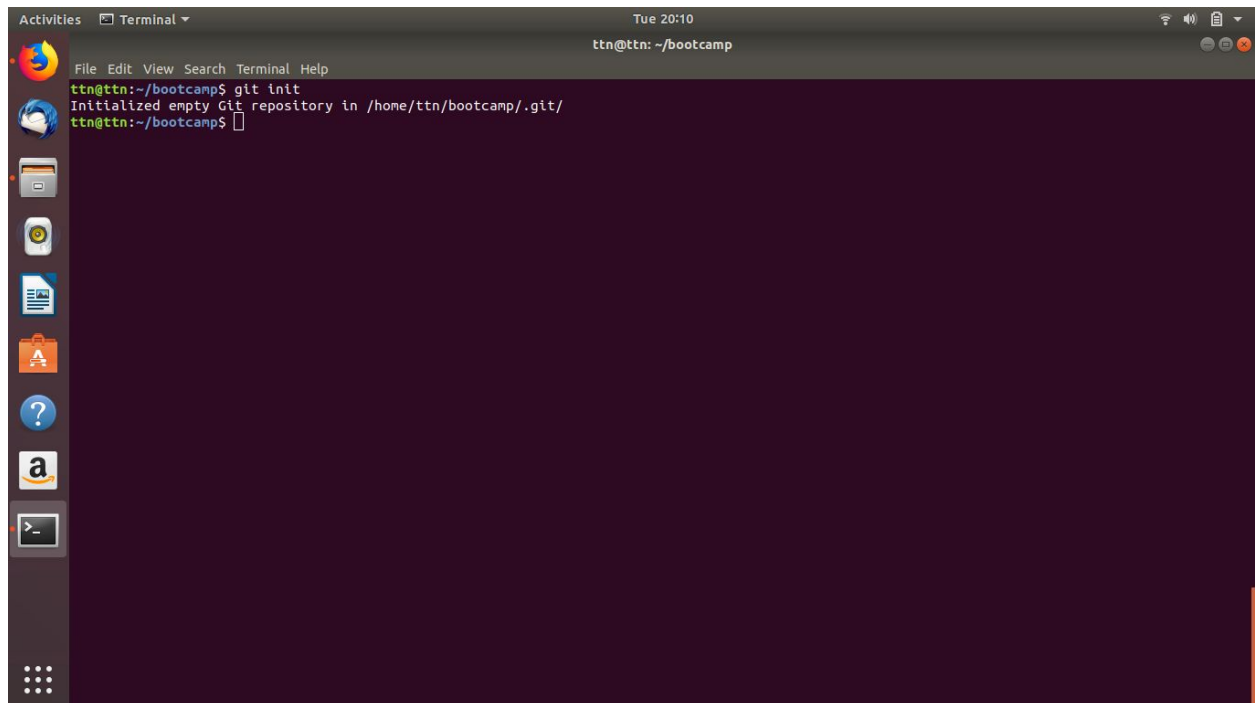
Version Control Session

Question1. Git Setup <https://confluence.atlassian.com/bitbucket/set-up-git-744723531.html>

A terminal window titled 'Terminal' with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Tue 20:07, ttn@ttn: ~). The terminal shows the command 'git --version' being executed, resulting in the output 'git version 2.17.1'. The prompt 'ttn@ttn:~\$' is visible on each line.

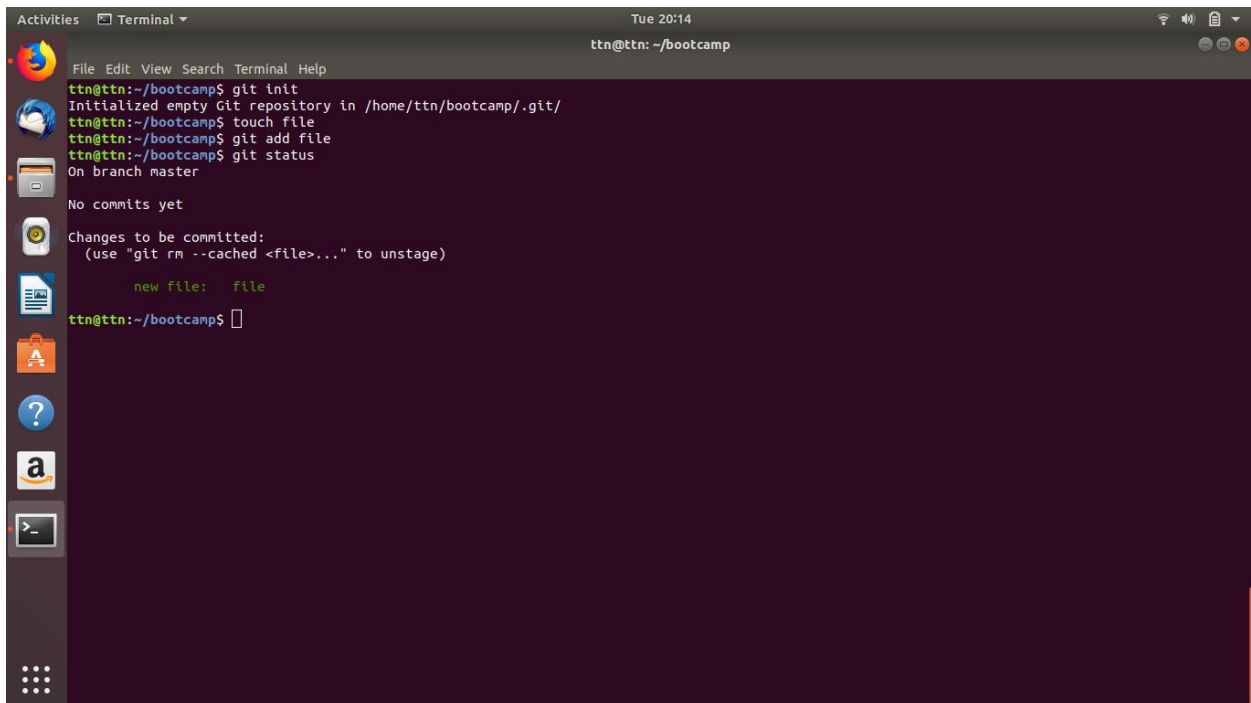
```
ttn@ttn:~$ git --version
git version 2.17.1
ttn@ttn:~$
```

Question3. Initialize a Git Repository

A terminal window titled 'Terminal' with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Tue 20:10, ttn@ttn: ~/bootcamp). The terminal shows the command 'git init' being executed, resulting in the output 'Initialized empty Git repository in /home/ttn/bootcamp/.git/'. The prompt 'ttn@ttn:~/bootcamp\$' is visible on each line.

```
ttn@ttn:~/bootcamp$ git init
Initialized empty Git repository in /home/ttn/bootcamp/.git/
ttn@ttn:~/bootcamp$
```

Question3.Add files to the repository

A terminal window titled 'Terminal' with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Tue 20:14, ttn@ttn: ~/bootcamp). The terminal shows the following commands and output:

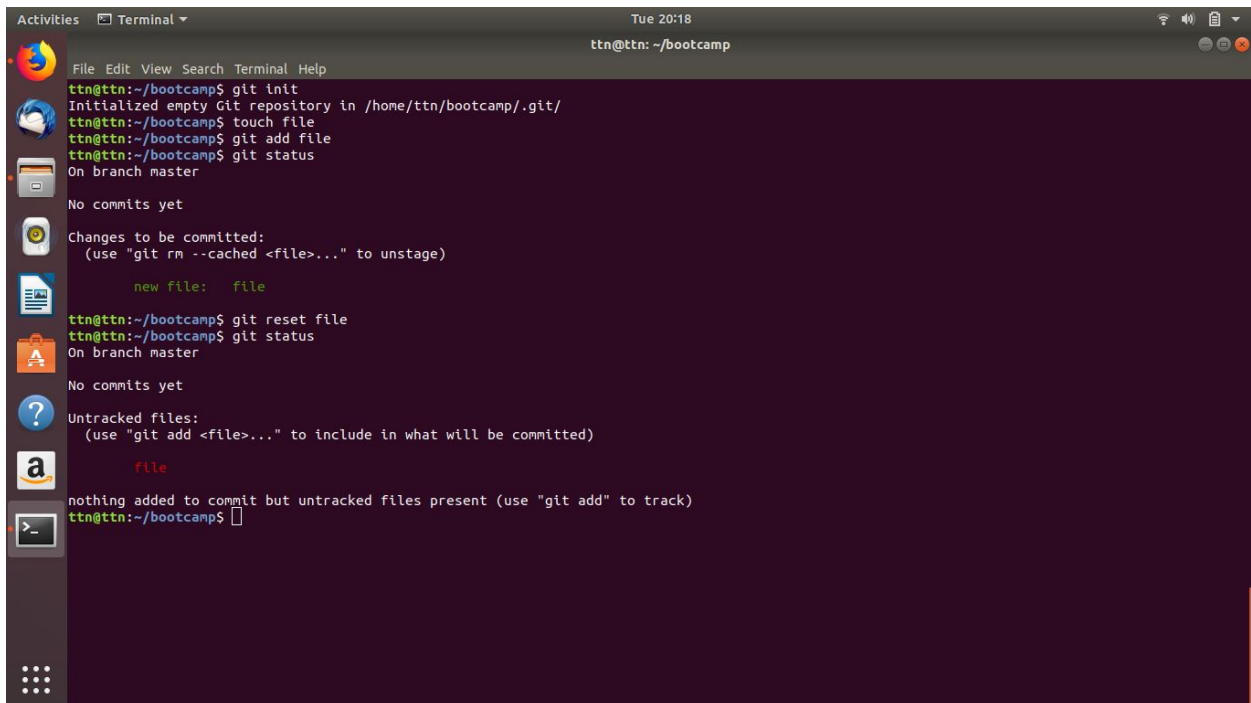
```
ttn@ttn:~/bootcamp$ git init
Initialized empty Git repository in /home/ttn/bootcamp/.git/
ttn@ttn:~/bootcamp$ touch file
ttn@ttn:~/bootcamp$ git add file
ttn@ttn:~/bootcamp$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

        new file:   file
ttn@ttn:~/bootcamp$
```

Question4.Unstage 1 file

A terminal window titled 'Terminal' with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Tue 20:18, ttn@ttn: ~/bootcamp). The terminal shows the following commands and output:

```
ttn@ttn:~/bootcamp$ git init
Initialized empty Git repository in /home/ttn/bootcamp/.git/
ttn@ttn:~/bootcamp$ touch file
ttn@ttn:~/bootcamp$ git add file
ttn@ttn:~/bootcamp$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

        new file:   file
ttn@ttn:~/bootcamp$ git reset file
ttn@ttn:~/bootcamp$ git status
On branch master

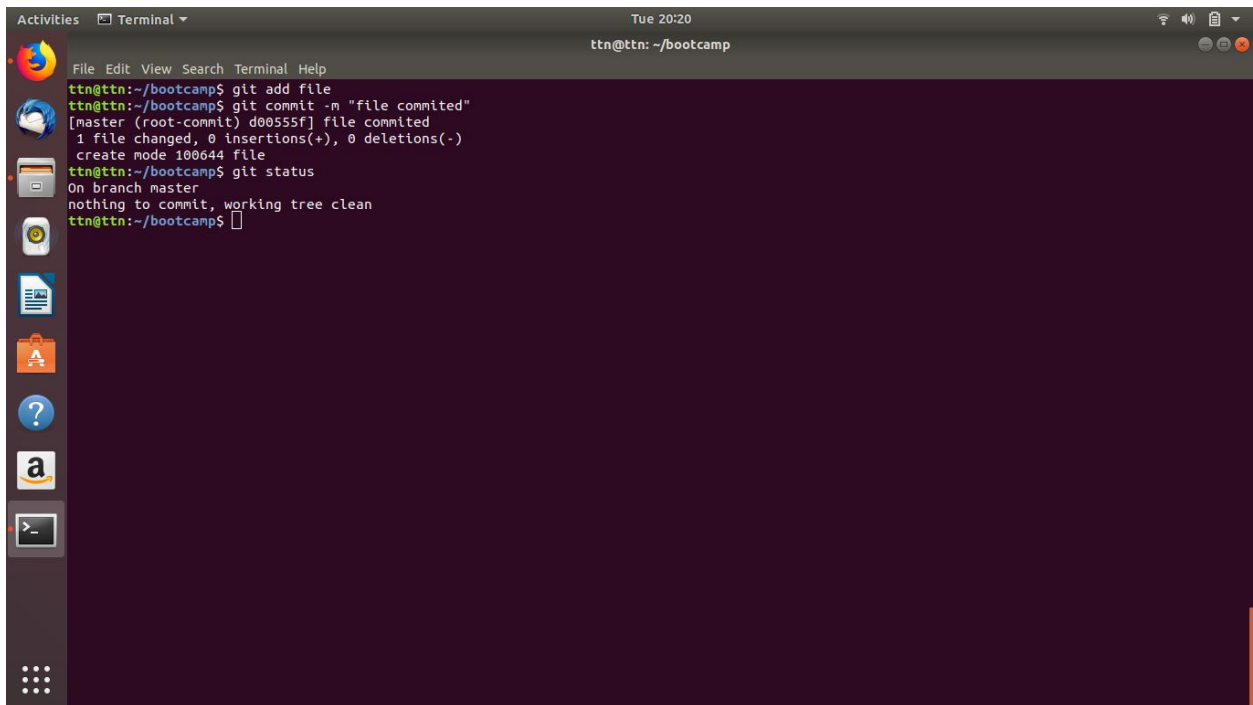
No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        file

nothing added to commit but untracked files present (use "git add" to track)
ttn@ttn:~/bootcamp$
```

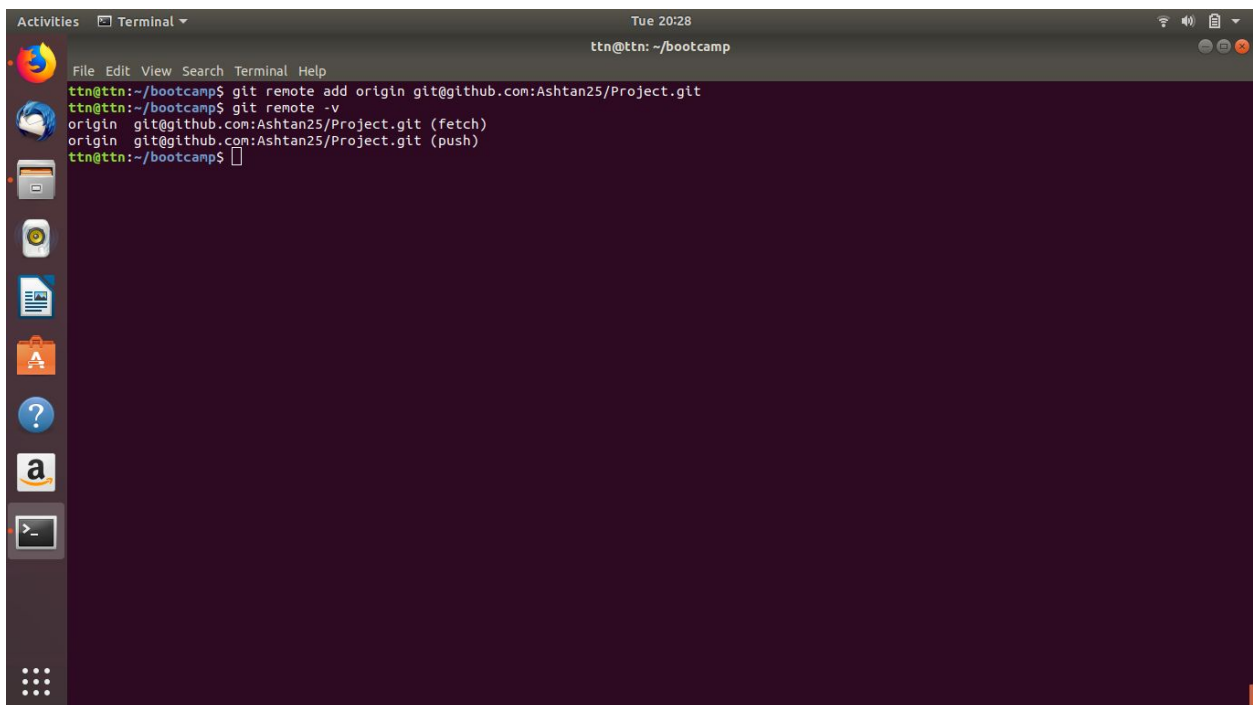
Question5.Commit the file



A terminal window titled "Terminal" with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Tue 20:20, ttn@ttn: ~/bootcamp). The terminal shows the following commands and output:

```
ttn@ttn:~/bootcamp$ git add file
ttn@ttn:~/bootcamp$ git commit -m "file committed"
[master (root-commit) d00555f] file committed
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file
ttn@ttn:~/bootcamp$ git status
On branch master
nothing to commit, working tree clean
ttn@ttn:~/bootcamp$
```

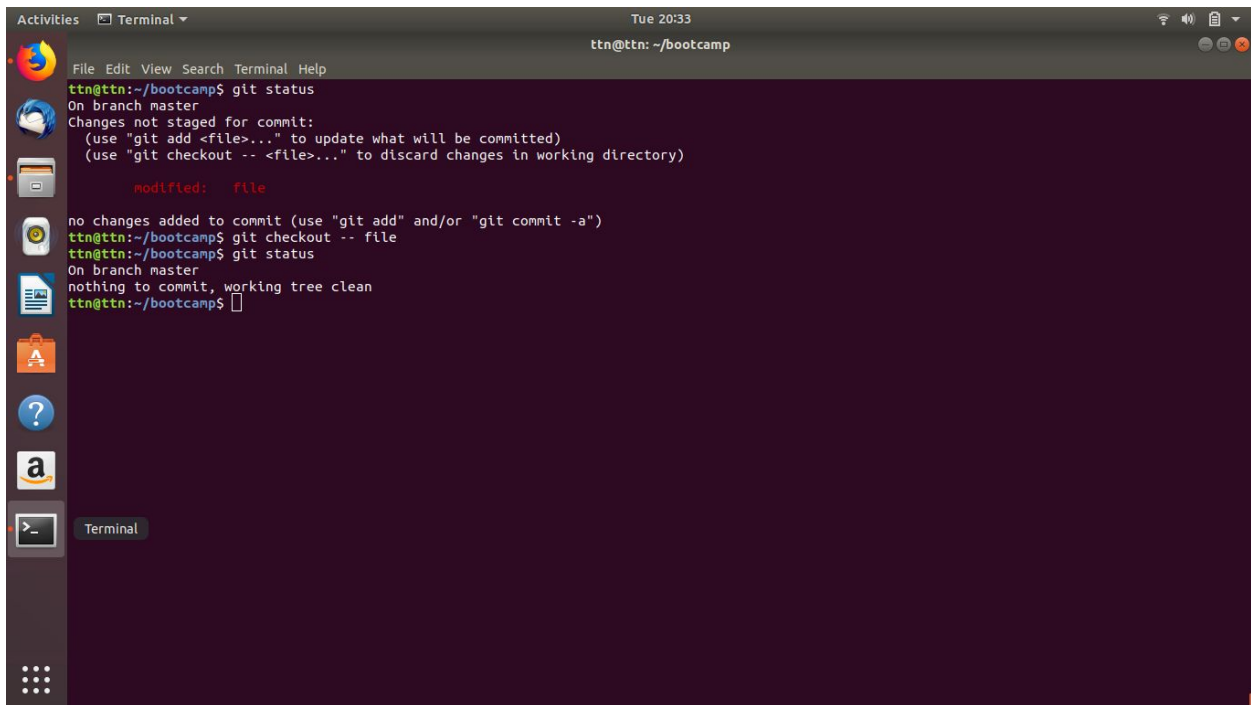
Question6.Add a remote



A terminal window titled "Terminal" with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Tue 20:28, ttn@ttn: ~/bootcamp). The terminal shows the following commands and output:

```
ttn@ttn:~/bootcamp$ git remote add origin git@github.com:Ashtan25/Project.git
ttn@ttn:~/bootcamp$ git remote -v
origin  git@github.com:Ashtan25/Project.git (fetch)
origin  git@github.com:Ashtan25/Project.git (push)
ttn@ttn:~/bootcamp$
```

Question7.Undo changes to a particular file



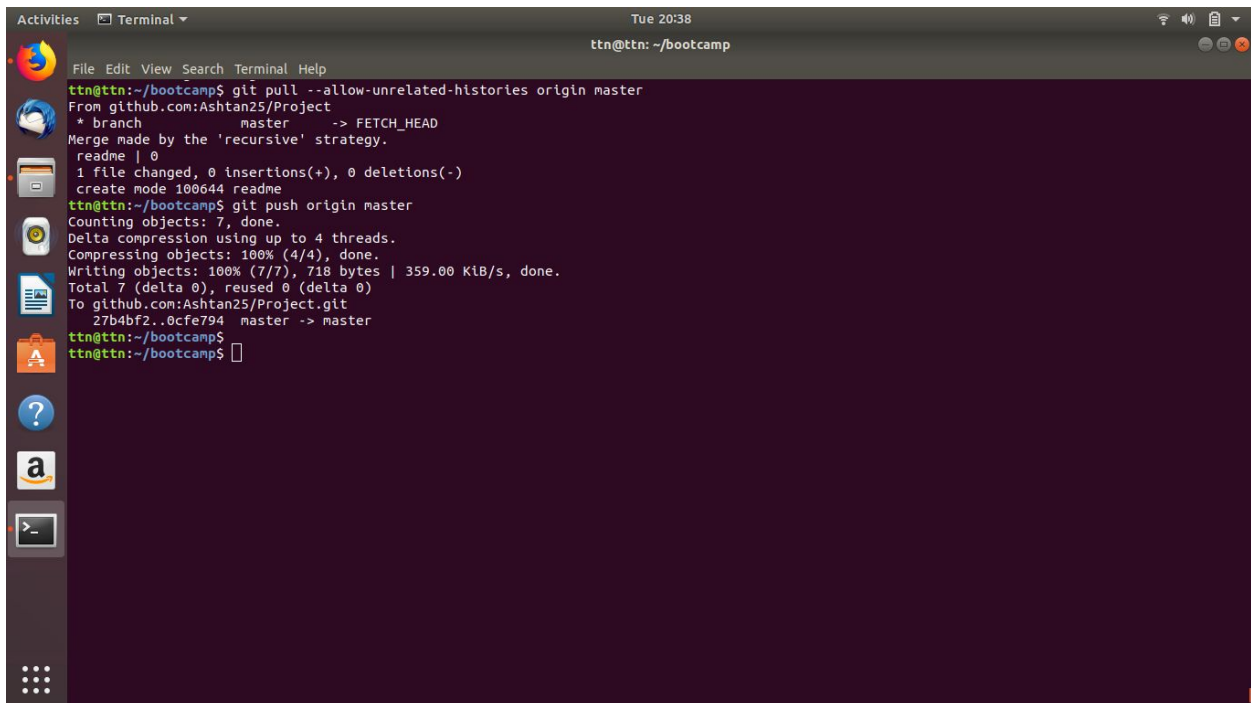
A terminal window titled "Terminal" showing the execution of git commands. The user is in the directory ~/bootcamp. The terminal output shows the current status of the repository, including the branch (master) and the files that have been modified. The user then runs the command "git checkout -- file" to undo changes to a specific file. The terminal output shows that the file has been checked out successfully and the working tree is now clean.

```
Activities Terminal Tue 20:33 ttn@ttn: ~/bootcamp
File Edit View Search Terminal Help
ttn@ttn:~/bootcamp$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

    modified:   file

no changes added to commit (use "git add" and/or "git commit -a")
ttn@ttn:~/bootcamp$ git checkout -- file
ttn@ttn:~/bootcamp$ git status
On branch master
nothing to commit, working tree clean
ttn@ttn:~/bootcamp$
```

Question8.Push changes to Github



A terminal window titled "Terminal" showing the execution of git commands. The user is in the directory ~/bootcamp. The terminal output shows the user pulling changes from the origin master branch. The user then runs the command "git push origin master" to push the changes to the remote repository. The terminal output shows that the push was successful and the remote repository is now up-to-date.

```
Activities Terminal Tue 20:38 ttn@ttn: ~/bootcamp
File Edit View Search Terminal Help
ttn@ttn:~/bootcamp$ git pull --allow-unrelated-histories origin master
From github.com:Ashtan25/Project
* branch          master       -> FETCH_HEAD
Merge made by the 'recursive' strategy.
 readme | 0
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 readme
ttn@ttn:~/bootcamp$ git push origin master
Counting objects: 7, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (7/7), 718 bytes | 359.00 KiB/s, done.
Total 7 (delta 0), reused 0 (delta 0)
To github.com:Ashtan25/Project.git
 27b4bf2..0cfe794  master -> master
ttn@ttn:~/bootcamp$
ttn@ttn:~/bootcamp$
```

Question9.Clone the repository

```
Activities Terminal Tue 20:41
ttn@ttn: ~/bootcamp/Project

File Edit View Search Terminal Help
ttn@ttn:~/bootcamp$ git clone git@github.com:Ashtan25/Project.git
Cloning into 'Project'...
remote: Enumerating objects: 13, done.
remote: Counting objects: 100% (13/13), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 13 (delta 0), reused 13 (delta 0), pack-reused 0
Receiving objects: 100% (13/13), done.
ttn@ttn:~/bootcamp$ ll
total 20
drwxr-xr-x 4 ttn ttn 4096 Feb 5 20:40 ./
drwxr-xr-x 20 ttn ttn 4096 Feb 5 20:10 ../
-rw-r--r-- 1 ttn ttn 23 Feb 5 20:34 file
drwxr-xr-x 8 ttn ttn 4096 Feb 5 20:37 .git/
drwxr-xr-x 3 ttn ttn 4096 Feb 5 20:40 Project/
-rw-r--r-- 1 ttn ttn 0 Feb 5 20:37 readme
ttn@ttn:~/bootcamp$ cd Project
ttn@ttn:~/bootcamp/Project$ ll
total 16
drwxr-xr-x 3 ttn ttn 4096 Feb 5 20:40 ./
drwxr-xr-x 4 ttn ttn 4096 Feb 5 20:40 ../
-rw-r--r-- 1 ttn ttn 23 Feb 5 20:40 file
drwxr-xr-x 8 ttn ttn 4096 Feb 5 20:40 .git/
-rw-r--r-- 1 ttn ttn 0 Feb 5 20:40 readme
ttn@ttn:~/bootcamp/Project$
```

Question10.Add changes to one of the copies and pull the changes in the other.

```
Activities Terminal Tue 20:52

ttn@ttn: ~/bootcamp
File Edit View Search Terminal Help
ttn@ttn:~/bootcamp$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   file

no changes added to commit (use "git add" and/or "git commit -a")
ttn@ttn:~/bootcamp$ git add file
ttn@ttn:~/bootcamp$ git commit -m "first line added"
[master 79a3473] first line added
 1 file changed, 1 insertion(+), 2 deletions(-)
ttn@ttn:~/bootcamp$ git push origin master
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 289 bytes | 289.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To github.com:Ashtan25/Project.git
   0cfe794..79a3473 master -> master
ttn@ttn:~/bootcamp$ cat file
Welcome assignmentttn@ttn:~/bootcamp$

ttn@ttn: ~/bootcamp/cloned/Project
File Edit View Search Terminal Help
ttn@ttn:~/bootcamp/cloned$ git clone git@github.com:Ashtan25/Project.git
Cloning into 'Project'...
remote: Enumerating objects: 16, done.
remote: Counting objects: 100% (16/16), done.
remote: Compressing objects: 100% (8/8), done.
remote: Total 16 (delta 0), reused 16 (delta 0), pack-reused 0
Receiving objects: 100% (16/16), done.
ttn@ttn:~/bootcamp/cloned$ ll
total 12
drwxr-xr-x 3 ttn ttn 4096 Feb 5 20:50 ./
drwxr-xr-x 4 ttn ttn 4096 Feb 5 20:50 ../
drwxr-xr-x 3 ttn ttn 4096 Feb 5 20:50 Project/
ttn@ttn:~/bootcamp/cloned$ cd Project
ttn@ttn:~/bootcamp/cloned/Project$ ll
total 16
drwxr-xr-x 3 ttn ttn 4096 Feb 5 20:50 ./
drwxr-xr-x 3 ttn ttn 4096 Feb 5 20:50 ../
-rw-r--r-- 1 ttn ttn 18 Feb 5 20:50 file
drwxr-xr-x 8 ttn ttn 4096 Feb 5 20:50 .git/
-rw-r--r-- 1 ttn ttn 0 Feb 5 20:50 readme
ttn@ttn:~/bootcamp/cloned/Project$ cat file
Welcome assignmentttn@ttn:~/bootcamp/cloned/Project$
```

Question11. Check differences between a file and its staged version

```
Activities Terminal Tue 21:58 ttn@ttn: ~/bootcamp
File Edit View Search Terminal Help
ttn@ttn:~/bootcamp$ nano file
ttn@ttn:~/bootcamp$ git add file
ttn@ttn:~/bootcamp$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    modified:   file

ttn@ttn:~/bootcamp$ nano file
ttn@ttn:~/bootcamp$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    modified:   file

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

    modified:   file

ttn@ttn:~/bootcamp$ git diff file
diff --git a/file b/file
index b2a8e3a..ae731d1 100644
--- a/file
+++ b/file
@@ -1,2 +1,3 @@
 Welcome assignment
 To See the difference
+again a difference
ttn@ttn:~/bootcamp$
```

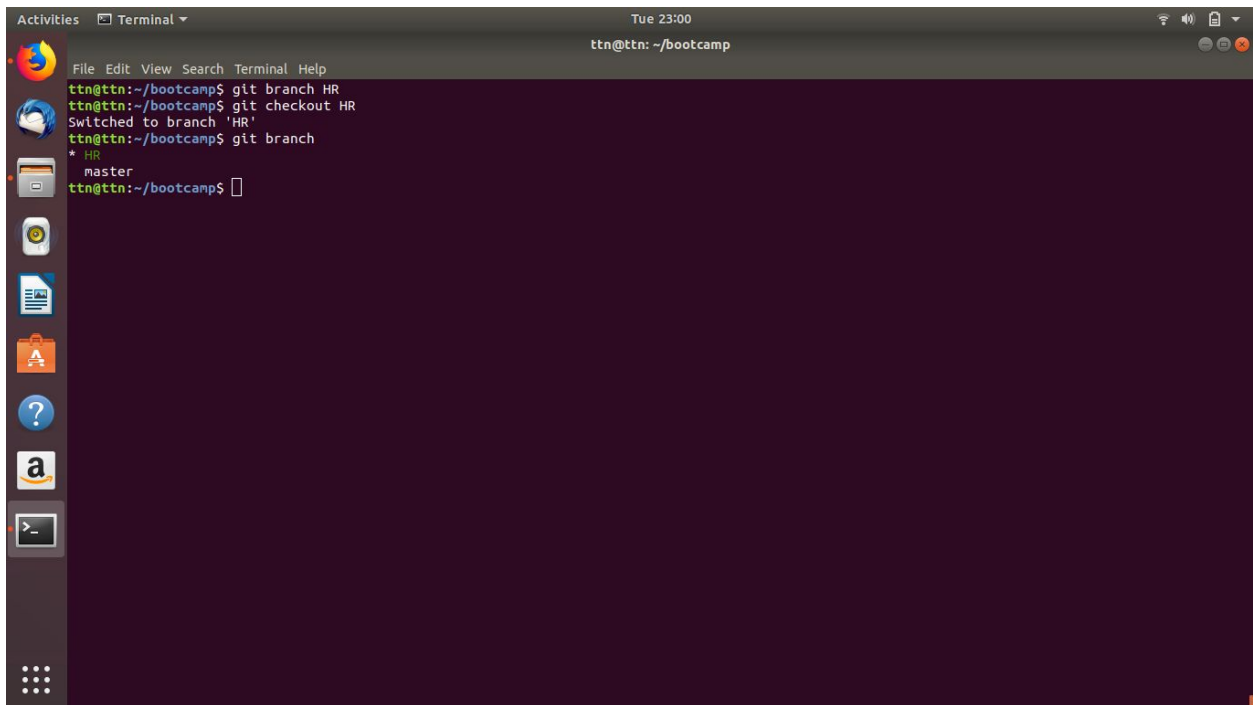
Question12. Ignore a few files to be checked in

```
Activities Terminal Tue 22:11 ttn@ttn: ~/bootcamp
File Edit View Search Terminal Help
ttn@ttn:~/bootcamp$ touch .gitignore
ttn@ttn:~/bootcamp$ nano .gitignore
ttn@ttn:~/bootcamp$ git add .gitignore
ttn@ttn:~/bootcamp$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    modified:   .gitignore

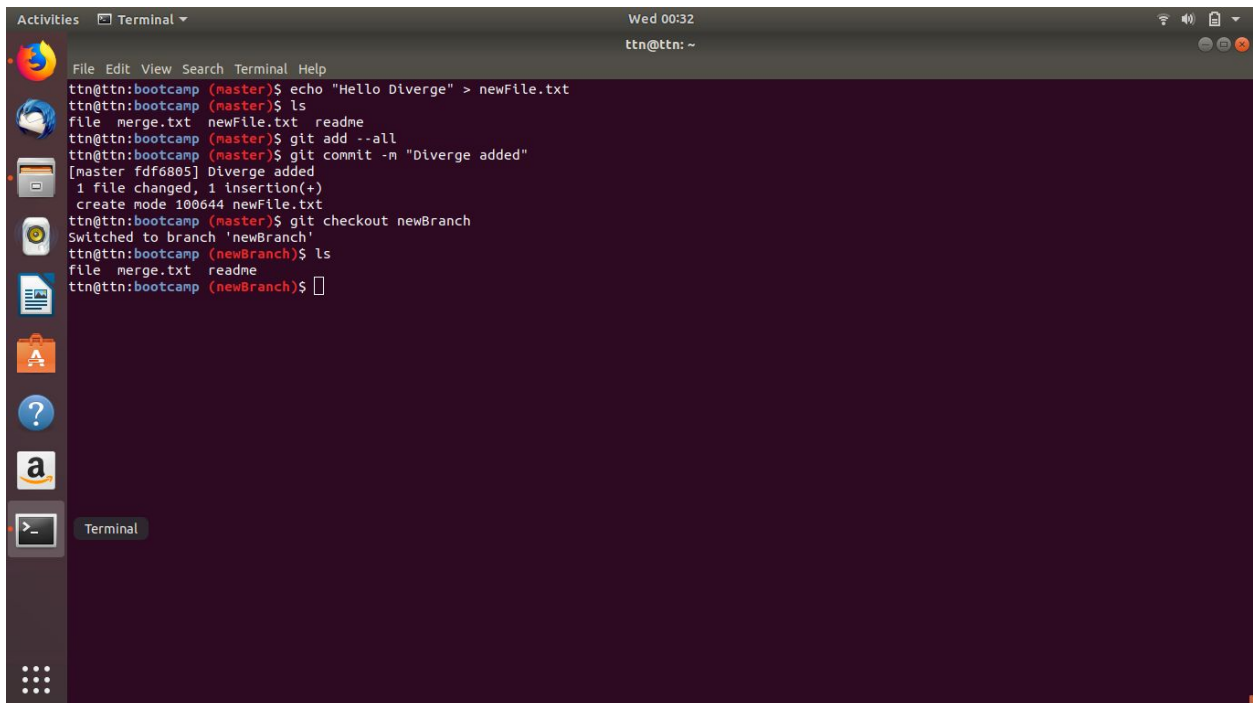
ttn@ttn:~/bootcamp$ touch ash.html
ttn@ttn:~/bootcamp$ git commit -m ".gitignore added"
[master 08f2ea4] .gitignore added
 1 file changed, 1 insertion(+)
ttn@ttn:~/bootcamp$ touch add.html
ttn@ttn:~/bootcamp$ git status
On branch master
nothing to commit, working tree clean
ttn@ttn:~/bootcamp$
```

Question13.Create a new branch

A terminal window titled 'Terminal' with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Tue 23:00, ttn@ttn: ~/bootcamp). The terminal shows the following commands and output:

```
ttn@ttn:~/bootcamp$ git branch HR
ttn@ttn:~/bootcamp$ git checkout HR
Switched to branch 'HR'
ttn@ttn:~/bootcamp$ git branch
* HR
master
ttn@ttn:~/bootcamp$
```

Question14.Diverge them with commits

A terminal window titled 'Terminal' with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Wed 00:32, ttn@ttn: ~). The terminal shows the following commands and output:

```
ttn@ttn:bootcamp (master)$ echo "Hello Diverge" > newFile.txt
ttn@ttn:bootcamp (master)$ ls
file merge.txt newFile.txt readme
ttn@ttn:bootcamp (master)$ git add --all
ttn@ttn:bootcamp (master)$ git commit -m "Diverge added"
[master fdf6805] Diverge added
1 file changed, 1 insertion(+)
create mode 100644 newFile.txt
ttn@ttn:bootcamp (master)$ git checkout newBranch
Switched to branch 'newBranch'
ttn@ttn:bootcamp (newBranch)$ ls
file merge.txt readme
ttn@ttn:bootcamp (newBranch)$
```


Question15. Edit the same file at the same line on both branches and commit

```
Activities Terminal Tue 23:33 ttn@ttn: ~/bootcamp
File Edit View Search Terminal Help

ttn@ttn:~/bootcamp$ echo "some content to produce conflict">merge.txt
ttn@ttn:~/bootcamp$ git add merge.txt
ttn@ttn:~/bootcamp$ git commit -am "committing the initial content"
[master 567a9eb] committing the initial content
2 files changed, 2 insertions(+)
create mode 100644 merge.txt
ttn@ttn:~/bootcamp$ git checkout -b newBranch
Switched to a new branch 'newBranch'
ttn@ttn:~/bootcamp$ echo "totally different content to cause conflict">merge.txt
ttn@ttn:~/bootcamp$ git commit -am "edited the content"
[newBranch dd96ecc] edited the content
1 file changed, 1 insertion(+), 1 deletion(-)
ttn@ttn:~/bootcamp$ git checkout master
Switched to branch 'master'
ttn@ttn:~/bootcamp$ echo "content to append">>merge.txt
ttn@ttn:~/bootcamp$ git commit -am "appended to merge.txt"
[master 5eaa34f] appended to merge.txt
1 file changed, 1 insertion(+)
ttn@ttn:~/bootcamp$ git merge newBranch
Auto-merging merge.txt
CONFLICT (content): Merge conflict in merge.txt
Automatic merge failed; fix conflicts and then commit the result.
ttn@ttn:~/bootcamp$ git status
On branch master
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)

        both modified:   merge.txt

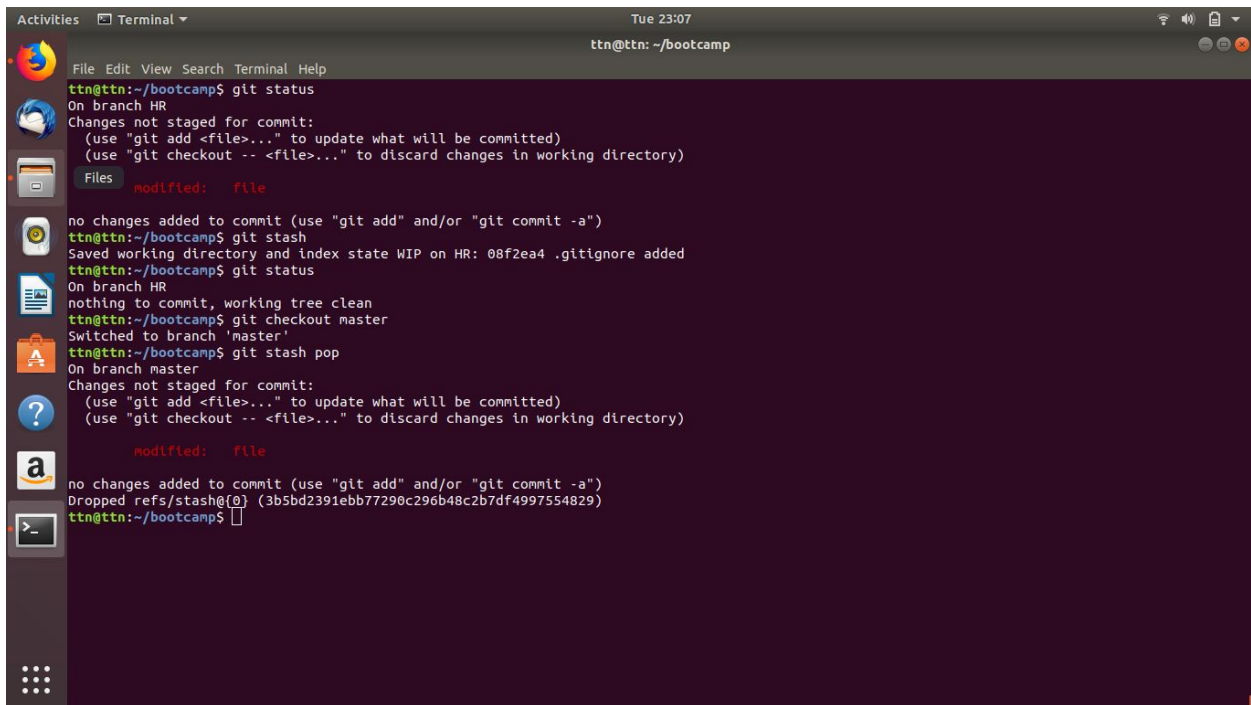
no changes added to commit (use "git add" and/or "git commit -a")
ttn@ttn:~/bootcamp$
```

Question16. Edit the same file at the same line on both branches and commit

```
Activities Terminal Tue 23:36 ttn@ttn: ~/bootcamp
File Edit View Search Terminal Help

ttn@ttn:~/bootcamp$ cat merge.txt
<<<<<< HEAD
some content to produce conflict
content to append
=====
totally different content to cause conflict
>>>>>> newBranch
ttn@ttn:~/bootcamp$ nano merge.txt
U
merge.txt
error: Committing is not possible because you have unmerged files.
hint: Fix them up in the work tree, and then use 'git add/rm <file>'
hint: as appropriate to mark resolution and make a commit.
fatal: Exiting because of an unresolved conflict.
ttn@ttn:~/bootcamp$ git add merge.txt
ttn@ttn:~/bootcamp$ git commit -m "merged and resolved"
[master 9397518] merged and resolved
ttn@ttn:~/bootcamp$ cat merge.txt
some content to produce conflict
content to append
totally different content to cause conflict
ttn@ttn:~/bootcamp$
```


Question17. Stash the changes and pop them



```
Activities Terminal Tue 23:07
ttn@ttn: ~/bootcamp

File Edit View Search Terminal Help
ttn@ttn:~/bootcamp$ git status
On branch HR
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

   modified:   file

no changes added to commit (use "git add" and/or "git commit -a")
ttn@ttn:~/bootcamp$ git stash
Saved working directory and index state WIP on HR: 08f2ea4 .gitignore added
ttn@ttn:~/bootcamp$ git status
On branch HR
nothing to commit, working tree clean
ttn@ttn:~/bootcamp$ git checkout master
Switched to branch 'master'
ttn@ttn:~/bootcamp$ git stash pop
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

   modified:   file

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (3b5bd2391ebb77290c296b48c2b7df4997554829)
ttn@ttn:~/bootcamp$
```

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

