

Q1. Git setup <https://confluence.atlassian.com/bitbucket/set-up-git-744723531.html>.

Ans

Git can be setup by the following code.

```
ttn@TTN:~$ sudo apt-get install git
[sudo] password for ttn:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  git-man liberror-perl
Suggested packages:
  git-daemon-run | git-daemon-sysvinit git-doc git-el git-email git-gui gitk
  gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  git git-man liberror-perl
0 upgraded, 3 newly installed, 0 to remove and 384 not upgraded.
Need to get 4,733 kB of archives.
After this operation, 33.9 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://in.archive.ubuntu.com/ubuntu bionic/main amd64 liberror-perl all
Get:2 http://in.archive.ubuntu.com/ubuntu bionic-updates/main amd64 git-man a
Get:3 http://in.archive.ubuntu.com/ubuntu bionic-updates/main amd64 git amd64
Fetched 4,733 kB in 2s (1,934 kB/s)
Selecting previously unselected package liberror-perl.
(Reading database ... 126213 files and directories currently installed.)
Preparing to unpack .../liberror-perl_0.17025-1_all.deb ...
```

```
ttn@TTN:~/learnings$ git config --global user.name "tayibah"
ttn@TTN:~/learnings$ git config --global user.email "tayibah.ayub@tothenew.com"
ttn@TTN:~/learnings$ git config user
error: key does not contain a section: user
ttn@TTN:~/learnings$ git config --global user.email "tayibah.ayub@tothenew.com"
ttn@TTN:~/learnings$ git config user
error: key does not contain a section: user
ttn@TTN:~/learnings$ git config user.name
tayibah
```

Q2. Initialize a Git Repository

Ans.

Git repository can be initialised

```
ttn@TTN:~/new$ git init
Initialized empty Git repository in /home/ttn/new/.git/
ttn@TTN:~/new$ $ git remote add origin git@github.com:geekycodi/tayibah
$: command not found
ttn@TTN:~/new$ git remote add origin git@github.com:geekycodi/tayibah
```

Q3.Add files to the repository

```
ttn@TTN:~/new$ touch test.txt
ttn@TTN:~/new$ vi test.txt
ttn@TTN:~/new$ git add test.txt
```

Q4. Unstage 1 file

Ans.

There are two ways to unstage a file:

1. Using the command

`git rm --cached filename`

This command makes git stop tracking the file completely (leaving it in the file system).

2. Using the command

`git reset HEAD -- filename`

```
ttn@TTN:~/new$ git reset test.txt
ttn@TTN:~/new$
```

Q5.Commit the file

Committing the file means saving the file but it does so by creating a unique id, hence each change is tracked.

```
ttn@TTN:~/new$ git commit
[master (root-commit) c76cc49] this is committed
1 file changed, 3 insertions(+)
```

Q6. Add a remote

To verify if remote repository has been added successfully or not, use the command:
git remote -v

```
ttn@TTN:~/new$ git remote add rem /home/ttn/new/.git/
ttn@TTN:~/new$ git remote -v

Command 'remote' not found, but can be installed with:

sudo snap install remote

ttn@TTN:~/new$ git remote -v
origin  git@github.com:geekycodi/tayibah (fetch)
origin  git@github.com:geekycodi/tayibah (push)
rem     /home/ttn/new/.git/ (fetch)
rem     /home/ttn/new/.git/ (push)
```

Q7. Undo changes to a particular file

After the file is added to staging, there are 4 ways to undo the local changes:

1. git reset HEAD <file>

This will unstage the file to current commit.

2. git reset

This will unstage all the files but changes will be retained.

3. git stash

This will discard all local changes, but they will be saved for later.

We stash the changes when we don't want to commit what we have been working on yet.

4. git reset --hard

To discard every change permanently.

Q8 Clone the repository

```
ttn@TTN:~/new$ git clone https://github.com/geekycodi/tayibah
Cloning into 'tayibah'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
ttn@TTN:~/new$
```

Q9 Push changes to Github

It moves the local directory into the server.

```
ttn@TTN:~/Desktop/learning/tayibah$ git add .
ttn@TTN:~/Desktop/learning/tayibah$ git commit -m "."
[master dcd6245] .
1 file changed, 0 insertions(+), 0 deletions(-)
delete mode 100644 Introduction to databases.pdf
ttn@TTN:~/Desktop/learning/tayibah$ git push origin master
Counting objects: 2, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 259 bytes | 259.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0)
To github.com:geekycodi/tayibah.git
72dedc8..dcd6245 master -> master
```


Q11. Check differences between a file and its staged version

```
ttn@TTN:~/learnings$ vi file1
ttn@TTN:~/learnings$ touch hello.txt
ttn@TTN:~/learnings$ vi hello.txt
ttn@TTN:~/learnings$ git status
On branch master
Your branch is based on 'origin/master', but the upstream is gone.
  (use "git branch --unset-upstream" to fixup)

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:   file1

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

        hello.txt
```

Q12. Ignore a few files to be checked in

```
ttn@TTN:~/Desktop/learning/tayibah$ cat >> .gitignore hello.txt
cat: hello.txt: No such file or directory
ttn@TTN:~/Desktop/learning/tayibah$ touch test.txt
ttn@TTN:~/Desktop/learning/tayibah$ vi test.txt
ttn@TTN:~/Desktop/learning/tayibah$ cat >> .gitignore test.txt
ttn@TTN:~/Desktop/learning/tayibah$ git status
On branch master
Your branch is up to date with 'origin/master'.

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

        .gitignore
        tayibah/
        test.txt

nothing added to commit but untracked files present (use "git add" to track)
```

Q13. Create a new branch.

Branch can be created by following :

```
@ [ttn]learnings -> git checkout -b tayib
Switched to a new branch 'tayib'
```

```
@ [ttn]learnings -> git branch
hello
master
* tayib
```

Q14. Diverge them with commits

```
@ [ttn]learnings -> git add hello.txt
__git_ps1: command not found
@ [ttn]learnings -> git commit
[tayib ec605d0] Commit
1 file changed, 3 insertions(+)
__git_ps1: command not found
@ [ttn]learnings -> git push origin tayib
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 320 bytes | 320.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects
To github.com:shobhitttn/learnings.git
7ee582f..ec605d0 tayib -> tayib
__git_ps1: command not found
@ [ttn]learnings -> git checkout hello
Switched to branch 'hello'
__git_ps1: command not found
@ [ttn]learnings -> git pull origin hello
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (1/1), done.
remote: Total 3 (delta 2), reused 3 (delta 2), pack-reused 0
Unpacking objects: 100% (3/3), done.
From github.com:shobhitttn/learnings
* branch hello -> FETCH_HEAD
7366b0a..48f4abd hello -> origin/hello
Updating 7366b0a..48f4abd
Fast-forward
 hello.txt | 2 +-
1 file changed, 1 insertion(+), 1 deletion(-)
```

Q15. Edit the same file at the same line on both branches and commit

Same as 14

Q16. Try merging and resolve merge conflicts

```
__git_ps1: command not found
@ [ttn]learnings -> git checkout tayib
Switched to branch 'tayib'
__git_ps1: command not found
@ [ttn]learnings -> git merge hello
Auto-merging hello.txt
CONFLICT (content): Merge conflict in hello.txt
Automatic merge failed; fix conflicts and then commit the result.
__git_ps1: command not found
```

```
@ [ttn]learnings -> vi hello.txt
__git_ps1: command not found
@ [ttn]learnings -> git add hello.txt
__git_ps1: command not found
@ [ttn]learnings -> git commit
[tayib a19d0f8] Merge branch 'hello' into tayib
__git_ps1: command not found
@ [ttn]learnings -> git merge
fatal: No remote for the current branch.
__git_ps1: command not found
@ [ttn]learnings -> git merge hello
Already up to date.
__git_ps1: command not found
@ [ttn]learnings -> git push origin hello
Everything up-to-date
__git_ps1: command not found
@ [ttn]learnings -> git merge hello
```

Q17. Stash the changes and pop them

```
@ [ttn]learnings -> git stash
Saved working directory and index state WIP on tayib: a19d0f8 Merge branch 'hello' into tayib
__git_ps1: command not found
@ [ttn]learnings ->
```



```

@ [ttn]learnings -> git stash pop
On branch tayib
You are currently rebasing branch 'master' on '0579323'.
  (all conflicts fixed: run "git rebase --continue")

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:   hello.txt

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

        .gitignore

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (959d27e989b2617f060329770dee9ced20bba195)
__git_ps1: command not found

```

Q18. 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[
else
PS1='\u@\h:\W $(parse_git_branch)\$ '
fi
unset color_prompt force_color_prompt

```

Ans

```

ttn@TTN:~/learnings$ parse_git_branch() {
> git branch 2> /dev/null | sed -e '/^[^*]/d' -e 's/* \(.*)/( \1)/'
> }
ttn@TTN:~/learnings$ if [ "$color_prompt" = yes ]; then
> PS1='${debian_chroot:+($debian_chroot)}\[\033[01;32m\]\u@\h\[\033[00m\]:\[\033[01;34m\]\w\[\033[01;31m\]$(parse_git_branch)'
> else
> PS1='${debian_chroot:+($debian_chroot)}\u@\h:\w$(parse_git_branch)\$ '
> fi
ttn@TTN:~/learnings(master)$ unset color_prompt force_color_prompt
ttn@TTN:~/learnings(master)$ RED="\e[0;31m"
ttn@TTN:~/learnings(master)$ GREEN="\e[0;92m"
ttn@TTN:~/learnings(master)$ BLACK="\e[m"
ttn@TTN:~/learnings(master)$ YELLOW="\e[0;93m"
ttn@TTN:~/learnings(master)$ export PS1='\[\e[0;96m\]@ [\[\e[0;94m\]\u\[\e[0;96m\]]\W\[\e[m\]\
> $(echo ${__git_ps1} "\[\'$GREEN'\] git:(\[\'$RED'\]%s\[\'$GREEN'\]\)")" \[\'$YELLOW'\]->\[\'$BLACK'\] '
__git_ps1: command not found

```



```
Activities Terminal Thu 22:09 ttn@TTN: ~/learnings
File Edit View Search Terminal Help
364 history
__git_ps1: command not found
@ [ttn]learnings -> git rebase --continue
hello.txt: needs update
You must edit all merge conflicts and then
mark them as resolved using git add
__git_ps1: command not found
@ [ttn]learnings -> git status
rebase in progress; onto 0579323
You are currently rebasing branch 'master' on '0579323'.
(all conflicts fixed: run "git rebase --continue")

Changes to be committed:
(use "git reset HEAD <file>..." to unstage)

    modified:   file1
    modified:   hello.txt

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:   hello.txt

__git_ps1: command not found
@ [ttn]learnings -> git rebase --continue
hello.txt: needs update
You must edit all merge conflicts and then
mark them as resolved using git add
__git_ps1: command not found
@ [ttn]learnings -> vi hello.txt
__git_ps1: command not found
@ [ttn]learnings -> git add hello.txt
__git_ps1: command not found
@ [ttn]learnings -> git status
rebase in progress; onto 0579323
You are currently rebasing branch 'master' on '0579323'.
(all conflicts fixed: run "git rebase --continue")
```