

Exercise

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

```
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 429 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 0.17025-1 [22.8 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu bionic-updates/main amd64 git-man all 1:2.17.1-1ubuntu0.4 [803 kB]
Get:3 http://in.archive.ubuntu.com/ubuntu bionic-updates/main amd64 git amd64 1:2.17.1-1ubuntu0.4 [3,907 kB]
Fetched 4,733 kB in 3s (1,649 kB/s)
Selecting previously unselected package liberror-perl.
(Reading database ... 128539 files and directories currently installed.)
Preparing to unpack .../liberror-perl_0.17025-1_all.deb ...
Unpacking liberror-perl (0.17025-1) ...
Selecting previously unselected package git-man.
Preparing to unpack .../git-man_1%3a2.17.1-1ubuntu0.4_all.deb ...
Unpacking git-man (1:2.17.1-1ubuntu0.4) ...
Selecting previously unselected package git.
Preparing to unpack .../git_1%3a2.17.1-1ubuntu0.4_amd64.deb ...
Unpacking git (1:2.17.1-1ubuntu0.4) ...
Setting up git-man (1:2.17.1-1ubuntu0.4) ...
Setting up liberror-perl (0.17025-1) ...
Processing triggers for man-db (2.8.3-2) ...
Setting up git (1:2.17.1-1ubuntu0.4) ...
ttn@TTN:~$ git --version
git version 2.17.1
ttn@TTN:~$
```

2. Initialize a Git Repository

```
ttn@ttn:~/Assignments$ cd Feb05_Git/
ttn@ttn:~/Assignments/Feb05_Git$ git init
Initialized empty Git repository in /home/ttn/Assignments/Feb05_Git/.git/
ttn@ttn:~/Assignments/Feb05_Git$
```

3. Add files to the repository

```
Initialized empty Git repository in /home/ttn/Assignments/Feb05_Git/.git/
ttn@ttn:~/Assignments/Feb05_Git$ git add .
ttn@ttn:~/Assignments/Feb05_Git$
```

4. Unstage 1 file

```
ttn@ttn:~/Assignments$ git status
On branch master
Your branch is up to date with 'origin/master'.

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

        renamed:    Feb04/Exercise.pdf -> Feb04_Linux/Exercise.pdf
        renamed:    Feb04/exercise/dir1/dir2/emptyFile1 -> Feb04_Linux/exercise/dir1/dir2/emptyFile1
        renamed:    Feb04/exercise/dir1/dir2/emptyFile2 -> Feb04_Linux/exercise/dir1/dir2/emptyFile2
        renamed:    Feb04/exercise/dir1/dir2/file1.txt -> Feb04_Linux/exercise/dir1/dir2/file1.txt
        renamed:    Feb04/exercise/output -> Feb04_Linux/exercise/output
        renamed:    Feb04/exercise/passwd_backup -> Feb04_Linux/exercise/passwd_backup
        renamed:    Feb04/exercise/tar_ball -> Feb04_Linux/exercise/tar_ball
        new file:   Feb05_Git/file1

ttn@ttn:~/Assignments$ git reset HEAD Feb05_Git/file1
ttn@ttn:~/Assignments$ git status
On branch master
Your branch is up to date with 'origin/master'.

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

        renamed:    Feb04/Exercise.pdf -> Feb04_Linux/Exercise.pdf
        renamed:    Feb04/exercise/dir1/dir2/emptyFile1 -> Feb04_Linux/exercise/dir1/dir2/emptyFile1
        renamed:    Feb04/exercise/dir1/dir2/emptyFile2 -> Feb04_Linux/exercise/dir1/dir2/emptyFile2
        renamed:    Feb04/exercise/dir1/dir2/file1.txt -> Feb04_Linux/exercise/dir1/dir2/file1.txt
        renamed:    Feb04/exercise/output -> Feb04_Linux/exercise/output
        renamed:    Feb04/exercise/passwd_backup -> Feb04_Linux/exercise/passwd_backup
        renamed:    Feb04/exercise/tar_ball -> Feb04_Linux/exercise/tar_ball

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

        Feb05_Git/

ttn@ttn:~/Assignments$
```

5. Commit the file

```
nothing added to commit but untracked files present (use "git add" to track)
ttn@ttn:~/Assignments/Feb05_Git$ git add .
ttn@ttn:~/Assignments/Feb05_Git$ git commit -m "File added."
[master 6899e08] File added.
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 Feb05_Git/file2
ttn@ttn:~/Assignments/Feb05_Git$
```

6. Add a remote

```
Set up remote as a mirror to push to or fetch from
ttn@ttn:~/Assignments$ git remote add origin https://github.com/Yatinajmani/Assignments.git
ttn@ttn:~/Assignments$
```

7. Undo changes to a particular file

```
added text
C
ttn@ttn:~/Assignments/Feb05_Git$ 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:   file1

no changes added to commit (use "git add" and/or "git commit -a")
ttn@ttn:~/Assignments/Feb05_Git$ git checkout -- file
error: pathspec 'file' did not match any file(s) known to git.
ttn@ttn:~/Assignments/Feb05_Git$ git checkout -- file1
ttn@ttn:~/Assignments/Feb05_Git$ git status
On branch master
nothing to commit, working tree clean
ttn@ttn:~/Assignments/Feb05_Git$
```

8. Push changes to Github

```
ttn@ttn:~/Assignments/Feb05_Git$ git push -u origin master
Username for 'https://github.com': yatinajmani
Password for 'https://yatinajmani@github.com':
Counting objects: 7, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (5/5), done.
Writing objects: 100% (7/7), 627 bytes | 627.00 KiB/s, done.
Total 7 (delta 0), reused 0 (delta 0)
To https://github.com/Yatinajmani/Assignments.git
   54bad9f..6899e08  master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

9. Clone the repository

```
y: Command not found
ttn@ttn:~$ git clone git@github.com:Yatinajmani/Assignments.git
Cloning into 'Assignments'...
remote: Enumerating objects: 19, done.
remote: Counting objects: 100% (19/19), done.
remote: Compressing objects: 100% (13/13), done.
remote: Total 19 (delta 0), reused 19 (delta 0), pack-reused 0
Receiving objects: 100% (19/19), 2.95 MiB | 1.59 MiB/s, done.
ttn@ttn:~$ cd Assignments/
```

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

```
ttn@ttn:~/exercise/Assignments$ git add Feb05_Git/file2
ttn@ttn:~/exercise/Assignments$ git commit -m "Checking Pulling."
[master 70e95d8] Checking Pulling.
1 file changed, 1 insertion(+)
ttn@ttn:~/exercise/Assignments$ git push
counting objects: 4, done.
delta compression using up to 4 threads.
compressing objects: 100% (3/3), done.
writing objects: 100% (4/4), 372 bytes | 372.00 KiB/s, done.
total 4 (delta 0), reused 0 (delta 0)
to github.com:Yatinajmani/Assignments.git
6899e08..70e95d8 master -> master
ttn@ttn:~/exercise/Assignments$ cd ..
ttn@ttn:~/exercise$ cd..
cd..: command not found
ttn@ttn:~/exercise$ cd..
cd..: command not found
ttn@ttn:~/exercise$ cd ..
ttn@ttn:~$ git Assignments/
git: 'Assignments/' is not a git command. See 'git --help'.
ttn@ttn:~$ cd Assignments/
ttn@ttn:~/Assignments$ git pull
remote: Enumerating objects: 7, done.
remote: Counting objects: 100% (7/7), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 0), reused 4 (delta 0), pack-reused 0
Unpacking objects: 100% (4/4), done.
from github.com:Yatinajmani/Assignments
6899e08..70e95d8 master -> origin/master
Updating 6899e08..70e95d8
Fast-forward
 Feb05_Git/file2 | 1 +
1 file changed, 1 insertion(+)
ttn@ttn:~/Assignments$
```

11. Check differences between a file and its staged version

```
ttn@ttn:~/Assignments/Feb05_Git$ cat>>file2
Checking Diff
^C
ttn@ttn:~/Assignments/Feb05_Git$ git diff
diff --git a/Feb05_Git/file2 b/Feb05_Git/file2
index ebb0328..b4c9f4a 100644
--- a/Feb05_Git/file2
+++ b/Feb05_Git/file2
@@ -1,2 @@
+added text for pulling
+Checking Diff
ttn@ttn:~/Assignments/Feb05_Git$
```

12. Ignore a few files to be checked in

```
HEAD is now at 70e95d8 Checking Pulling.
ttn@ttn:~/Assignments$ cat>>.gitignore
*.class
^C
ttn@ttn:~/Assignments$ cd Feb05_Git/
ttn@ttn:~/Assignments/Feb05_Git$ cat>file1.class
something ignored^C
usage: cat: too many arguments
ttn@ttn:~/Assignments$ git add .
ttn@ttn:~/Assignments$ git status
On branch master
Your branch is up to date with 'origin/master'.

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

        new file:   .gitignore

ttn@ttn:~/Assignments$ git commit -m "Checking gitignore."
[master 2fa1a04] Checking gitignore.
1 file changed, 1 insertion(+)
create mode 100644 .gitignore
ttn@ttn:~/Assignments$
```


13. Create a new branch.

```
create mode 100644 .gitignore
ttn@ttn:~/Assignments$ git checkout -b newBranch
Switched to a new branch 'newBranch'
ttn@ttn:~/Assignments$
```

14. Diverge them with commits

```
ttn@ttn:~/Assignments$ cd Feb05_Git/
ttn@ttn:~/Assignments/Feb05_Git$ cat>file3
File on new branch to diverge branches^C
ttn@ttn:~/Assignments/Feb05_Git$ cd ..
ttn@ttn:~/Assignments$ git add Feb05_Git/file3
ttn@ttn:~/Assignments$ git commit -m "Diverding branches."
[newBranch 0c8807f] Diverding branches.
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Feb05_Git/file3
ttn@ttn:~/Assignments$
```

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

```
already on newBranch
ttn@ttn:~/Assignments$ git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)
ttn@ttn:~/Assignments$ cd Feb05_Git/
ttn@ttn:~/Assignments/Feb05_Git$ cat>file3
making conflicts^C
ttn@ttn:~/Assignments/Feb05_Git$ cd ..
ttn@ttn:~/Assignments$ git add Feb05_Git/file3
ttn@ttn:~/Assignments$ git commit -m "Adding merge conflict."
[master 42e514b] Adding merge conflict.
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Feb05_Git/file3
ttn@ttn:~/Assignments$
```

16. Try merging and resolve merge conflicts

```
making conflicts
ttn@ttn:~/Assignments$ git merge newBranch
Auto-merging Feb05_Git/file3
CONFLICT (add/add): Merge conflict in Feb05_Git/file3
Automatic merge failed; fix conflicts and then commit the result.
ttn@ttn:~/Assignments$ cat Feb05_Git/file3
<<<<<<< HEAD
making conflicts
=====
File on new Branch to diverge branches
>>>>>>> newBranch
ttn@ttn:~/Assignments$ vi Feb05_Git/file3
ttn@ttn:~/Assignments$ cat Feb05_Git/file3
Resolved Conflicts
ttn@ttn:~/Assignments$ git add Feb05_Git/file3
ttn@ttn:~/Assignments$ git merge --continue
[master 87e4817] Merge branch 'newBranch'
ttn@ttn:~/Assignments$
```

17. Stash the changes and pop them

```
[master 87e4817] Merge branch 'newBranch'
ttn@ttn:~/Assignments$ cat >>Feb05_Git/file3
added some text to stash
^C
ttn@ttn:~/Assignments$ cat Feb05_Git/file3
Resolved Conflicts
added some text to stash
ttn@ttn:~/Assignments$ git status
On branch master
Your branch is ahead of 'origin/master' by 5 commits.
(use "git push" to publish your local commits)

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:   Feb05_Git/file3

no changes added to commit (use "git add" and/or "git commit -a")
ttn@ttn:~/Assignments$ git stash
Saved working directory and index state WIP on master: 87e4817 Merge branch 'newBranch'
ttn@ttn:~/Assignments$ git status
On branch master
Your branch is ahead of 'origin/master' by 5 commits.
(use "git push" to publish your local commits)

nothing to commit, working tree clean
ttn@ttn:~/Assignments$ git stash pop
On branch master
Your branch is ahead of 'origin/master' by 5 commits.
(use "git push" to publish your local commits)

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:   Feb05_Git/file3

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (658895b72839e1957d842d0d98af0fed8ceba277)
ttn@ttn:~/Assignments$ git status
On branch master
Your branch is ahead of 'origin/master' by 5 commits.
(use "git push" to publish your local commits)

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:   Feb05_Git/file3

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

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
```

```
ttn@ttn: ~/Assignments
ttn@yatin:Assignments (master)$
```