

Git/Vim Assignments Module -1

Assignment 1:

1. Commit a repository with a single file change.

```
MINGW64/c/Users/Admin/Desktop/GIT DEMO FILES/dir1

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ mkdir dir1

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ cd dir1

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir1 (master)
$ echo "hello world" >> file.txt

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir1 (master)
$ git config --global -l
user.email=vrshasolachi@gmail.com
user.name=Vrsha

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir1 (master)
$ git init
Initialized empty Git repository in C:/Users/Admin/Desktop/GIT DEMO FILES/dir1/.git/

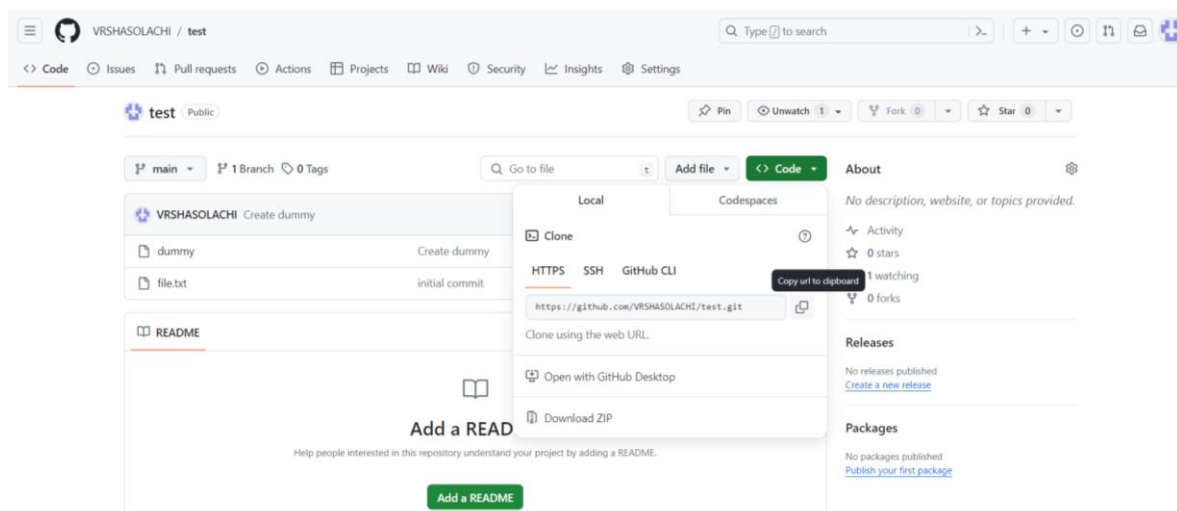
Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir1 (master)
$ git add .
warning: in the working copy of 'file.txt', LF will be replaced by CRLF the next time Git touches it

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir1 (master)
$ git commit -m "initial commit"
[master (root-commit) b125418] initial commit
1 file changed, 1 insertion(+)
create mode 100644 file.txt

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir1 (master)
$ git status
On branch master
nothing to commit, working tree clean

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir1 (master)
$
```

2. Create 2 new directories in local PC and clone the repository on both directories.



MINGW64:/c:/Users/Admin/Desktop/GIT DEMO FILES/dir3

```
Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ mkdir dir1

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ cd dir1

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir1 (master)
$ echo "hello world" >> file.txt

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir1 (master)
$ git config --global -l
user.email=vrshasolachi@gmail.com
user.name=Vrsha

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir1 (master)
$ git init
Initialized empty Git repository in C:/Users/Admin/Desktop/GIT DEMO FILES/dir1/.git/

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir1 (master)
$ git add .
warning: in the working copy of 'file.txt', LF will be replaced by CRLF the next time Git touches it

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir1 (master)
$ git commit -m "initial commit"
[master (root-commit) b125418] initial commit
1 file changed, 1 insertion(+)
create mode 100644 file.txt

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir1 (master)
$ git status
On branch master
nothing to commit, working tree clean

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir1 (master)
$ mkdir dir2 dir2
mkdir: cannot create directory 'dir2': File exists

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir1 (master)
$ cd ..

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ mkdir dir2 dir3

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ cd dir2

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir2 (master)
$ git clone https://github.com/VRSHASOLACHI/test.git
Cloning into 'test'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 6 (delta 0), reused 3 (delta 0), pack-reused 0
Receiving objects: 100% (6/6), done.
```

```
Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir2 (master)
$ cd ..

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ cd dir3

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES/dir3 (master)
$ git clone https://github.com/VRSHASOLACHI/test.git
Cloning into 'test'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 6 (delta 0), reused 3 (delta 0), pack-reused 0
Receiving objects: 100% (6/6), done.
```

3. Make changes in one of the directories and commit and merge the changes.

```
MINGW64:/c/Users/Admin/Desktop/GIT DEMO FILES

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (feature-a)
$ git checkout master
Switched to branch 'master'
D      demol.txt
Your branch is up to date with 'origin/master'.

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ git checkout -b feature-b
Switched to a new branch 'feature-b'

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (feature-b)
$ git checkout master
Switched to branch 'master'
D      demol.txt
Your branch is up to date with 'origin/master'.

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ git branch
  feature-a
  feature-b
* master

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ git merge feature-a
Already up to date.

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ git merge feature-b
Already up to date.

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ git checkout feature-a
Switched to branch 'feature-a'
D      demol.txt

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (feature-a)
$ mkdir dir1

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (feature-a)
$ echo "This file is at feature-a" >> a.txt

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (feature-a)
$ git checkout feature-b
Switched to branch 'feature-b'
D      demol.txt
A      dir1

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (feature-b)
$ echo "This file is at feature-b" >> b.txt

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (feature-b)
$ git checkout -b feature-c
Switched to a new branch 'feature-c'

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (feature-c)
$ git checkout master
Switched to branch 'master'
```

MINGW64:/c/Users/Admin/Desktop/GIT DEMO FILES

```
Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (feature-c)
$ git checkout master
Switched to branch 'master'
D      demo1.txt
A      dir1
Your branch is up to date with 'origin/master'.

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ git add .
warning: in the working copy of 'a.txt', LF will be replaced by CRLF the next ti
me Git touches it
warning: in the working copy of 'b.txt', LF will be replaced by CRLF the next ti
me Git touches it

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ git add .
warning: in the working copy of 'b.txt', LF will be replaced by CRLF the next ti
me Git touches it

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ git commit -m "added now"
[master 79a0051] added now
4 files changed, 3 insertions(+), 1 deletion(-)
create mode 100644 a.txt
create mode 100644 b.txt
delete mode 100644 demo1.txt
create mode 160000 dir1
```

3. Parallely make a change in the same file in another directory and when the changes are pushed, there should be merge conflict noticed which should be resolved and pushed changes again.

```
Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ git checkout feature-c
Switched to branch 'feature-c'

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (feature-c)
$ git add .
warning: in the working copy of 'b.txt', LF will be replaced by CRLF the next ti
me Git touches it

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (feature-c)
$ git commit -m "added here as well"
[feature-c 7822e30] added here as well
1 file changed, 1 insertion(+)
create mode 100644 b.txt

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (feature-c)
$ git checout master
git: 'checout' is not a git command. See 'git --help'.

The most similar command is
    checkout

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (feature-c)
$ 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)

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
```

MINGW64:/c/Users/Admin/Desktop/GIT DEMO FILES

warning: in the working copy of 'b.txt', LF will be replaced by CRLF the next time Git touches it

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (feature-c)

```
$ git commit -m "added here as well"
[feature-c 7822e30] added here as well
1 file changed, 1 insertion(+)
create mode 100644 b.txt
```

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (feature-c)

```
$ git checout master
git: 'checout' is not a git command. See 'git --help'.
```

The most similar command is
checkout

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (feature-c)

```
$ 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)
```

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)

```
$ git push -u origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (4/4), 358 bytes | 358.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/VRSHASOLACHI/DEMO1
   bd4620a..79a0051 master -> master
branch 'master' set up to track 'origin/master'.
```

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)

```
$ git merge feature-c
Auto-merging b.txt
CONFLICT (add/add): Merge conflict in b.txt
Automatic merge failed; fix conflicts and then commit the result.
```

MINGW64:/c/Users/Admin/Desktop/GIT DEMO FILES

Merge branch 'feature-c'

```
# Conflicts:
#   b.txt
#
# It looks like you may be committing a merge.
# If this is not correct, please run
#   git update-ref -d MERGE_HEAD
# and try again.
#
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# Your branch is up to date with 'origin/master'.
#
# All conflicts fixed but you are still merging.
#
# Changes to be committed:
#   modified:   b.txt
#
```

.git/COMMIT_EDITMSG [unix] (09:31 07/06/2024)

~/Desktop/GIT DEMO FILES/ .git/COMMIT_EDITMSG" [unix] 221 491B

4. Merge the changes

```
Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ git merge feature-c
Auto-merging b.txt
CONFLICT (add/add): Merge conflict in b.txt
Automatic merge failed; fix conflicts and then commit the result.

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master|MERGING)
$ git add .
warning: in the working copy of 'b.txt', LF will be replaced by CRLF the next time Git touches it

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master|MERGING)
$ git commit
[master 617449f] Merge branch 'feature-c'

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$ git log --oneline
617449f (HEAD -> master) Merge branch 'feature-c'
7822e30 (feature-c) added here as well
79a0051 (origin/master) added now
bd4620a (feature-b, feature-a) first commit

Admin@HP MINGW64 ~/Desktop/GIT DEMO FILES (master)
$
```

Assignment 2:

Create a story.txt file using below lines, save and quit the file

=====

The quick brown fox jumps over the lazy dog.

The quick brown fox is fast and clever.

Lazy dogs often lie in the sun all day.

The dog chased the fox through the field.

A quick fox can outsmart a lazy dog.

Dogs are loyal and friendly animals.

Foxes are cunning and quick in nature.

The field was green and full of life.

The fox found food in the field.

Dogs and foxes rarely become friends.

=====

Open story.txt file and do the following,

1. Replace fox with wolf, save and quit the file. Open and see if the changes get reflected.

```
1 The quick brown fox jumps over the lazy dog.
2 The quick brown fox is fast and clever.
3 Lazy dogs often lie in the sun all day.
4 The dog chased the fox through the field.
5 A quick fox can outsmart a lazy dog.
6 Dogs are loyal nd friendly animals.
7 Foxes are cunning and quick in nature.
8 The field was green and full of life.
9 The fox found food in the field.
10 Dogs and foxes rarely becomes friends.
```

%s/fox/wolf/g

```
1 The quick brown wolf jumps over the lazy dog.
2 The quick brown wolf is fast and clever.
3 Lazy dogs often lie in the sun all day.
4 The dog chased the wolf through the field.
5 A quick wolf can outsmart a lazy dog.
6 Dogs are loyal nd friendly animals.
7 Foxes are cunning and quick in nature.
8 The field was green and full of life.
9 The wolf found food in the field.
10 Dogs and wolves rarely becomes friends.
```

substitutions on 6 lines

10,1

All

```
1 The quick brown wolf jumps over the lazy dog.
2 The quick brown wolf is fast and clever.
3 Lazy dogs often lie in the sun all day.
4 The dog chased the wolf through the field.
5 A quick wolf can outsmart a lazy dog.
6 Dogs are loyal nd friendly animals.
7 Foxes are cunning and quick in nature.
8 The field was green and full of life.
9 The wolf found food in the field.
10 Dogs and wolves rarely becomes friends.
```

wq

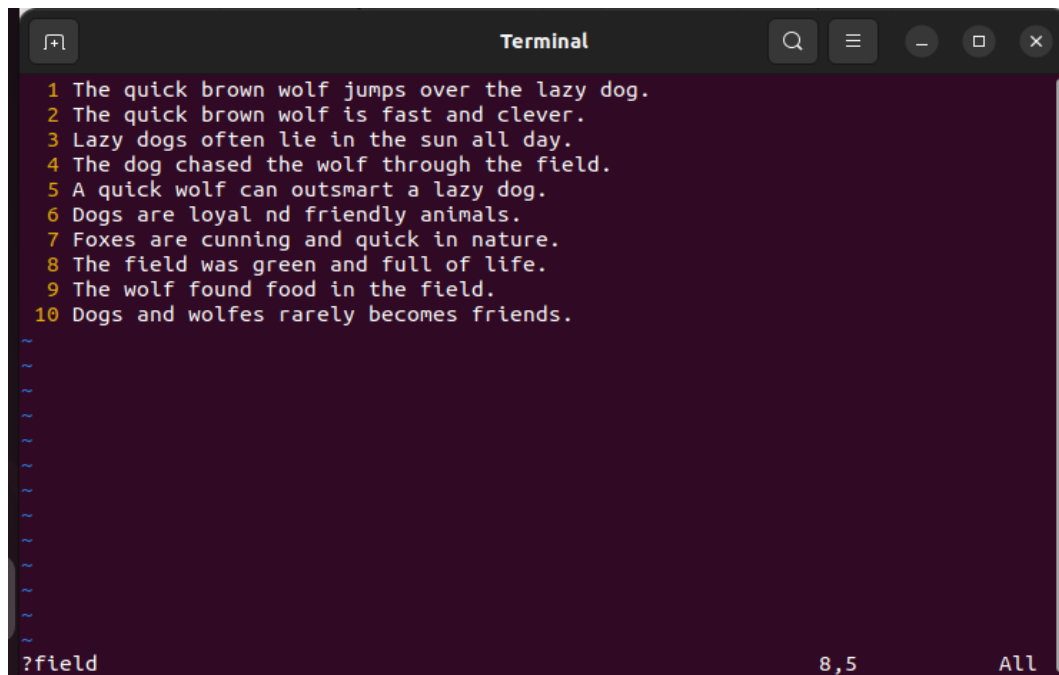
2. Search for string "field" in the file and move to next and prev occurrence of the string.

Move to next occurrence:

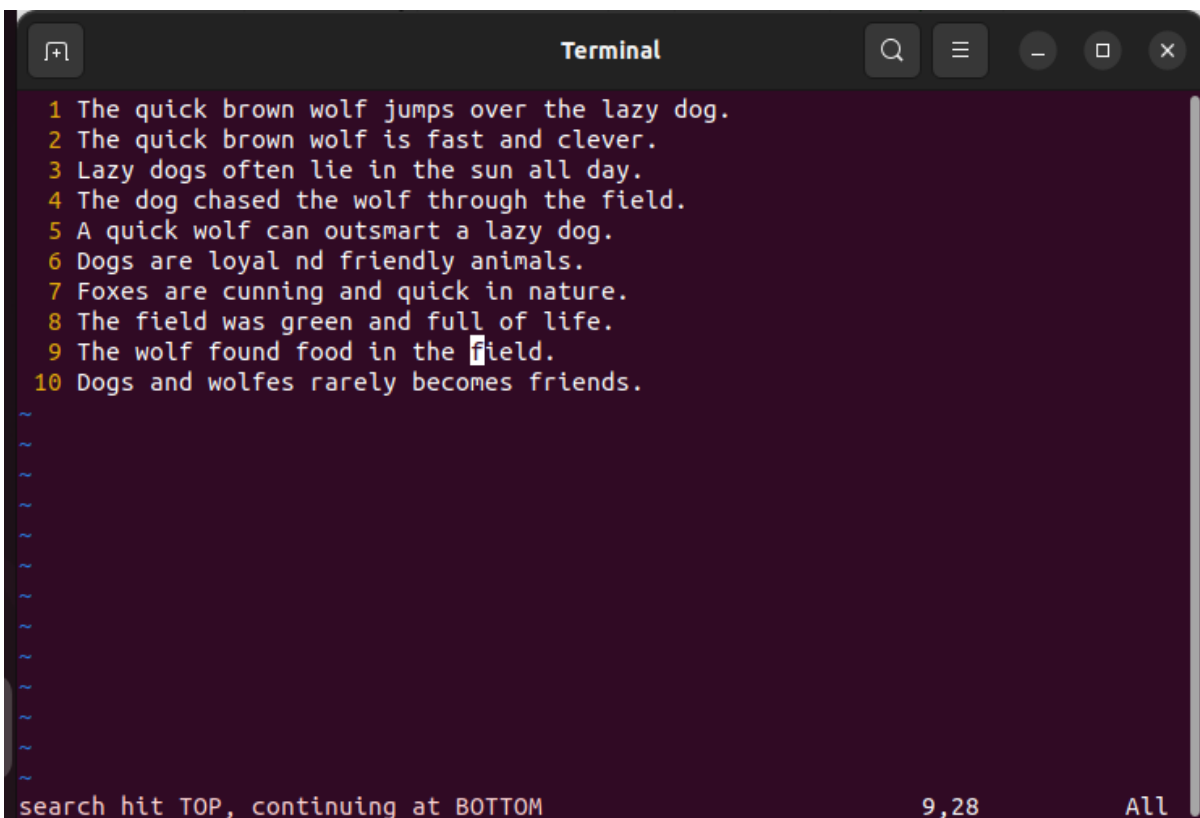
n

Move to previous occurrence:

N



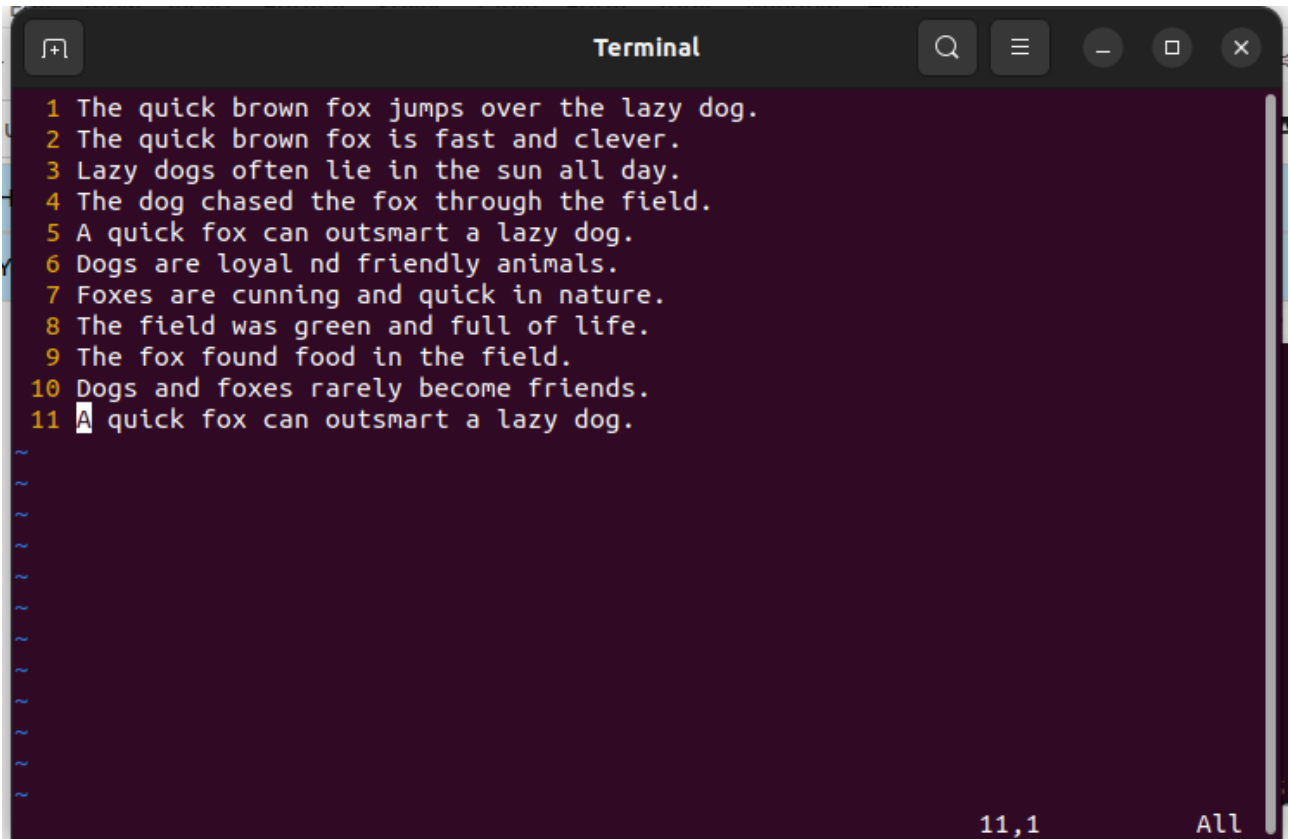
```
Terminal
1 The quick brown wolf jumps over the lazy dog.
2 The quick brown wolf is fast and clever.
3 Lazy dogs often lie in the sun all day.
4 The dog chased the wolf through the field.
5 A quick wolf can outsmart a lazy dog.
6 Dogs are loyal nd friendly animals.
7 Foxes are cunning and quick in nature.
8 The field was green and full of life.
9 The wolf found food in the field.
10 Dogs and wolfes rarely becomes friends.
~
~
~
~
~
~
~
~
~
~
?field 8,5 All
```



```
Terminal
1 The quick brown wolf jumps over the lazy dog.
2 The quick brown wolf is fast and clever.
3 Lazy dogs often lie in the sun all day.
4 The dog chased the wolf through the field.
5 A quick wolf can outsmart a lazy dog.
6 Dogs are loyal nd friendly animals.
7 Foxes are cunning and quick in nature.
8 The field was green and full of life.
9 The wolf found food in the field.
10 Dogs and wolfes rarely becomes friends.
~
~
~
~
~
~
~
~
~
~
search hit TOP, continuing at BOTTOM 9,28 All
```


3. Copy the entire line "A quick fox can outsmart a lazy dog" and paste at the end of file and save it.

yy, G, P



```
Terminal
1 The quick brown fox jumps over the lazy dog.
2 The quick brown fox is fast and clever.
3 Lazy dogs often lie in the sun all day.
4 The dog chased the fox through the field.
5 A quick fox can outsmart a lazy dog.
6 Dogs are loyal nd friendly animals.
7 Foxes are cunning and quick in nature.
8 The field was green and full of life.
9 The fox found food in the field.
10 Dogs and foxes rarely become friends.
11 A quick fox can outsmart a lazy dog.
~
~
~
~
~
~
~
~
~
~
11,1 All
```

4. Replace all the vowels with letter "x" in the file.

```
:%s/[aeiou]/x/g
```

```

1 Thx qxxck brxwn fxx jmps xvvr thx lxzy dxg.
2 Thx qxxck brxwn fxx xs fxst xnd clxvvr.
3 Lxzy dxgs xftxn lxx xn thx sxn xll dxy.
4 Thx dxg chxsxd thx fxx thrxxgh thx fxxld.
5 A qxxck fxx cxn xxtsmxrt x lxzy dxg.
6 Dxgs xrx lxyxl nd frxxndly xnxmxls.
7 Fxxxs xrx cxnnxng xnd qxxck xn ntxtxrx.
8 Thx fxxld wxs grxxn xnd fxll xf lxfx.
9 Thx fxx fxxnd fxxd xn thx fxxld.
10 Dxgs xnd fxxxs rxrxly bxcxmx frxxnds.
11 A qxxck fxx cxn xxtsmxrt x lxzy dxg.
~
~
~
~
~
~
~
~
~
~
~
118 substitutions on 11 lines

```