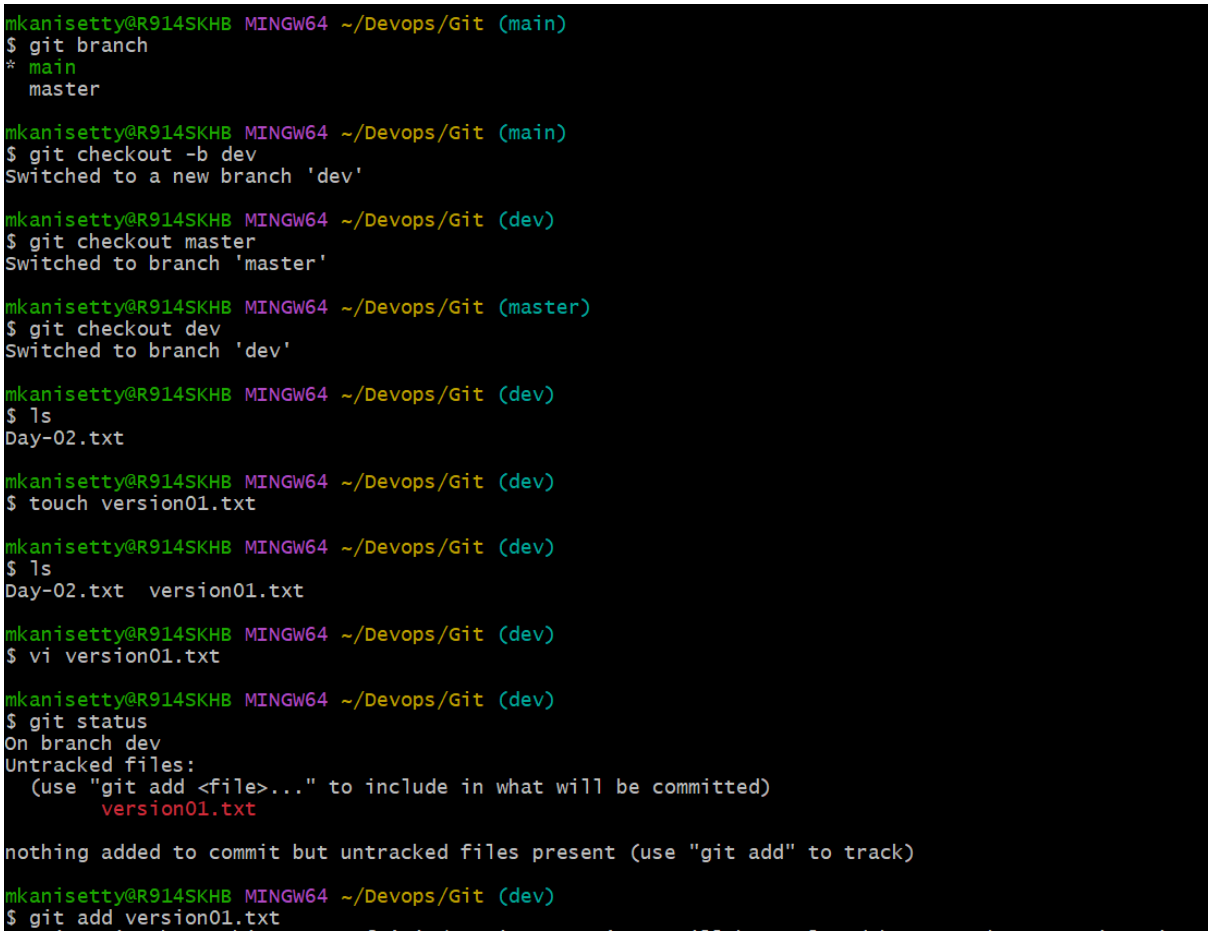
Task 1:

Create branch "dev" from master, add "version01.txt" with "This is first feature of our application", commit as "Added new feature", push to remote. Add "This is the bug fix in development branch" with commit message "Added feature2 in development branch", "This is gadbad code" with commit message "Added feature3 in development branch", "This feature will gadbad everything from now" with commit message "Added feature4 in development branch". Restore "version01.txt" to "This is the bug fix in development branch" using git revert/reset.

Below are the screenshots of these task, which will help you to complete this task.



```
mkanisetty@R914SKHB MINGW64 ~/Devops/Git (dev)
$ cat version01.txt
This is first feature of our application

This is the bug fix in development branch

This is gadbad code

This feature will gadbad everything from now



mkanisetty@R914SKHB MINGW64 ~/Devops/Git (dev)
$ git status
On branch dev
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   version01.txt

no changes added to commit (use "git add" and/or "git commit -a")

mkanisetty@R914SKHB MINGW64 ~/Devops/Git (dev)
$ git add version01.txt

mkanisetty@R914SKHB MINGW64 ~/Devops/Git (dev)
$ git commit -m "added new feature"
[dev 94f018f] added new feature
 1 file changed, 4 insertions(+)
```

```
mkanisetty@R914SKHB MINGW64 ~/Devops/Git (dev)
$ git revert HEAD~2..HEAD
[dev 6964f3d] Revert "added new feature"
 1 file changed, 4 deletions(-)
[dev 88043c0] Reapply "added feature2 in development branch"
 1 file changed, 1 insertion(+), 1 deletion(-)

mkanisetty@R914SKHB MINGW64 ~/Devops/Git (dev)
$ cat version01.txt
This is first feature of our application

This is the bug fix in development branch

This is gadbad code

mkanisetty@R914SKHB MINGW64 ~/Devops/Git (dev)
$ git revert HEAD~1..HEAD
[dev 11856a0] Revert "Reapply "added feature2 in development branch""
 1 file changed, 1 insertion(+), 1 deletion(-)

mkanisetty@R914SKHB MINGW64 ~/Devops/Git (dev)
$ cat version01.txt
This is first feature of our application

This is the bug fix in development branch
```

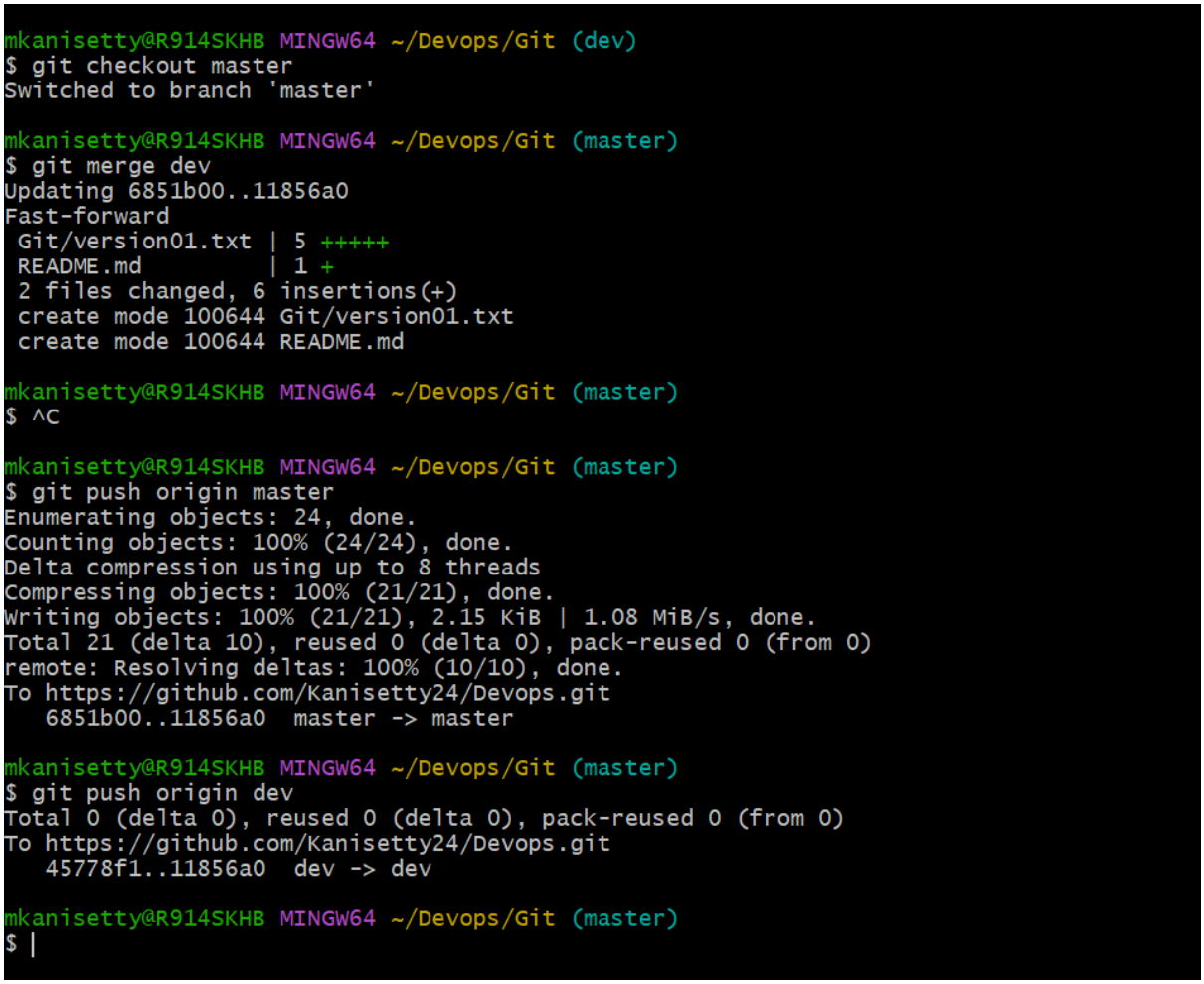
Task 2:

A.Demonstrate the concept of branches with 2 or more branches with screenshot.

add some changes to dev branch and merge that branch in master

B. As a practice try git rebase too, see what difference you get.

In above task I already made some changes in dev branch, now am merge that with master branch.



Now am trying git Rebase also

```
mkanisetty@R914SKHB MINGW64 ~/Devops/Git (dev)
$ ls
Day-02.txt  cat  version01.txt

mkanisetty@R914SKHB MINGW64 ~/Devops/Git (dev)
$ cat Day-02.txt
This is a file in my local repository

Now lets see whatever changes i did will i be able to push it to remote repository.

mkanisetty@R914SKHB MINGW64 ~/Devops/Git (dev)
$ vi Day-02.txt

mkanisetty@R914SKHB MINGW64 ~/Devops/Git (dev)
$ cat Day-02.txt
This is a file in my local repository

Now lets see whatever changes i did will i be able to push it to remote repository.

Let's try git rebase

mkanisetty@R914SKHB MINGW64 ~/Devops/Git (dev)
$ git status
On branch dev
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   Day-02.txt

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

no changes added to commit (use "git add" and/or "git commit -a")

mkanisetty@R914SKHB MINGW64 ~/Devops/Git (dev)
$ git add Day-02.txt

mkanisetty@R914SKHB MINGW64 ~/Devops/Git (dev)
$ git commit -m "modified the file Day-02.txt"
[dev 6be5975] modified the file Day-02.txt
 1 file changed, 2 insertions(+)
```

```
mkanisetty@R914SKHB MINGW64 ~/Devops/Git (dev)
$ git checkout master
Switched to branch 'master'

mkanisetty@R914SKHB MINGW64 ~/Devops/Git (master)
$ git rebase dev
Successfully rebased and updated refs/heads/master.

mkanisetty@R914SKHB MINGW64 ~/Devops/Git (master)
$ ls
Day-02.txt  cat  version01.txt

mkanisetty@R914SKHB MINGW64 ~/Devops/Git (master)
$ cat Day-02.txt
This is a file in my local repository

Now lets see whatever changes i did will i be able to push it to remote repository.

Let's try git rebase
```

```
mkanisetty@R914SKHB MINGW64 ~/Devops/Git (master)
$ git log --oneline
6be5975 (HEAD -> master, dev) modified the file Day-02.txt
11856a0 (origin/master, origin/dev) Revert "Reapply "added feature2 in development branch""
88043c0 Reapply "added feature2 in development branch"
6964f3d Revert "added new feature"
94f018f added new feature
eba4560 Revert "added feature2 in development branch"
f2826ca Revert "Added feature4 in development branch"
1a392d0 Added feature4 in development branch
8d53504 added feature2 in development branch
e706e33 Added feature2 in development branch
45778f1 added new versionfile
f6d4237 (origin/main, main) merge branch 'master'
6851b00 added new version
ffb04d9 added new file
6f66dd2 Create README.md
```

Merging combines the histories of two branches, preserving their individual paths. Rebasing integrates the changes of one branch onto another, resulting in a linear sequence of commits.