Gaurang Pawar
501852
Sem 8 IT

DevOps Exp 2

Aim : Version control system using git

Theory:

Git is a software for tracking changes in any set of files, usually used for coordinating work among programmers collaboratively developing source code during software development. Its goals include speed, data integrity, and support for distributed, non-linear workflows (thousands of parallel branches running on different systems). As with most other distributed version control systems, and unlike most client–server systems, every Git directory on every computer is a full-fledged repository with complete history and full version-tracking abilities, independent of network access or a central server.Git is free and open-source software distributed under GNU General Public License Version 2.

Git thinks of its data like a series of snapshots of a miniature filesystem. With Git, every time a commit is made, or the project's state is saved, Git basically takes a picture of what all the files look like at that moment and stores a reference to that snapshot. To be efficient, if files have not changed, Git doesn't store the file again, just a link to the previous identical file it has already stored. Git thinks about its data more like a stream of snapshots.

Steps:

Check git version

Create new file in local directory

Initialize local git repository

```
azureuser@fyp:~/devops$ git init
Initialized empty Git repository in /home/azureuser/devops/.git/
azureuser@fyp:~/devops$
```

Add newly created file on local git repository

```
azureuser@fyp:~/devops$ git status
On branch master

No commits yet

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

        test.txt

nothing added to commit but untracked files present (use "git add" to track)
azureuser@fyp:~/devops$ git add test.txt
azureuser@fyp:~/devops$ git status
On branch master

No commits yet

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

        new file:   test.txt

azureuser@fyp:~/devops$
```

Commit changes

```
azureuser@fyp:~/devops$ git commit -m "Created a text file"
[master (root-commit) 71168e5] Created a text file
 Committer: Ubuntu <azureuser@fyp.c3ei4bkqyh5e5d0g23arjuajva.rx.internal.cloudapp.net>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 1 file changed, 1 insertion(+)
 create mode 100644 test.txt
```
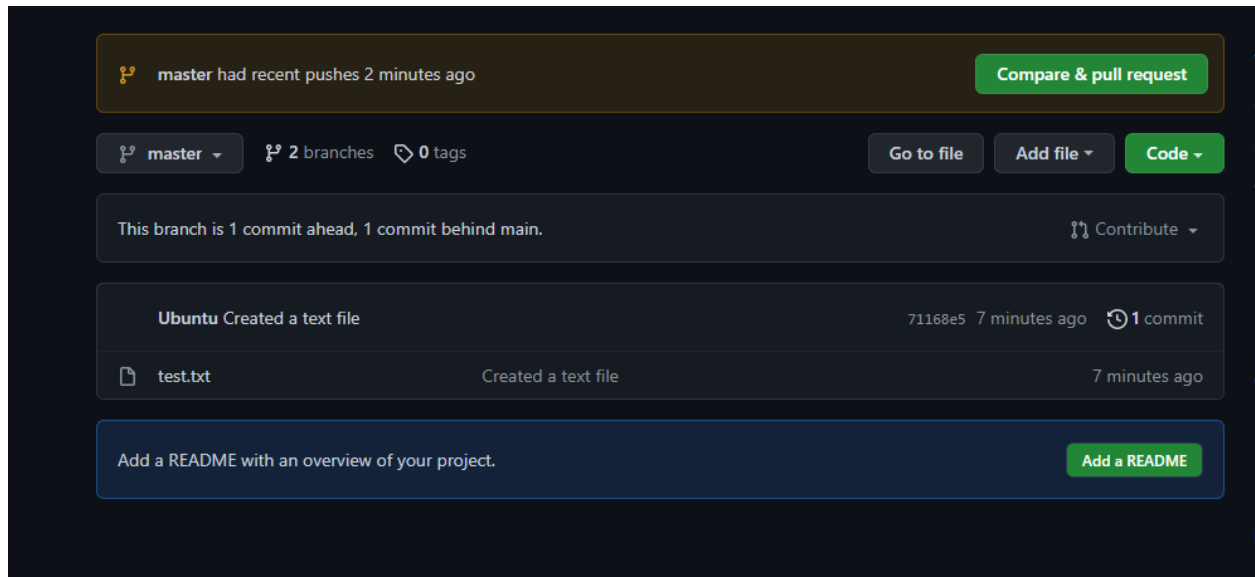
Push the changes on remote repository

```
azureuser@fyp:~/devops$ git add
Nothing specified, nothing added.
Maybe you wanted to say 'git add .'?
azureuser@fyp:~/devops$ git add .
azureuser@fyp:~/devops$ git status
On branch master
nothing to commit, working tree clean
azureuser@fyp:~/devops$ git push https://github.com/gaurang98671/devops_pracs m
Username for 'https://github.com': gaurang98671
Password for 'https://gaurang98671@github.com':
Counting objects: 3, done.
Writing objects: 100% (3/3), 271 bytes | 271.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:        https://github.com/gaurang98671/devops_pracs/pull/new/master
remote:
To https://github.com/gaurang98671/devops_pracs
 * [new branch]      master -> master
azureuser@fyp:~/devops$
```

Conclusion : Therefore we have successfully studied about Git, Github and how to create a repository and add files to it.