

Jan Sher Khan
Assignment -1 (Git Tasks)

Read about the difference between Git and GitHub?

Git is a version control system for the source code which keeps the track of all the history of code changes. GitHub is a cloud-based storage service which lets you manage the Git repositories.

How many types of version control systems are there?

There are two types: The Centralized version control system and the other is distributed version control system. Centralized doesn't have a local copy other than the server but the distributed VC can have both central(server) and local copy.

Explain Branching concept in Git

Git doesn't store data as a series of changesets or differences, but instead as a series of snapshots. So, every time a branch is created that branch object points to a snapshot and when there is a change committed it makes a new snapshot and the same branch object points to the newly created snapshot.

Explain Forking Workflow in Git.

Forking workflow is different than other Git workflows. So, instead of using a single server-side repository to act as the "central" codebase, it provides every developer their own server-side repository. This means that each contributor has not one, but two Git repositories: a private local one and a public server-side one.

Practical Task

Make a repository on GitHub with the name "InnovationPython_yourname"
eg: "InnovationPython_Ankush".

 [jansher4](#) / [InnovationPython_Jansher](#) Private

Initialize an empty Git repository on your local machine with the name "gitlocal" and make a README.md file in that directory which should contain your name as a heading and a hello message.

```
gitlocal — -bash — 80x24
Jans-MacBook-Air:Desktop jansherkhan$ mkdir gitlocal
Jans-MacBook-Air:Desktop jansherkhan$ cd gitlocal/
Jans-MacBook-Air:gitlocal jansherkhan$ echo "#Jan Sher Khan" > readme.md
Jans-MacBook-Air:gitlocal jansherkhan$ echo "Hello" >> readme.md
Jans-MacBook-Air:gitlocal jansherkhan$ git init
Initialized empty Git repository in /Users/jansherkhan/Desktop/gitlocal/.git/
Jans-MacBook-Air:gitlocal jansherkhan$
```

Now check the status of your git directory and push all the files in that directory to your GitHub repo which you have made in the first step. With a message “First Commit to Git Repo”.

```
Jans-MacBook-Air:gitlocal jansherkhan$ git status
On branch master

No commits yet

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

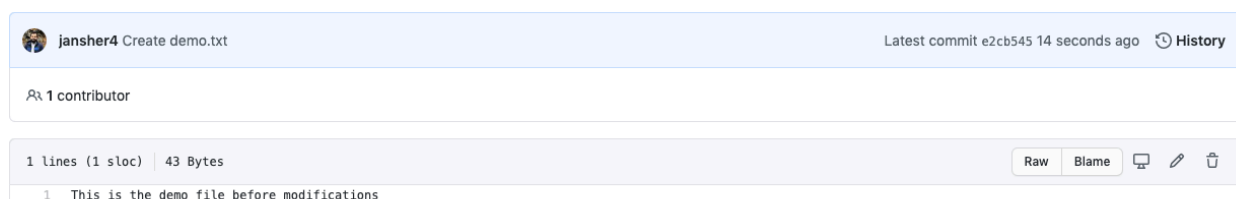
nothing added to commit but untracked files present (use "git add" to track)
Jans-MacBook-Air:gitlocal jansherkhan$ 
Jans-MacBook-Air:gitlocal jansherkhan$ git add .
Jans-MacBook-Air:gitlocal jansherkhan$ git status
On branch master

No commits yet

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

Jans-MacBook-Air:gitlocal jansherkhan$ git commit -m "First Commit to Git Repo"
[master (root-commit) c71fa95] First Commit to Git Repo
 1 file changed, 2 insertions(+)
 create mode 100644 readme.md
Jans-MacBook-Air:gitlocal jansherkhan$ git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 238 bytes | 79.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/jansher4/InnovationPython_Jansher.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
Jans-MacBook-Air:gitlocal jansherkhan$
```

Now add a file to your Github repo named “demo.txt” from the github console with content : “This is the demo file before modifications”.



Pull the changes in your git repo to your local machine git directory named "gitlocal" and check the status for the modifications done in that repo. This time demo should be visible in your local machine.

```
gitlocal — -bash — 81x29
Jans-MacBook-Air:gitlocal jansherkhan$ git pull
warning: Pulling without specifying how to reconcile divergent branches is
discouraged. You can squelch this message by running one of the following
commands sometime before your next pull:

    git config pull.rebase false  # merge (the default strategy)
    git config pull.rebase true   # rebase
    git config pull.ff only       # fast-forward only

You can replace "git config" with "git config --global" to set a default
preference for all repositories. You can also pass --rebase, --no-rebase,
or --ff-only on the command line to override the configured default per
invocation.

remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), 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), 688 bytes | 344.00 KiB/s, done.
From https://github.com/jansher4/InnovationPython_Jansher
   c71fa95..e2cb545  master       -> origin/master
Updating c71fa95..e2cb545
Fast-forward
 demo.txt | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 demo.txt
Jans-MacBook-Air:gitlocal jansherkhan$ ls
demo.txt      readme.md
Jans-MacBook-Air:gitlocal jansherkhan$
```

Now make a new branch in your local machine with the name "developer".

```
gitlocal — -bash — 81x29
Jans-MacBook-Air:gitlocal jansherkhan$ git branch developer
Jans-MacBook-Air:gitlocal jansherkhan$
```

Edit that demo file and write some content in that eg: "This is a demo file after modification" and push the modifications to your GitHub repo from the developer branch with a commit message "Commit from developer branch".

```
gitlocal — -bash — 81x29
Jans-MacBook-Air:gitlocal jansherkhan$ vim demo.txt
Jans-MacBook-Air:gitlocal jansherkhan$

gitlocal — -bash — 86x27
Jans-MacBook-Air:gitlocal jansherkhan$ git commit -a -m "Commit from developer branch"

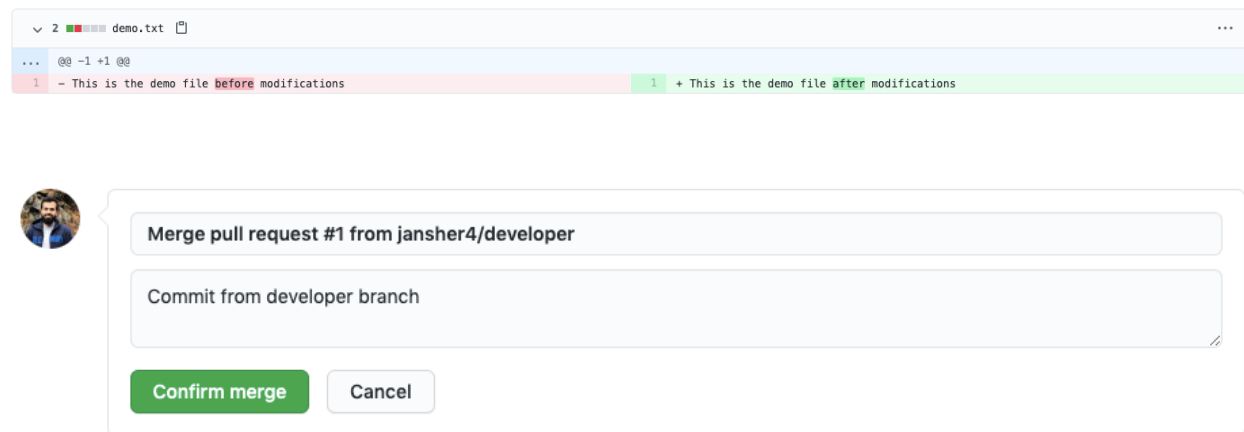
[developer 3f36e5e] Commit from developer branch
 1 file changed, 1 insertion(+), 1 deletion(-)
Jans-MacBook-Air:gitlocal jansherkhan$
```

```

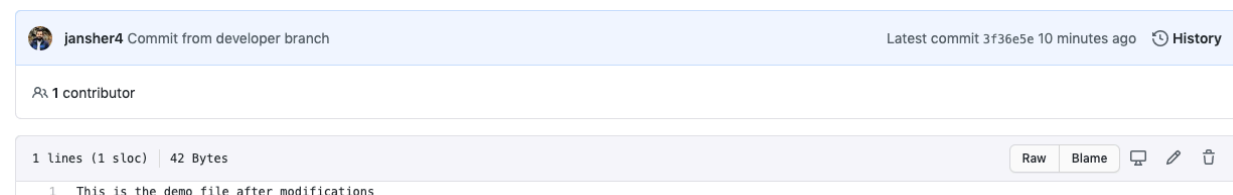
Jans-MacBook-Air:gitlocal jansherkhan$ git push -u origin developer
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 320 bytes | 106.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'developer' on GitHub by visiting:
remote:   https://github.com/jansher4/InnovationPython_Jansher/pull/new/developer
remote:
To https://github.com/jansher4/InnovationPython_Jansher.git
 * [new branch]      developer -> developer
Branch 'developer' set up to track remote branch 'developer' from 'origin'.
Jans-MacBook-Air:gitlocal jansherkhan$

```

Go to the GitHub console and generate a merge request to master branch after checking the modifications.



After merging you could see the modified content in the demo file. Now revert back to the previous version from terminal. After switching back to the previous version your demo file should have the content: This is a demo file before modification.



```
Jans-MacBook-Air:gitlocal jansherkhan$ git log
commit 3f36e5e8fb21016a80be970089f464f611282c8a (HEAD -> developer, origin/developer)
Author: jansher4 <jansher4@gmail.com>
Date:   Wed Oct 28 23:33:04 2020 -0400

    Commit from developer branch

commit e2cb5451b80860ff11eba04e89794cce93d1bf85 (origin/master, master)
Author: Jan Sher Khan <jansher4@gmail.com>
Date:   Wed Oct 28 23:08:27 2020 -0400

    Create demo.txt

commit c71fa952940469d49ed4d1863968e9ee9582e2f1
Author: jansher4 <jansher4@gmail.com>
Date:   Wed Oct 28 22:52:05 2020 -0400

    First Commit to Git Repo
```

```
Jans-MacBook-Air:gitlocal jansherkhan$ git revert HEAD
[developer 46c59ca] Revert "Commit from developer branch"
 1 file changed, 1 insertion(+), 1 deletion(-)
Jans-MacBook-Air:gitlocal jansherkhan$
```

```
Jans-MacBook-Air:gitlocal jansherkhan$ cat demo.txt
This is the demo file before modifications
Jans-MacBook-Air:gitlocal jansherkhan$
```

At the end delete the developer branch.

```
Jans-MacBook-Air:gitlocal jansherkhan$ git branch -d developer
error: The branch 'developer' is not fully merged.
If you are sure you want to delete it, run 'git branch -D developer'.
Jans-MacBook-Air:gitlocal jansherkhan$
Jans-MacBook-Air:gitlocal jansherkhan$
Jans-MacBook-Air:gitlocal jansherkhan$ git branch -D developer
Deleted branch developer (was 46c59ca).
Jans-MacBook-Air:gitlocal jansherkhan$
```