

Steps involved-

1. Create a public repository named surprise-test by forking it.
2. To add remote use git remote add the origin URL of the forked repo and for pushing use git push -u origin main.

```
ayush@LAPTOP-14JQ3IMU MINGW64 ~  
$ mkdir st  
  
ayush@LAPTOP-14JQ3IMU MINGW64 ~  
$ cd st  
  
ayush@LAPTOP-14JQ3IMU MINGW64 ~/st  
$ git remote add origin https://github.com/Ayushi121104/surprise-test  
fatal: not a git repository (or any of the parent directories): .git  
  
ayush@LAPTOP-14JQ3IMU MINGW64 ~/st  
$ git init  
Initialized empty Git repository in c:/Users/ayush/st/.git/  
  
ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (main)  
$ git remote add origin https://github.com/Ayushi121104/surprise-test
```

```
ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (main)  
$ git push -ff origin main  
Enumerating objects: 3, done.  
Counting objects: 100% (3/3), done.  
Writing objects: 100% (3/3), 209 bytes | 209.00 KiB/s, done.  
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)  
To https://github.com/Ayushi121104/surprise-test.git  
+ de322ad...445727f main -> main (forced update)
```

3. With the help of the settings of the repository we can add collaborators.
4. Verify by using git remote -v.

```
ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (main)  
$ git remote -v  
origin https://github.com/Ayushi121104/surprise-test.git (fetch)  
origin https://github.com/Ayushi121104/surprise-test.git (push)
```

5. To make a local commit we can open a file named b1 then we can add it in the staging area and then commit it.
 - Open a file named b1.
 - git add b1.
 - git commit -m "local commit"

```

ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (main)
$ vi b1

ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (main)
$ git add b1
warning: in the working copy of 'b1', LF will be replaced by CRLF the next time you commit

ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (main)
$ git commit -m"local commit"
[main (root-commit) 445727f] local commit
1 file changed, 1 insertion(+)
create mode 100644 b1

ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (main)
$ git log
commit 445727f3215b1543d7f3481fbacb13df310c3aef (HEAD -> main)
Author: Ayushi <ayushi121104@gmail.com>
Date: Fri Apr 12 14:39:36 2024 +0530

    local commit

```

- Push it by using the git push -u origin main command.

```

ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (main)
$ git push -ff origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 209 bytes | 209.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Ayushi121104/surprise-test.git
+ de322ad...445727f main -> main (forced update)

```

- To verify use git pull origin main.

```

ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (main)
$ git pull origin main
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 1.60 KiB | 116.00 KiB/s, done.
From https://github.com/Ayushi121104/surprise-test
* branch                main          -> FETCH_HEAD
   445727f..06470b8      main          -> origin/main
Updating 445727f..06470b8
Fast-forward
 22BDO10055.md | 11 ++++++++
1 file changed, 11 insertions(+)
create mode 100644 22BDO10055.md

```

- We can check the git status and commit history to check the status of the file.

```

ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (main)
$ git log
commit 06470b805795941e9fb38625161a80f99b4f6d7a (HEAD -> main, origin/main)
Author: Ayushi <135798277+Ayushi121104@users.noreply.github.com>
Date:   Fri Apr 12 14:42:22 2024 +0530

    Create 22BDO10055.md

commit 445727f3215b1543d7f3481fbacb13df310c3aef
Author: Ayushi <ayushi121104@gmail.com>
Date:   Fri Apr 12 14:39:36 2024 +0530

    local commit

```

6. Use the git log to see the commit history –

- It includes the following-
 - Commit id
 - Author name
 - Timestamp
 - SHA

7. Two members will add different files so one file we are adding in the main is named c1 and the next one in the branch named k file named c2.

```

ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (main)
$ vi c1

ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (main)
$ git add c1
warning: in the working copy of 'c1', LF will be rep

ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (main)
$ git commit -m"first file"
[main e896ac8] first file
1 file changed, 1 insertion(+)
create mode 100644 c1

```

Now, add both the files in the staging area and commit it with the commit message as first commit and second commit and push it. Now, edit both the files separately.

```

ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (main)
$ git checkout -b k
Switched to a new branch 'k'

ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (k)
$ vi c2

ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (k)
$ git add c2
warning: in the working copy of 'c2', LF wi

ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (k)
$ git commit -m"second file"
[k db4bac8] second file
1 file changed, 1 insertion(+)
create mode 100644 c2

```

8. Now, merge branch k with the main branch and use git pull origin main to pull the changes. Use the git log to see the commit history.

```

ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (main)
$ git merge k
Updating e896ac8..db4bac8
Fast-forward
 c2 | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 c2

ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (main)
$ git commit -m"file merged"
On branch main
nothing to commit, working tree clean

ayush@LAPTOP-14JQ3IMU MINGW64 ~/st (main)
$ git push -u origin main
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (8/8), 721 bytes | 721.00 KiB/s, done.
Total 8 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/Ayushi121104/surprise-test.git
 06470b8..db4bac8  main -> main
branch 'main' set up to track 'origin/main'.

```

9. Repeat the last 2 steps couple of times.

Screenshots are added below-

wid. and
steps
photo

Surprise Test

- 5) To make a local commit -
~~make a~~ → open a file named b1 write some content in it.
 → add that file in the staging area (git add b1)
 → now commit it (git commit -m "local commit")
 (b) Then push it by using git push -u origin main.

- (c) To verify use `git pull origin main`
(d) By checking `git status` & commit history (`git log`) we can easily tell that commit is added to local or not.

- (6) Use git log
history verify
last commit
command to see the commit
that ~~everyone~~ ~~or~~ has done the
is same or not.

(6) SHA-1 is given with the help of metadata
metadata include → SHA code (Hash code)
→ Author's name
→ Time stamp
→ commit id

It is important as it provides the following details of the commit that took place.

(7) Two members should now add different files & edit it differently & push it
To make the commit locally now open a file,
add it in staging area and then commit it.

→ vi b2

→ git add b2

→ git commit -m "commit locally"

Then push it -

→ git push -u origin main

Now edit the file by following steps -

→ vi b2 (edit it) exit :wq

→ add & commit it

(8) For pulling, use → git pull origin main
Use git log to see the log history (commit history)
→ After git pull all the changes are now available locally.

(9) Repeat the last 2 steps couple of times.