1. Establish a local repository for your project with a descriptive name. Set up Git configurations by adding a username and email address. Display and modify the Git configurations, and verify if the changes are reflected in the .git/config file post-editing. Showcase the utilization of both Git Local and Global configurations.

```
root@DESKTOP-PDOJQQA:.git# cat config

[core]

        repositoryformatversion = 0

        filemode = true

        bare = false

        logallrefupdates = true

[user]

        name = vivek

        email = vivekmore1992@gmail.com


remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*

root@DESKTOP-PDOJQQA:.git# git config --local user.name "vivek"

root@DESKTOP-PDOJQQA:.git# git config --global user.email "vivekmore1992@gmail.com.com"

root@DESKTOP-PDOJQQA:.git# git config --global --list

user.name=vivek more

user.email=vivekmore1992@gmail.com.com
```

2.Set up a local repository for your project with a meaningful name. Explore the default branch in the local Git repository. Sequentially create five files, committing each with informative messages. Share the Git log output through multiple methods once the process is complete.

```
root@DESKTOP-PDOJQQA:.git# cd my_project/

root@DESKTOP-PDOJQQA:my_project# ll
```

```
total 32
drwxr-xr-x 3 root root 4096 Jan  3 13:35 ./
drwxr-xr-x 9 root root 4096 Jan  3 14:24 ../
drwxr-xr-x 8 root root 4096 Jan  3 13:35 .git/
-rw-r--r-- 1 root root   15 Jan  3 13:35 file1.txt
-rw-r--r-- 1 root root   15 Jan  3 13:35 file2.txt
-rw-r--r-- 1 root root   15 Jan  3 13:35 file3.txt
-rw-r--r-- 1 root root   15 Jan  3 13:35 file4.txt
-rw-r--r-- 1 root root   15 Jan  3 13:35 file5.txt

root@DESKTOP-PDOJQQA:my_project# git log
commit aeac8ee284c0cdc33976c53a9c7b6015a8583dbd (HEAD -> master)
Author: vivek more <vivekmore1992@gmail.com.com>
Date:   Wed Jan 3 13:35:27 2024 +0530

    Add File 5

commit 290767551bb115c8766717874e1e284b17b1bd12
Author: vivek more <vivekmore1992@gmail.com.com>
Date:   Wed Jan 3 13:35:27 2024 +0530

    Add File 4

commit a98fcd9a2b18a1b37f53e4441aeb4048c7bdee51
Author: vivek more <vivekmore1992@gmail.com.com>
Date:   Wed Jan 3 13:35:27 2024 +0530
```

```
    Add File 3

commit 5514a5ed90d0c79cb39c98b66fa04a28eeed1a64
Author: vivek more <vivekmore1992@gmail.com.com>
Date:   Wed Jan 3 13:35:27 2024 +0530

    Add File 2

commit 1d92b39e9f3f1508f22370a0d51874398ec3a5d3
Author: vivek more <vivekmore1992@gmail.com.com>
Date:   Wed Jan 3 13:35:27 2024 +0530
```

---

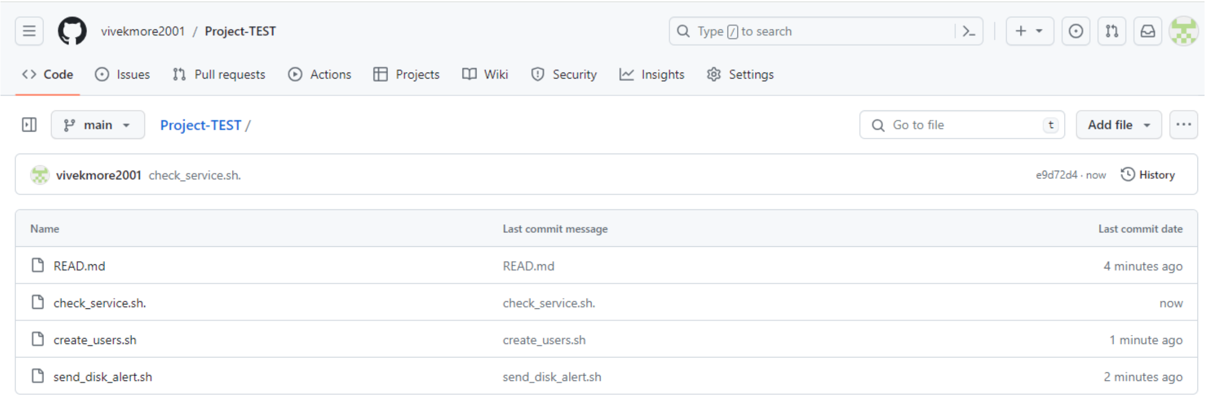
3. Open a GitHub account and initiate a remote Git repository. Examine the default branch in the remote repository. Create and commit the following files one by one with meaningful commit messages:

i. README.md

ii. send_disk_alert.sh

iii. create_users.sh

iv. check_service.sh.



vivekmore2001 check_service.sh. e9d72d4 · now

Name	Last commit message	Last commit date
READ.md	READ.md	4 minutes ago
check_service.sh.	check_service.sh.	now
create_users.sh	create_users.sh	1 minute ago
send_disk_alert.sh	send_disk_alert.sh	2 minutes ago

4. Establish a local repository for your project with a descriptive name. Illustrate the implementation of a Git-Ignore file by committing multiple files to the local repository.

```
root@DESKTOP-PDOJQQA:~# mkdir brave
root@DESKTOP-PDOJQQA:~# cd brave/
root@DESKTOP-PDOJQQA:brave# ll
total 8
drwxr-xr-x 2 root root 4096 Jan  3 16:13 ./
drwx------ 8 root root 4096 Jan  3 16:13 ../
root@DESKTOP-PDOJQQA:brave# git init
Initialized empty Git repository in /root/brave/.git/
root@DESKTOP-PDOJQQA:brave# ls
root@DESKTOP-PDOJQQA:brave# touch abc.txt
root@DESKTOP-PDOJQQA:brave# touch xyz.txt
root@DESKTOP-PDOJQQA:brave# touch pqr.txt
root@DESKTOP-PDOJQQA:brave# touch wwy.txt
root@DESKTOP-PDOJQQA:brave# git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        abc.txt
        pqr.txt
        wwy.txt
        xyz.txt
```

```
nothing added to commit but untracked files present (use "git add" to track)
root@DESKTOP-PDOJQQA:brave# vi .gitignore
root@DESKTOP-PDOJQQA:brave# git add .
root@DESKTOP-PDOJQQA:brave# git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   .gitignore
        new file:   pqr.txt
        new file:   wwy.txt

root@DESKTOP-PDOJQQA:brave# git log
fatal: your current branch 'master' does not have any commits yet
```

---

5. Create a local repository with a meaningful project name. Demonstrate the utilization of GIT working area, staging area, and commit area. Experiment with committing, deleting, and restoring files from the repository. Additionally, add files to the staging area and restore them from the staging area.

```
root@DESKTOP-PDOJQQA:brave#
root@DESKTOP-PDOJQQA:brave# ll
total 16
drwxr-xr-x 3 root root 4096 Jan  3 16:16 ./
drwx------ 8 root root 4096 Jan  3 16:16 ../
drwxr-xr-x 7 root root 4096 Jan  3 16:16 .git/
```

```
-rw-r--r-- 1 root root   17 Jan  3 16:16 .gitignore
-rw-r--r-- 1 root root    0 Jan  3 16:13 abc.txt
-rw-r--r-- 1 root root    0 Jan  3 16:14 pqr.txt
-rw-r--r-- 1 root root    0 Jan  3 16:14 wwy.txt
-rw-r--r-- 1 root root    0 Jan  3 16:14 xyz.txt
root@DESKTOP-PDOJQQA:brave# rm -rf wwy.txt
root@DESKTOP-PDOJQQA:brave# git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   .gitignore
        new file:   pqr.txt
        new file:   wwy.txt

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    wwy.txt

root@DESKTOP-PDOJQQA:brave# ls
abc.txt  pqr.txt  xyz.txt
root@DESKTOP-PDOJQQA:brave# git restore wwy.txt
```