

Experiment no. 3 : To perform various git operations on local and remote repositories using GIT cheatsheet

Git Cheat Sheet: Local & Remote Repository Operations

Basic Git Commands

1. Git Configuration

- `git config --global user.name "Your Name"` → Set your name
- `git config --global user.email "youremail@example.com"` → Set your email
- `git config --list` → View Git configurations

Local Repository Operations

2. Initialize and Clone

- `git init` → Initialize a new Git repository
- `git clone <repo_url>` → Clone a remote repository

3. Basic Workflow

- `git status` → Check the status of changes
- `git add <file>` → Stage a specific file
- `git add .` → Stage all files
- `git commit -m "Commit message"` → Commit changes
- `git log` → View commit history
- `git diff` → View unstaged changes

4. Branching and Merging

- `git branch` → List branches
- `git branch <branch_name>` → Create a new branch
- `git checkout <branch_name>` → Switch to a branch
- `git merge <branch_name>` → Merge a branch into the current branch
- `git branch -d <branch_name>` → Delete a branch

Remote Repository Operations

5. Connect to Remote Repository

- `git remote add origin <repo_url>` → Connect local repo to remote
- `git remote -v` → View remote connections

6. Push and Pull Changes

- `git push origin <branch_name>` → Push local changes to remote
- `git pull origin <branch_name>` → Fetch and merge changes from remote
- `git fetch` → Fetch changes from remote without merging
- `git add <file> & git commit -m "Resolved conflicts"` → Finalize merge

8. Undo Changes

- `git reset --hard <commit_id>` → Reset to a specific commit
- `git revert <commit_id>` → Create a new commit that undoes previous changes
- `git checkout -- <file>` → Discard changes in a file

Useful Commands

- `git stash` → Save uncommitted changes temporarily
- `git stash pop` → Restore stashed changes
- `git tag -a v1.0 -m "Version 1.0"` → Create a tag
- `git show <commit_id>` → View commit details

By using these commands, we can efficiently manage both local and remote repositories with Git.

Output of the executed commands

```

ROHAN@LAPTOP-F68SP2JU MINGW64 ~/Desktop/DegreeAIDS/sem 6/SEPM Lab (master)
$ git init
Reinitialized existing Git repository in C:/Users/ROHAN/Desktop/DegreeAIDS/sem 6/SEPM Lab/.git/

ROHAN@LAPTOP-F68SP2JU MINGW64 ~/Desktop/DegreeAIDS/sem 6/SEPM Lab (master)
$ git commit -m "first commit"
On branch master

Initial commit

nothing to commit (create/copy files and use "git add" to track)

ROHAN@LAPTOP-F68SP2JU MINGW64 ~/Desktop/DegreeAIDS/sem 6/SEPM Lab (master)
$ git branch -M main
  
```

```

$ git add .

ROHAN@LAPTOP-F68SP2JU MINGW64 ~/Desktop/DegreeAIDS/sem 6/SEPM Lab (main)
$ git commit -m "first commit"
On branch main

Initial commit

nothing to commit (create/copy files and use "git add" to track)
  
```

```
ROHAN@LAPTOP-F68SP2JU MINGW64 ~/Desktop/DegreeAIDS/sem 6/SEPM Lab/git-folder (main)
$ git config --global user.name "rohanfukat"

ROHAN@LAPTOP-F68SP2JU MINGW64 ~/Desktop/DegreeAIDS/sem 6/SEPM Lab/git-folder (main)
$ git config --global user.email "rohanfukat123@gmail.com"

ROHAN@LAPTOP-F68SP2JU MINGW64 ~/Desktop/DegreeAIDS/sem 6/SEPM Lab/git-folder (main)
$ git init
Initialized empty Git repository in C:/Users/ROHAN/Desktop/DegreeAIDS/sem 6/SEPM Lab/git-folder/.git/

ROHAN@LAPTOP-F68SP2JU MINGW64 ~/Desktop/DegreeAIDS/sem 6/SEPM Lab/git-folder (master)
$ ls -a
./ ../ .git/

ROHAN@LAPTOP-F68SP2JU MINGW64 ~/Desktop/DegreeAIDS/sem 6/SEPM Lab/git-folder (master)
$
```

```
File Edit Selection View Go Run Terminal Help Experiment 1.docx - SEPM Lab - Visual Studio Code [Administrator]

SEPM LAB
  Experiment 1.docx

ROHAN@LAPTOP-F68SP2JU MINGW64 ~/Desktop/DegreeAIDS/sem 6/SEPM Lab (main)
$ git add .
ROHAN@LAPTOP-F68SP2JU MINGW64 ~/Desktop/DegreeAIDS/sem 6/SEPM Lab (main)
$ git commit -m "first commit"
[main (root-commit) d2b379c] first commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Experiment 1.docx

ROHAN@LAPTOP-F68SP2JU MINGW64 ~/Desktop/DegreeAIDS/sem 6/SEPM Lab (main)
$ git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 216 bytes | 108.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/rohanfukat/SEPM.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

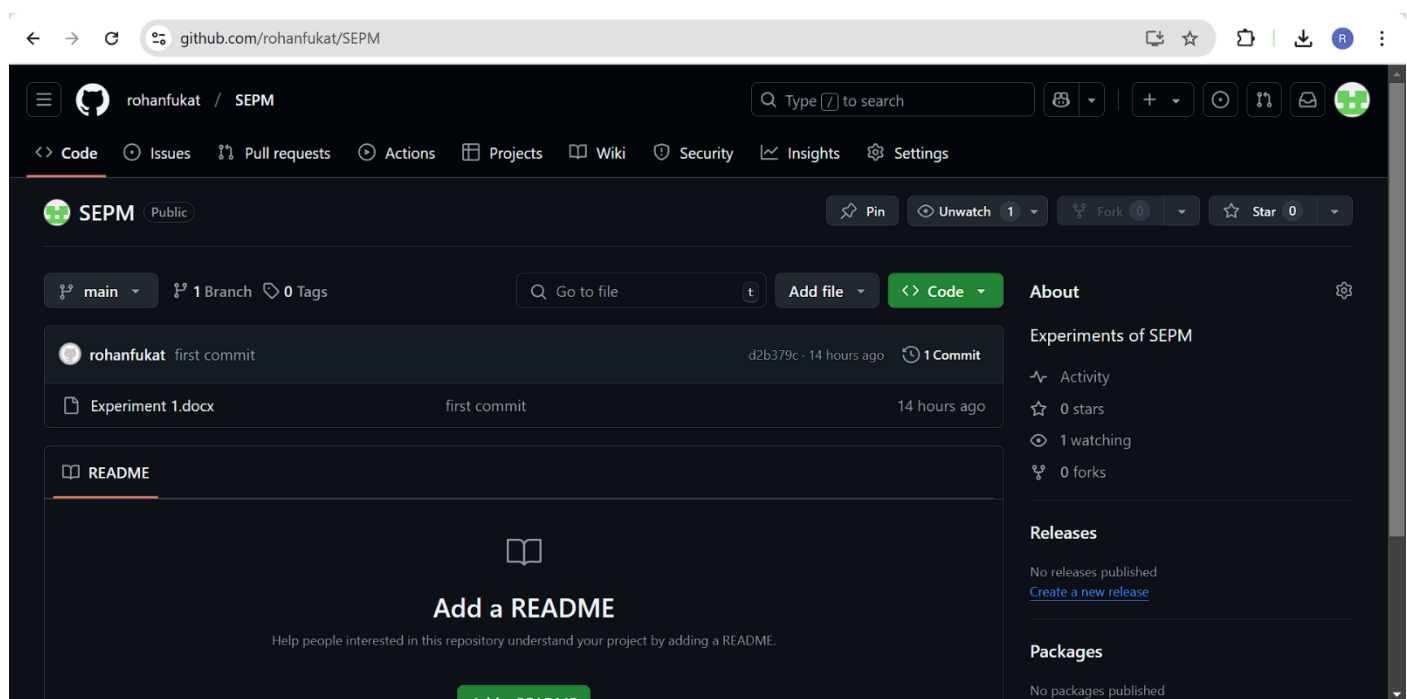
ROHAN@LAPTOP-F68SP2JU MINGW64 ~/Desktop/DegreeAIDS/sem 6/SEPM Lab (main)
$
```



The screenshot shows the Visual Studio Code interface with a terminal window open. The terminal displays the following commands and output:

```
ROHAN@LAPTOP-F68SP2JU MINGW64 ~/Desktop/DegreeAIDS/sem 6/SEPM Lab (main)
$ mkdir git-folder
ROHAN@LAPTOP-F68SP2JU MINGW64 ~/Desktop/DegreeAIDS/sem 6/SEPM Lab (main)
$ cd git-folder
```

The file explorer on the left shows the project structure with a folder named 'git-folder' and a file named 'Experiment 1.docx'.



Conclusion: Thus, we have successfully executed git operations on local and remote repositories