Lab 1

Title: Introduction to Source Code Management using Git

Objective:

- To understand the concept and importance of source code management.
- To install and configure Git.
- To learn basic Git commands for version control.
- To create and manage a local Git repository.

Theory:

Source Code Management (SCM) is the practice of tracking and managing changes to software code. Git is a free, open-source distributed version control system widely used in software development. It helps teams collaborate efficiently, manage project versions, and track changes over time.

Git allows developers to:

- Maintain multiple versions of code.
- Collaborate with others without overwriting each other’s work.
- Roll back to previous versions when needed.
- Work offline with local repositories.

Code

Step 1 : Initialize a Git repository

```
git init
```

Step 2: Check the status of files

```
git status
```

Step 3: Add files to staging area

`git add <filename> -` To add a specific file

`git add .` – To add all files

Step 4 : Commit the changes

```
git commit –m “Initial commit”
```

Step 5: Create a new branch and switch to the branch

```
git checkout –b feature-login
```

Step 6 : Merge changes back to main

```
git checkout main git merge feature-login
```

Step 7 : Push to remote GitHub repo

```
git remote add origin https://github.com.itsmesubas/agile.git
git push –u origin main
```

Conclusion

In this lab, we successfully learned how to manage source code using Git. We covered key commands like init, add, commit, branch, checkout, and push. Understanding Git enables developers to manage projects efficiently and collaborate with others. It also provides a safety net by allowing easy rollback to previous versions.