

**SOURCE CODE MANAGEMENT**

Subject Code: CS181

Cluster: Alpha

Department: CSE

Submitted By:

Submitted To: Dr. Shikha

Shreya Chauhan

2110991326

G-18

**INDEX**

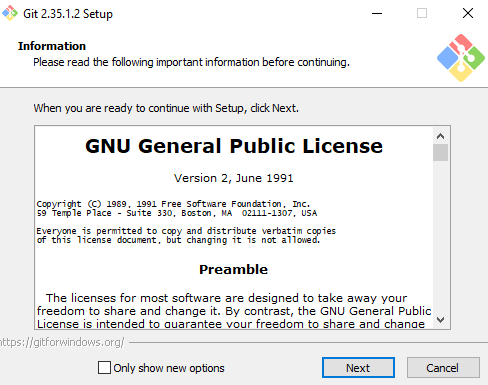
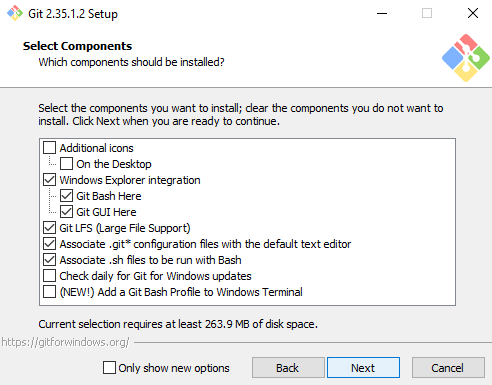
|  |  |  |
| --- | --- | --- |
| **S.no** | **TASK** | **Page No.** |
| **1.** | **How to install GitBash** |  |
| **2.** | **Setting up GitHub** |  |
| **3.** | **Generating Logs** |  |
| **4.** | **Creating and Visualising branches** |  |
| **5.** | **Git lifecycle description** |  |
|  | | |

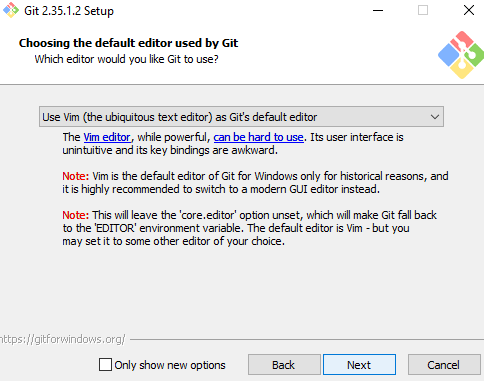
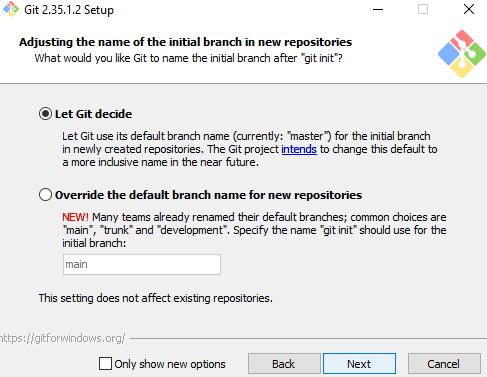
**Aim: Install GitBash**

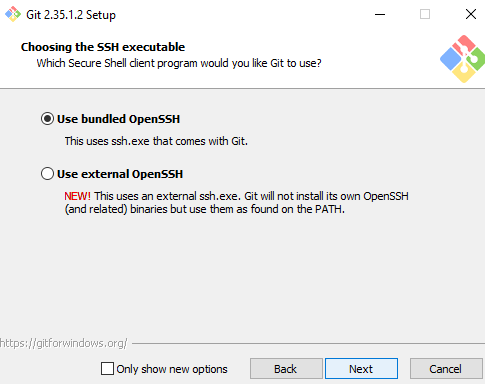
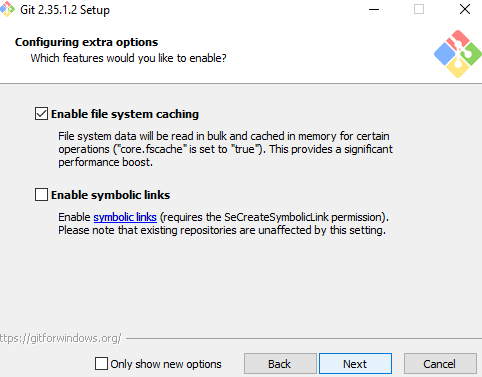
**Download Git for Windows:**

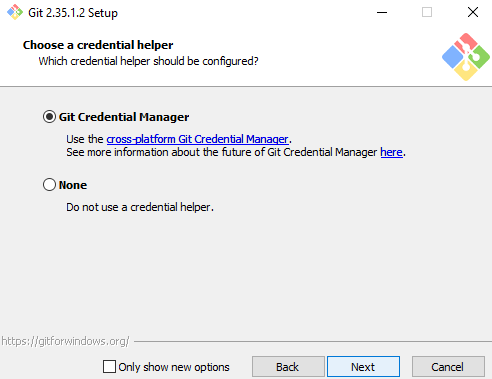
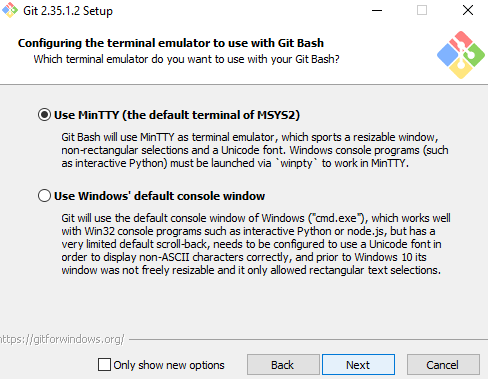
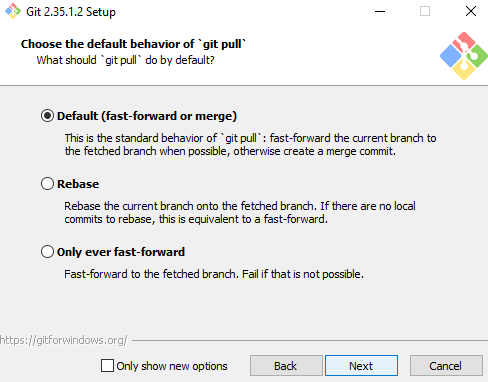
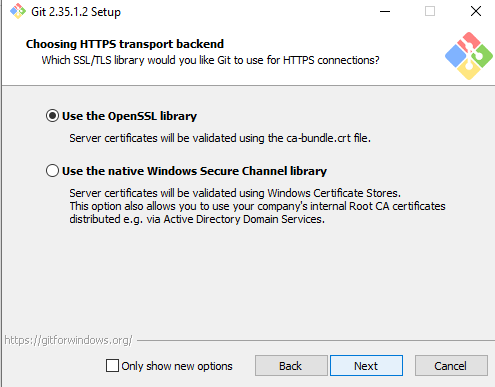
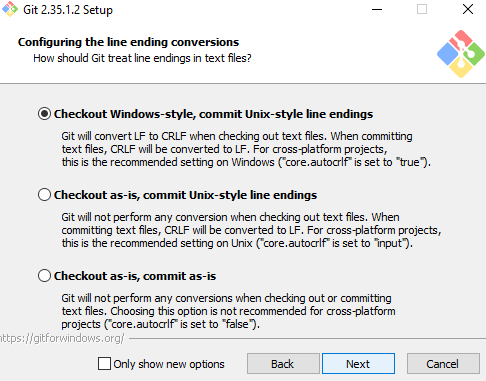
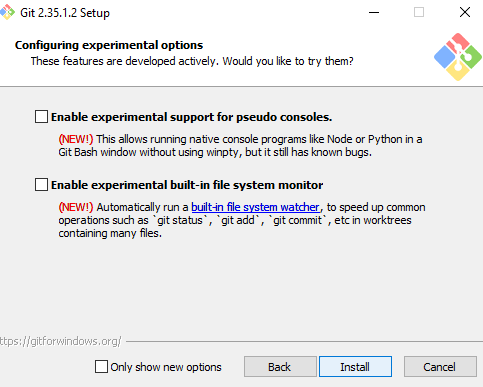
1.Browse to the official Git website: <https://git-scm.com/downloads>

2. Click the download link for Windows and allow the download to complete.

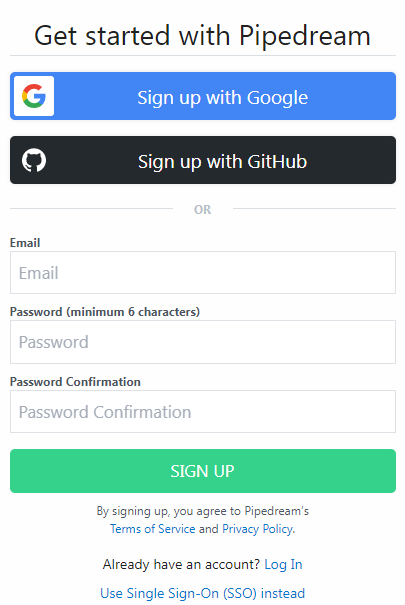
   

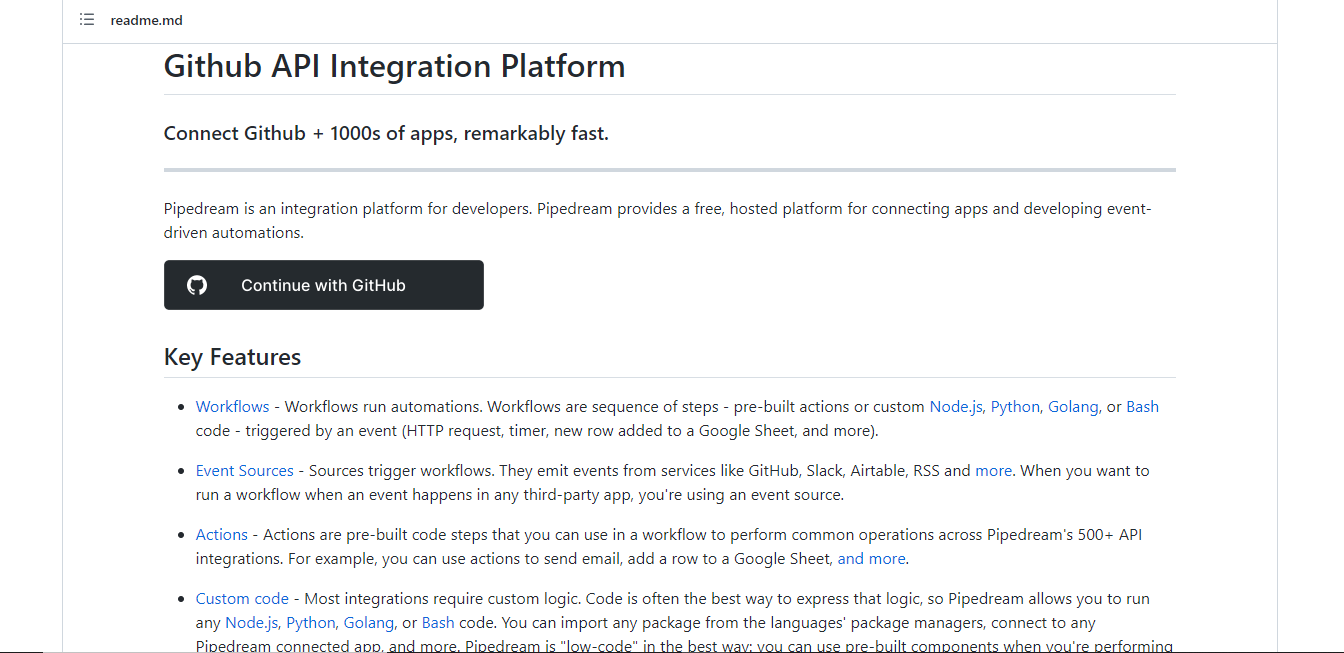
Aim: Setting up GitHub account.

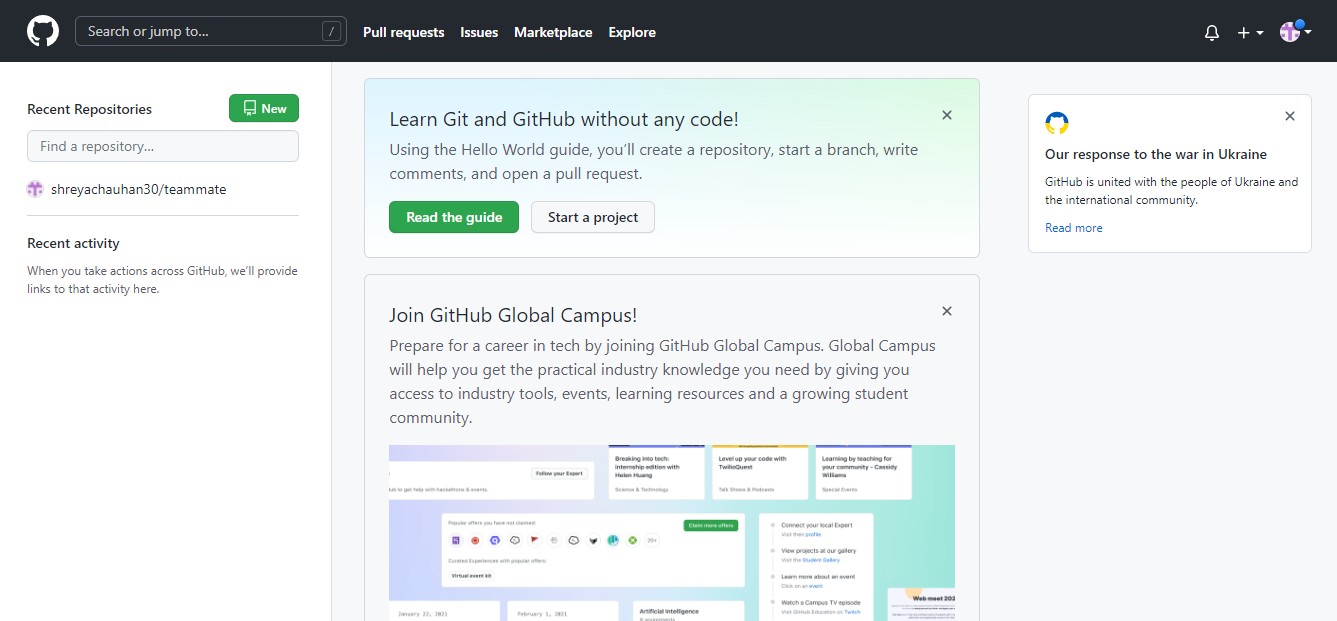
You need a GitHub username and password for this next step.

To ensure you can use all the features in your GitHub , verify your email address after signing up for a new account.

Git is responsible for everything GitHub-related that happens locally on your computer. To effectively collaborate on GitHub, you'll write in issues and pull requests using GitHub Flavoured Markdown.



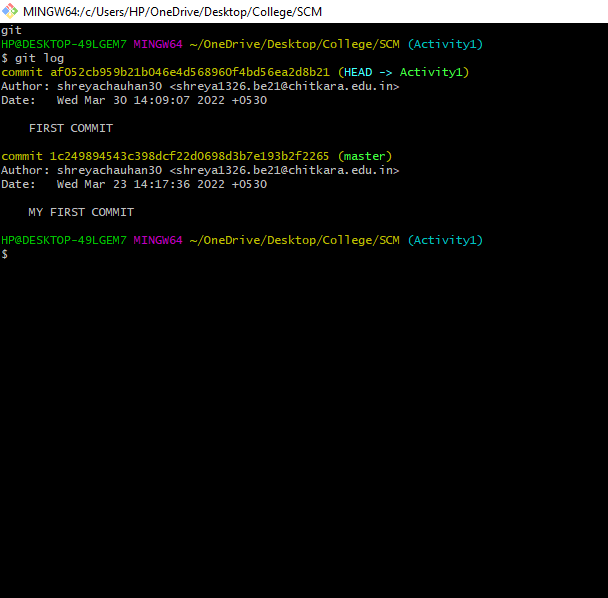




**Aim: Generating logs.**

GIT log Command

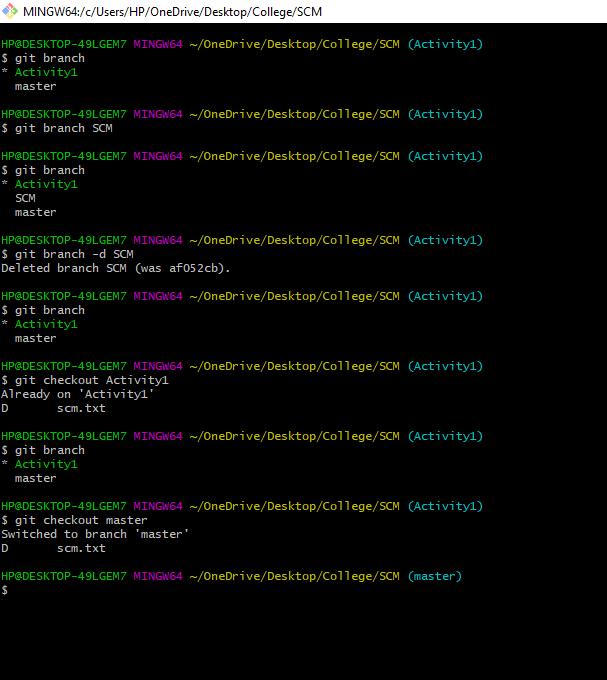
After you have created several commits, or if you have cloned a repository with an existing commit history, you’ll probably want to look back to see what has happened. The most basic and powerful tool to do this is the git log command.



Aim: Creating and Visualizing branches

A branch represents an independent line of development. Branches serve as an abstraction for the edit/stage/commit process. You can think of them as a way to request a brand new working directory, staging area, and project history. New commits are recorded in the history for the current branch, which results in a fork in the history of the project.

**git branch**



Aim: Git lifecycle description

The life cycle of Git.

General workflow is as follows −

* You clone the Git repository as a working copy.
* You modify the working copy by adding/editing files.
* If necessary, you also update the working copy by taking other developer's changes.
* You review the changes before commit.
* You commit changes. If everything is fine, then you push the changes to the repository.
* After committing, if you realize something is wrong, then you correct the last commit and push the changes to the repository.

Git Commands

**git config**

**Usage: git config –global user.name “[name]”**

**Usage: git config –global user.email “[email address]”**

This command sets the author name and email address respectively to be used with your commits.

**git init**

**Usage: git init [repository name]**

This command is used to start a new repository.

**git clone**

This command is used to obtain a repository from an existing URL

**git add**

**Usage: git add [file]**

This command adds a file to the staging area.

**git commit**

**Usage: git commit -m “[ Type in the commit message]”**

This command records or snapshots the file permanently in the version history.

**Usage: git commit -a**

This command commits any files you’ve added with the git add command and also commits any files you’ve changed since then.

**git status**

**Usage: git status**

This command lists all the files that have to be committed.

