.

Subject Name: **Source Code Management**

Subject Code: **CS181** Cluster: **Zeta** Department: D**CSE**



**Submitted By:** Ranvir Kumar

2110992143

G28

Submitted To:

Dr. Dimple Nagpal

**Table of Content**

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Title** | **Page No.** |
| 1 | Version control with Git | 2-3 |
| 2 | Setting up Github account | 4-5 |
| 3 | Objective |  |
| 4 | Resources Requirements – Frontend / Backend |  |
| 5 | Concepts and commands |  |
| 6 | Reference |  |
| 7 | Snapshots |  |

 **Experiment No. 01**

**Aim:** Version Control with Git

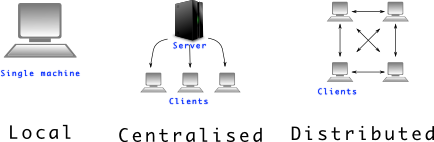
# Theory:

**What is Git?**

Git is a software used for tracking changes in any set of files, usually used for coordinating work among members of a team.

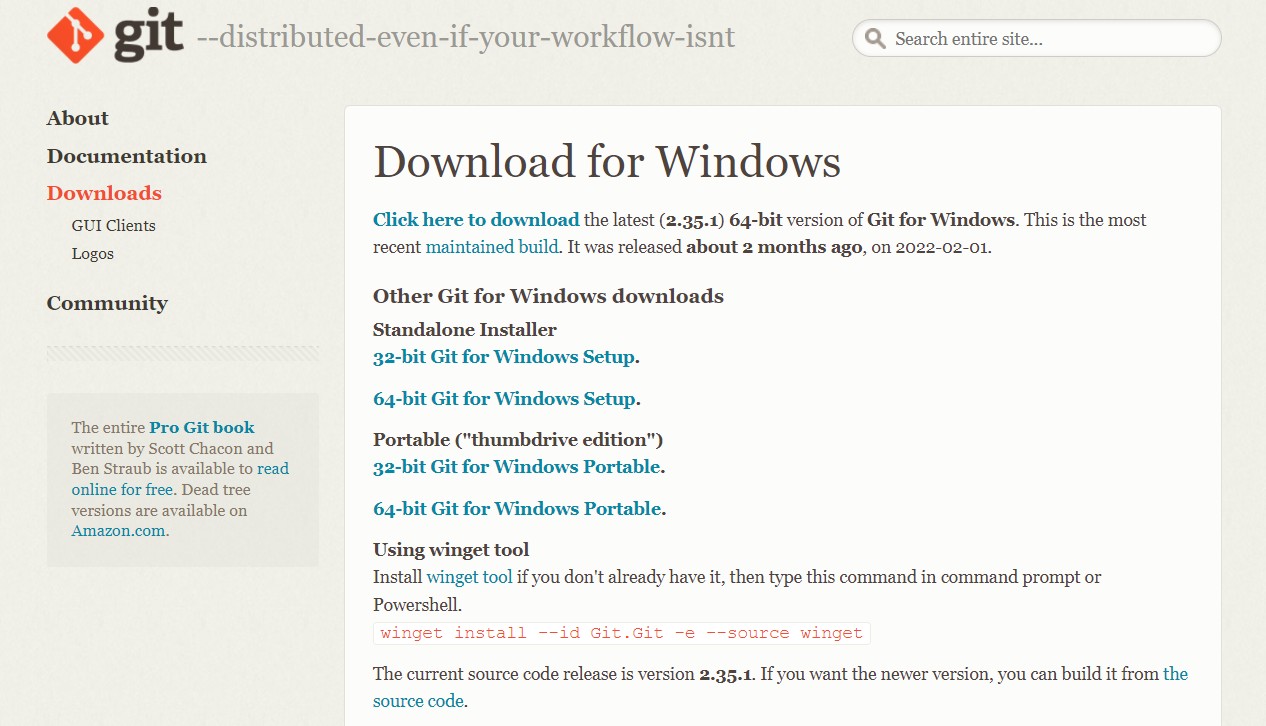
# History of VCS:

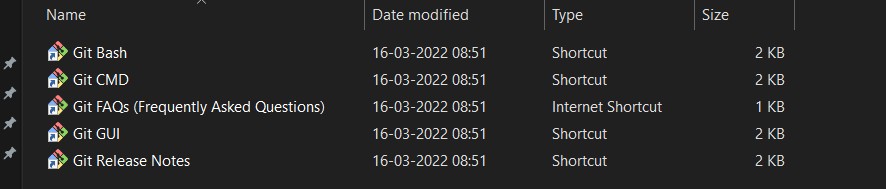
* **Local VCS:** No internet is needed because it uses a database to keep track of files. It is located in your local machine. If the local machine crashes it would not be possible to retreat the files and all the information will be lost. If anything happens to a single version all the version made after that will be lost
* **Centralized VCS:** The next major issue that people encounter is that they need to collaborate with developers on other systems. To deal with this problem, Centralized Control System(CVS) were developed. These system have a single server that contains all the versions files, and a number of clients that check out files from that central place.
* **Distributed VCS:** A type of version control where the complete codebase including its full version history is mirrored on every developer's computer. If any server dies, and these system were collaborating via that server, any one of the client repositories can be copied backup to the server to restore it.



# How to install GIT on Windows?

There are also a few ways to install Git on Windows. The most official build is available for download on the Git website. Just go to [https://git-](https://git-scm.com/download/win) [scm.com/download/win](https://git-scm.com/download/win) and the download will start automatically. Note that this is a project called Git for Windows, which is separate from Git itself; for more information on it, go to [https://gitforwindows.org](https://gitforwindows.org/).





Check version of git by using git –version command.

**Experiment No. 02 **

**Aim:** Setting up GitHub Account

# Theory:

**What is GitHub?**

GitHub is a code hosting platform for version control and collaboration. In other words, it manages repositories.

# Advantages:

* It makes it easy to contribute to Open-Source projects.
* Track changes in your code across versions.

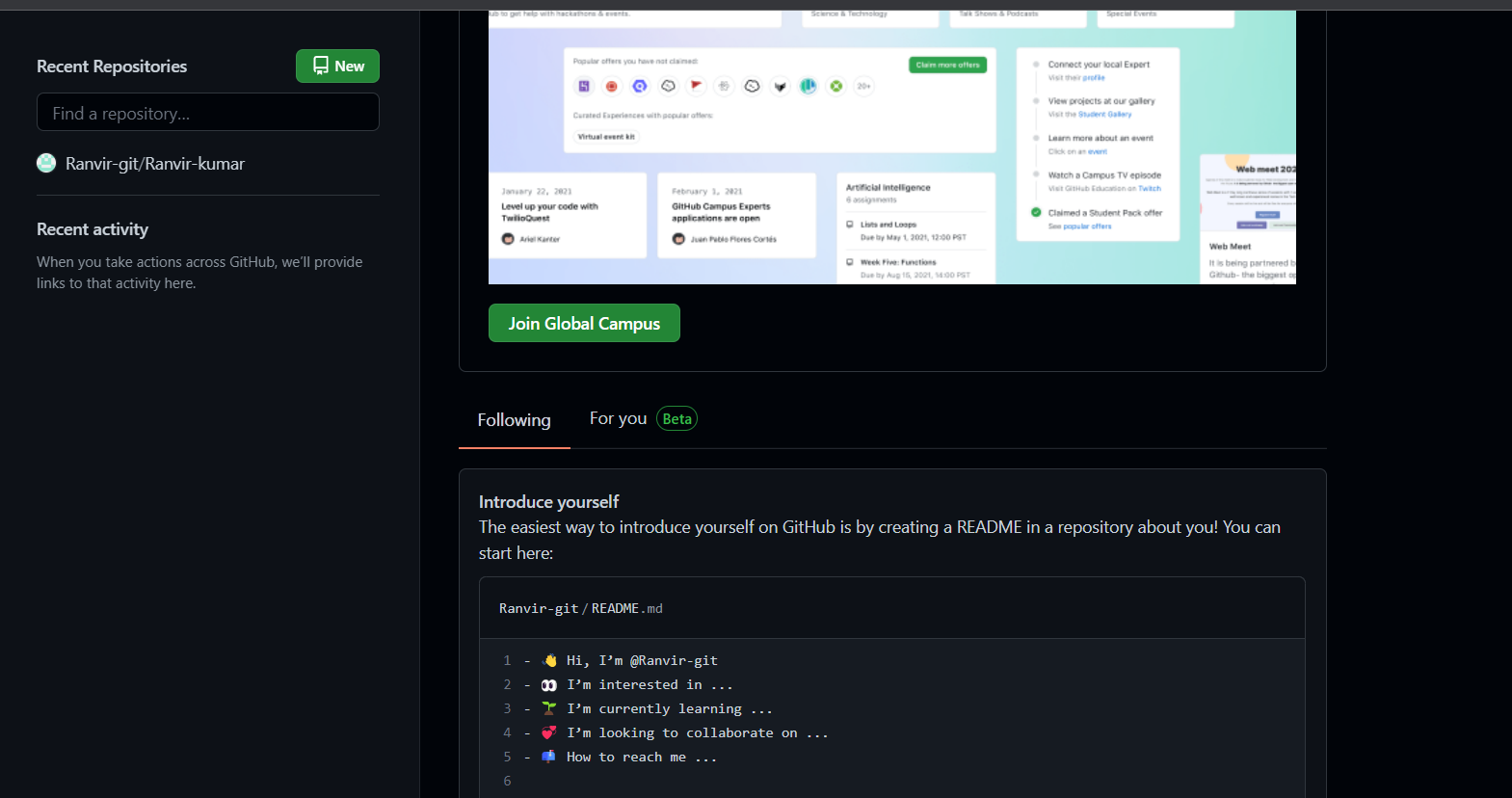
# Procedure:

Search for GitHub in any search engine or <https://github.com/signup>



If you’re a new user add your email and click on **Sign up for GitHub**. Otherwise click on **Sign In** at the top right corner, if u have pre existing account.

# Signing into GitHub:

****

**Linking GitHub account with Git Bash:**

**Username:**

git config --global user.name “username in github”

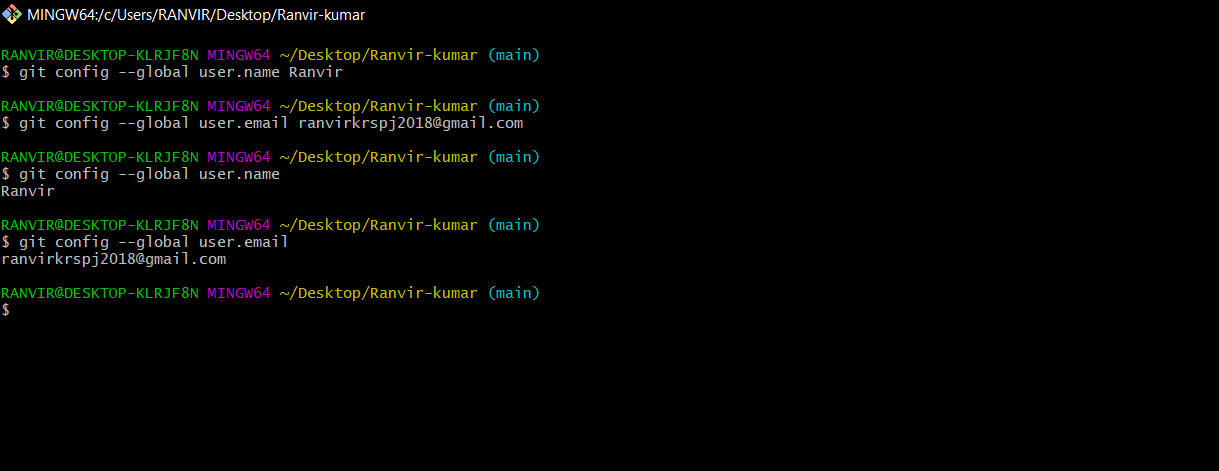
**Email:**

git config --global user.email “your email in github”

**Check Username & Email:**

git config user.name

git config user.email



**Experiment No. 03 **

**Aim:** Program to Generate log

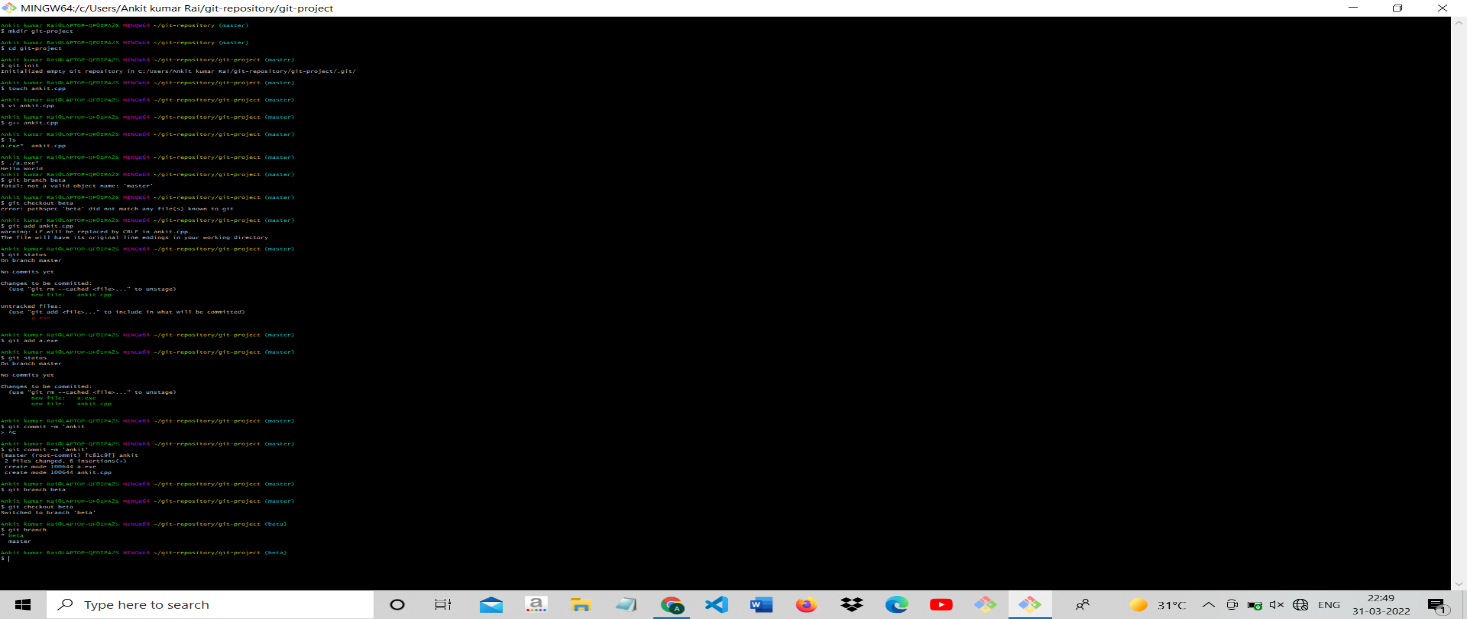
# Theory:

**Git Logs:**

Logs are nothing but the history which we can see in Git by using the code Git log. It contains all the past commits, insertions and deletions which can be seen anytime.

**Why do we need logs?**

Logs help us to check the changes made in code or files and by whom. It also contains the details of insertions and deletions and also the time it was changed at.



* **Use command** git log **to access logs.**

**Experiment No. 4 **

**Aim:** Git Concepts and Commands.

# Theory:

## **Advantages of Git:**

### Free and open source

Git is released under GPL’s open source license. It is available freely over the internet. You can use Git to manage property projects without paying a single penny. As it is an open source, you can download its source code and also perform changes according to your requirements.

### Fast and small

As most of the operations are performed locally, it gives a huge benefit in terms of speed. Git does not rely on the central server; that is why, there is no need to interact with the remote server for every operation. The core part of Git is written in C, which avoids runtime overheads associated with other high-level languages. Though Git mirrors entire repository, the size of the data on the client side is small. This illustrates the efficiency of Git at compressing and storing data on the client side.

### Implicit backup

The chances of losing data are very rare when there are multiple copies of it. Data present on any client side mirrors the repository, hence it can be used in the event of a crash or disk corruption.

**Commands**

1. Git clone - Bring a repository i.e hosted somewhere on the platform like Github into a folder on your machine.
2. Commit - Save your files in Git.
3. Push – Upload Git comments to a remote repository like Github.
4. Pull – Download changes from remote repository to your local machine.
5. Status – Tells the state of working directory and staging area.
6. Log – To view the history of commands executed on git bash.

**Experiment No. 05 **

**Aim:** Git lifecycle description

# Theory:

**Stages in GIT Life Cycle:**

Files in a Git project have various stages like Creation, Modification, Refactoring, and Deletion and so on. Irrespective of whether this project is tracked by Git or not, these phases are still prevalent. However, when a project is under Git version control system, they are present in three major Git states in addition to these basic ones. Here are the three Git states:

* Working directory
* Staging area
* Git directory

**Working Directory:**

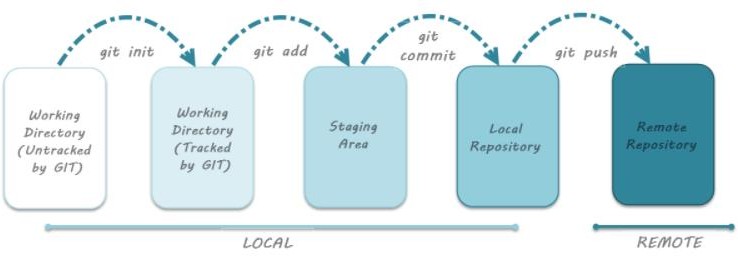
Consider a project residing in your local system. This project may or may not be tracked by Git. In either case, this project directory is called your Working directory.

**Staging Area:**

Staging area is the playground where you group, add and organize the files to be committed to Git for tracking their versions.

**Git Directory:**

Now that the files to be committed are grouped and ready in the staging area, we can commit these files. So, we commit this group of files along with a commit message explaining what is the commit about. Apart from commit message, this step also records the author and time of the commit. Now, a snapshot of the files in the commit is recorded by Git. The information related to this commit is stored in the Git directory.

**Remote Repository:** means mirror or clone of the local Git repository in GitHub. And pushing means uploading the commits from local Git repository to remote repository hosted in GitHub

# Reference:

* Search for GitHub in any search engine or <https://github.com/signup>

# To install Git on windows visit [https://git-](https://git-scm.com/download/win) [scm.com/download/win](https://git-scm.com/download/win)

# For more information visit [https://gitforwindows.org](https://gitforwindows.org/).

# Snapshots:

