**SCM File**

Source Code Management

Submitted By: Aayush Mittal

Roll No: 2110990030

Cluster: Beta

Submitted To:

Monit Sir

Introduction

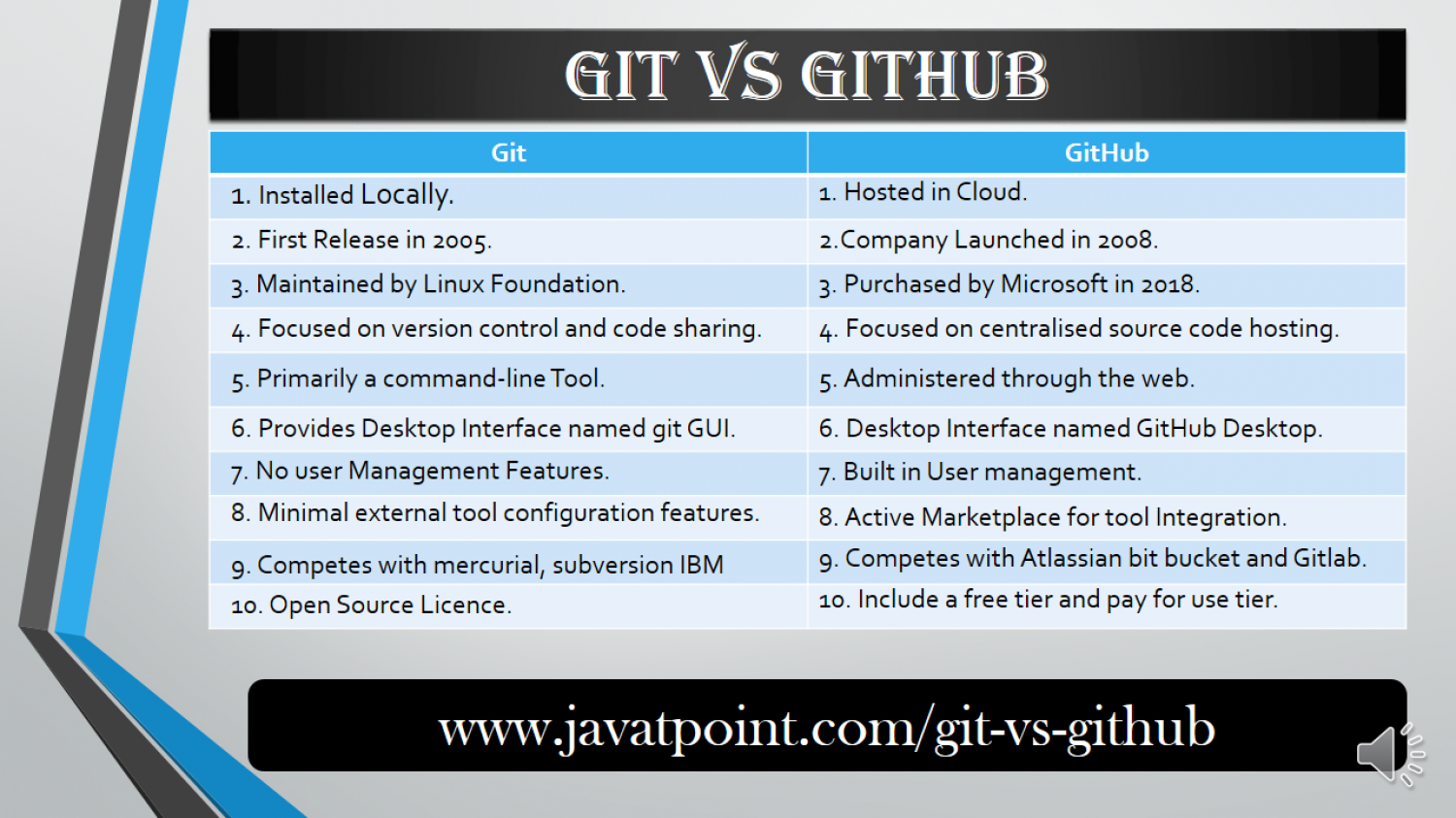
**Git**

Git is a free and open source distributed version control system that records changes to a file or a set of files overtime so that you can recall specific versions later. The changes are stored in special database called repository. It was created by Linus Torvalds in 2005. It provides the flexibility to view source code according to user’s need.

**GitHub**

GitHub is an online portal or a cloud-based online service that allows users to keep a track of the files. It enables developers to upload their own code files and to collaborate with fellow developers on open-source projects.

GitHub also serves as a social networking site in which developers can openly network, collaborate, and pitch their work.



**What is Repository?**

In git, the repository is like a data structure used by VCS to store metadata for a set of files and directories. It contains a collection of files as well as the history of changes made to those files. Repository in Git is considered as your project folder and repository has all project related data

**What is Version Control System (VCS)**

* **Centralized Version Control System**

In a centralized version control system (CVCS), server acts as the main repository which stores every version of code. Using centralized source control, every user commits directly to the main branch, so this type of version control often works well for small teams, because team members have the ability to communicate quickly so that no two developers want to work on the same piece of code simultaneously. Strong communication and collaboration are important to ensure a centralized workflow is successful.

* **Distributed Version Control System**

**Distributed version control** is a form of version control in which the complete codebase, , including its full history, is mirrored on every developer's computer

Task1

Setting up Git Client

For installing git go to [https://gitscm.com/download/win](https://git-scm.com/download/win). The most official build is available here

1.Click the download link for windows and allow the download to complete



2. Select the CPU for your system now(most of the computers now have 65-bit processors)



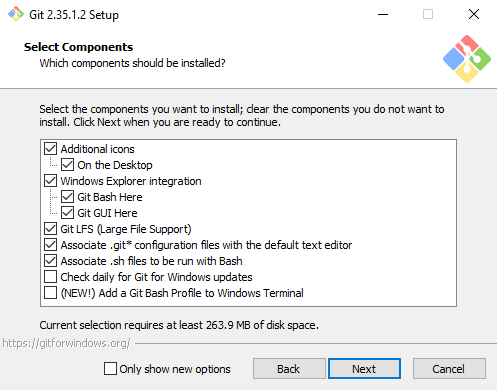
git2.png

**3. Once the git gets downloaded click the installer to install git**

**4. Review the GNU General Public License, and when you’re ready to install, click Next.**

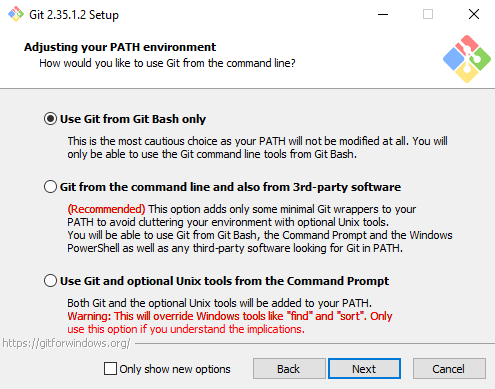


5.  **A component selection screen will appear. Leave the defaults unless you have a specific need to change them and click Next.**



6. Continue clicking next for few steps

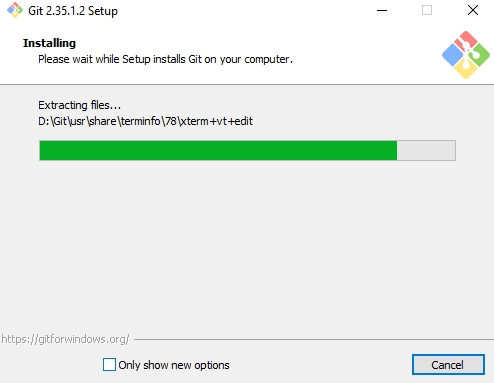
7. The next step is very important. It allows you to change the path environment



8. Continue clicking next for few more steps

9. Now click on the finish option

Now the git is installed in your local pc



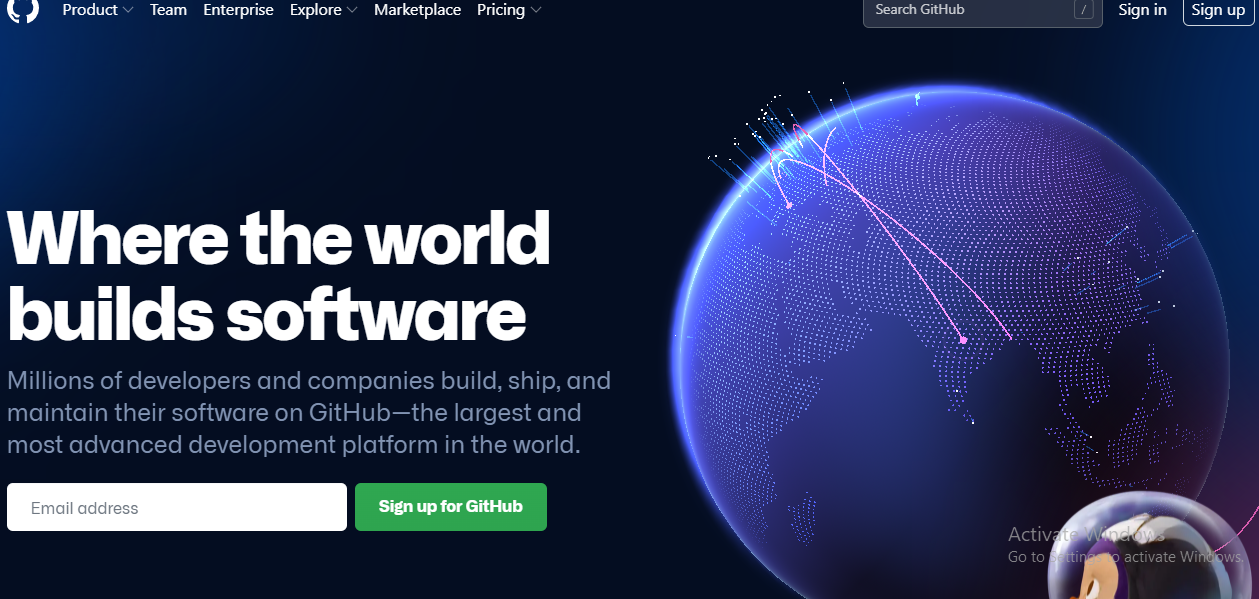
**Checking Git Version**

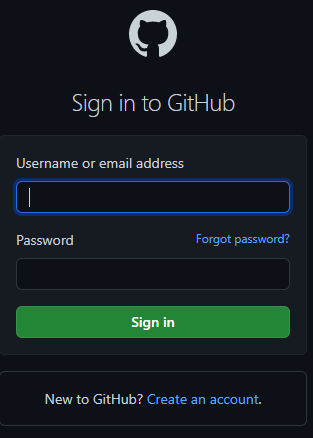
The command to check git version is :

* git --version**Task2**

**Setting Up GitHub Account**

* Go to the official page of github. ie <https://github.com>.

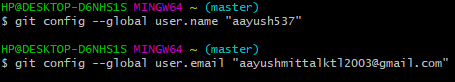


* If you already have an account then fill the required details. click on sign 
* If you are new to github then click on create new account. A dialog box appears



* Fill in the required details and your account will be created

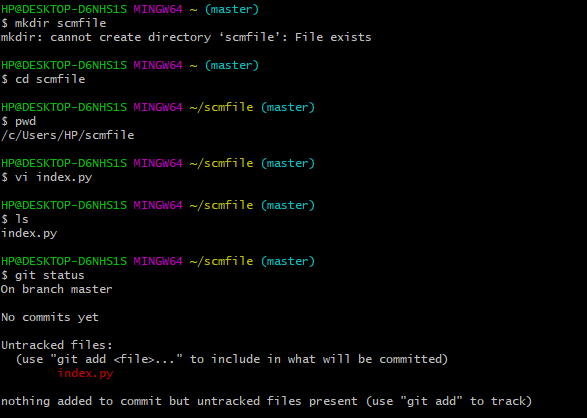
**Setting Up Username and email**

****

**Some Important Git Commands**

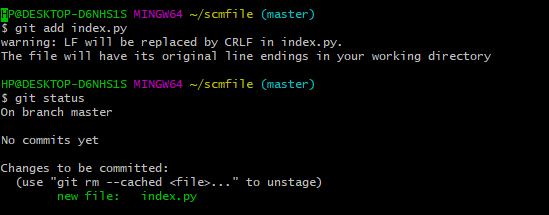
* **git config: It allows us to specify username and email address that will be used with our commits**
* **git init: It creates a new git repository**
* **git clone: It takes the part to the the git repository we want to clone**
* **git status: Display the status of working directoryand staging area. It let us see what changes have been staged , which files aren’t be tracked by git.**
* **git add: It is used to move file from working directory to staging area**
* **git commit: It saves a log message along with a commit id of the modifications made to the git repository**
* **git push : It push the content of local repository to remote repository we have added**
* **git branch: It is used to perform operartion on the parent branch**
* **git checkout : It is used to switch to an existing branch or to create or add new branch**
* **git merge :** **It joins the existing branch to the main branch**
* **rm –rf.git : It removes the repository**
* **touch filename: It creates a new file in the repository**

**Initializing Empty Git Repository**

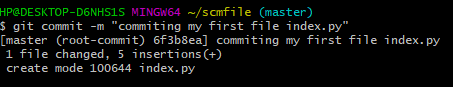
****

* **mkdir: create or make new directory**
* **git init : initialize empty git repository.**
* **Cd: change current directory**
* **Ls: list files or directory in current folder.**

**Adding Files to Staging Area**

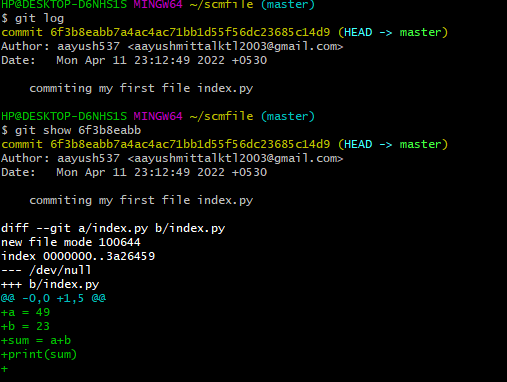
****

**Commiting Files**

****

**Generate logs**

* **git log:** It shows a list of all the commands made in a repository along with a hash id. It is used for displaying the history of a repository.
* **git log –onelne:**  Itis used to display the output as one commit per line.

****

**Task 1.4**

**Create and Visualize Branches**

**Git Branching:**

Branching is the practice of creating copies of programs in development to work in parallel versions, retaining the original and working on the branch or making different changes to each. The default branch is the master branch. Each copy is considered a branch; the original program from which the branch is taken is referred to as the trunk.



**Few Commands**

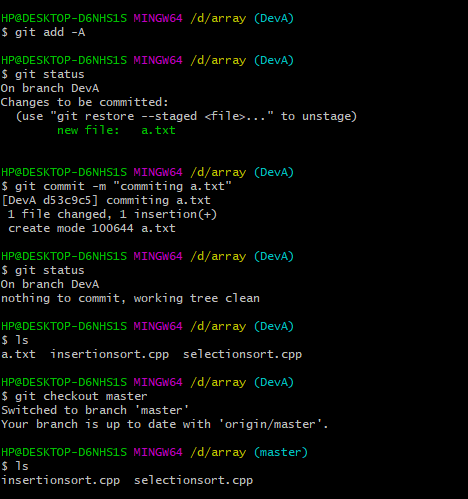
* **git branch:** show all existing branches
* **git branch<branch name>:** creating new branch
* **git checkout<branch name>:** use to switch branch

****

In this we can see that the default branch is master branch which is highlighted in green. Using git branch <branchname> command we have created a new command but the default branch is still master branch using git checkout command we changed the default branch. The current branch is now DevA.

**Parallel Branching**

**.Now let us create a new file in DevA andf compare it with the master branch**

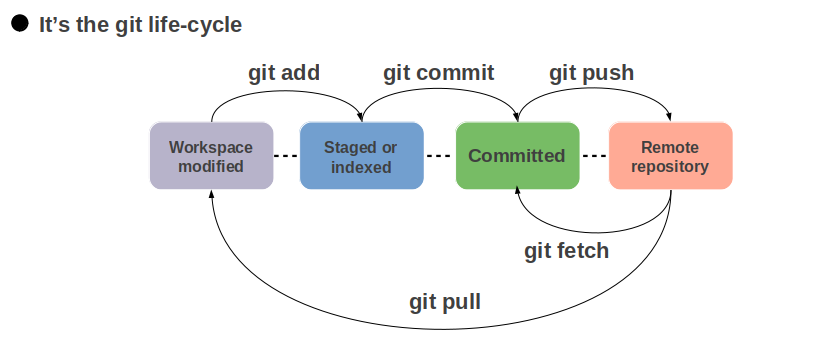
****

As you can see a.txt is present in DevA but not in master branch. This is how we can create parallel branches. Using git merge command we can merge DevA with master branch.

Git Lifecycle Discription

It is important to have a brief introduction about git before diving into much details. Git has three main stages that our file reside in:

1. Modified 2. Staged 3.Committed



****

**Working Directory:**

**It is a place where our project resides in local disk. Here the poject may or may not be tracked by git. The project can be tracked by the git using the command git init.**

**Staging Area:**

**It consists of the files which are to be a part of next commited. It is a place where different versions of our file are stored. It let git knows what changes in the fileare going to occur for the next commit. We can, however, choose which files we need to add to the staging area because in our working directory there are some files that we don’t want to get tracked. The command we use to stage file is git add<filename>**

**Git Directory**

**The .git folder contains all the information that is necessary for the project and allinformation related commits, remote reository address etc. It also contains a log that stores a commit history. This log can helpyou to rollback to the desired versions of the code**