**Git and GitHub part 1 ( 18.09.23 , Monday )**

**Git:**

Git is a distributed version control system (VCS) that allows developers to track changes in their codebase over time. It was created by Linus Torvalds in 2005 and has since become the de facto standard for version control in the software development industry.

Git enables developers to:

Record and track changes to their code, making it easy to revert to previous versions if needed.

Collaborate with others by merging changes made by different team members.

Work on different branches to develop new features or fix bugs independently.

Maintain a history of all changes made to the code, including who made the changesand when.

**GitHub:**

GitHub is a web-based platform built on top of Git, providing a centralized hub for code hosting, collaboration, and project management. It was founded in 2008 by Tom Preston-Werner, Chris Wanstrath, and PJ Hyett.

GitHub offers the following features:

Code hosting: Developers can store their Git repositories on GitHub, making it easy to share code with others.

Collaboration tools: GitHub provides features like pull requests, issues, and code reviews to facilitate collaboration among team members.

Version history: Every change made to a repository on GitHub is tracked, allowing users to view the complete history of a project.

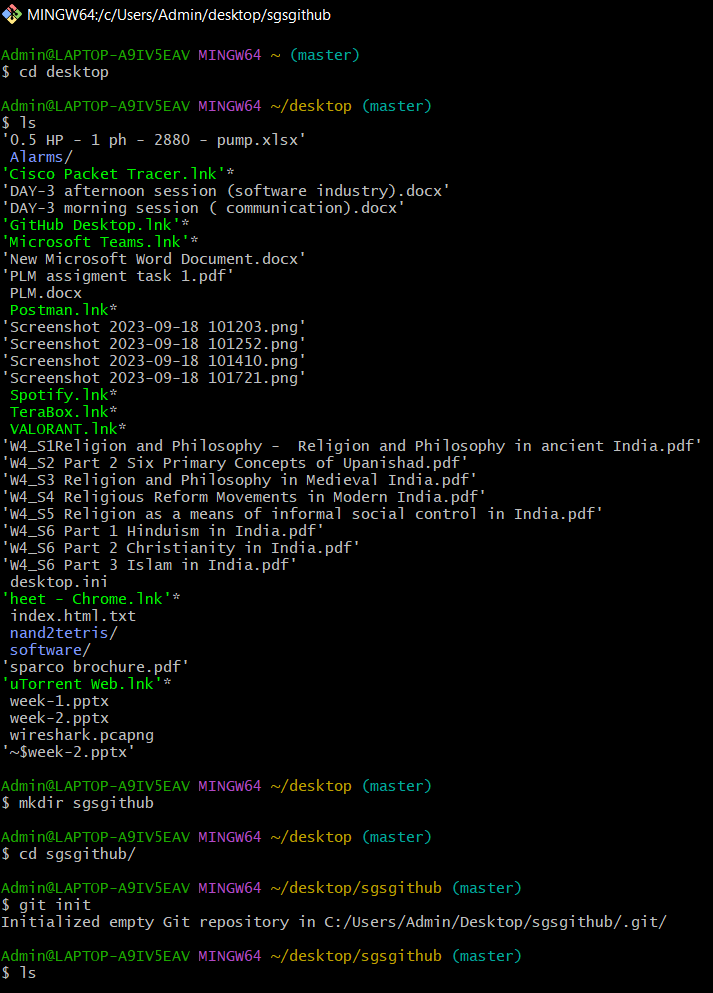
Access control: GitHub offers granular access control, allowing project owners to specify who can view, edit, or contribute to a repository.

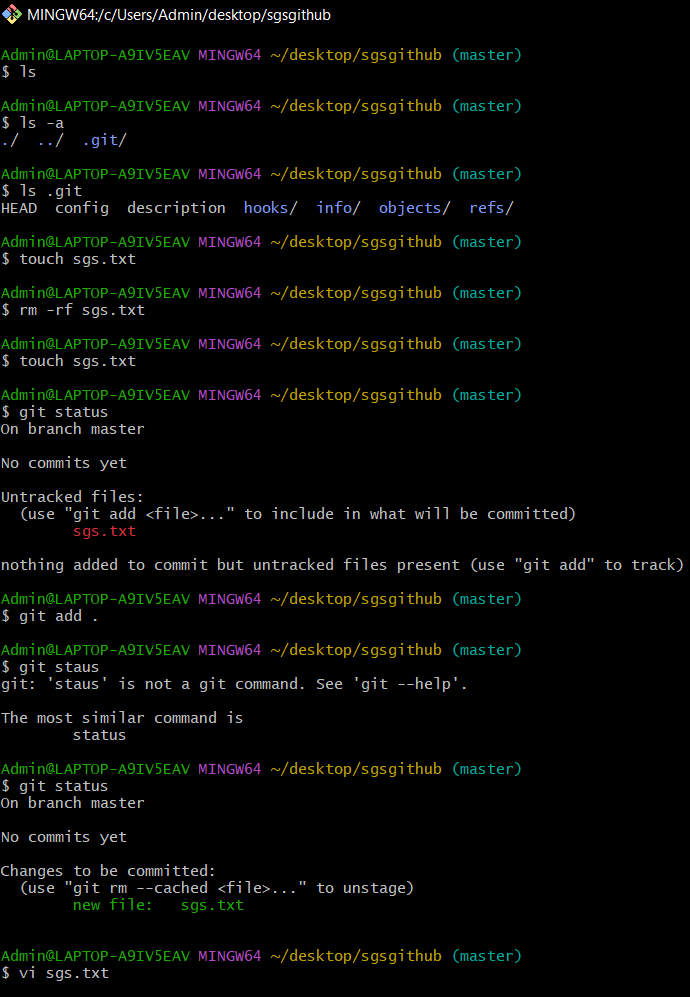
Continuous integration and deployment (CI/CD) integration: GitHub can be integrated with CI/CD tools to automate testing and deployment workflows.

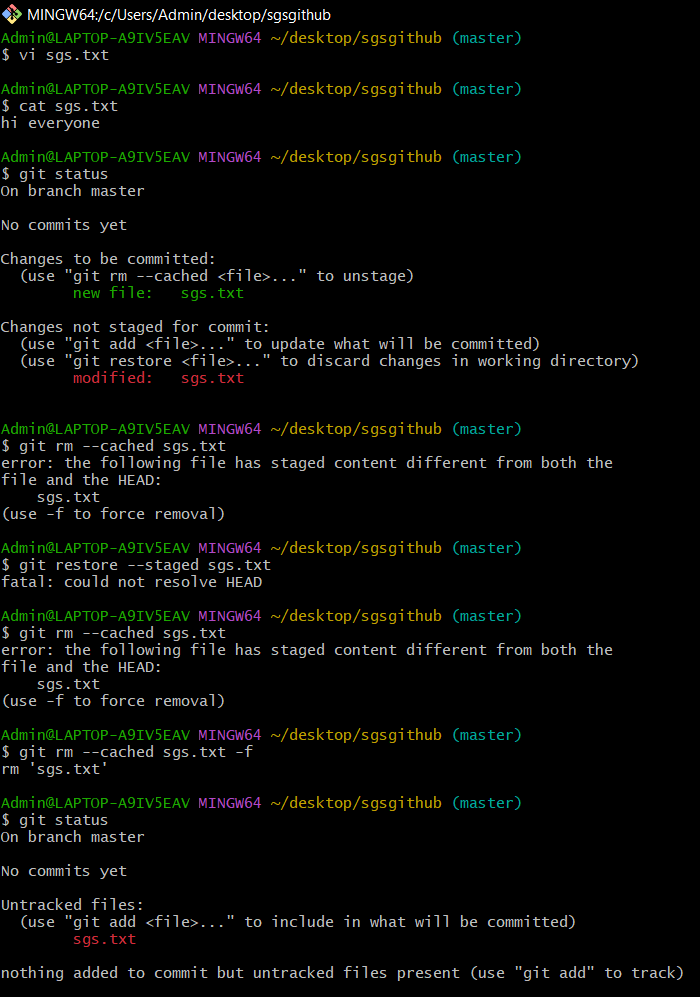
**Difference between add , push , commit**

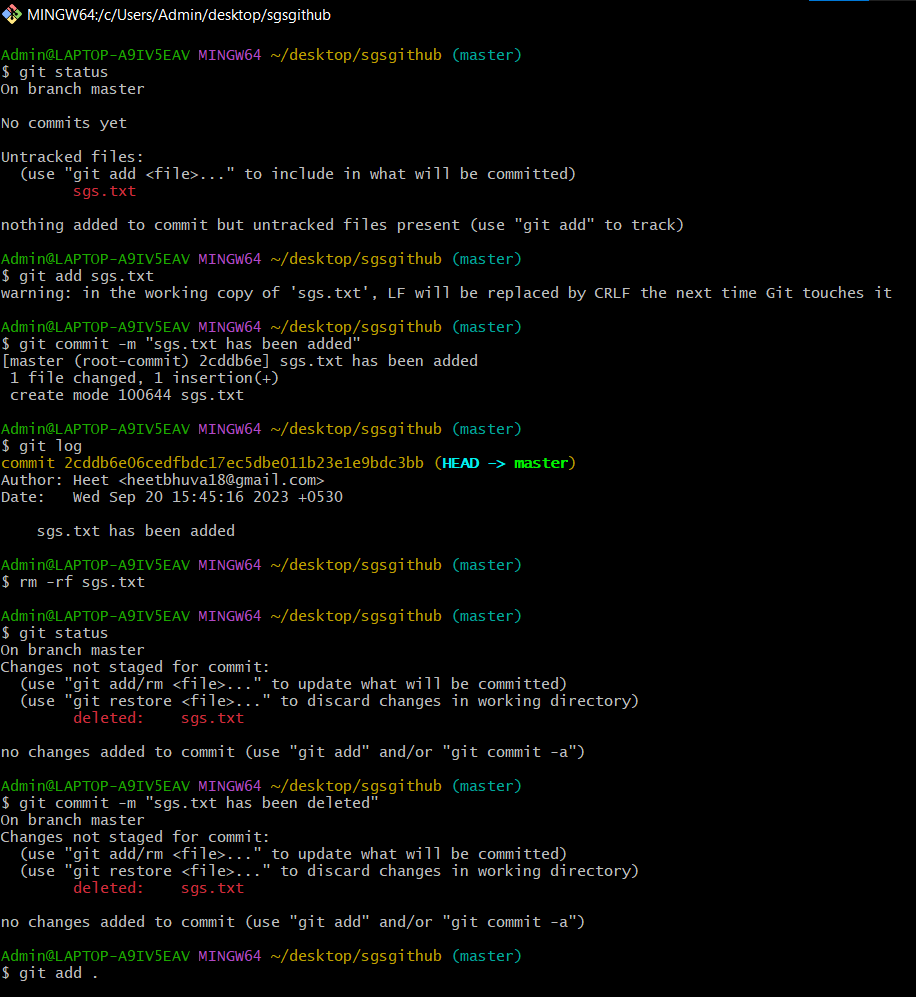
* **Add -** The git add command stages changes in your working directory for the next commit . Staging means that you are telling Git to include these changes in the next commit. It prepares the changes to be committed but doesn't actually create a new commit.
* **Commit -** The git commit command creates a new commit with the changes that have been staged using git add. Each commit represents a snapshot of the project at a specific point in time. It includes a commit message that describes the changes made in that commit. Commits are essential for tracking the history of your project and providing a record of what was done and why.
* **Push -** The git push command is used to upload (or "push") the commits from your local repository to a remote repository, typically on a server (like GitHub, GitLab, or Bitbucket). Pushing is necessary when you want to share your changes with others or keep your remote repository up-to-date. It's important to note that git push is used for working with remote repositories, whereas git add and git commit are used for managing changes in your local repository.

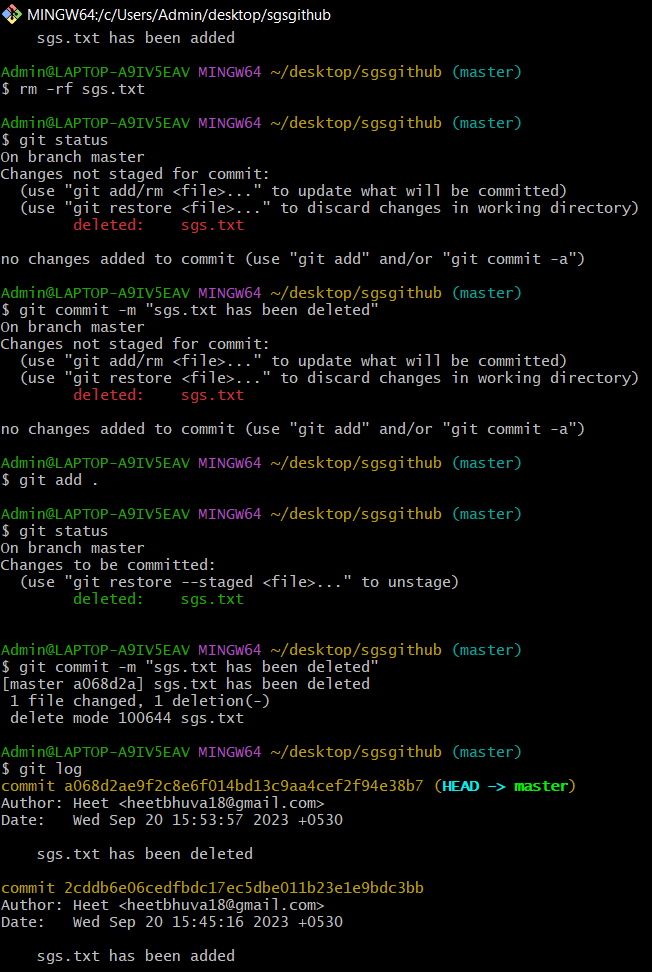
**Some commands what sir taught us and I practiced it**











**Cd – change directory**

**Ls – list all files of that directory**

**Mkdir – make directory**

**git init - initialize an existing directory as a Git repository**

**git status - show modified files in working directory, staged for your next commit**

**git add [file] - add a file as it looks now to your next commit (stage)**

**git commit -m “[descriptive message]” - commit your staged content as a new commit snapshot.**

**git branch - list your branches. a \* will appear next to the currently active branch**

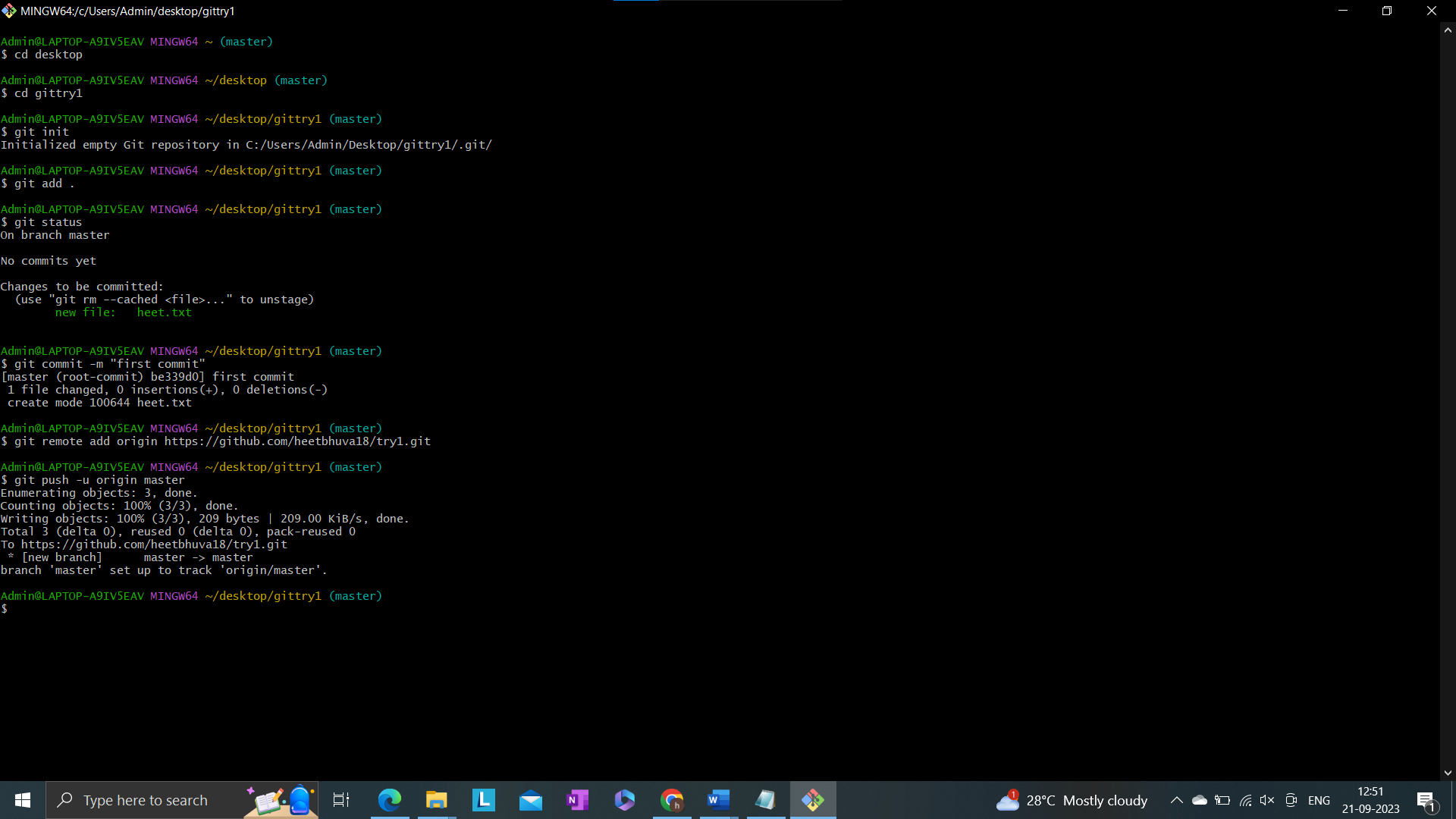
**git branch [branch-name] - create a new branch at the current commit**

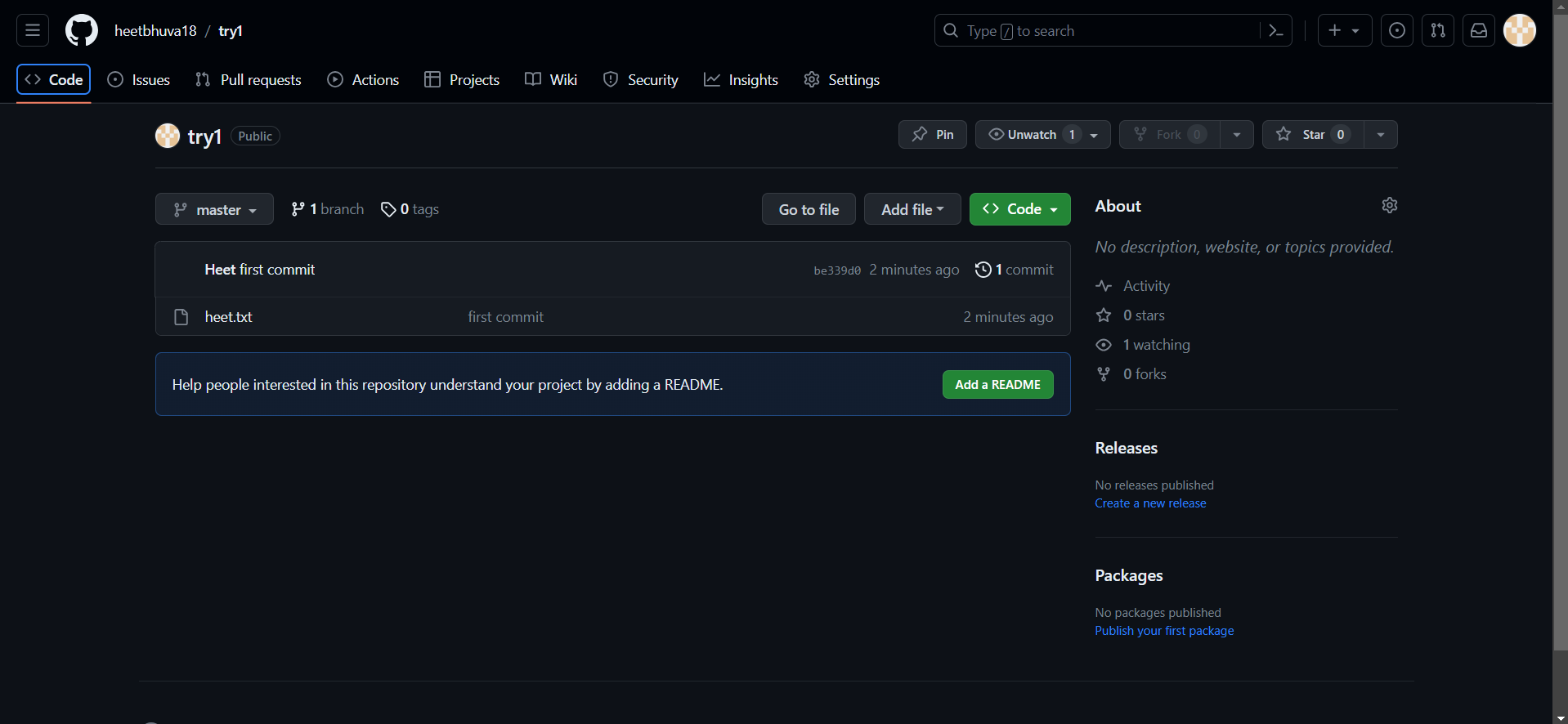
**git checkout - switch to another branch and check it out into your working directory.**

**git log - show all commits in the current branch’s history.**

**Git push -u .. – to push a file into the repository.**

**Then we did that how to push a file in the GitHub repository -**





**Stashing –** In the context of version control systems like Git, "stashing" refers to the action of temporarily saving changes in your working directory that are not ready to be committed. Stashing is useful when you are working on a branch and need to switch to another branch or apply changes from one branch to another without committing your current changes.