**GIT ASSIGNMENT**

**Git:** It is a version control system for tracking changes in the source code during software development. Git is used to collaborate on code. The version control allows us to track and work together with our team members at the same workspace. Every time we make changes in files of an existing project, we can push those changes to a repository. Other developers are allowed to pull your changes from the repository and continue to work with the updates that you added to the project files.

**Git Bash**: It is an application that provides a command line interface on the operating system

**Git Bash commands:**

1. **$ git config –global user.name username**

**$ git config –global user.email user\_email**

Before using Git you need to set your Global Username and Email using the above commands.

1. **$ mkdir FolderName**

Using this command you can create an empty folder for your project.

1. **$ cd FolderName**

After creating a folder,we use cd command to change the current directory to the newly created folder.

1. **$ git init**

The git init command is used to create a new blank repository. It is used to make an existing project as a Git project.

1. **$ git status**

The git status command displays the state of the working directory and the staging area. It lets you see which changes have been staged, which haven't, and which files aren't being tracked by Git.

1. **$ touch f1.txt**

The touch command creates new, empty files.

1. **$ start f1.txt**

This command will open the file in the supported application

1. **$ git add .**

The git add command is used to add file contents to the Staging area.

However, git add doesn't really affect the repository in any significant way, changes are not actually recorded until you run git commit .

1. **$ git status**

This command will now show you the changes made in the working directory. It will display that the newly created file is in the staging area, and the changes are not yet committed.

1. **$ git commit -m “Comment”**

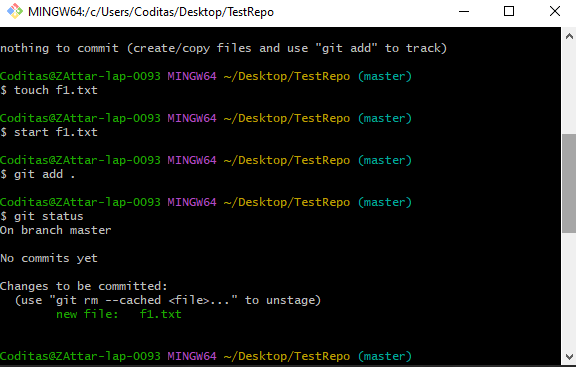
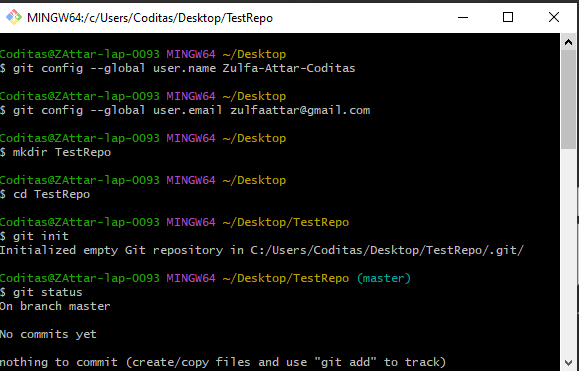
This command will commit and save all the changes you have made in the directory. Using -m we can add a message to describe our commit.

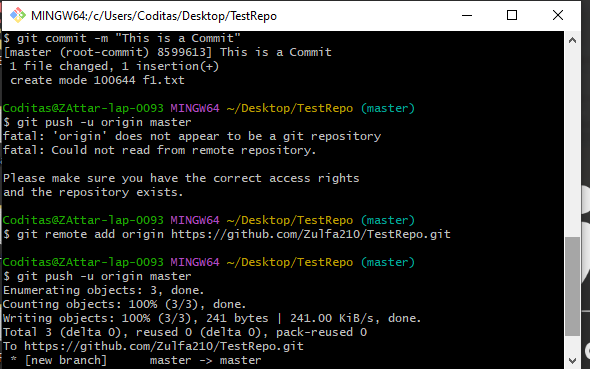
1. **$ git push -u origin master**

Whenever we need to push the changes to a remote repository, we use git push along with the remote repository “origin” and “master” branches. Origin is simply the name given to any remote repository available on GitHub.

1. **$ git pull**

The git pull command is used to fetch and download content from a remote repository and immediately update the local repository to match that content.





**Q. 1) How to add an existing folder in a Git repository?**

1.Create a repository on Github.

2. Go to your project folder :

$ cd /path/to/my/project

3. Add your project files to the repository by adding these commands in git Bash:

$ git init

$ git add .

$ git commit -m "Commit Message"

4. Write the following command, replace <username> with your Github username and <repository> with your repository name :

$ git remote add origin https://github.com/<username>/<repository>.git

5. Push the files on the remote server

$ git push -u origin master

**GITHUB FUN ACTIVITY**

We conducted a fun activity on github, in order to understand the concepts of github in a better way. We created a Git Repository and all the team members were added as contributors. The repository had 2 text files. One file had Names and the other had characters. All the team members were told to push their content randomly, which led to various issues and conflicts. Later on the conflicts were resolved.

