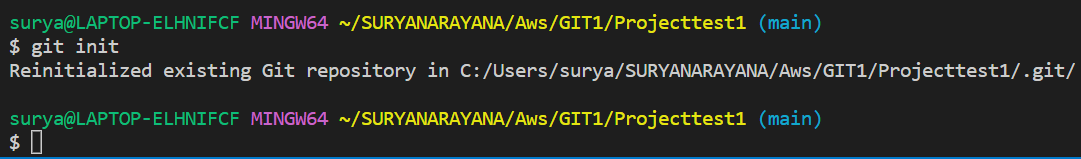
**Task 1**

* Demonstrate minimum 15 basic Git command with explanation and screenshot.

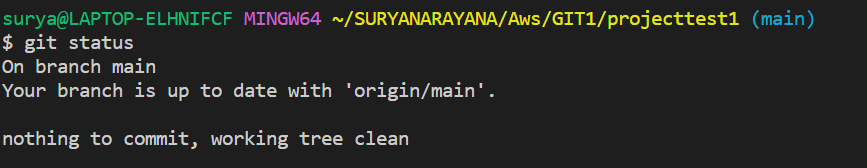
1. **Command: git init [repository name]**

our project directory and type the command **git init** to initialize a Git repository for our local project folder.



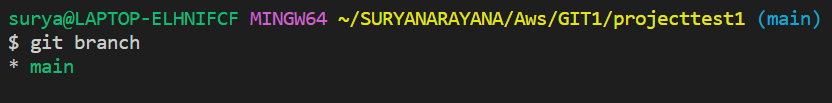
1. **Command: git status**

This command will show the modified status of an existing file and the file addition status of a new file, if any, that has to be committed.



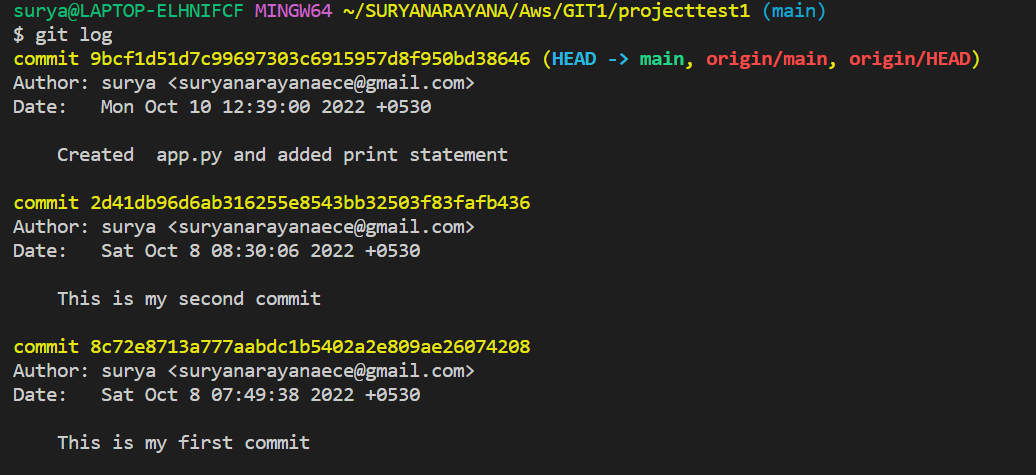
1. **Command: git branch**

Multiple developers working on the same project or repository! To handle the workspace of multiple developers, we can use branches.



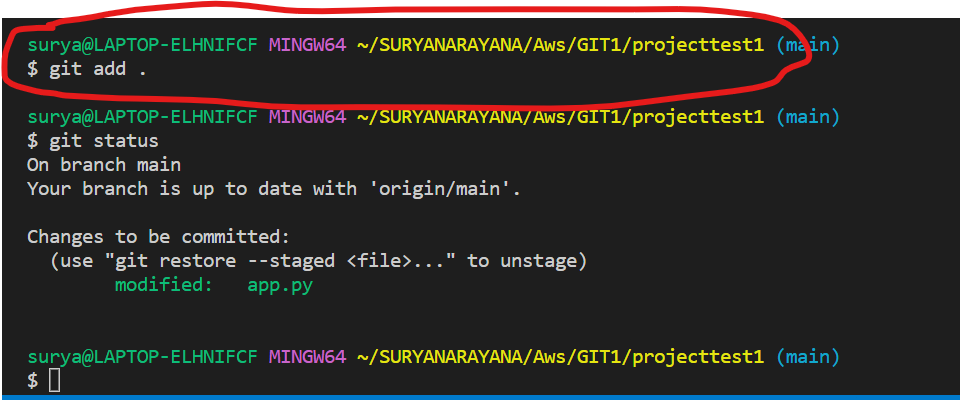
1. **Command: git log**

This command is used when we want to check the log for every commit in detail in our repository

****

1. **Command: git add. (All files) or git add (specific file name)**

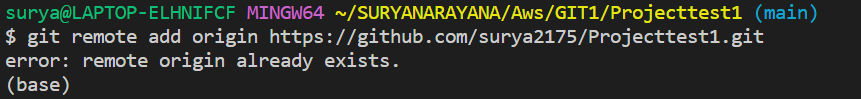
This will add the specified file into the Git repository, the staging area, where they are already being tracked by Git and now ready to be committed.

****

1. **Command : git remote add origin “url”**

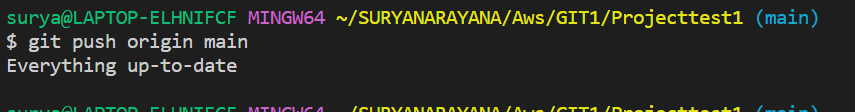
Once everything is ready on our local system, we can start pushing our code to the remote (central) repository of the project. For that, follow the below steps:

****

****

1. **Command: git push origin “branch name”**

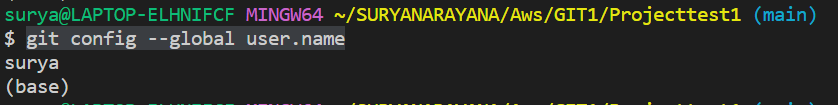
Suppose, we have made some changes in the file and want to push the changes to our remote repository on a particular branch. By using the command ‘git push,’ the local repository’s files can be synced with the remote repository on Github.

****

1. **Command: git config --global user.name “username”:**

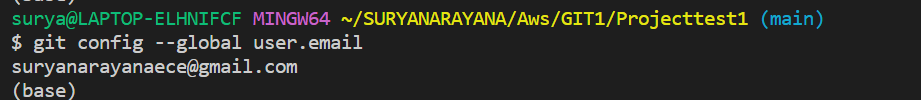
**To set Git username, run the git config –global user.name command**. One should specify both first and last name but username can be anything that want to attach

commits.

****

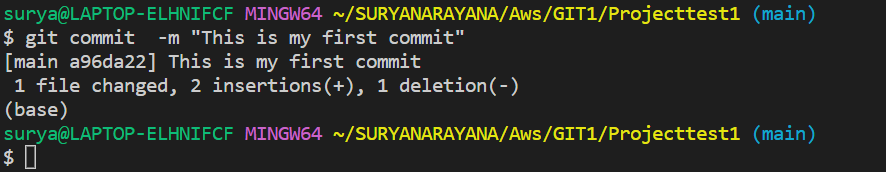
1. **Command: git config --global user.email “emailadress”:**

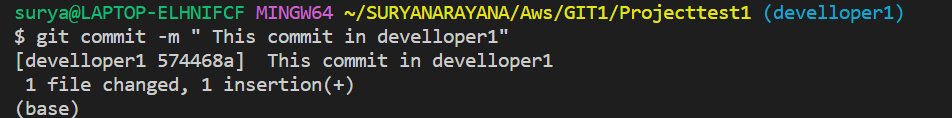
The global git email address are **associated with commits on all repositories on system that don't have repository-specific values**. We can also edit the file with any text editor, but it is recommended to use the git config command.



10)**Command: git commit -m "This is my first commit"**

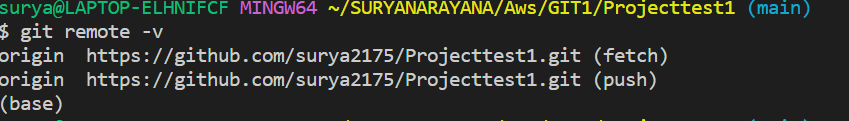
Commit a snapshot of all changes in the working directory. this only includes modification to tracked files.



****

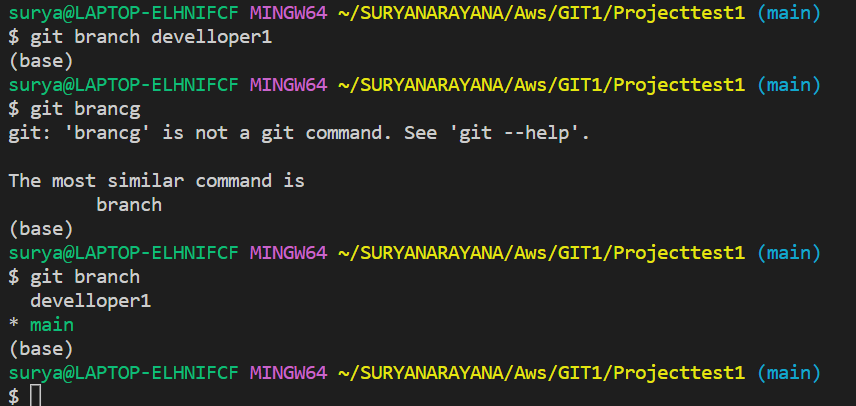
* 1. **Command:** **git remote -v**

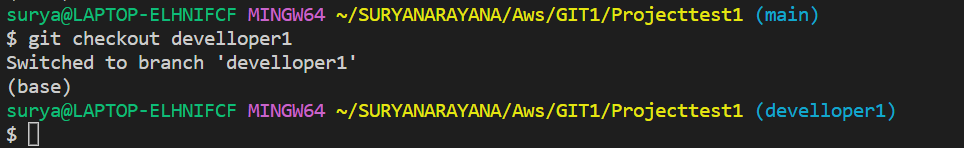
If we want to see what tracking branches which we set. which are fetch and push branches.

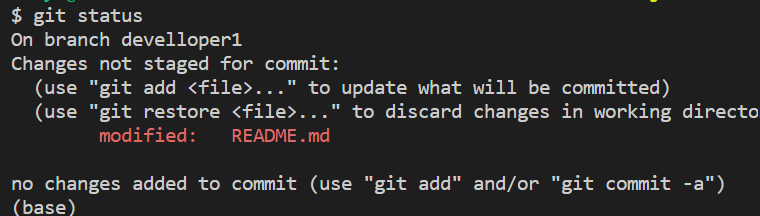


* 1. **Command: git checkout developer1 (git branch develloper1 )**

This command created new branch which developer1 and once we created checkout command switches dvelloper1 and main branches. The git checkout command lets navigate between the branches created by git branch. Checking out a branch updates the files in the working directory to match the version stored in that branch, and it tells Git to record all new commits on that branch.

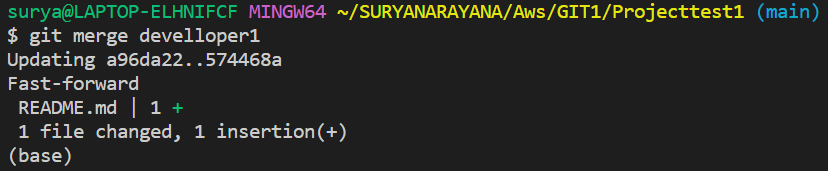
****

****

****

* 1. **Command: git merge “branchname”**

To merge branches locally, use git checkout to switch to the branch you want to merge into. This branch is typically the main branch. Next, use git merge and specify the name of the other branch to bring into this branch

****

* **14) Command :** **git push origin main**

To push the branch or you can say to push the changes in the branch to the Github repo you have to run this command “**git push origin <the branch name>**” in our case the branch name is “main”. After pushing the changes, the repo will look like and this is how you can push a branch to a remotely hosted GitHub repository

****

**15) Command : Git pull origin main**

the git pull origin master will pull changes from the origin remote, master branch and merge them to the locally checked-out branch.

****