INTRODUCTION TO DEVOPS

Assignment 1

**Github**

Prepared by :

Session : 2023-2024

**Table of Contents**

Contents

[**Installation: GIT** 3](#_Toc164559399)

[Creating account on GitHub 3](#_Toc164559400)

[**Create repository in github and clone it in local environment** 4](#_Toc164559401)

[ **Configuring git on Git Bash** 4](#_Toc164559402)

[ **Create a repository and working directory:** 4](#_Toc164559403)

[ **Initiate the repository and create file masterbranch:** 5](#_Toc164559404)

[ **Check the status of the file** 5](#_Toc164559405)

[**Create a new repository “wimsrepo”** 6](#_Toc164559406)

[ **In Git Bash connect Git hub** 6](#_Toc164559407)

[**Add new file in local environment and push it to github** 7](#_Toc164559408)

[**Create a branch in local environment and make changes** 8](#_Toc164559409)

[ **Using Pull command download the files from github** 8](#_Toc164559410)

[**Clone the github into local git** 10](#_Toc164559411)

[ **Git log** 11](#_Toc164559412)

[**Merge changes with main branch and push it to github** 12](#_Toc164559413)

[ **Git checkout for moving to another branch** 12](#_Toc164559414)

[ **Create a new file mul.cs under multiply branch** 13](#_Toc164559415)

[ Assume there is issue in sub.cs 13](#_Toc164559416)

[ **Find the difference between two branches “main” and “newissueinsub” after merging** 14](#_Toc164559417)

[**Github Part2** 15](#_Toc164559418)

[Introduction 15](#_Toc164559419)

[Clone Repo 15](#_Toc164559420)

[ **Create a branch and Push to remote** 16](#_Toc164559421)

[ **Switch to Main branch** 17](#_Toc164559422)

[ **Merge Branch1** 18](#_Toc164559423)

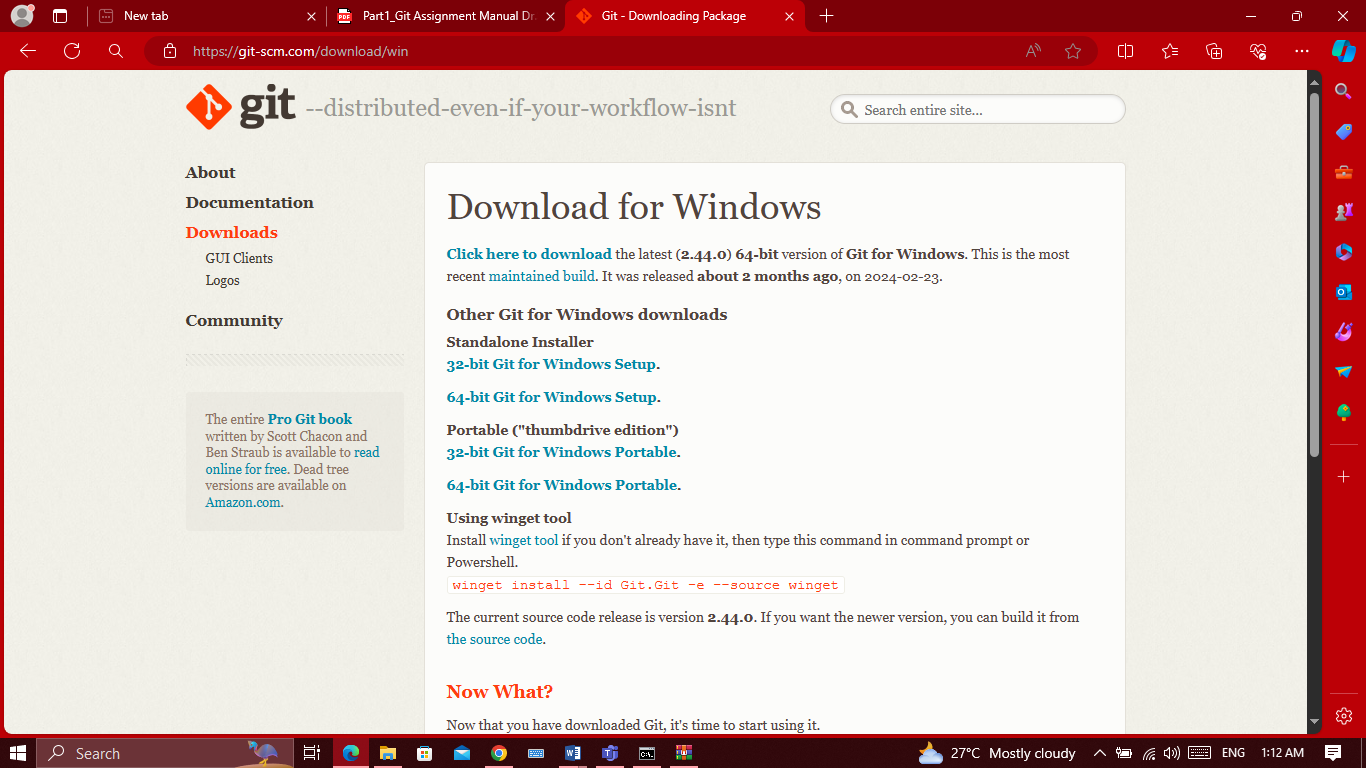
[**Resolve conflict** 19](#_Toc164559424)

[**Git push : Final conclusion** 20](#_Toc164559425)

# **Installation: GIT**

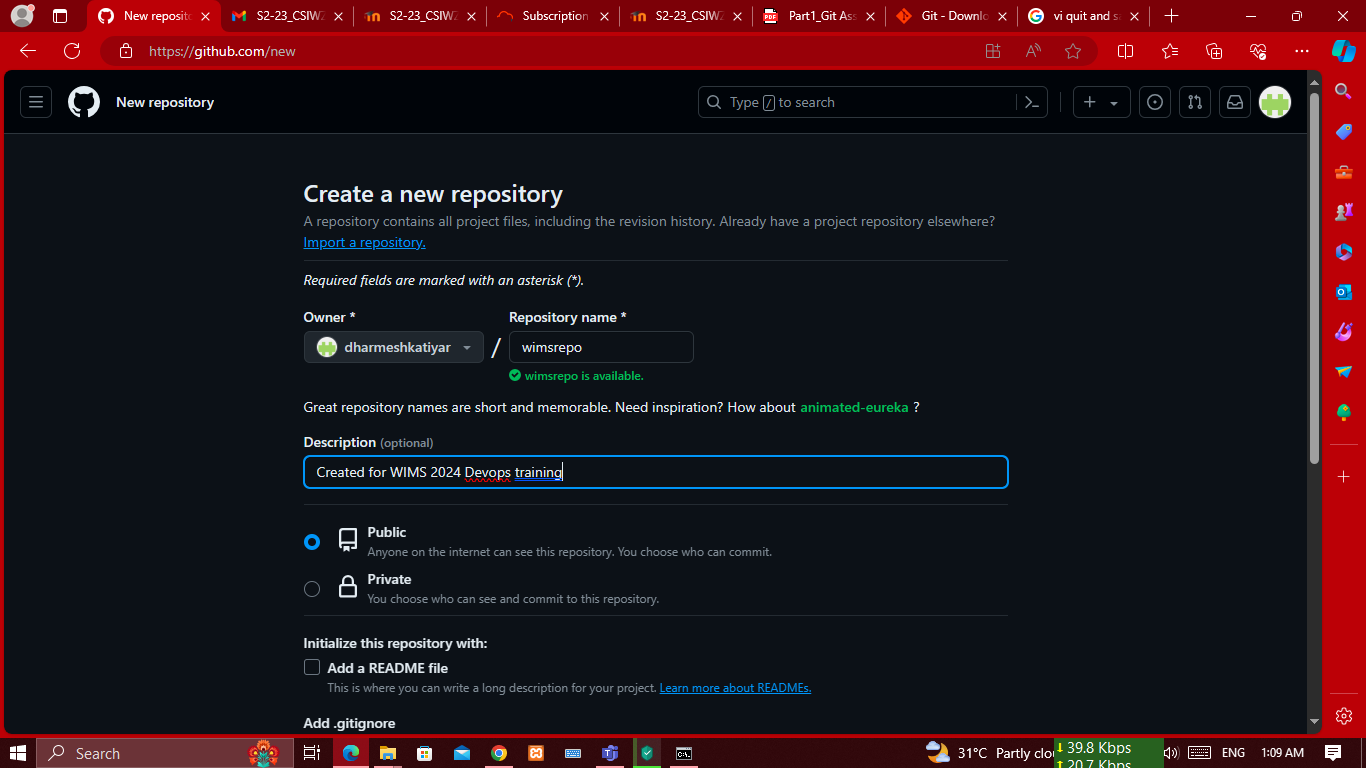
As mentioned in the pdf, GIT has been installed in the windows local environment with the help of snapshots of each step.

Link to download git : <https://git-scm.com/>



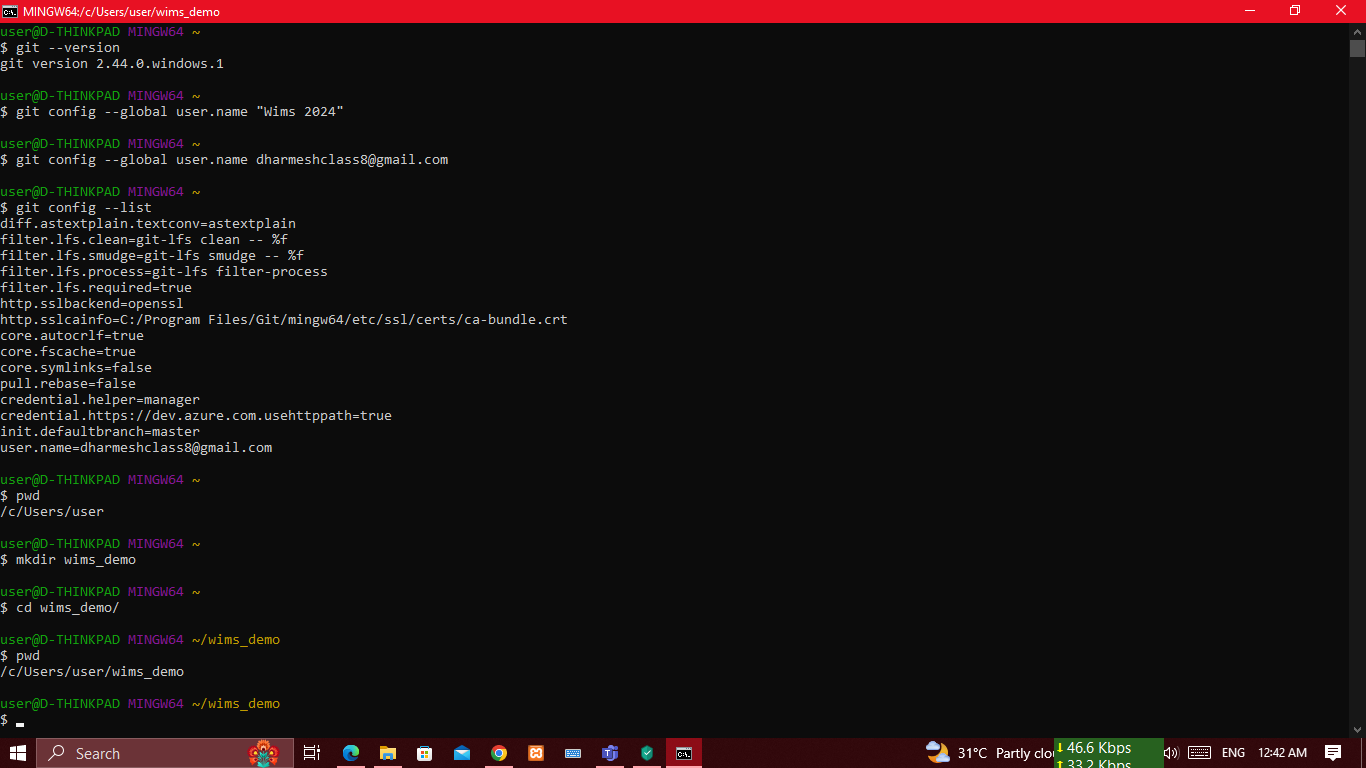
# Creating account on GitHub

For pushing and pulling the repository, we need to create account on github as mentioned below:

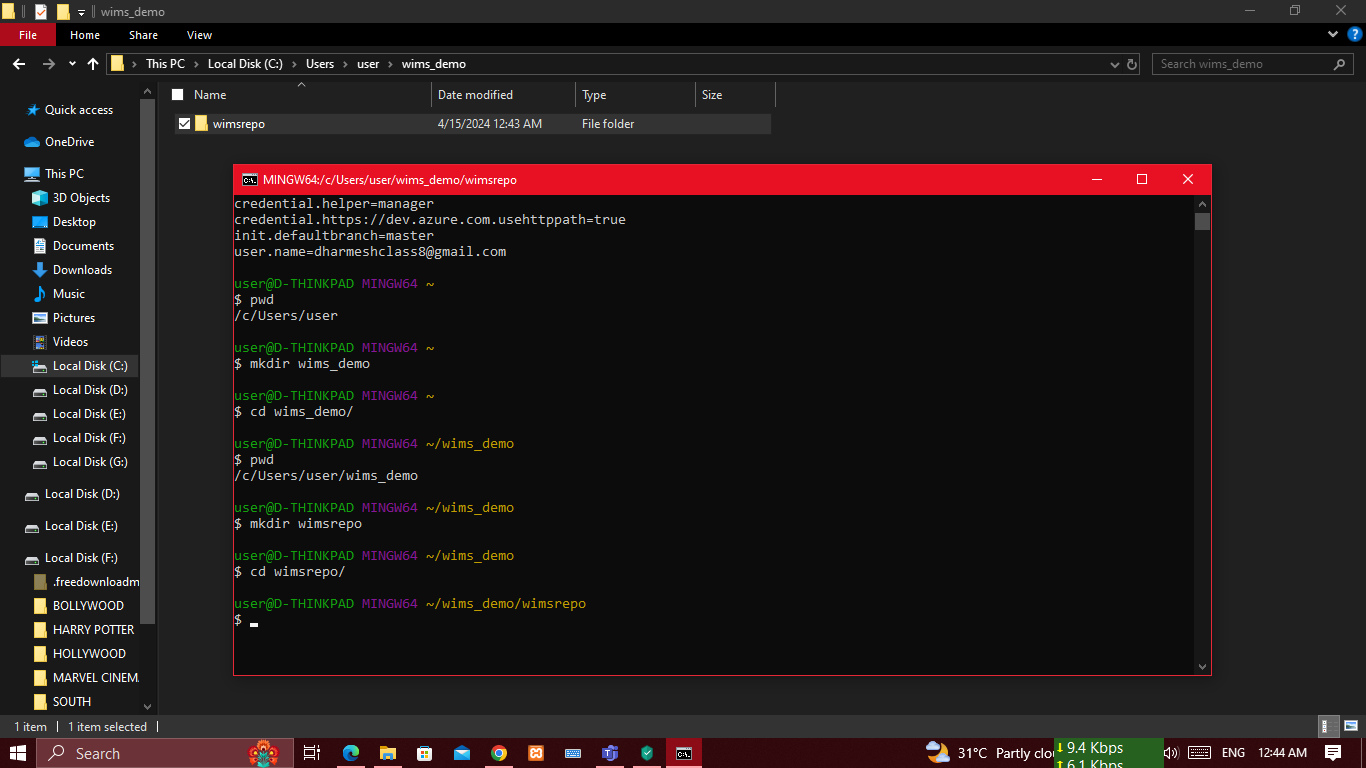


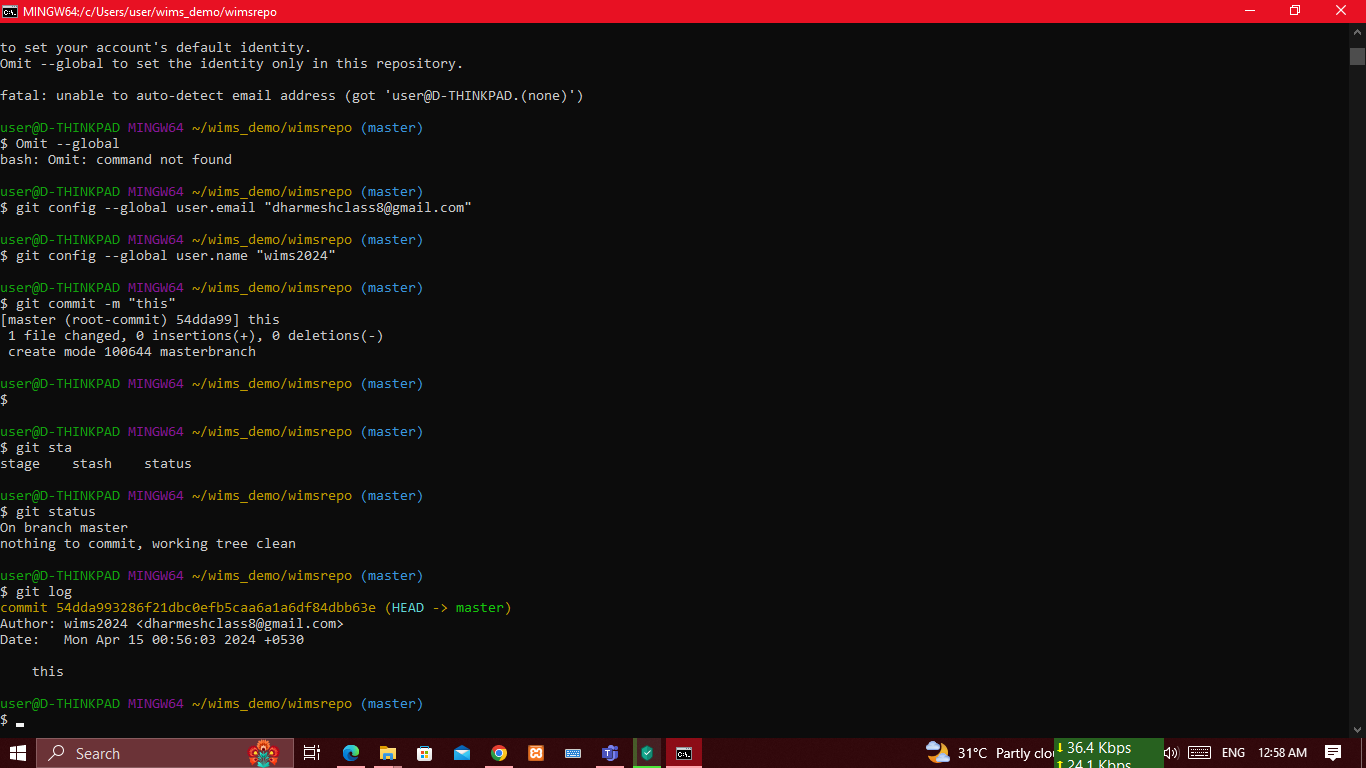
# **Create repository in github and clone it in local environment**

### **Configuring git on Git Bash**



### **Create a repository and working directory:**





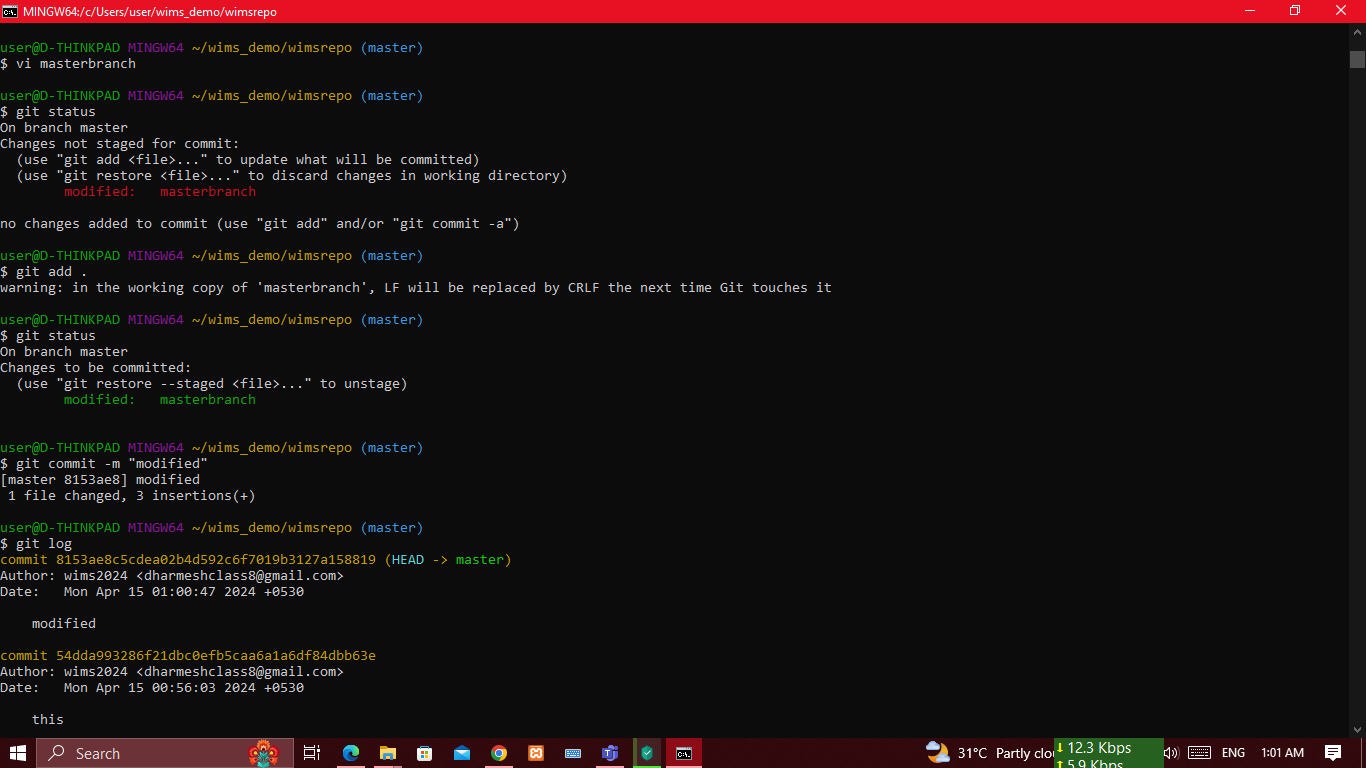
### **Initiate the repository and create file masterbranch:**

We can initiate the repository by following command -

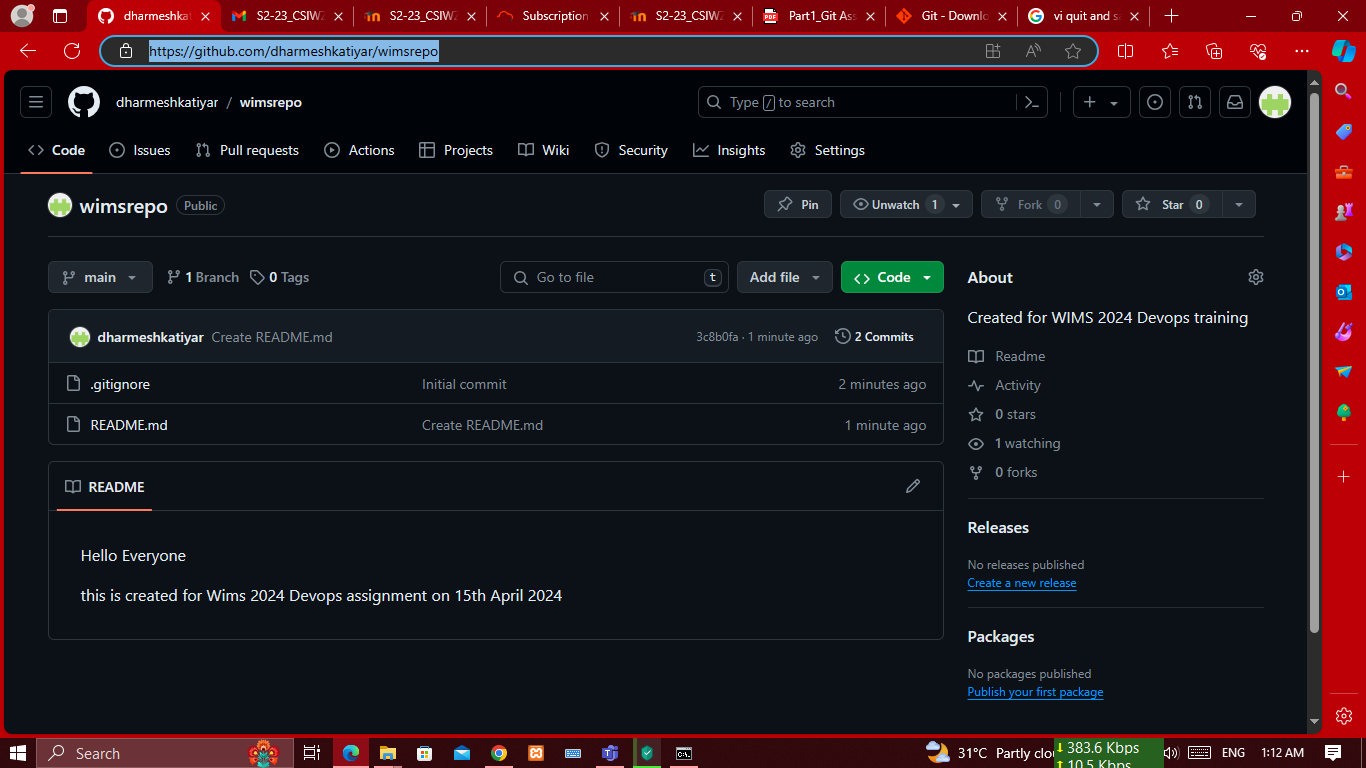
Cmd> git init

$ touch masterbranch

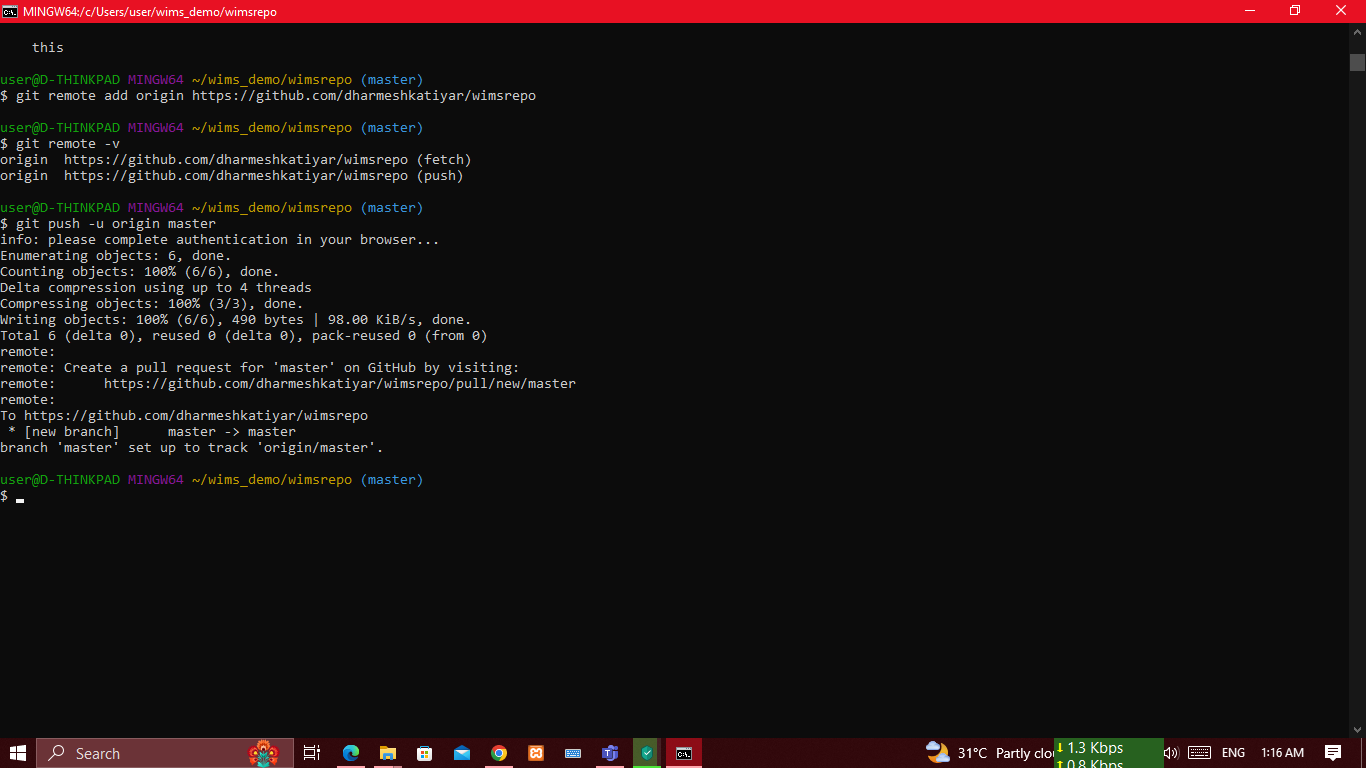
### **Check the status of the file**



# **Create a new repository “wimsrepo”**

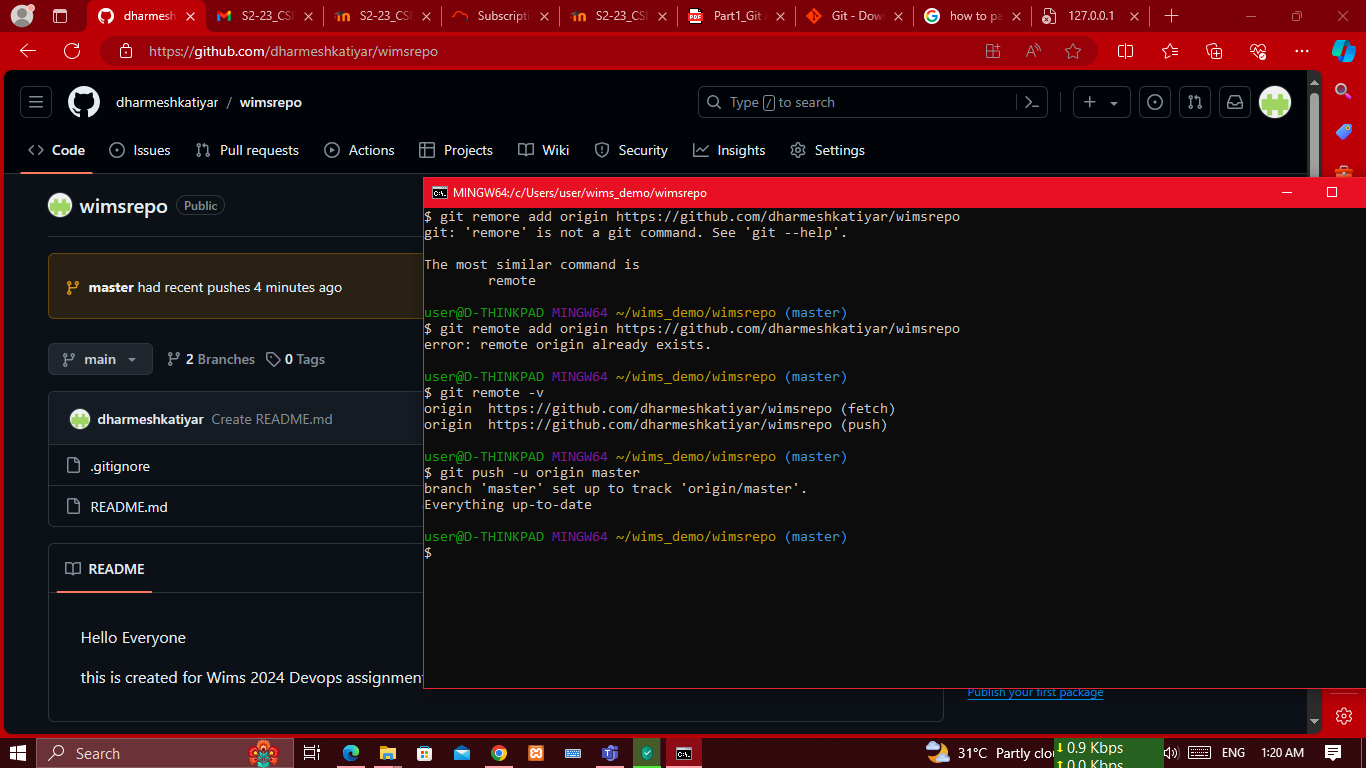
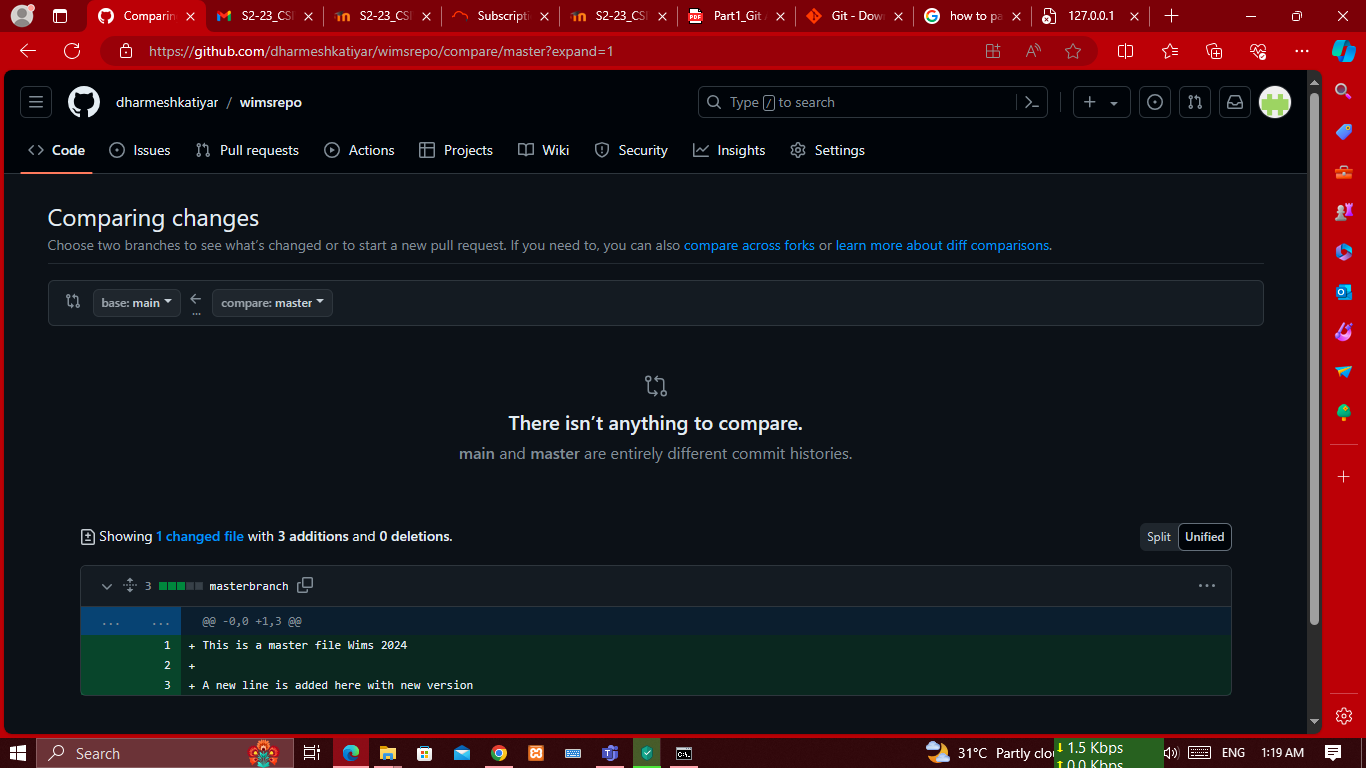
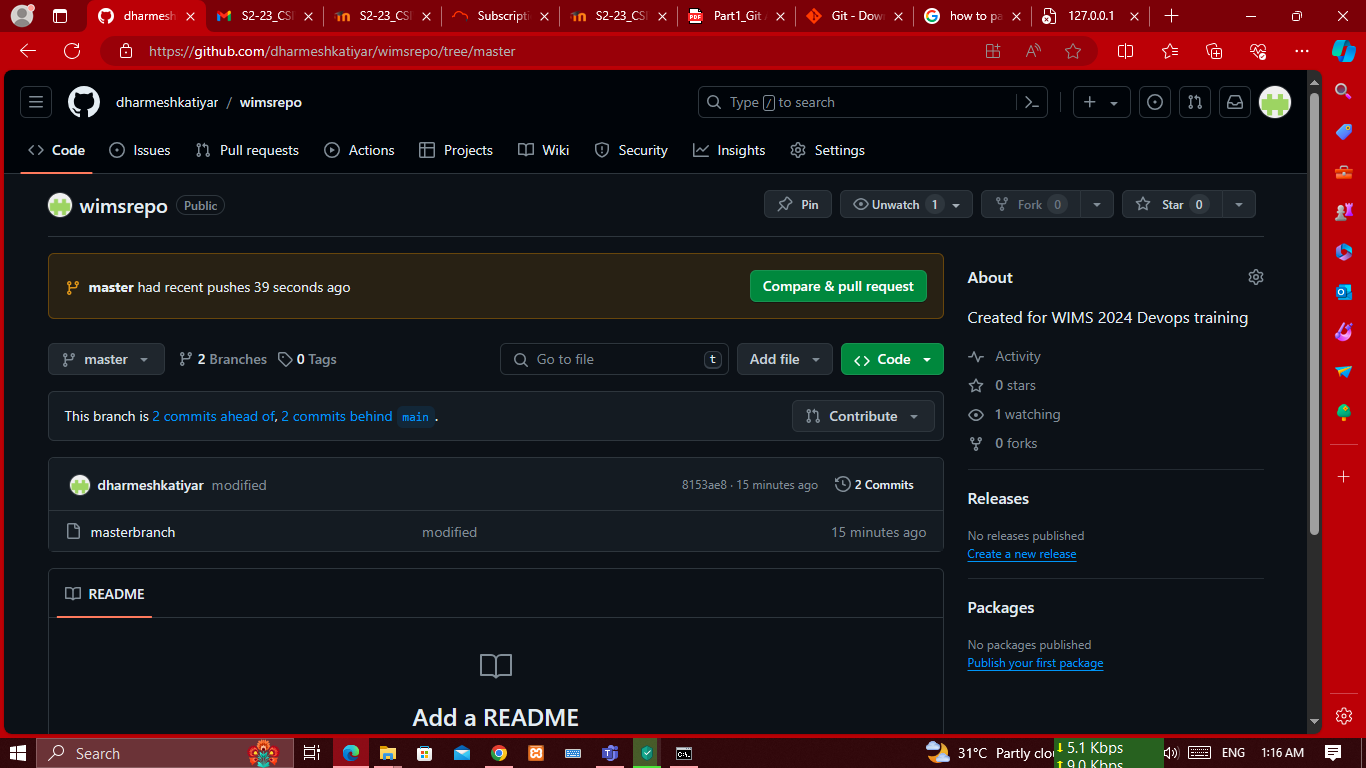


### **In Git Bash connect Git hub**

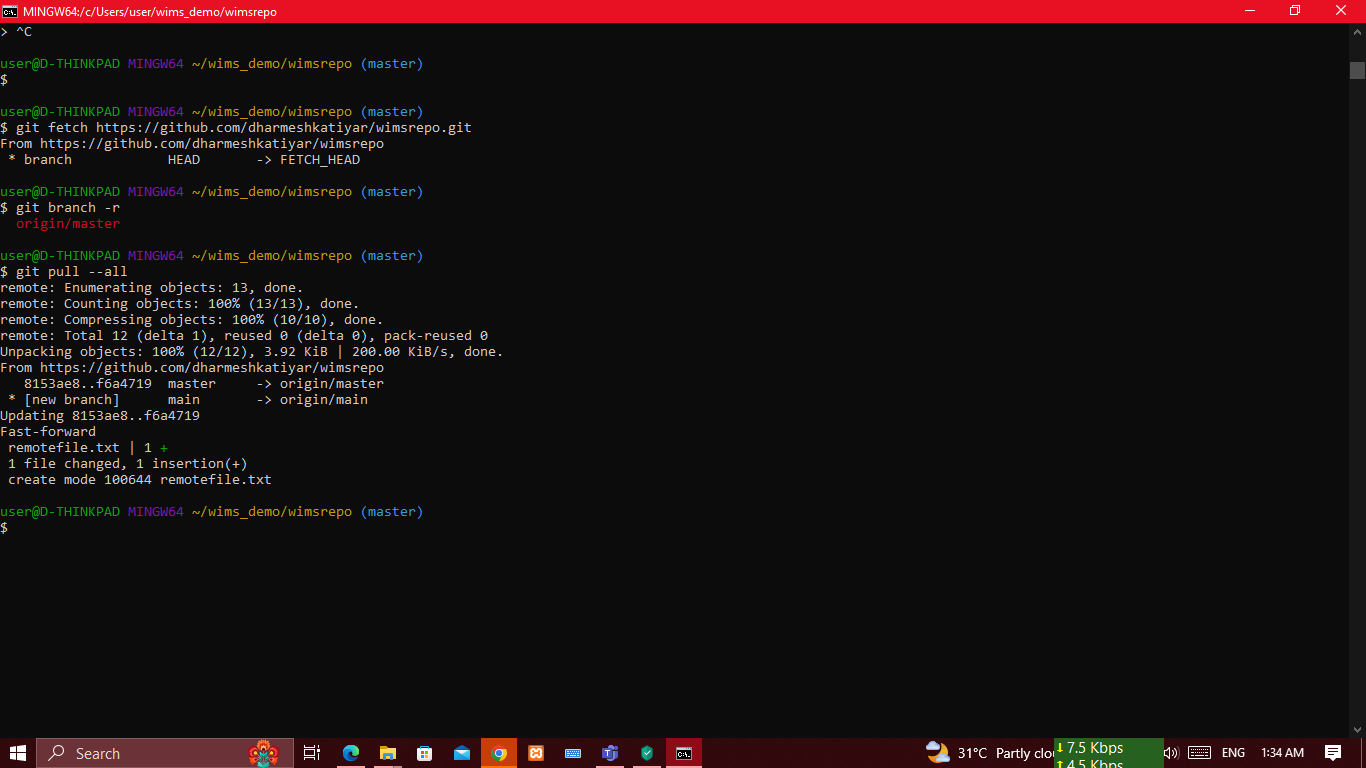


# **Add new file in local environment and push it to github**

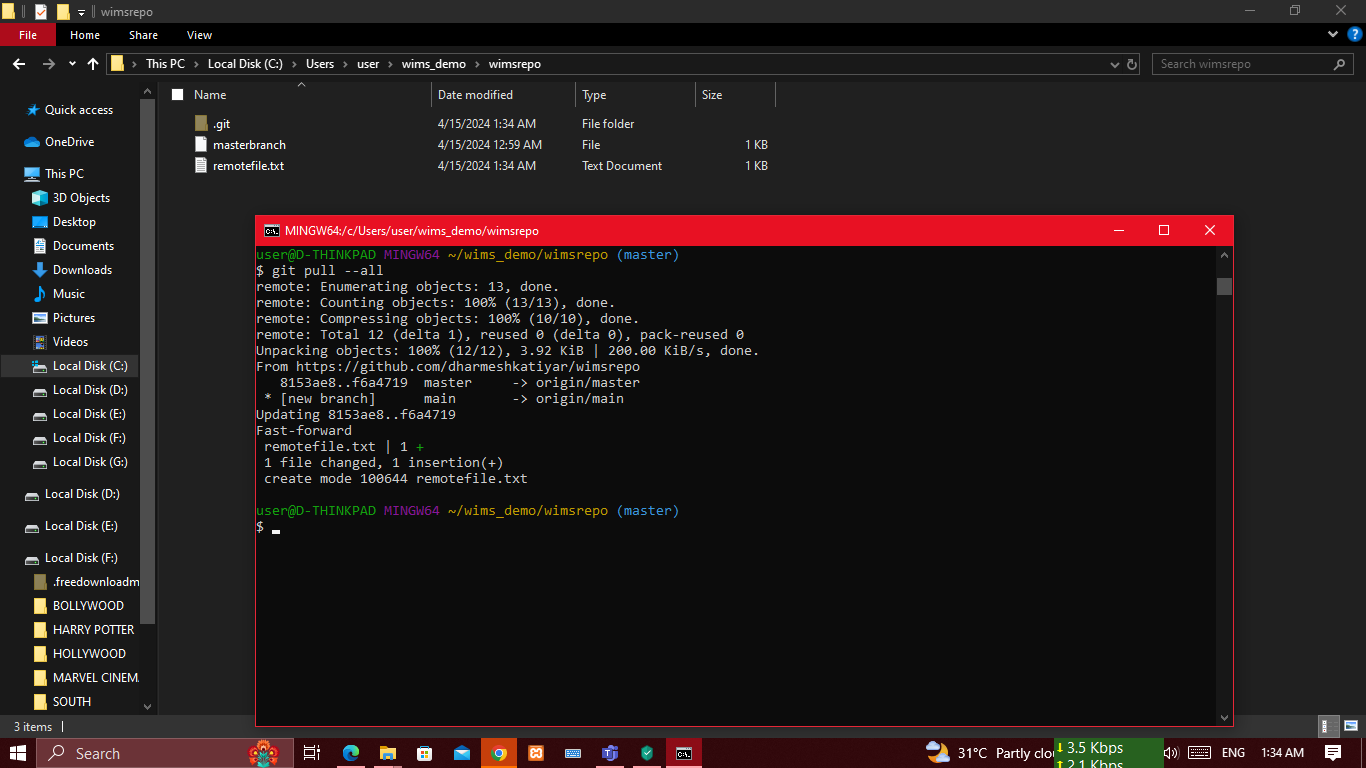
After push command, we can see the response on git hub

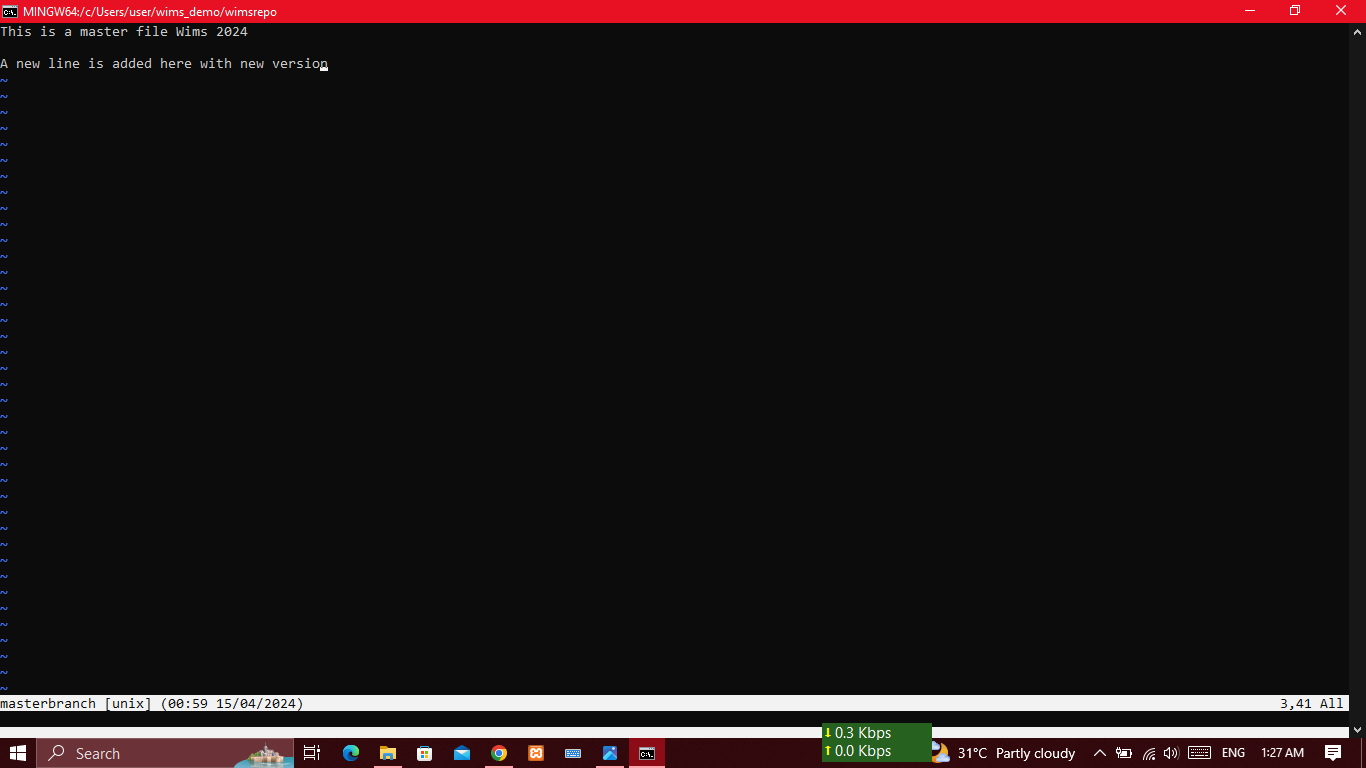
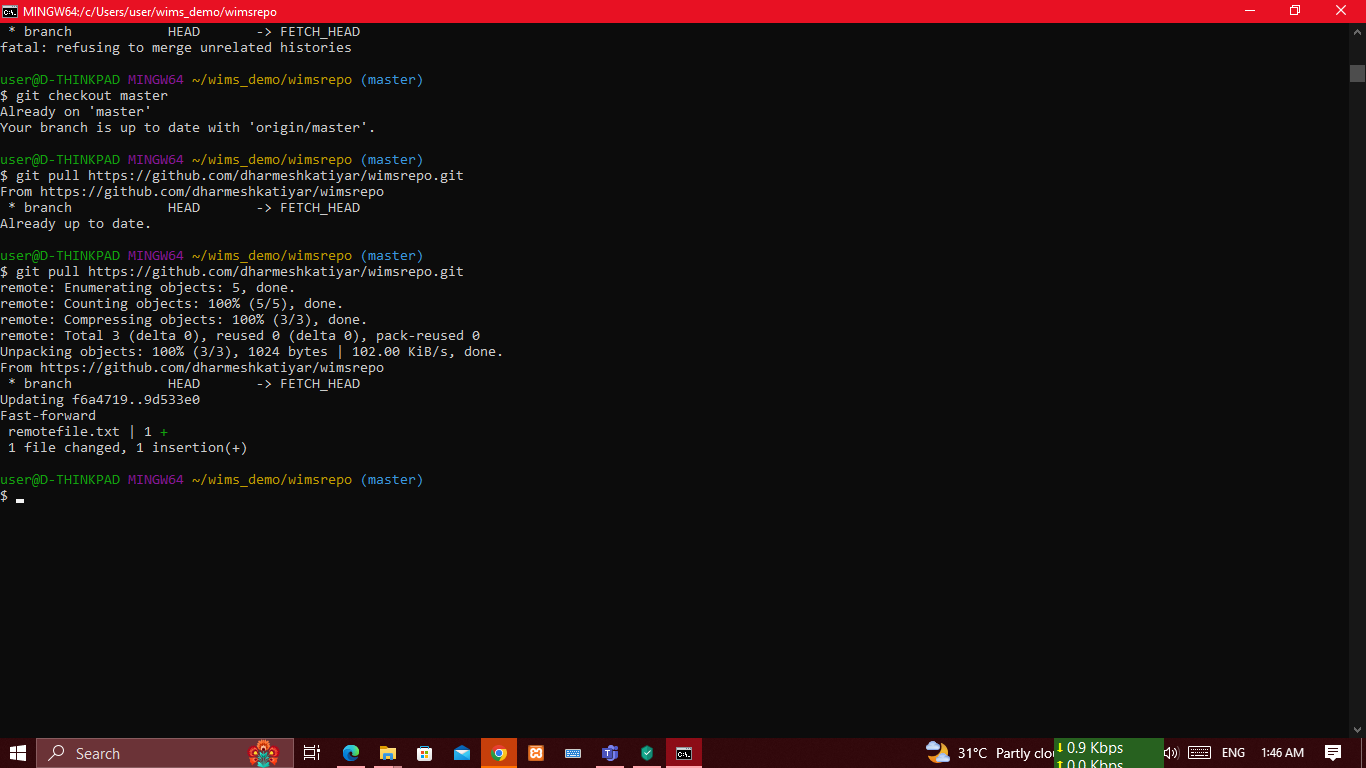


# **Create a branch in local environment and make changes**



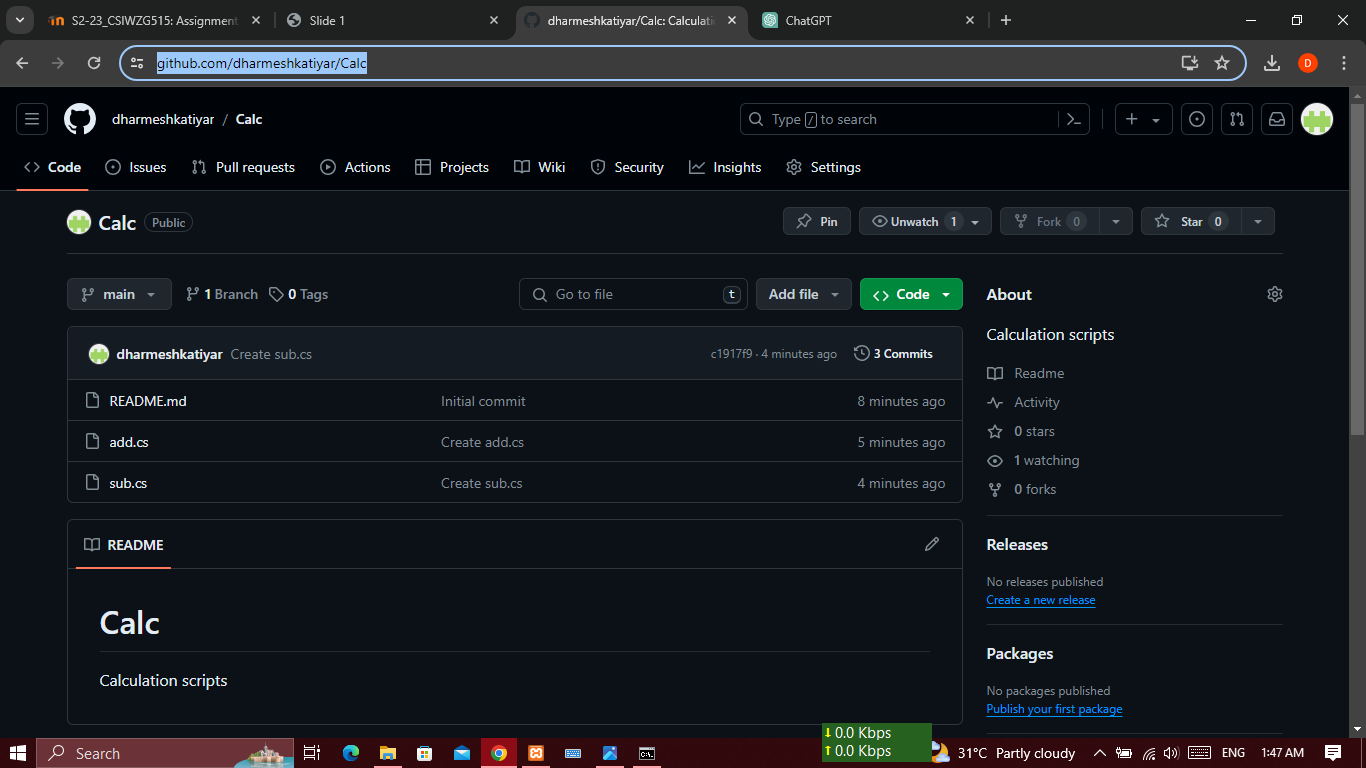
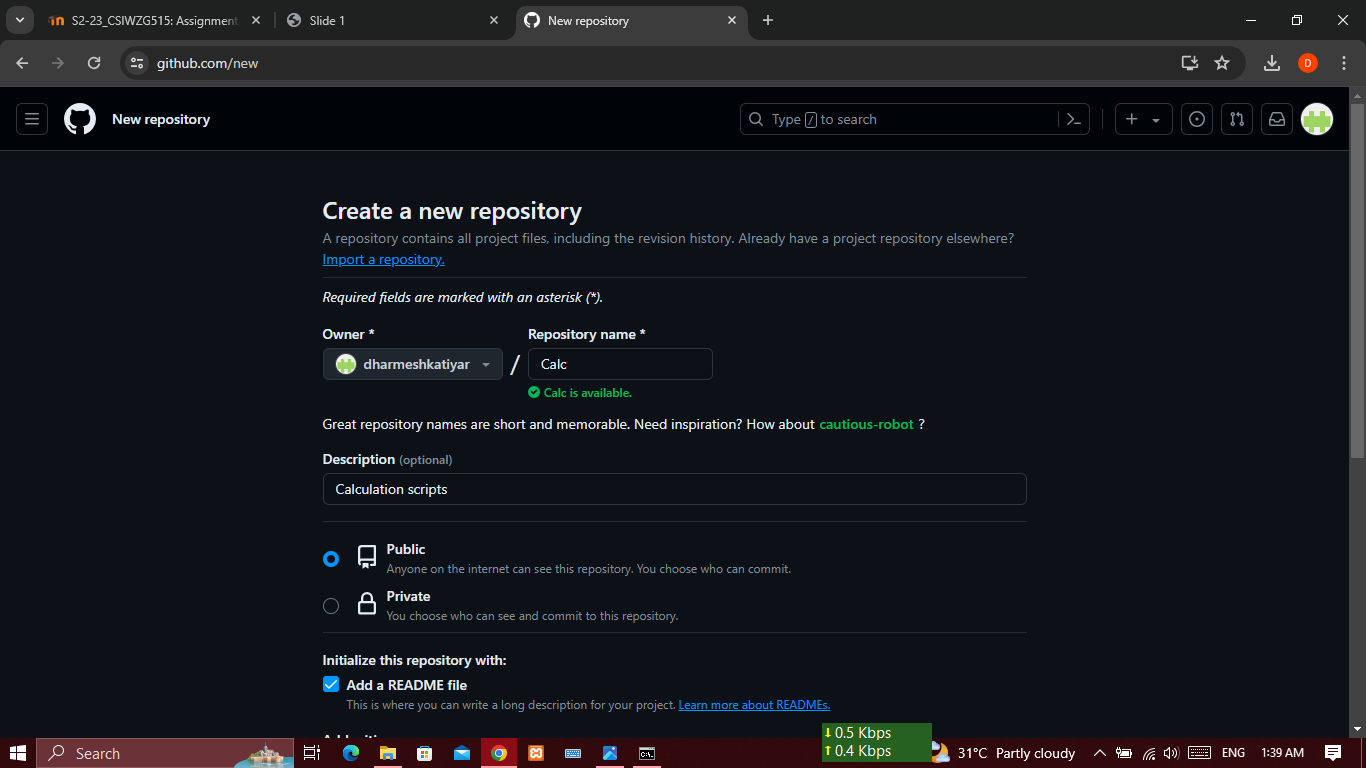
### **Using Pull command download the files from github**



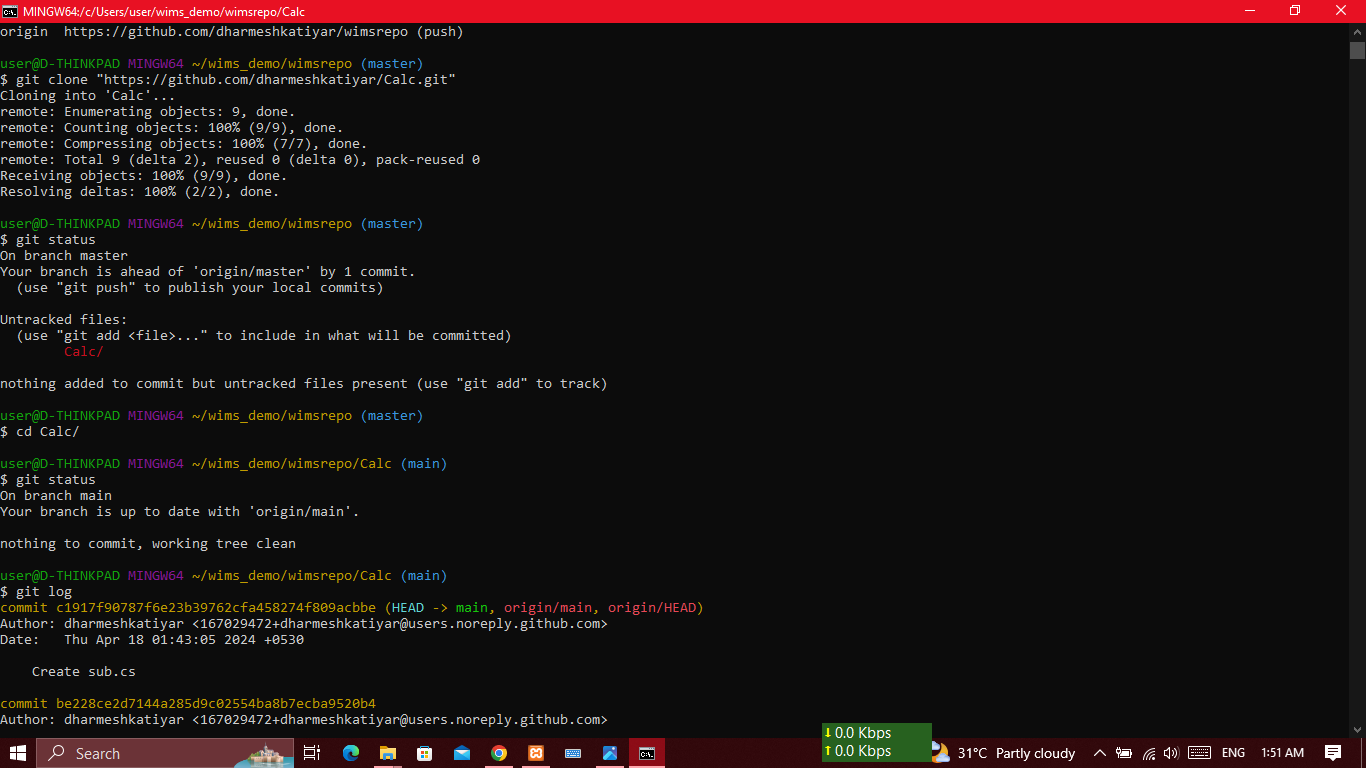


# **Clone the github into local git**

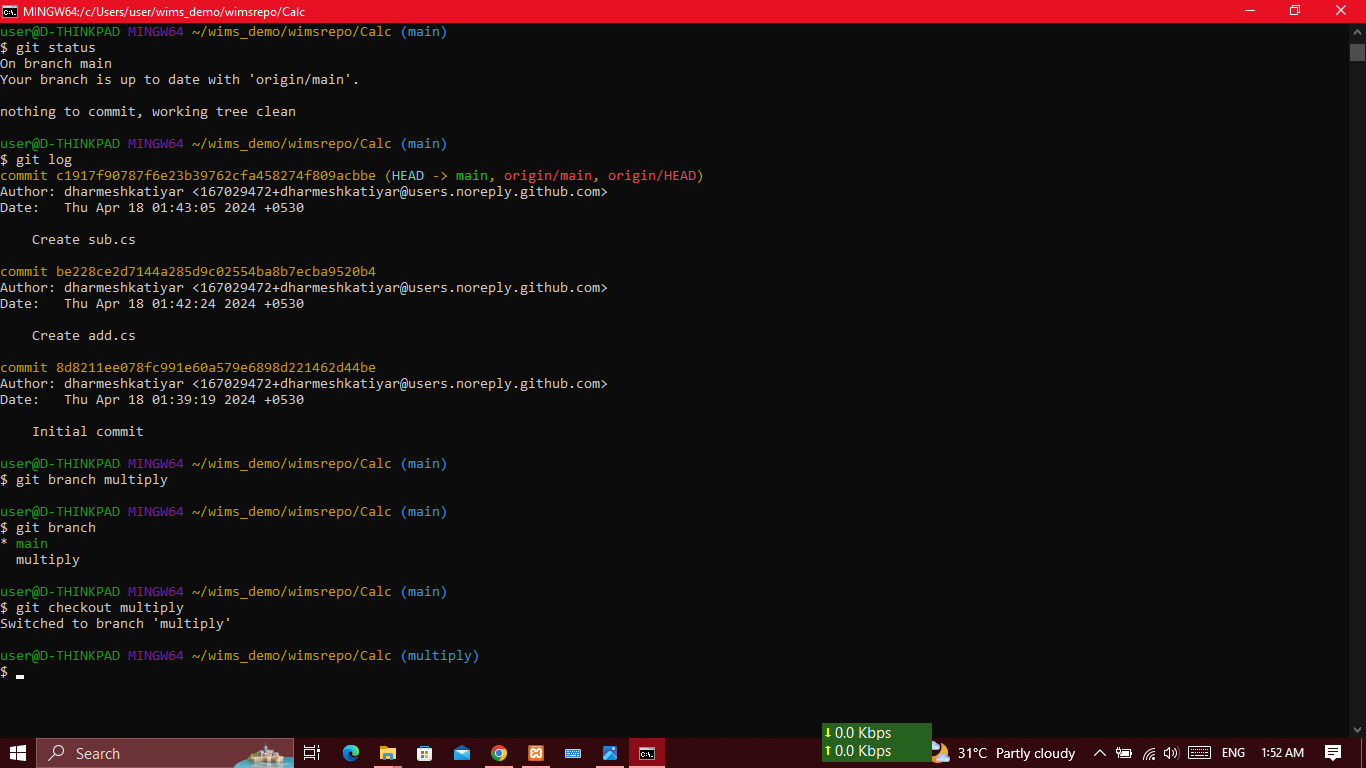
Here we are creating a new repository as Calc and cloning to local git (make sure to select the new repository as default branch.



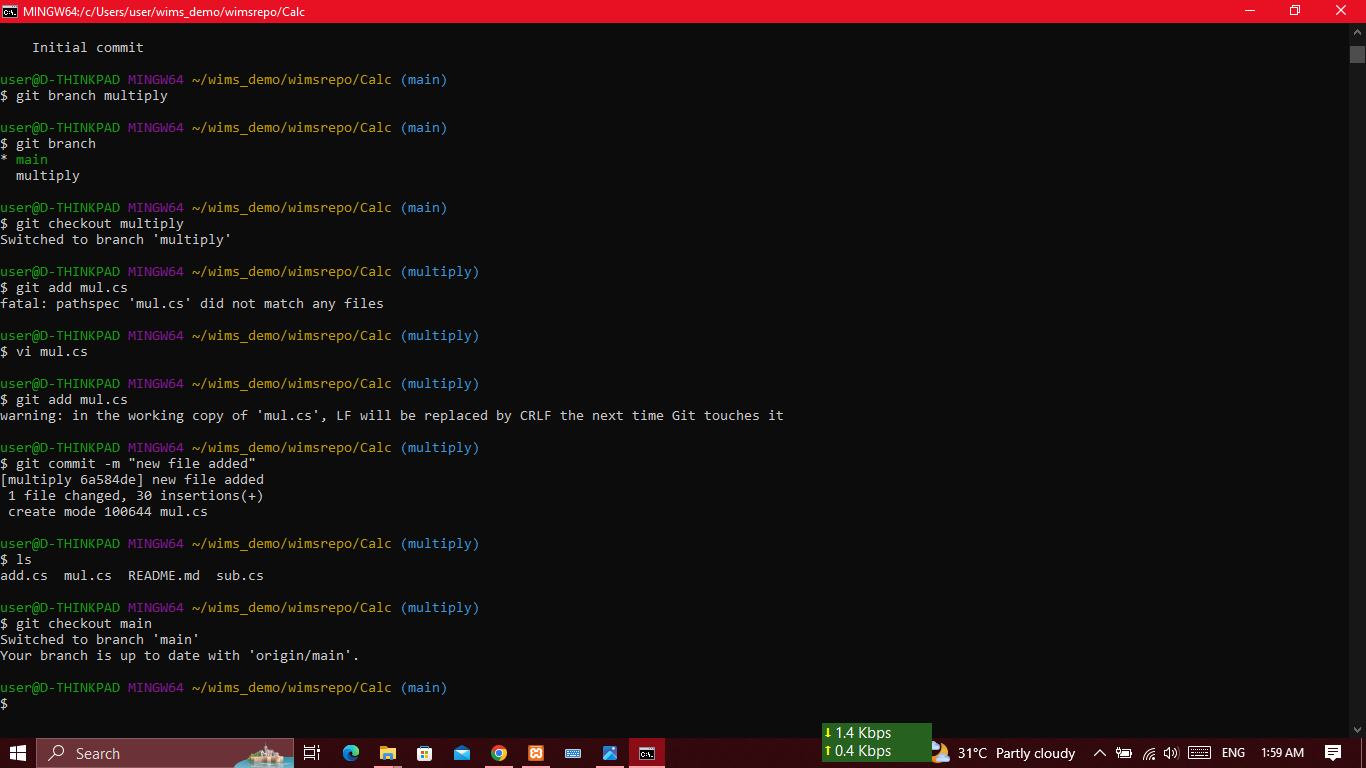
### **Git log**



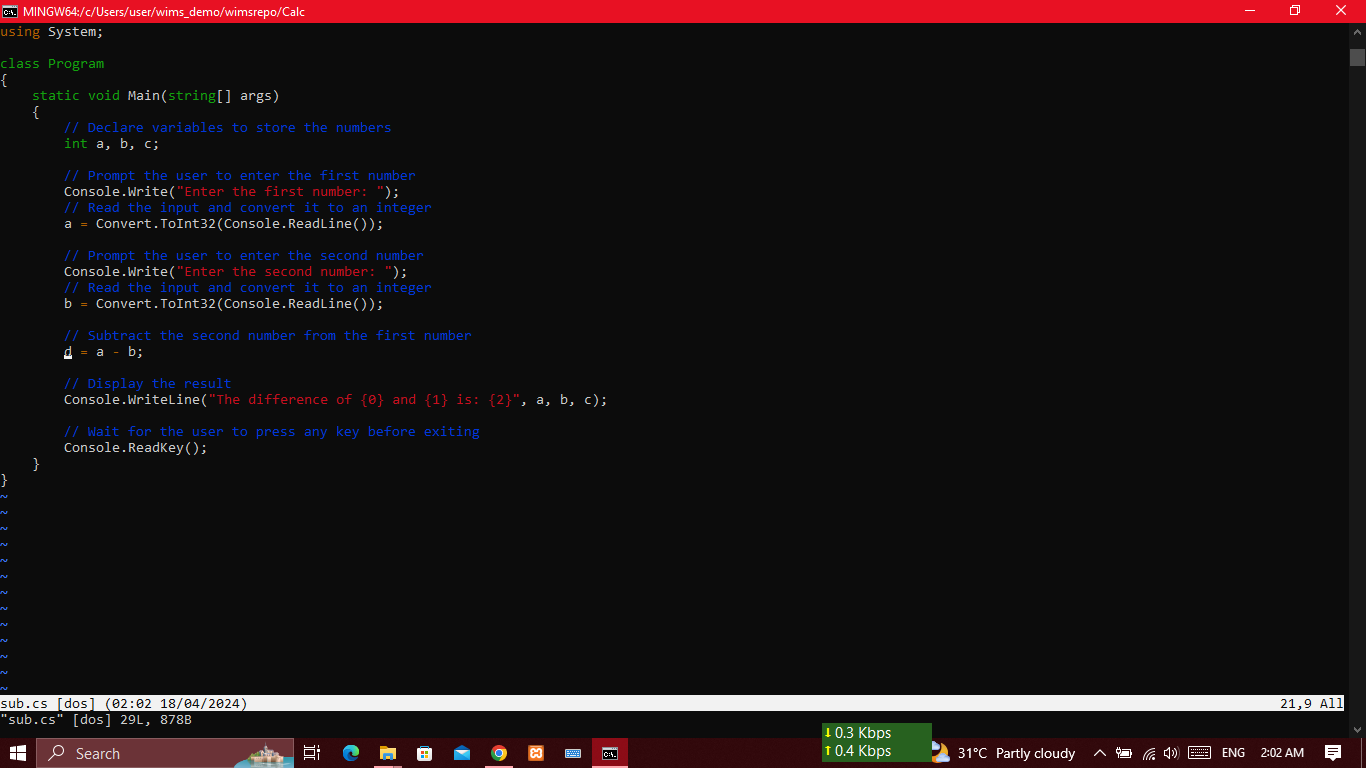
# **Merge changes with main branch and push it to github**



### **Git checkout for moving to another branch**

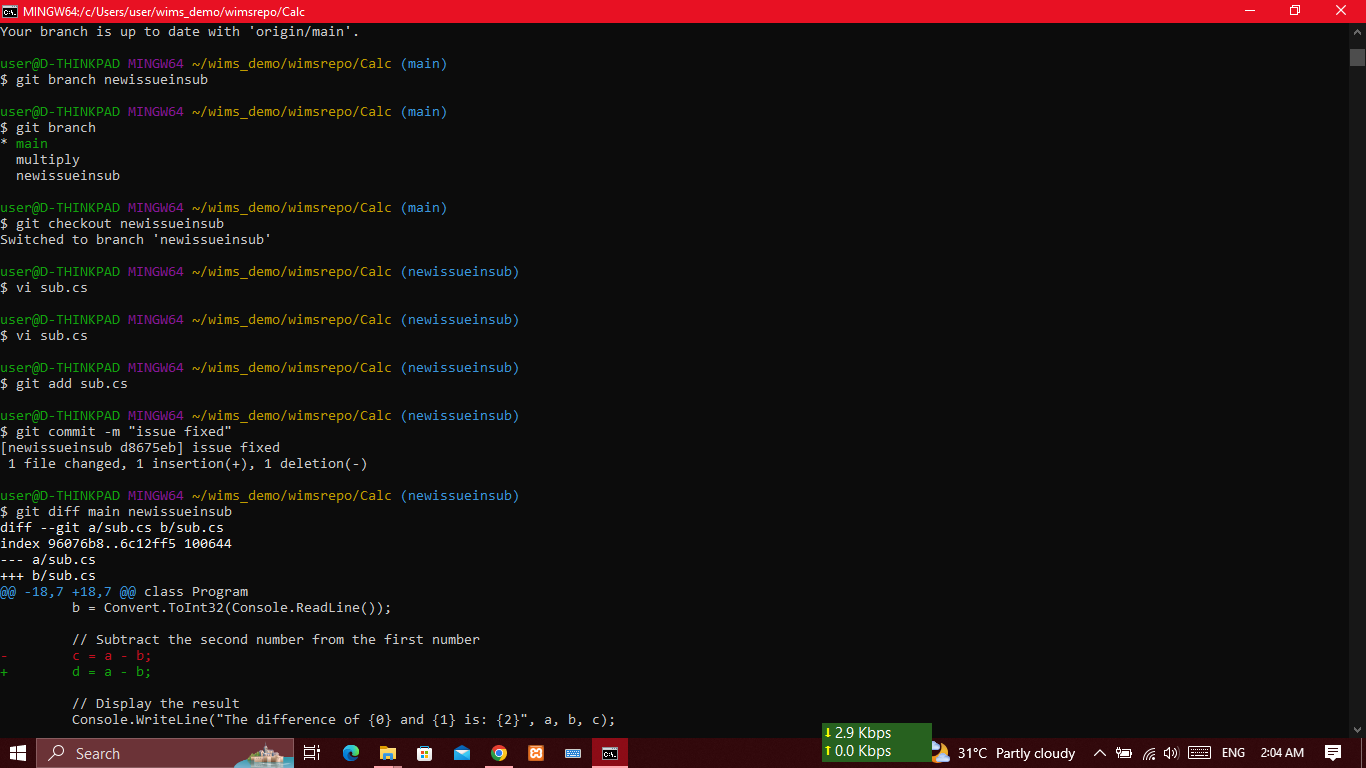


### **Create a new file mul.cs under multiply branch**

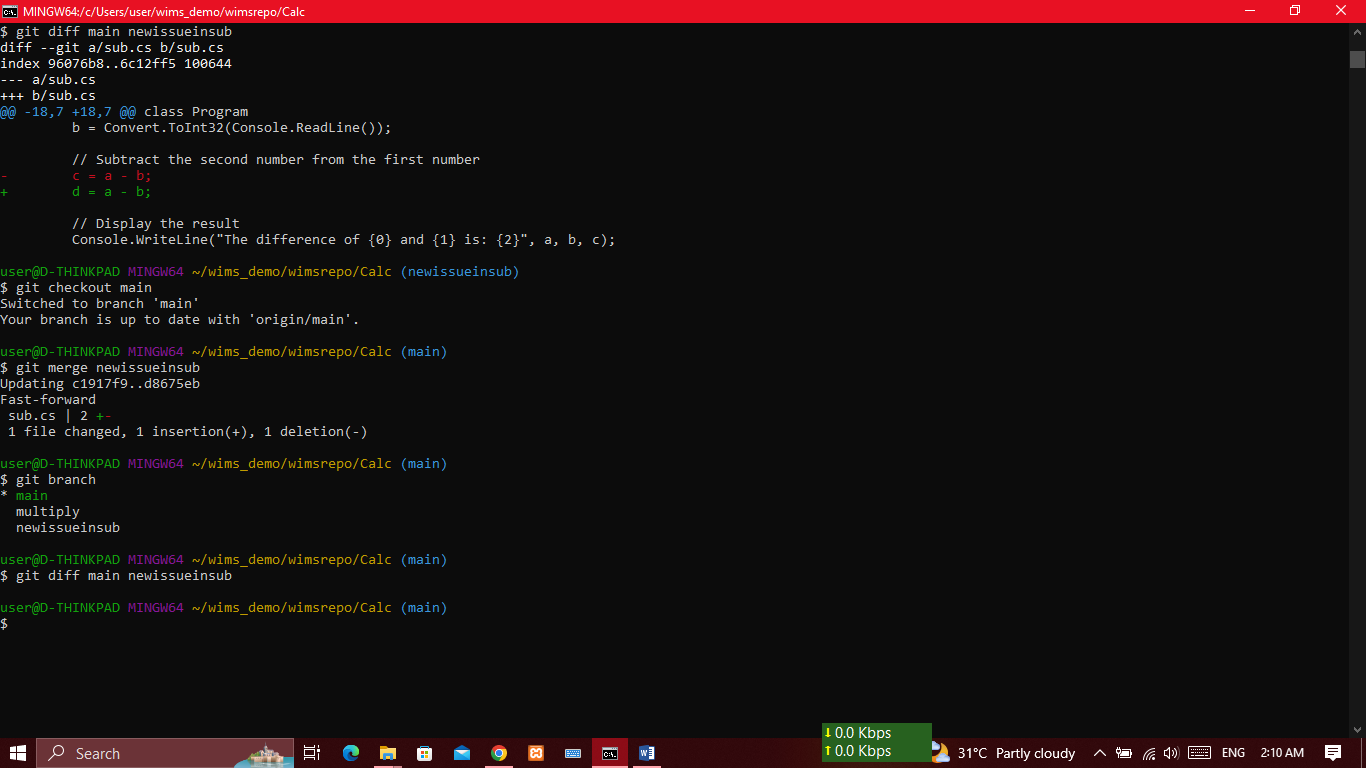


### Assume there is issue in sub.cs

Assume that there is an mistake in sub.cs Now we change the variable c to d in sub.cs and save it.



### **Find the difference between two branches “main” and “newissueinsub” after merging**



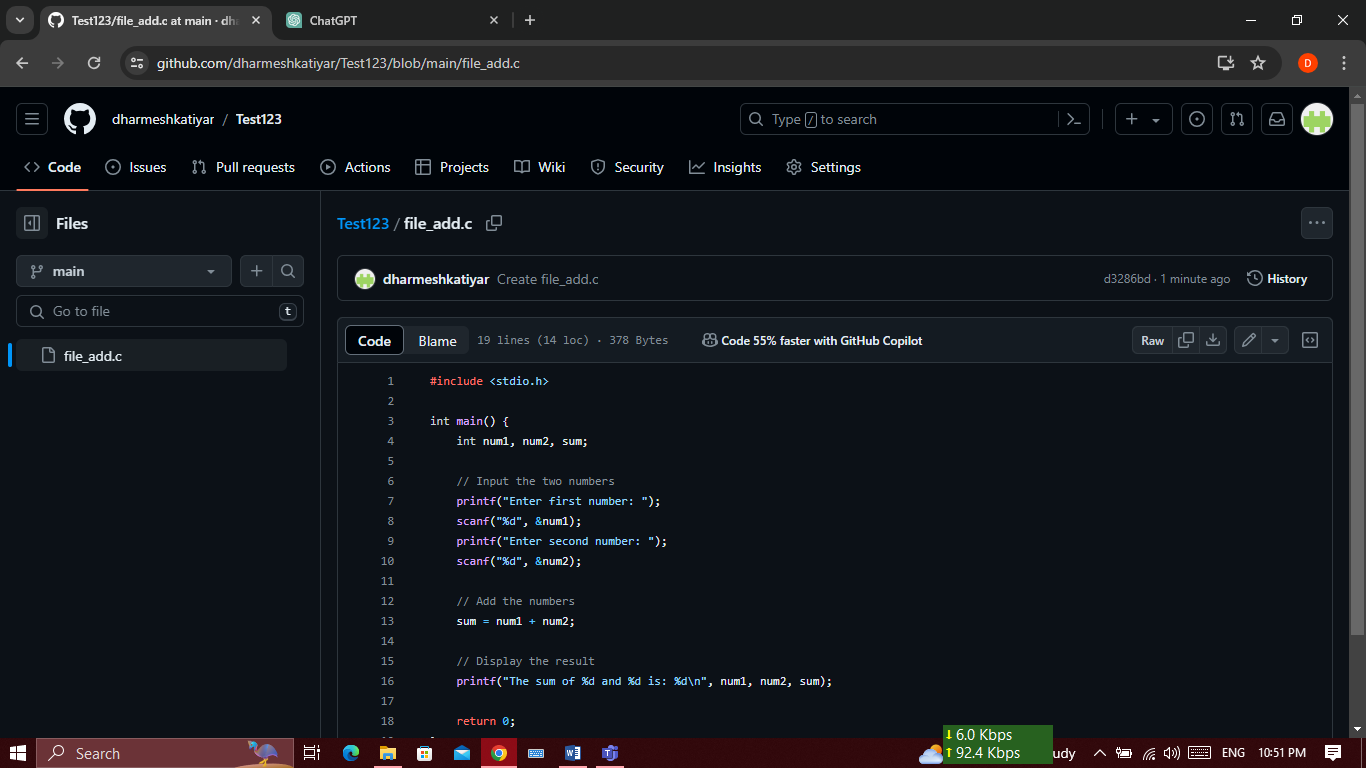
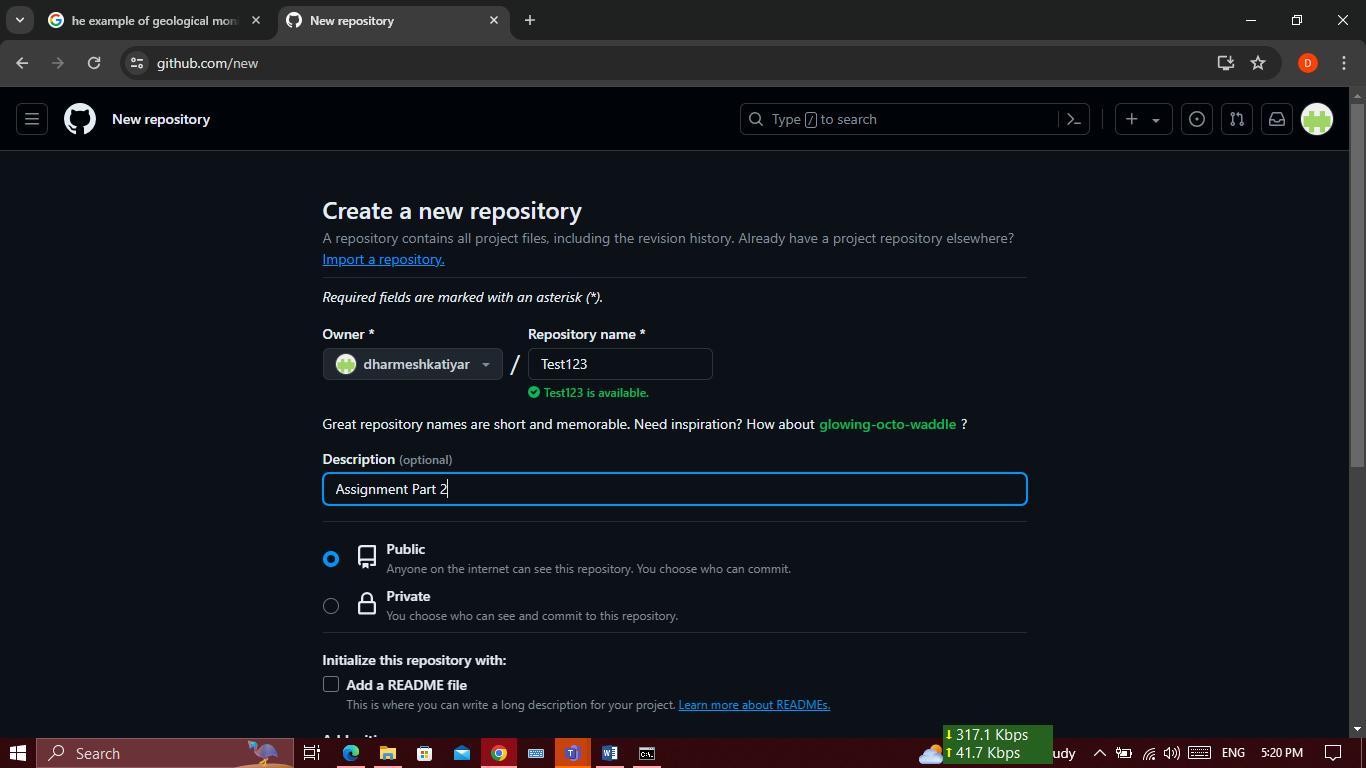
# **Github Part2**

## Introduction

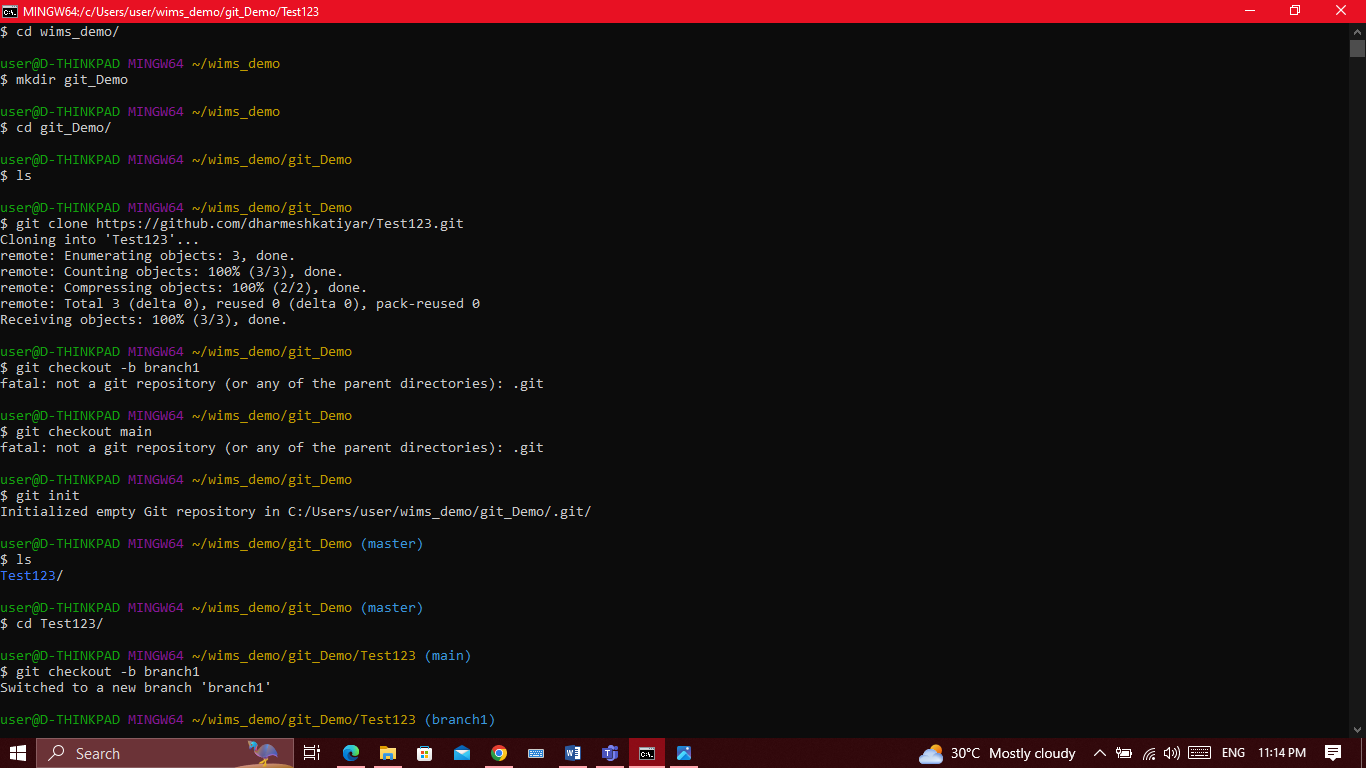
In part1 of github assignment we have understood how to create a github repo and branches and merge. In this part we will take it further to just one more level. I.e., how to resolve merge conflict while merging.

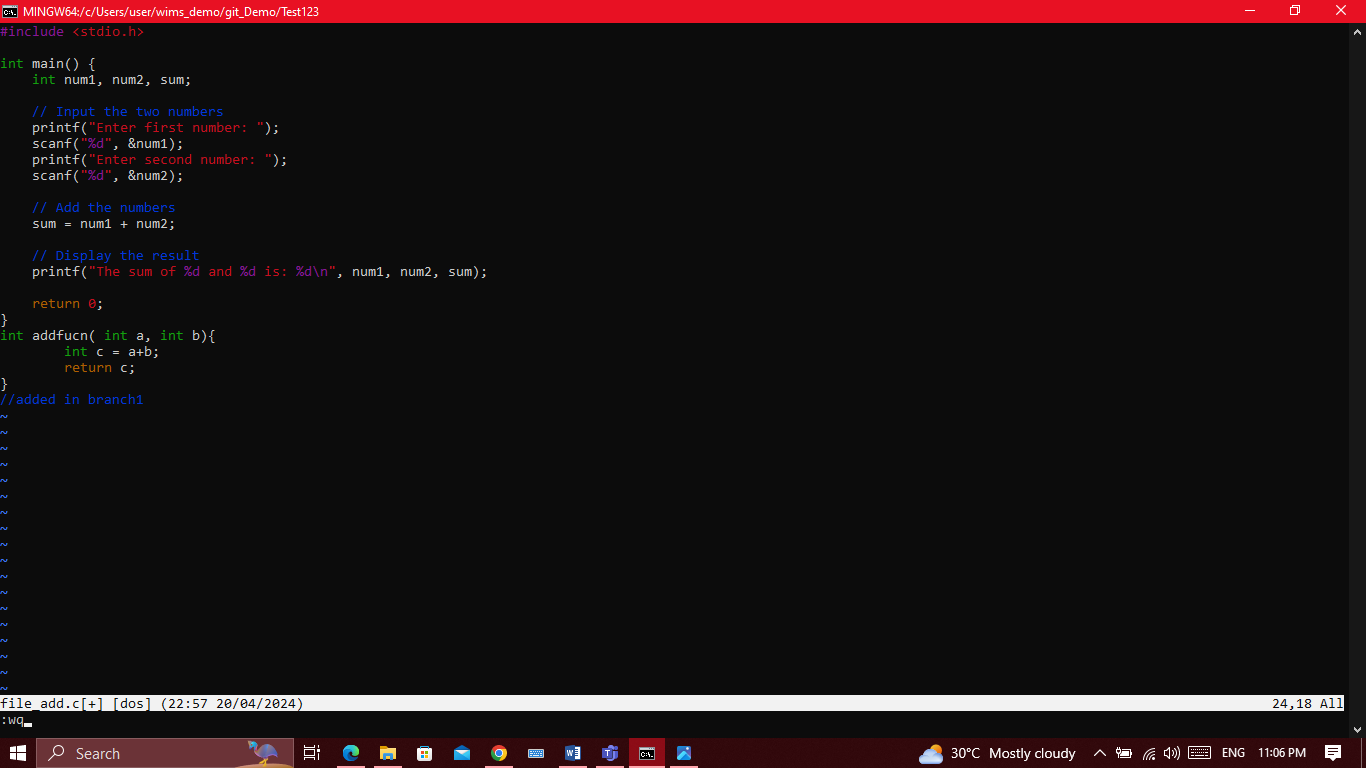
## Clone Repo

Let us clone a existing repo using git clone command

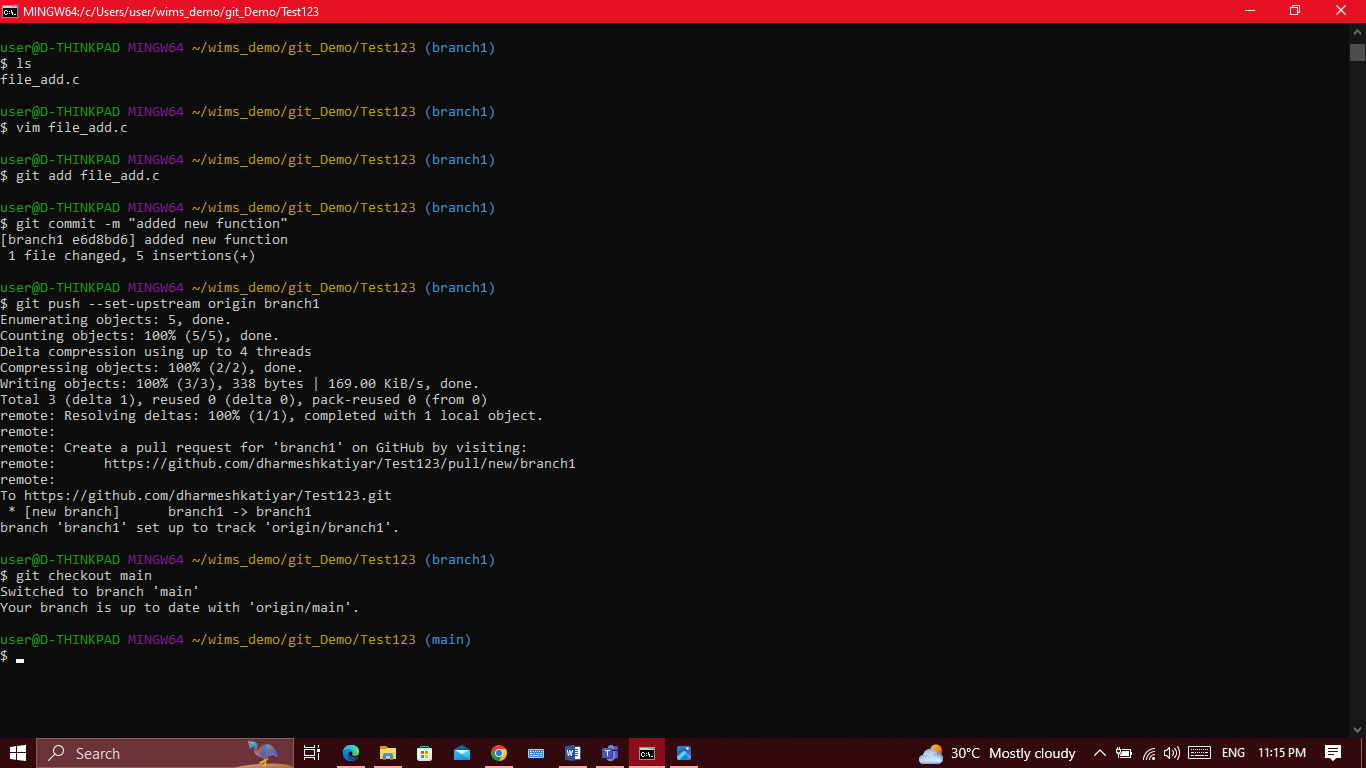


## **Create a branch and Push to remote**





## **Switch to Main branch**



Now we will switch to main branch and make changes to file\_1.c in the same place where you have made changes in the branch also. This is important to trigger merge conflicts

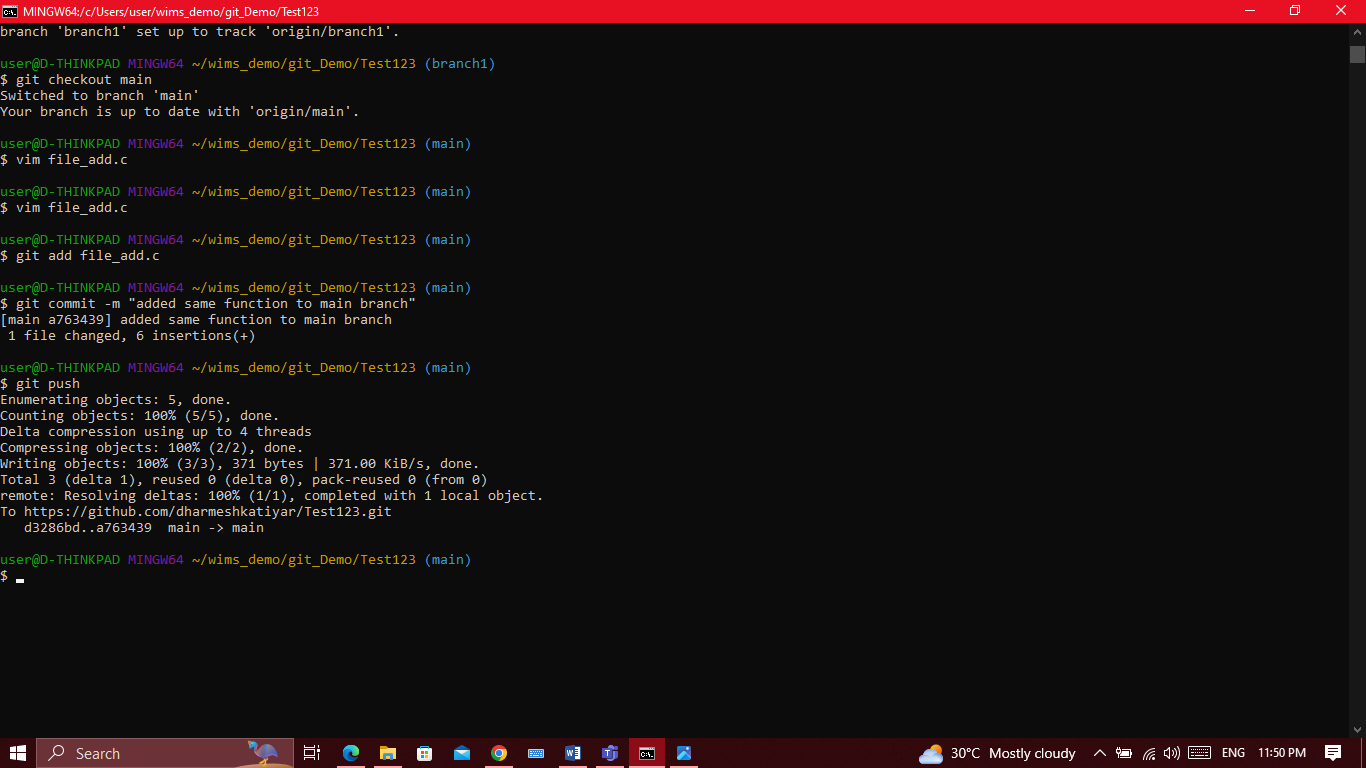
*git checkout main*

*vim file\_1.c*

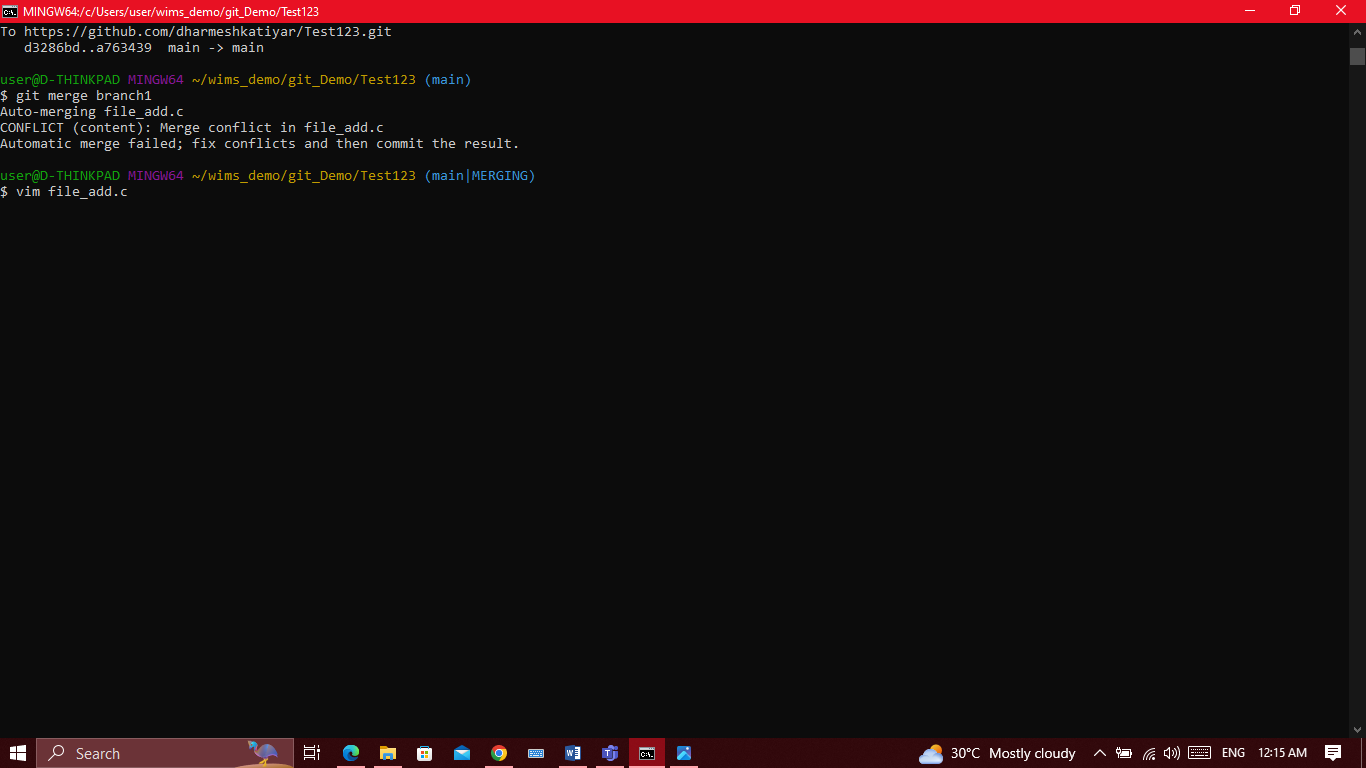
*git add file\_1.c*

*git commit -m "adding same function from main branch"*

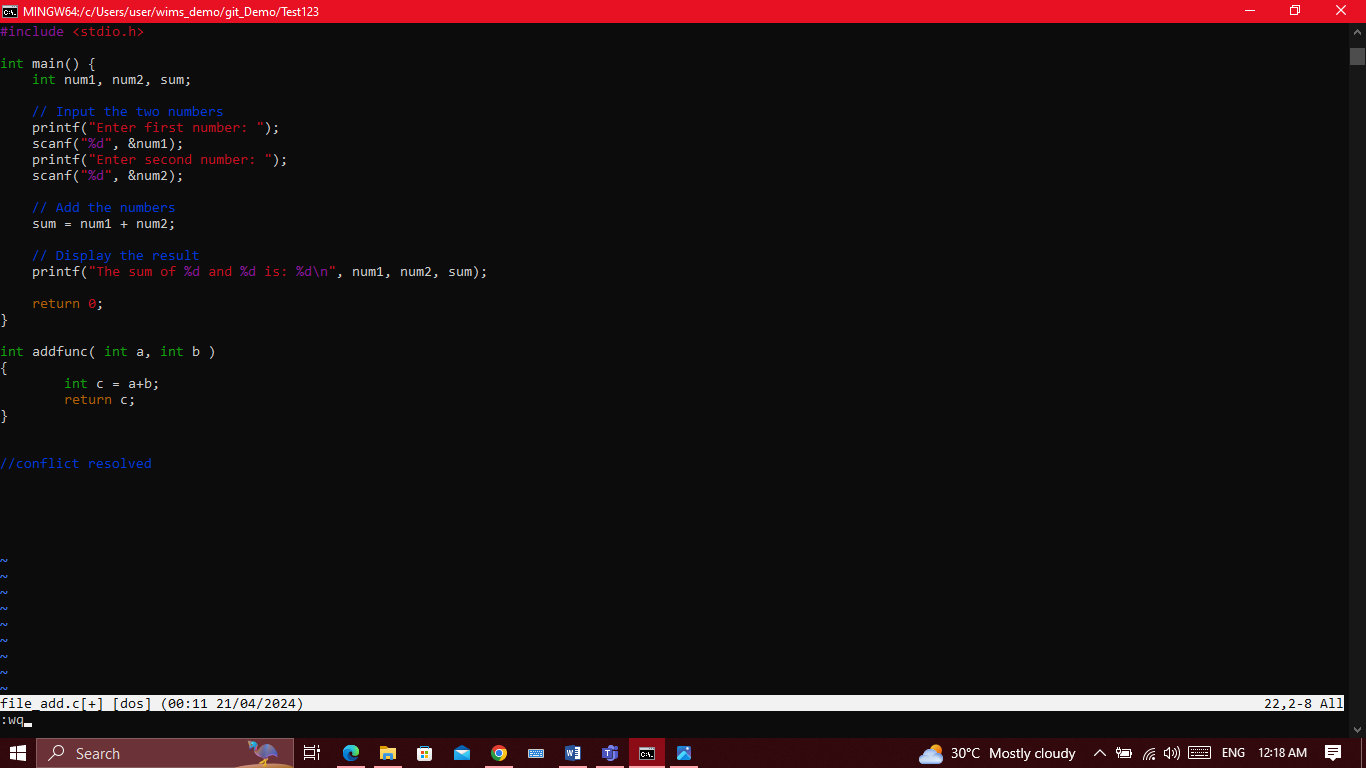
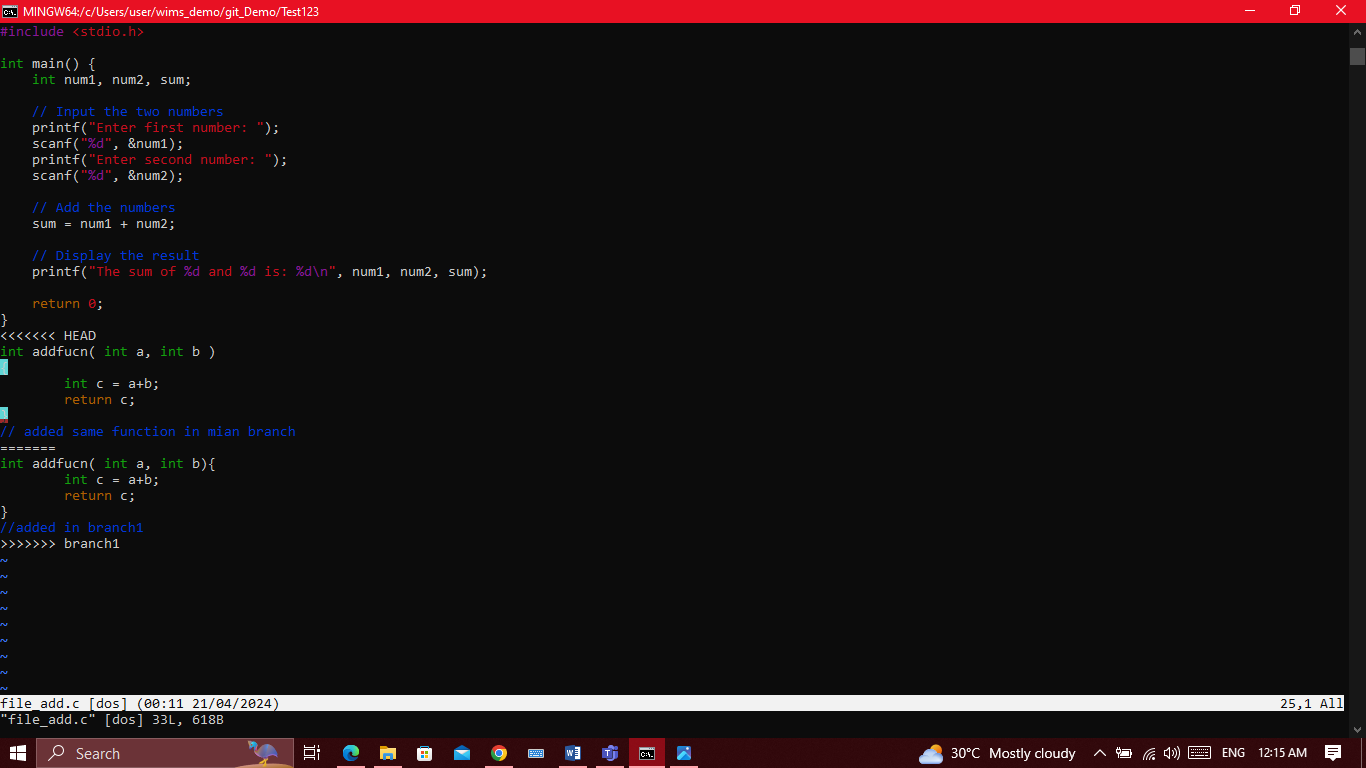
*git push*



## **Merge Branch1**



Below we can observe merge conflict on file\_add.c and we will fix it manually on main branch (by removing the duplicate function on vim editor)



# **Resolve conflict**

To resolve conflict we have to open the conflicting file (file\_1.c) and edit it in such a way that we remove the unwanted part of the code and retain only required code.

Here we can see the code from both branch, we have to decide which one to keep and which one to remove.

Above we can see changes from both the branch main(Head) and branch1. We resolve the conflict manually and save the file.

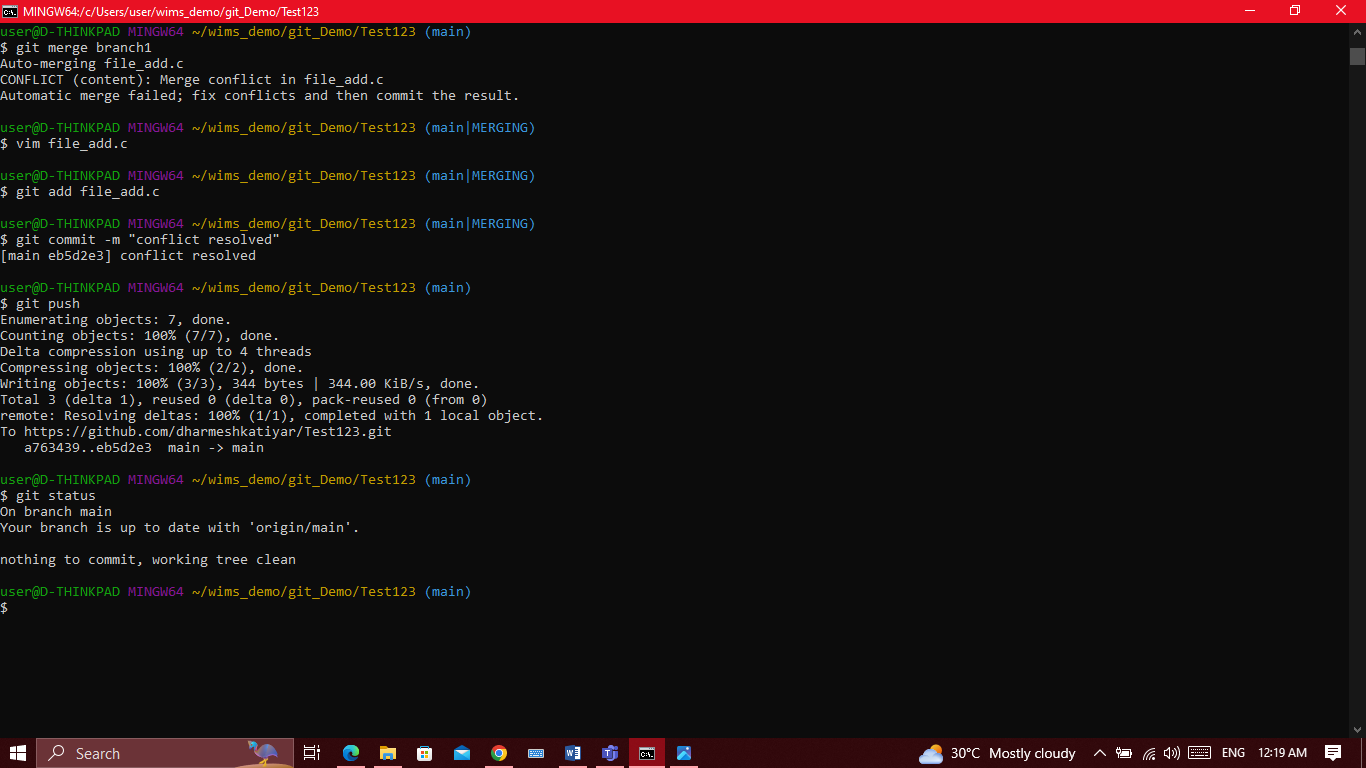
# **Git push : Final conclusion**

Now we should push our changes to main branch like the same way we push other changes

*git add file\_1.c*

*git commit -m "conflict resolved"*

*git push*



**Thank You !!**