Assignment No. 1

Email: harshalgite888@gmail.com

Tasks:

1. Based on what you have learnt in the class, do the following steps:

    a. Create a new folder

    b. Put the following files in the folder:

        i) Code.txt

        ii) Log.txt

        iii) Output.txt

    c. Stage the Code.txt and Output.txt files

    d. Commit them

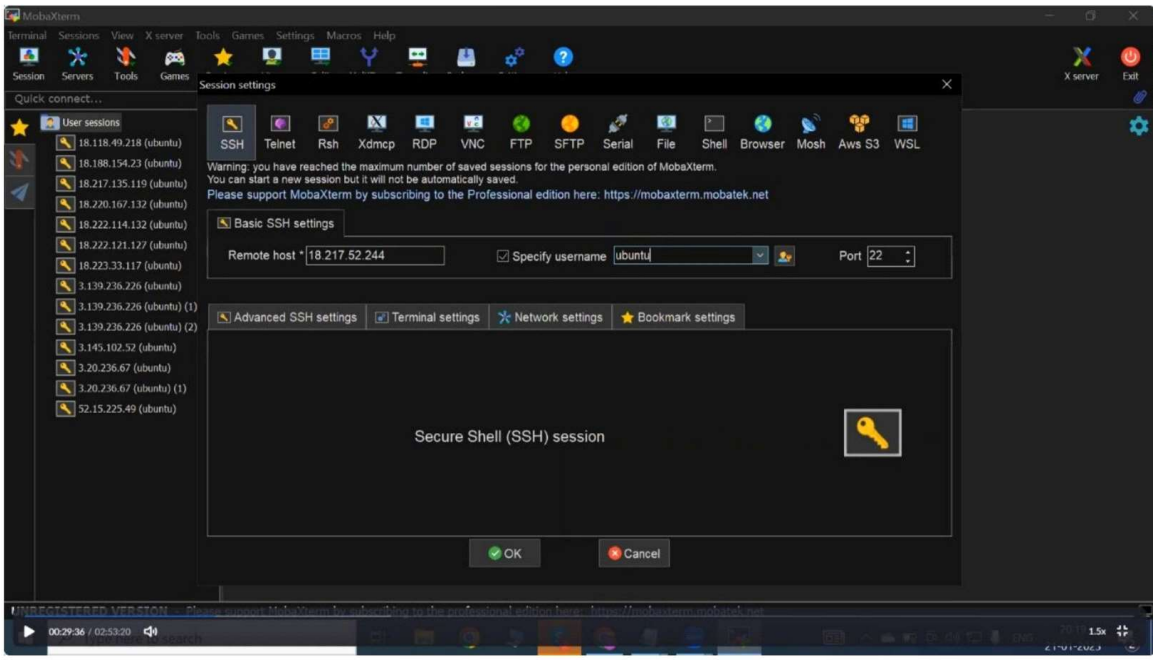
    e. And finally push them to GitHub

Solution:

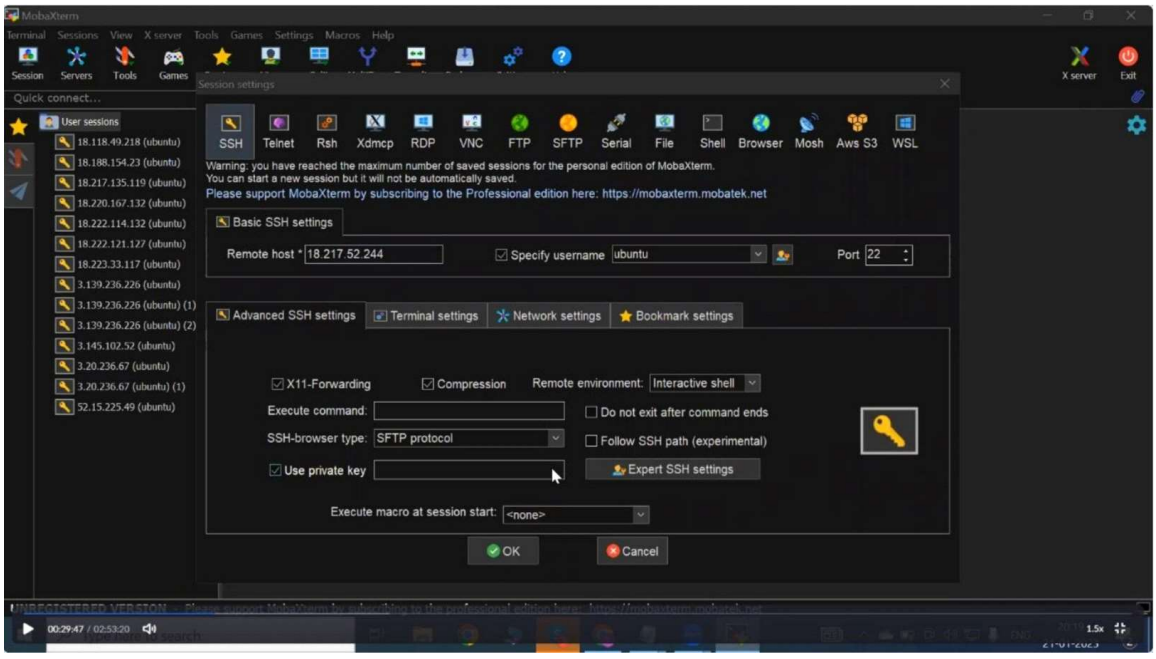
Steps:

1. Create an ec2 instance using Ubuntu AMI . Once the instance is running copy thepublic IP address.

Open putty or mobaxterm app and paste the IP address in remote host . Type inubuntu as specify username . Then click on advanced ssh settings.



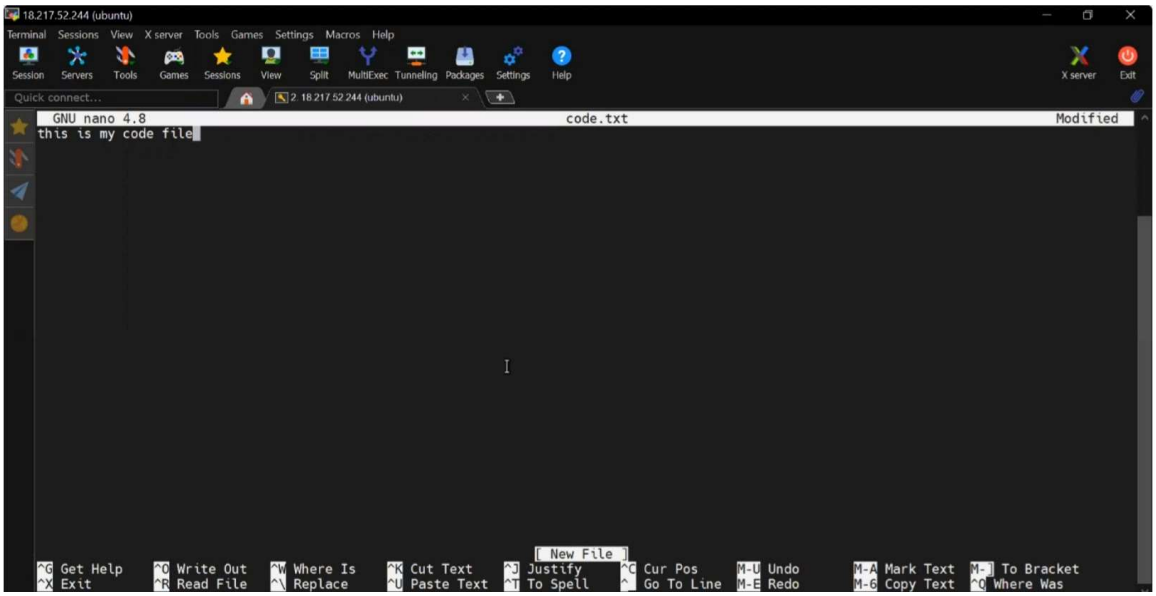
2. In the private key option , select the private key that you have used in your EC2 instance and then click on OK.



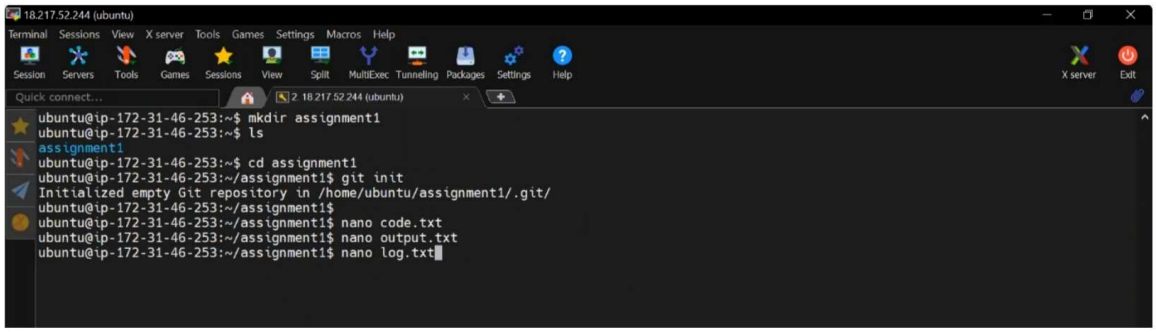
3. Once the terminal is running , create a folder by typing “mkdir assignment1”. Then type “cd assignment1” to go inside the assignment1 folder . To make your folder accept all the git commands type “git init” and click enter .

4. Now type “nano code.txt” to create files inside your assignment1 folder and click enter .

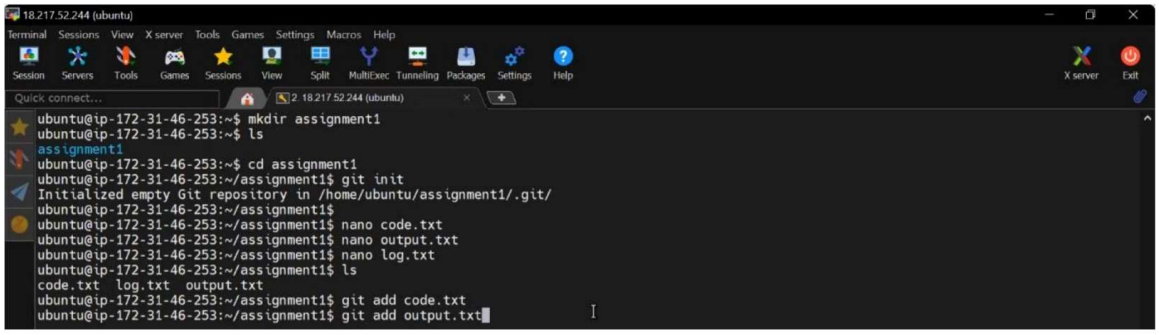
5. Enter any text inside your file . Once done click ctrl+s to save and ctlr+x to exit .



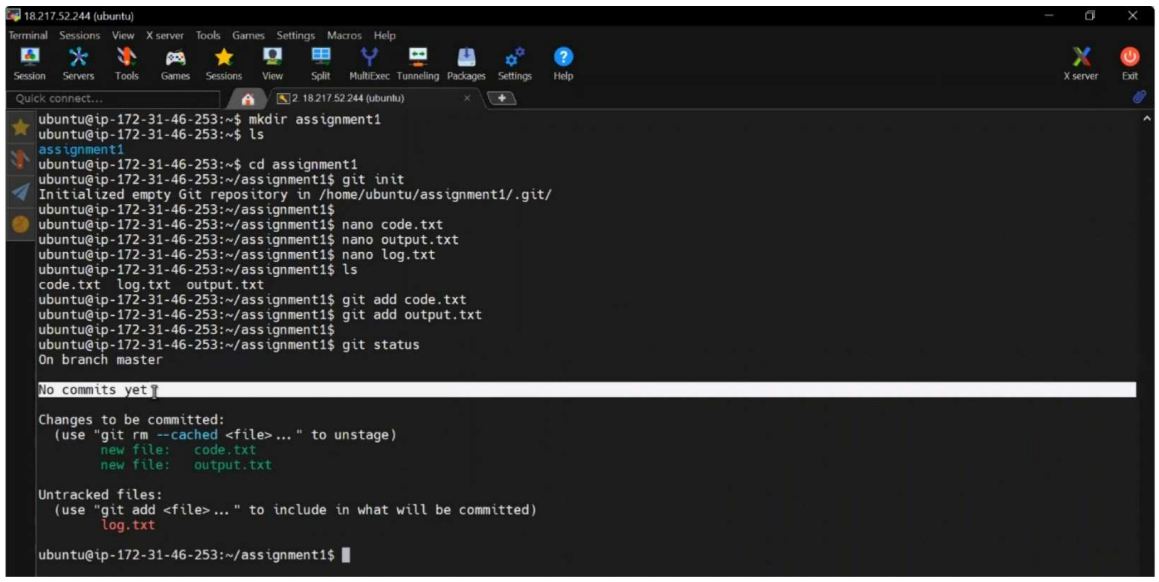
6. Similarly create remaining two folders output.txt and log .txt.



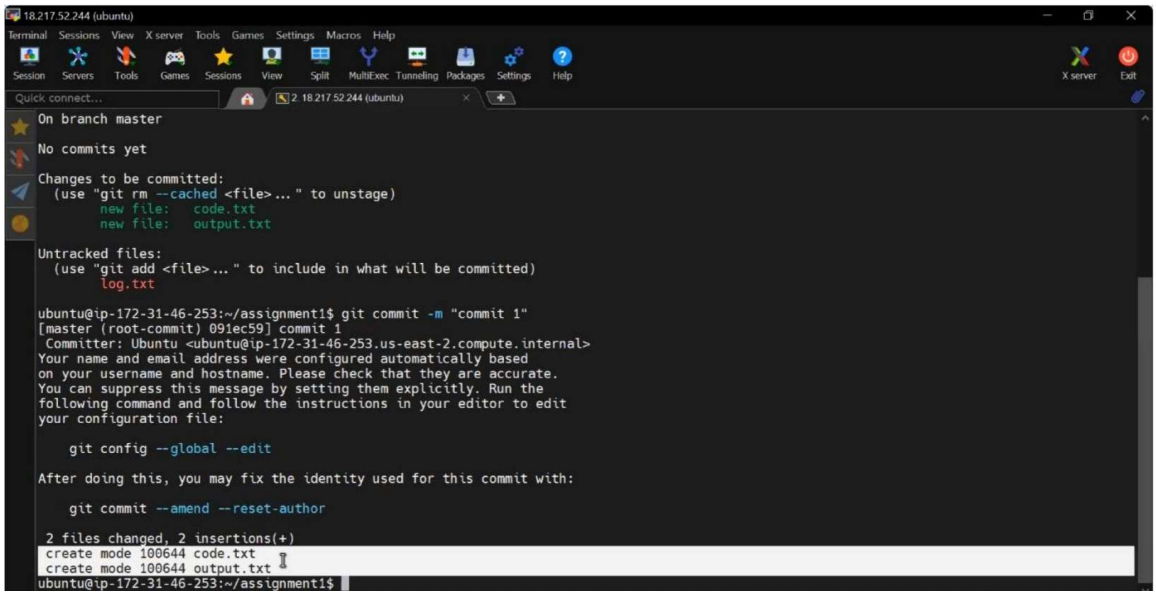
7. Now to add the two folder type in “git add code.txt” and “git add output.txt” . Hence the two files are staged.



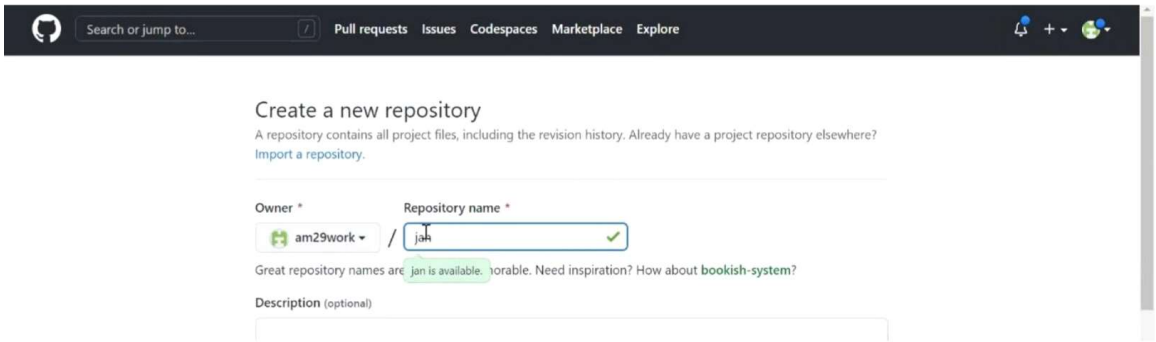
8. You will see the output as seen below once the files are staged successfully.



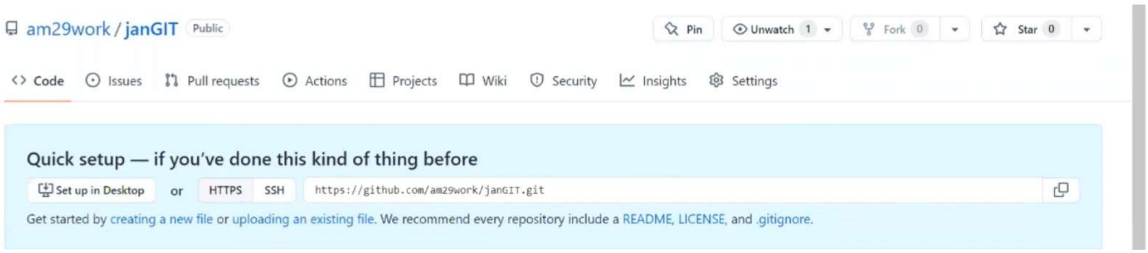
9. Now to commit the files type in “git commit -m “commit 1” and hit enter . Hence the files will get commited.



10. Now go to your GitHub account . Click on New to create a new repository .
    - Give you repository a name and scroll down and click on create .

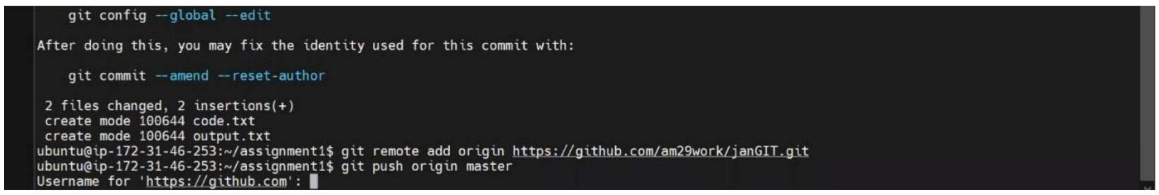


11. Once it is created you will see an HTTPS link . Copy it .

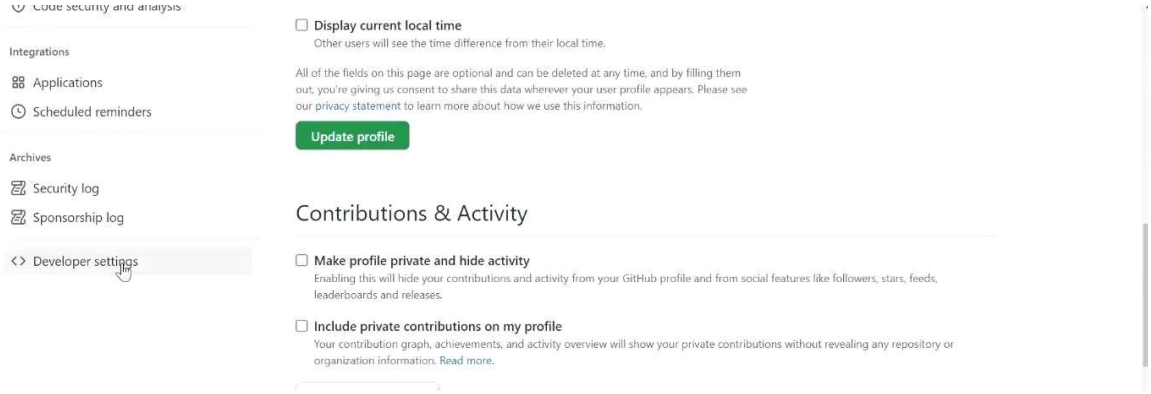


12. Come back to your terminal and type in “git add remote origin ” and paste the copied link and hit enter. Then type in “git push origin master” and hit enter to connect your folder to your GitHub repository.

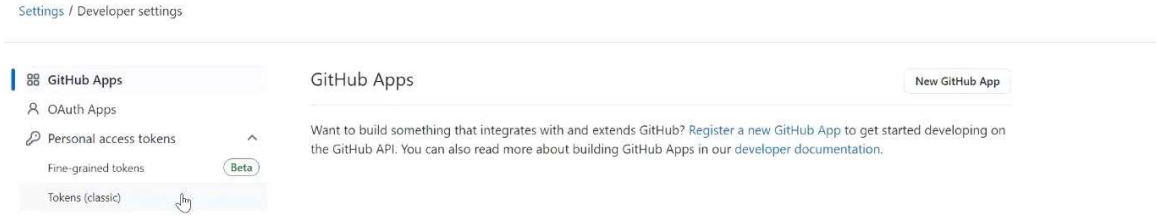
Enter your GitHub username below and hit enter.



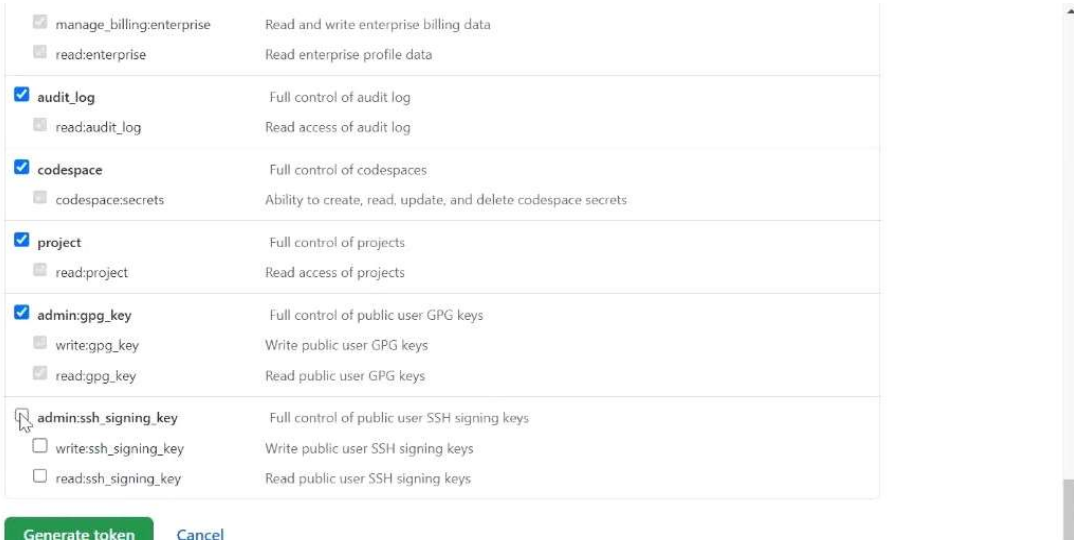
13. Then go to your repository settings > select developer settings .



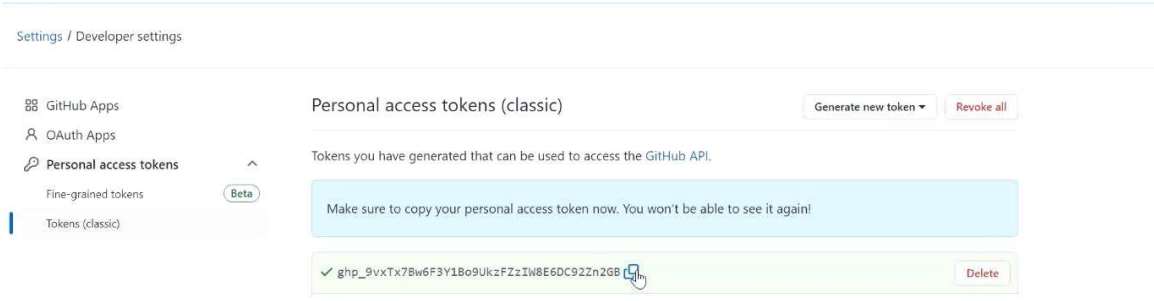
14. Click on Personal access tokens > tokens (classic) > Generate new tokens > Generatenew tokens (classic) and hit enter.



15. Give your token a name and select all the options from the drop down then click ongenerate token.



16. After that you will be able to see the token . Copy it and paste it on your terminal asyour GitHub password.



17. Click enter and your will see the below output on your terminal .

```
        log.txt
ubuntu@ip-172-31-46-253:~/assignment1$ git commit -m "commit 1"
[master (root-commit) 091ec59] commit 1
 Committer: Ubuntu <ubuntu@ip-172-31-46-253.us-east-2.compute.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 2 files changed, 2 insertions(+)
 create mode 100644 code.txt
 create mode 100644 output.txt
ubuntu@ip-172-31-46-253:~/assignment1$ git remote add origin https://github.com/am29work/janGIT.git
ubuntu@ip-172-31-46-253:~/assignment1$ git push origin master
Username for 'https://github.com': am29work
Password for 'https://am29work@github.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (4/4), 307 bytes | 307.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0)
To https://github.com/am29work/janGIT.git
 * [new branch]      master -> master
ubuntu@ip-172-31-46-253:~/assignment1$
```

18. Go back to your GitHub repository and there you will be able to see the staged filescode.txt and output.txt . Hence the files are commited successfully.