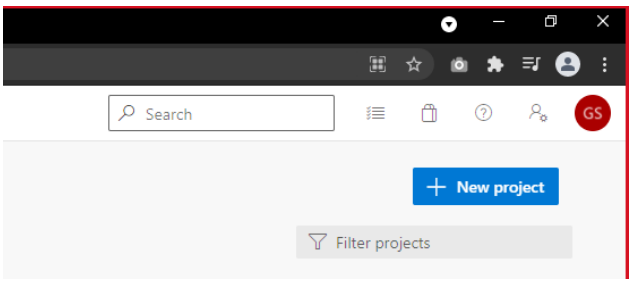
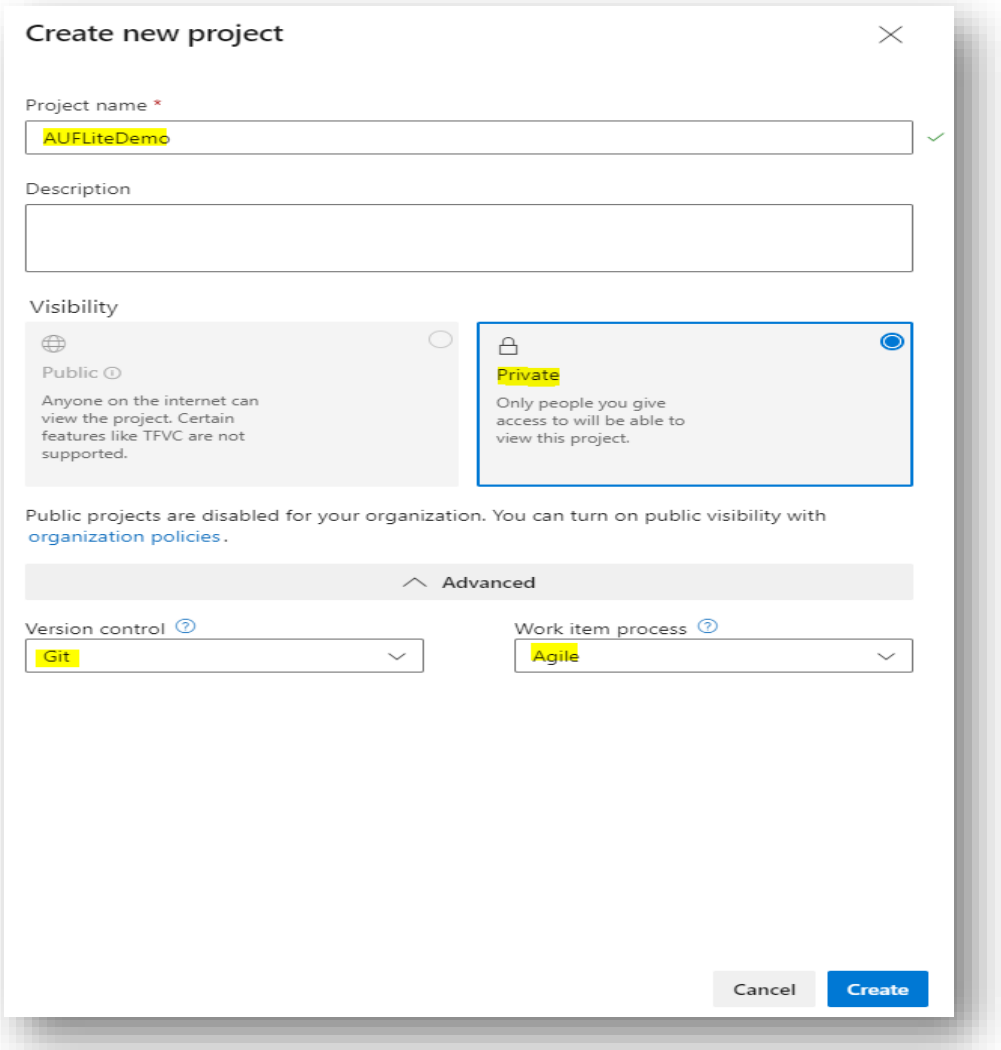
Create Azure Project

Go to the organization page: For example → https://dev.azure.com/ApteanSandbox/

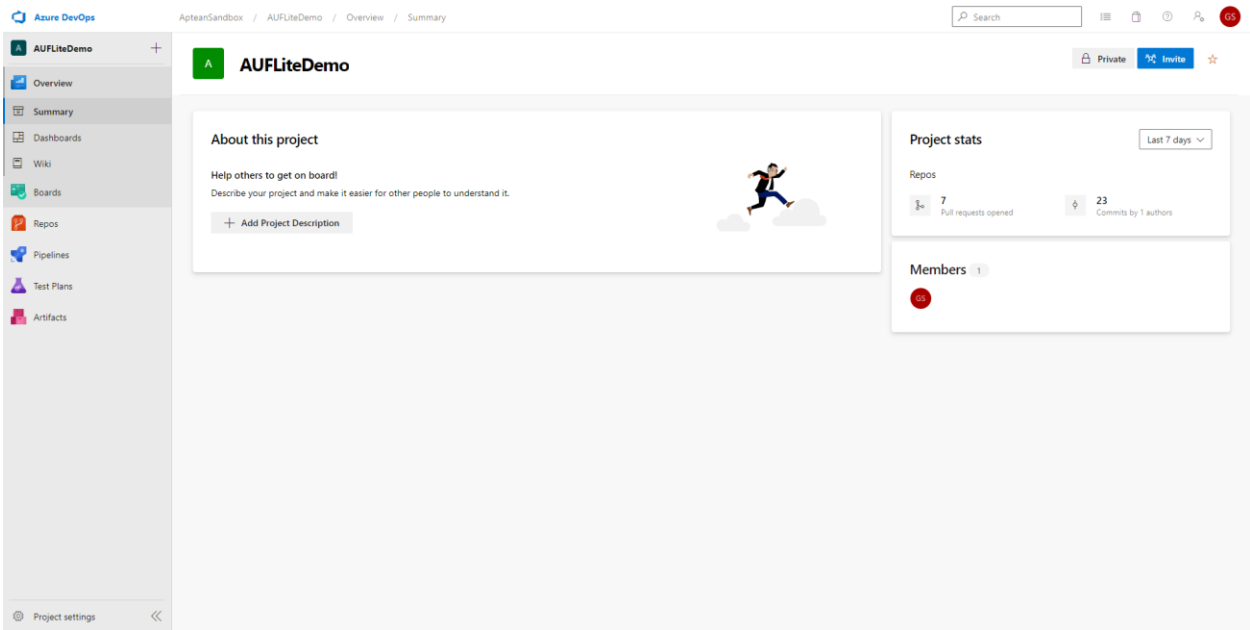
Click on New project



Enter values and select options as shown in the image below:



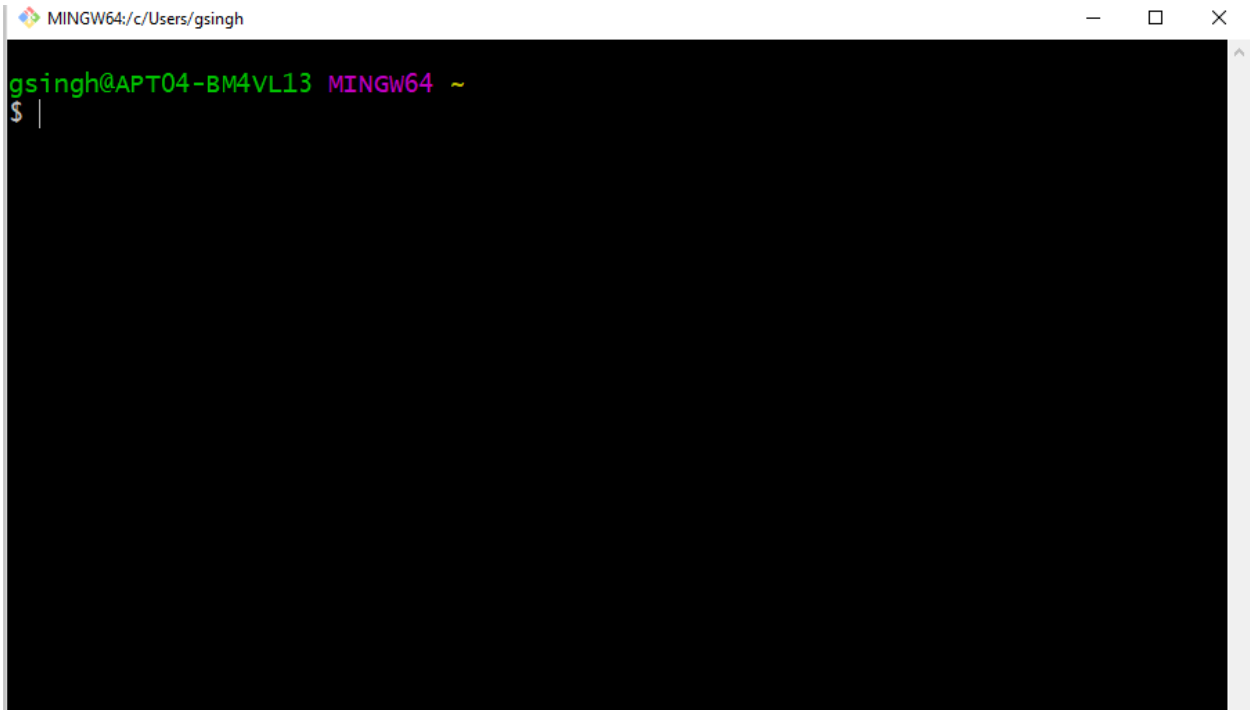
After clicking on Create button, new project will be created successfully.



Create Git repo and initial commit:

Prerequisite: Git should be installed on your local machine

1.) Open git bash in local machine



2.) Execute these two commands to add your username and email to git global configuration

Git config –global user.name <<Username>>

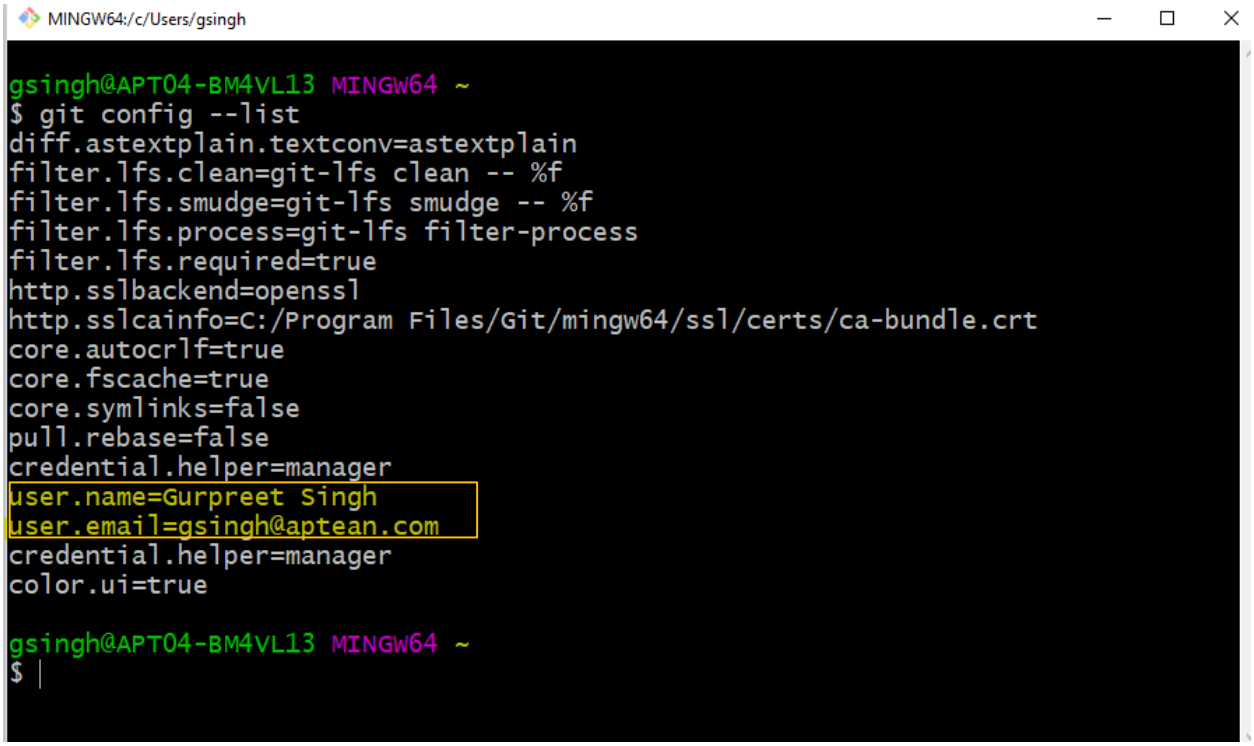
Example: Git config –global user.name Gurpreet Singh

Git config –global user.email <<email id>>

Example: Git config –global user.email gsingh@aptean.com

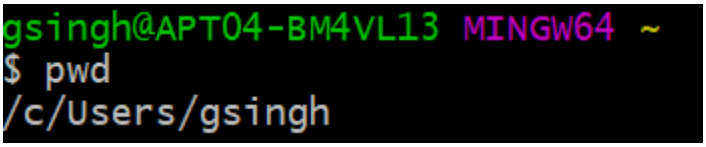
3.) Check if username and email added successfully by executing this command

Git config --list



4.) Check working directory:

Command → pwd



5.) Change directory where you want to create git repo

Command → cd “location”

```
gsingh@APT04-BM4VL13 MINGW64 ~
$ cd "C:\Automation\GitTraining"

gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining
$
```

6.) Create folder with the repo name and go in this folder:

Command → mkdir “repo name” and cd “location”

```
gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining
$ mkdir AUFLiteDemo
```

```
gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining
$ cd AUFLiteDemo

gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining/AUFLiteDemo
$
```

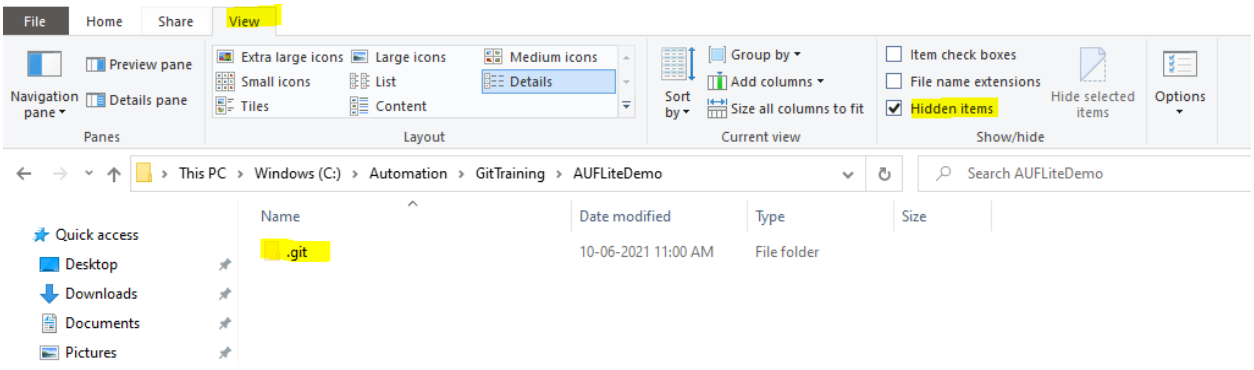
7.) Create git repo at this location:

Command → git init

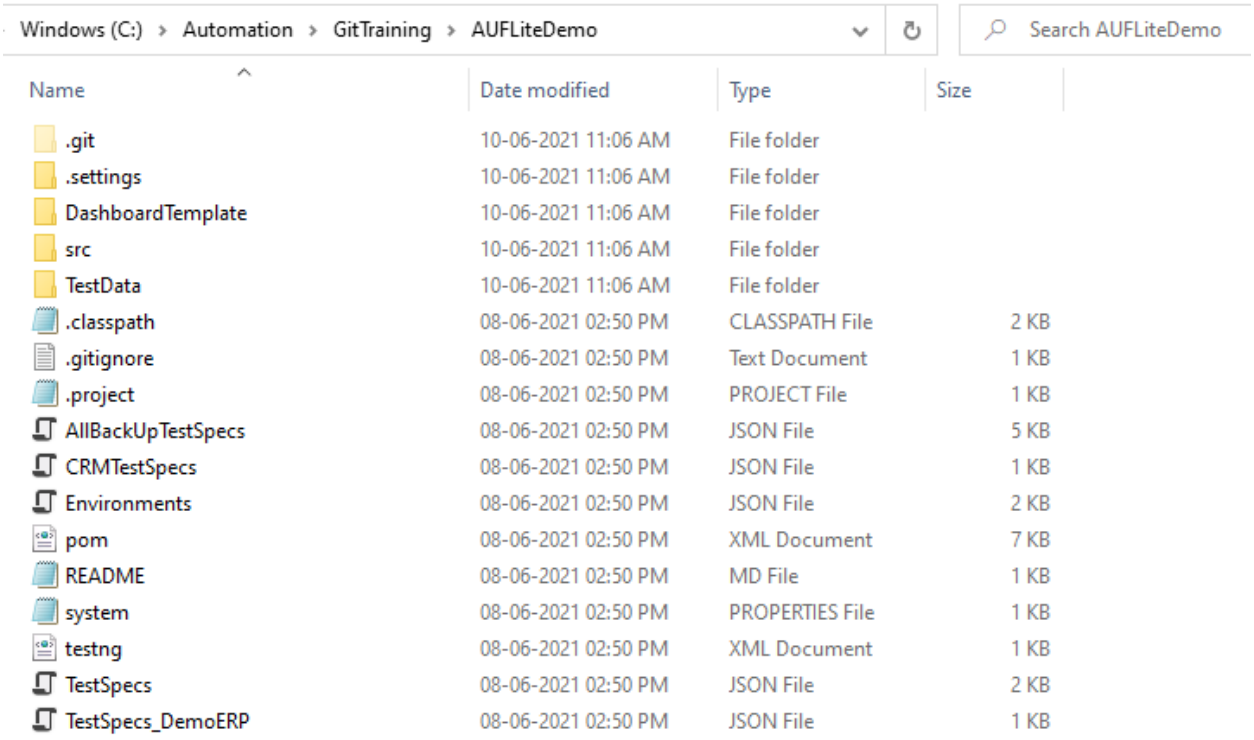
```
gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining/AUFLiteDemo
$ git init
Initialized empty Git repository in C:/Automation/GitTraining/AUFLiteDemo/.git/
```

8.) After executing this command, go to your repo → view → check hidden items checkbox as shown in the image below.

After performing these steps, you will see .git folder. That means git repo has been created successfully in the local machine.



9.) Now copy and paste all AUF Lite code inside this git repo.



Name	Date modified	Type	Size
.git	10-06-2021 11:06 AM	File folder	
.settings	10-06-2021 11:06 AM	File folder	
DashboardTemplate	10-06-2021 11:06 AM	File folder	
src	10-06-2021 11:06 AM	File folder	
TestData	10-06-2021 11:06 AM	File folder	
.classpath	08-06-2021 02:50 PM	CLASSPATH File	2 KB
.gitignore	08-06-2021 02:50 PM	Text Document	1 KB
.project	08-06-2021 02:50 PM	PROJECT File	1 KB
AllBackUpTestSpecs	08-06-2021 02:50 PM	JSON File	5 KB
CRMTestSpecs	08-06-2021 02:50 PM	JSON File	1 KB
Environments	08-06-2021 02:50 PM	JSON File	2 KB
pom	08-06-2021 02:50 PM	XML Document	7 KB
README	08-06-2021 02:50 PM	MD File	1 KB
system	08-06-2021 02:50 PM	PROPERTIES File	1 KB
testng	08-06-2021 02:50 PM	XML Document	1 KB
TestSpecs	08-06-2021 02:50 PM	JSON File	2 KB
TestSpecs_DemoERP	08-06-2021 02:50 PM	JSON File	1 KB

10.) Go to git bash and execute these commands:

Command → git status

```
gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining/AUFLiteDemo (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .classpath
        .gitignore
        .project
        .settings/
        AllBackUpTestSpecs.json
        CRMTestSpecs.json
        DashboardTemplate/
        Environments.json
        README.md
        TestData/
        TestSpecs.json
        TestSpecs_DemoERP.json
        pom.xml
        src/
        system.properties
        testng.xml

nothing added to commit but untracked files present (use "git add" to track)
```

Command → git add .

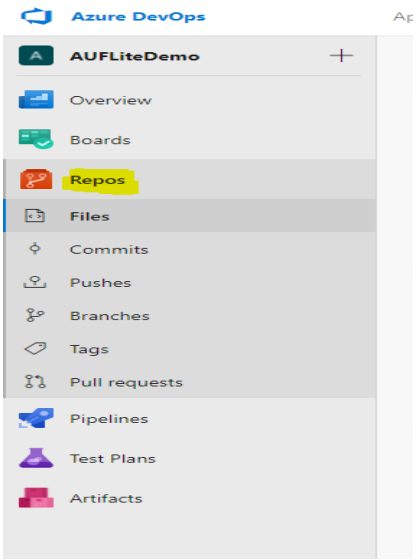
```
gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining/AUFLiteDemo (master)
$ git add .
```

Command → git commit –m “Initial Commit”

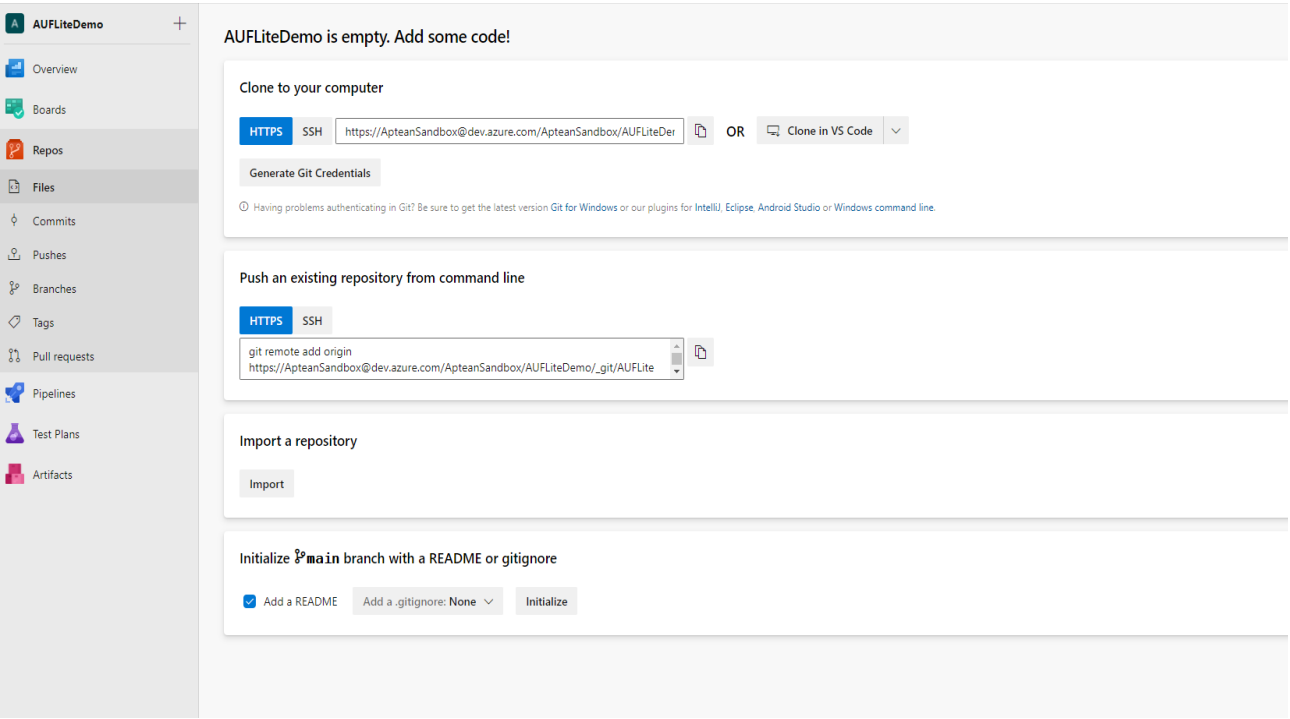
```
gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining/AUFLiteDemo (master)
$ git commit -m "Initial Commit"
[master (root-commit) 7aee090] Initial Commit
 196 files changed, 32423 insertions(+)
 create mode 100644 .classpath
 create mode 100644 .gitignore
 create mode 100644 .project
 create mode 100644 .settings/org.eclipse.core.resources.prefs
 create mode 100644 .settings/org.eclipse.jdt.core.prefs
 create mode 100644 .settings/org.eclipse.m2e.core.prefs
 create mode 100644 AllBackUpTestSpecs.json
 create mode 100644 CRMTestSpecs.json
 create mode 100644 DashboardTemplate/MacroTemplate-V1.xlsx
 create mode 100644 Environments.json
 create mode 100644 README.md
 create mode 100644 TestData/ApiPro - Gina.xlsx
 create mode 100644 TestData/ApiPro - Julia.xlsx
 create mode 100644 TestData/ApiPro - Murugan.xlsx
 create mode 100644 TestData/ApiPro.xlsx
 create mode 100644 TestData/Apprise.xlsx
 create mode 100644 TestData/ApteanPay.xlsx
 create mode 100644 TestData/ApteanPayAssessment.xlsx
 create mode 100644 TestData/CRM.xlsx
 create mode 100644 TestData/Covid19.xlsx
 create mode 100644 TestData/DemoERP_Inventory.xlsx
 create mode 100644 TestData/DemoERP_Inventory_Driver.xlsx
```

11.) Push this local repo to azure git server:

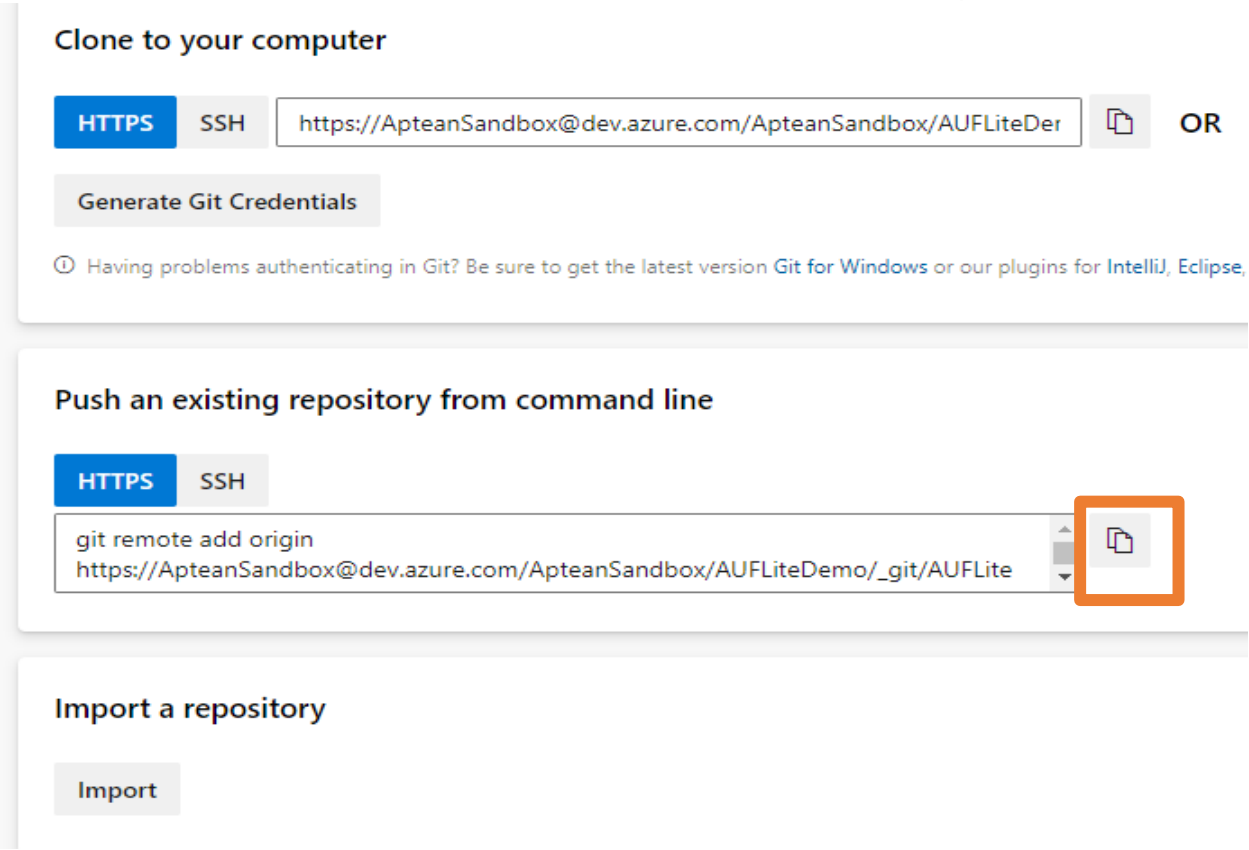
Go to azure project and click on repo in the left panel.



12.) After clicking on repo, this page will open.



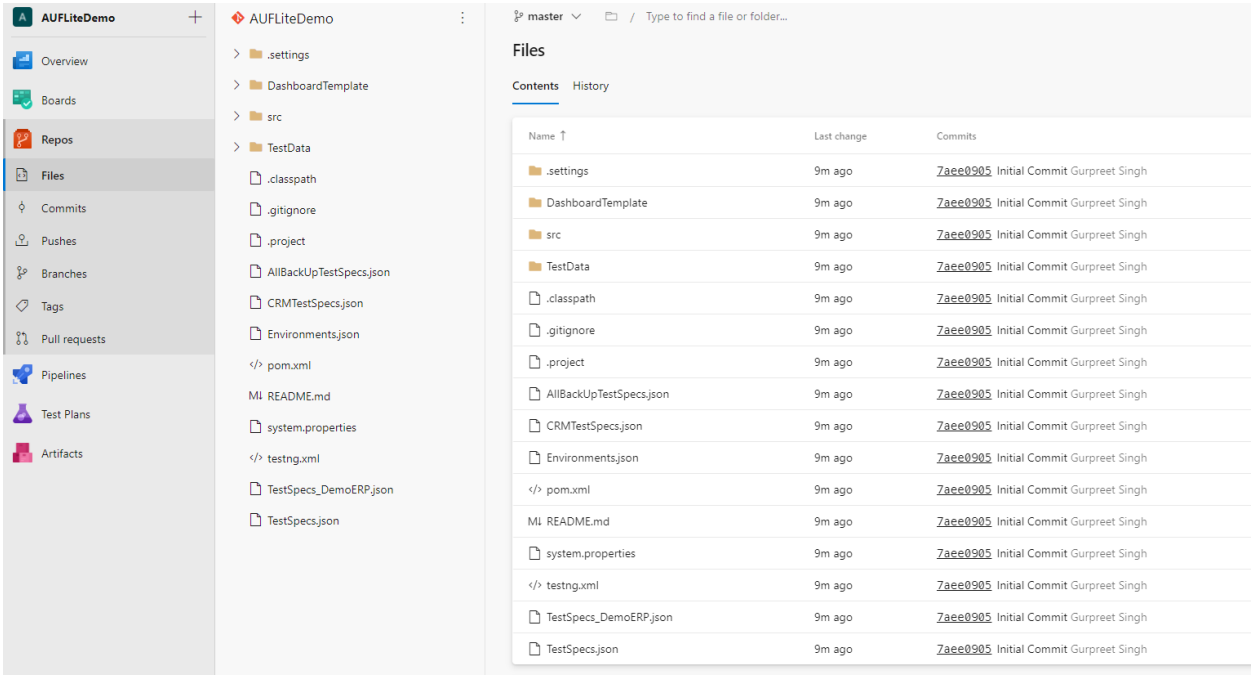
Click on the outlined button as shown in the image below to copy the command



13.) Go to git bash, paste this command and execute

```
gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining/AUFLiteDemo (master)
$ git remote add origin https://ApteanSandbox@dev.azure.com/ApteanSandbox/AUFLiteDemo/_git/AUFLiteDemo

gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining/AUFLiteDemo (master)
$ git push -u origin --all
Enumerating objects: 296, done.
Counting objects: 100% (296/296), done.
Delta compression using up to 8 threads
Compressing objects: 100% (253/253), done.
Writing objects: 100% (296/296), 750.95 KiB | 8.34 MiB/s, done.
Total 296 (delta 86), reused 0 (delta 0), pack-reused 0
remote: Analyzing objects... (296/296) (732 ms)
remote: Storing packfile... done (94 ms)
remote: Storing index... done (59 ms)
To https://dev.azure.com/ApteanSandbox/AUFLiteDemo/_git/AUFLiteDemo
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.

gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining/AUFLiteDemo (master)
$
```

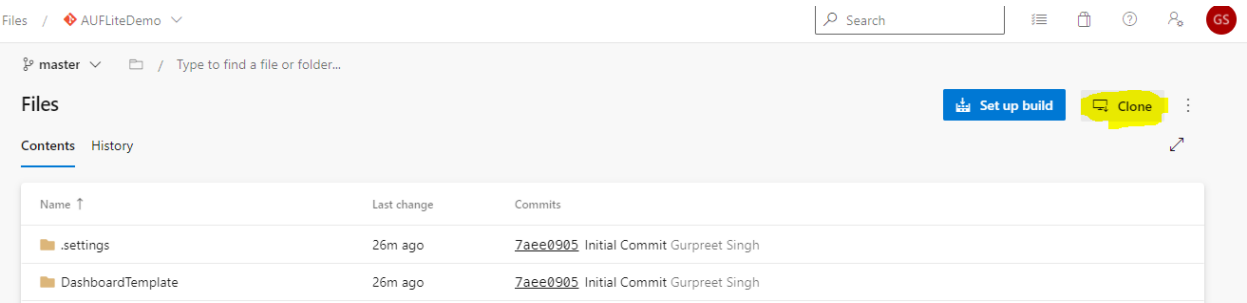
Go to azure project, refresh the page and you will see the code inside the master branch.

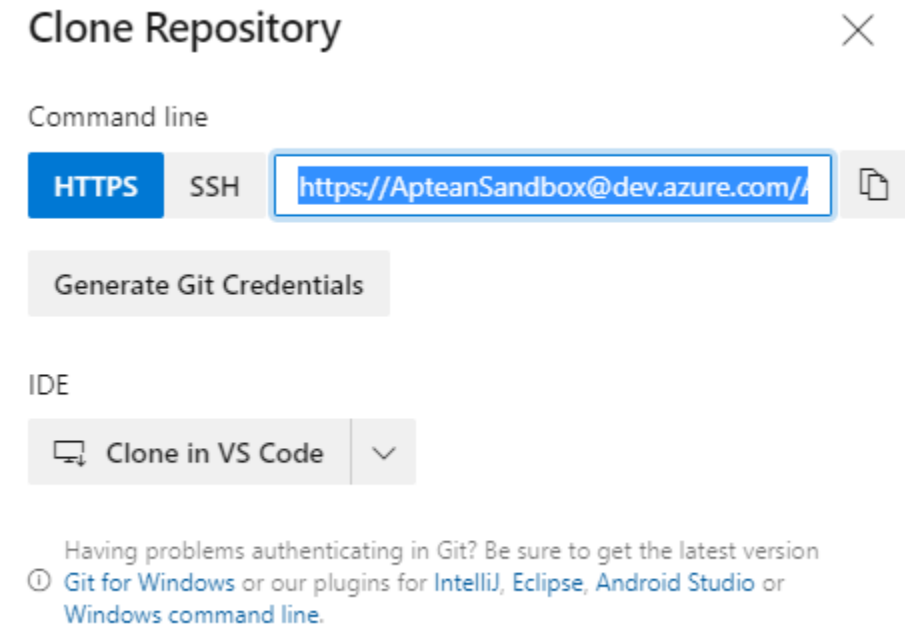


Git flow

1.) Clone the repo in local machine.

Go to the repo you want to clone.
Click on the clone button.

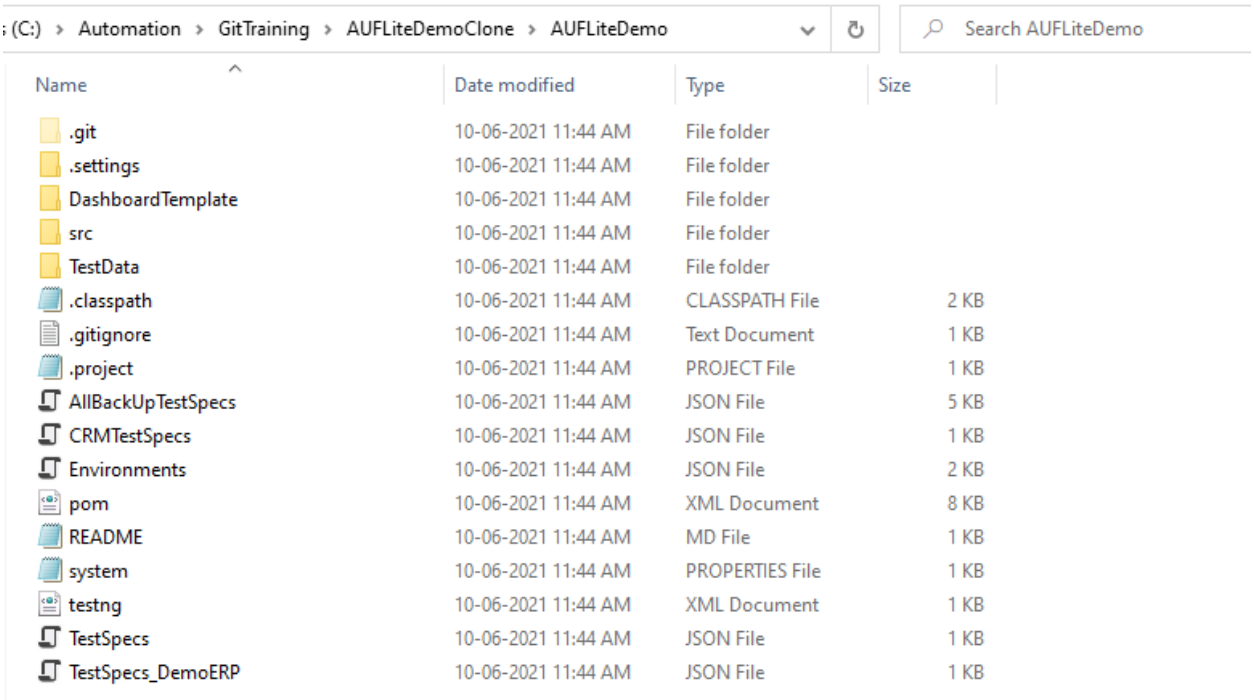




2.) Copy the URL, open git bash in the location where you want to clone the code and execute this command.

```
gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining/AUFLiteDemoClone
$ git clone https://ApteanSandbox@dev.azure.com/ApteanSandbox/AUFLiteDemo/_git/AUFLiteDemo
Cloning into 'AUFLiteDemo'...
remote: Azure Repos
remote: Found 296 objects to send. (27 ms)
Receiving objects: 100% (296/296), 750.75 KiB | 1.04 MiB/s, done.
Resolving deltas: 100% (87/87), done.

gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining/AUFLiteDemoClone
$
```

3.) After executing this command, local git repo will be created at the specified location.



4.) Create Branch.

Command → git branch <<branch name>>

```
gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining/AUFLiteDemoClone/AUFLiteDemo (master)
$ git branch feature
```

5.) Checkout to specific branch.

Command → git checkout <<branch name>>

```
gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining/AUFLiteDemoClone/AUFLiteDemo (master)
$ git checkout feature
Switched to branch 'feature'
```

6.) Check the status of your local repo.

Command → git status

```
gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining/AUFLiteDemoClone/AUFLiteDemo (feature)
$ git status
On branch feature
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   TestSpecs.json

no changes added to commit (use "git add" and/or "git commit -a")
```

7.) Stage changes for commit:

Command → git add .

```
gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining/AUFLiteDemoClone/AUFLiteDemo (feature)
$ git add .
```

8.) Commit changes:

Command → git commit –m “commit message”

```
gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining/AUFLiteDemoClone/AUFLiteDemo (feature)
$ git commit -m "feature created"
[feature c3dde86] feature created
 1 file changed, 4 insertions(+), 4 deletions(-)
```

9.) Get log of the current branch:

Command → git log

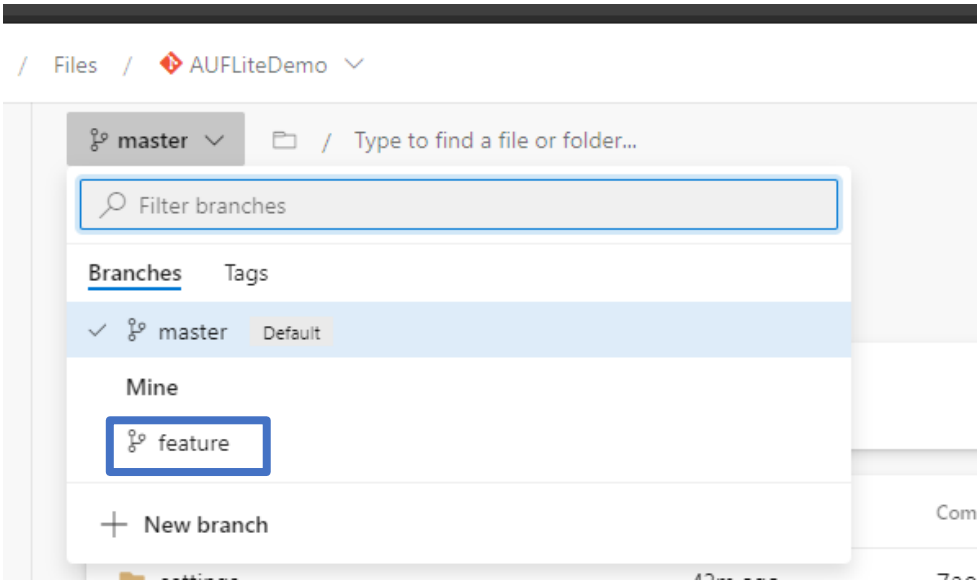
```
gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining/AUFLiteDemoClone/AUFLiteDemo (feature)
$ git log
commit c3dde86f3726d3f18495f16e2ef941fb72761df7 (HEAD -> feature)
Author: Gurpreet Singh <gsingh@aptean.com>
Date:   Thu Jun 10 11:52:50 2021 +0530

    feature created

commit 7aee0905e61b259002161e6401f03c9a33c2f638 (origin/master, origin/HEAD, master)
Author: Gurpreet Singh <gsingh@aptean.com>
Date:   Thu Jun 10 11:14:52 2021 +0530

    Initial Commit
```

10.) Push current branch and all the changes to azure git server:

Command → git push –u origin <<branch name>>

```
gsingh@APT04-BM4VL13 MINGW64 /c/Automation/GitTraining/AUFLiteDemoClone/AUFLiteDemo (feature)
$ git push -u origin feature
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 311 bytes | 311.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Analyzing objects... (3/3) (87 ms)
remote: Storing packfile... done (57 ms)
remote: Storing index... done (50 ms)
To https://dev.azure.com/ApteanSandbox/AUFLiteDemo/_git/AUFLiteDemo
 * [new branch]      feature -> feature
Branch 'feature' set up to track remote branch 'feature' from 'origin'.
```

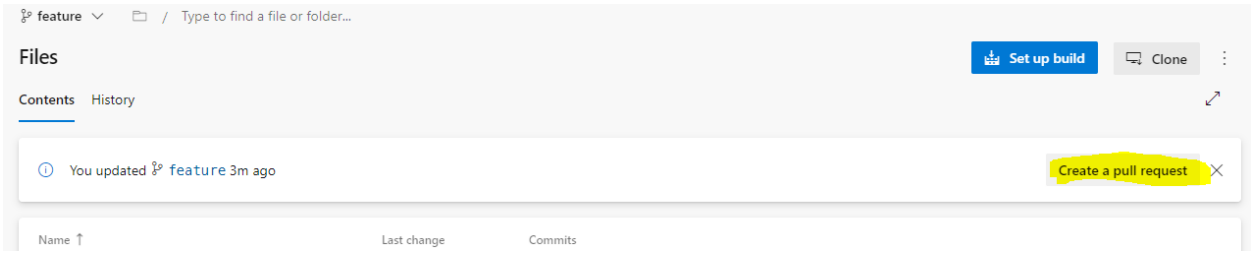
11.) Go to azure project and check if the branch has been created or not.



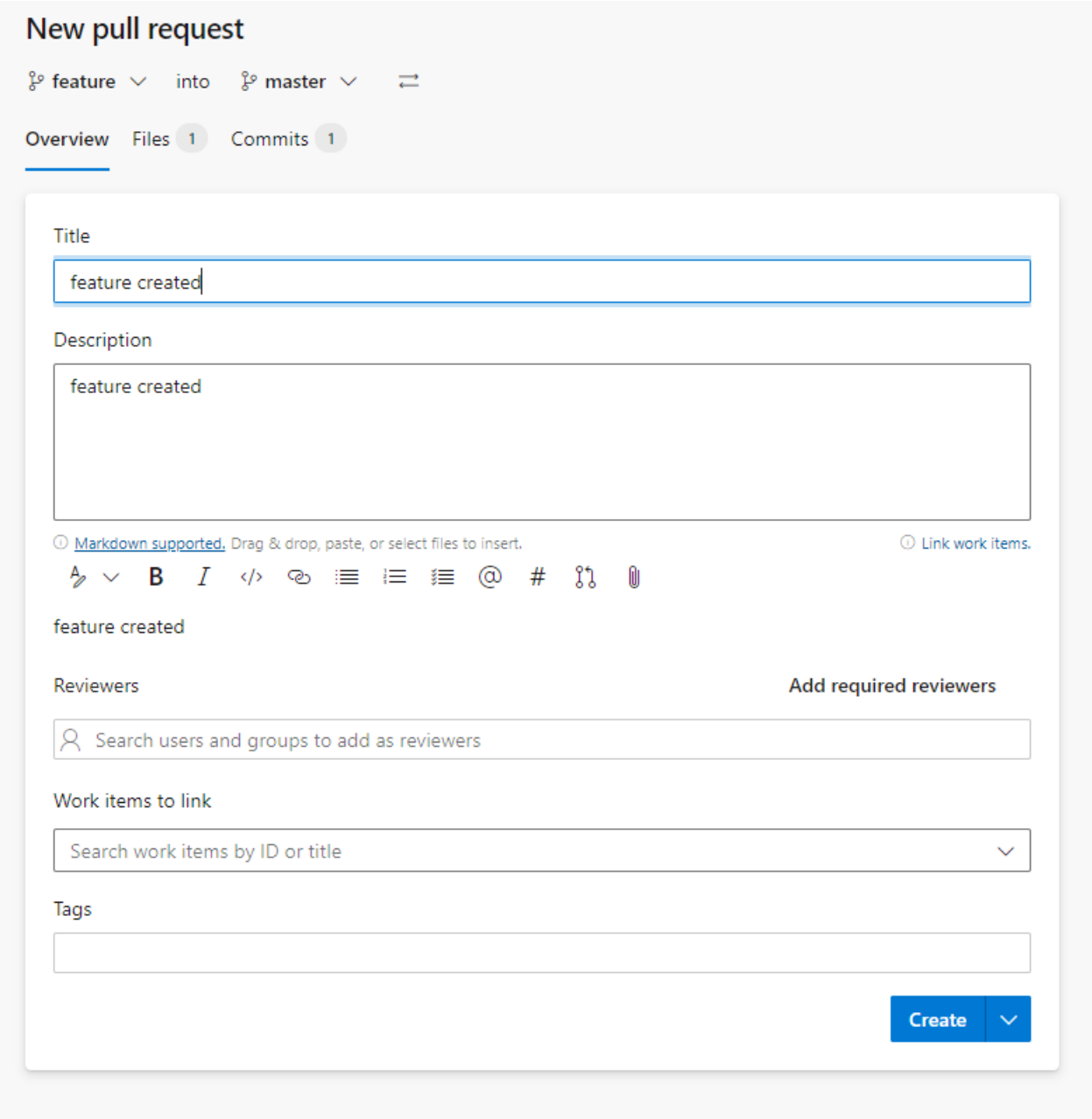
12.) Merge feature branch in master branch

Create a pull request

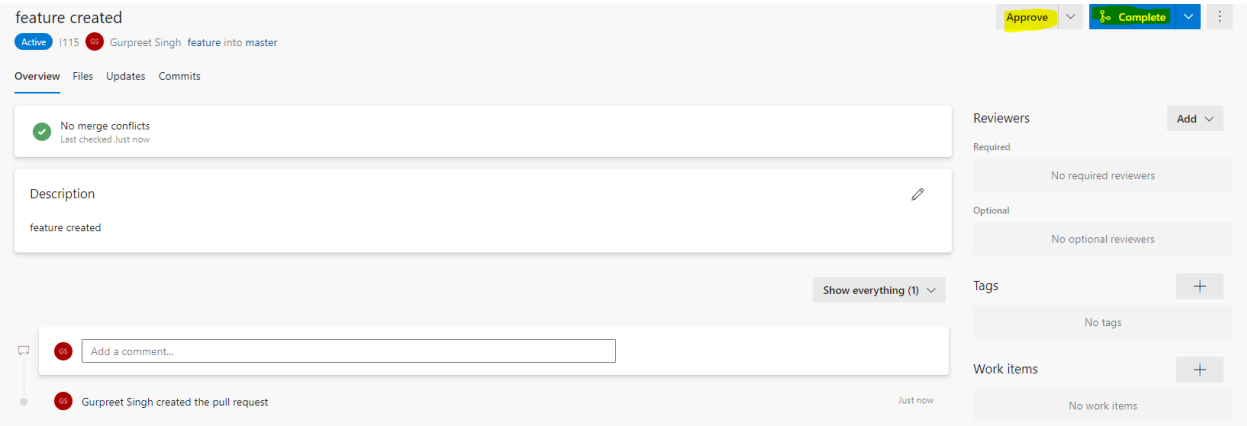
Click on the button highlighted in the image shown below



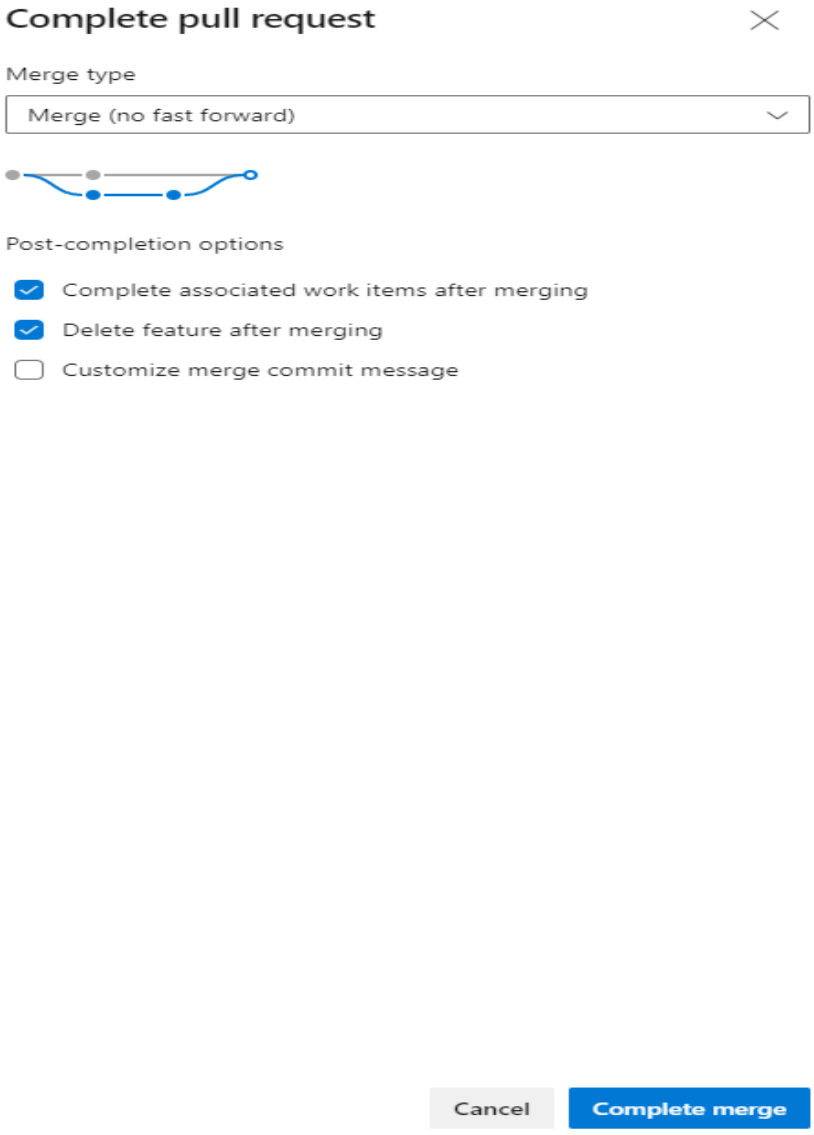
Click on create button to create pull request.



Click on approve and complete button to complete the pull request.



Select these options and click on complete merge.



13.) Retrieve the latest meta-data info from the origin:

Command: git fetch

14.) Update the local version of a repository from a remote:

Command: git pull

15.) Moving or combining a sequence of commits to a new base commit:

Command: git rebase <<branch name>>

If there are some conflicts in the branch, resolve them, and perform below commands to continue changes:

git status

git rebase --continue