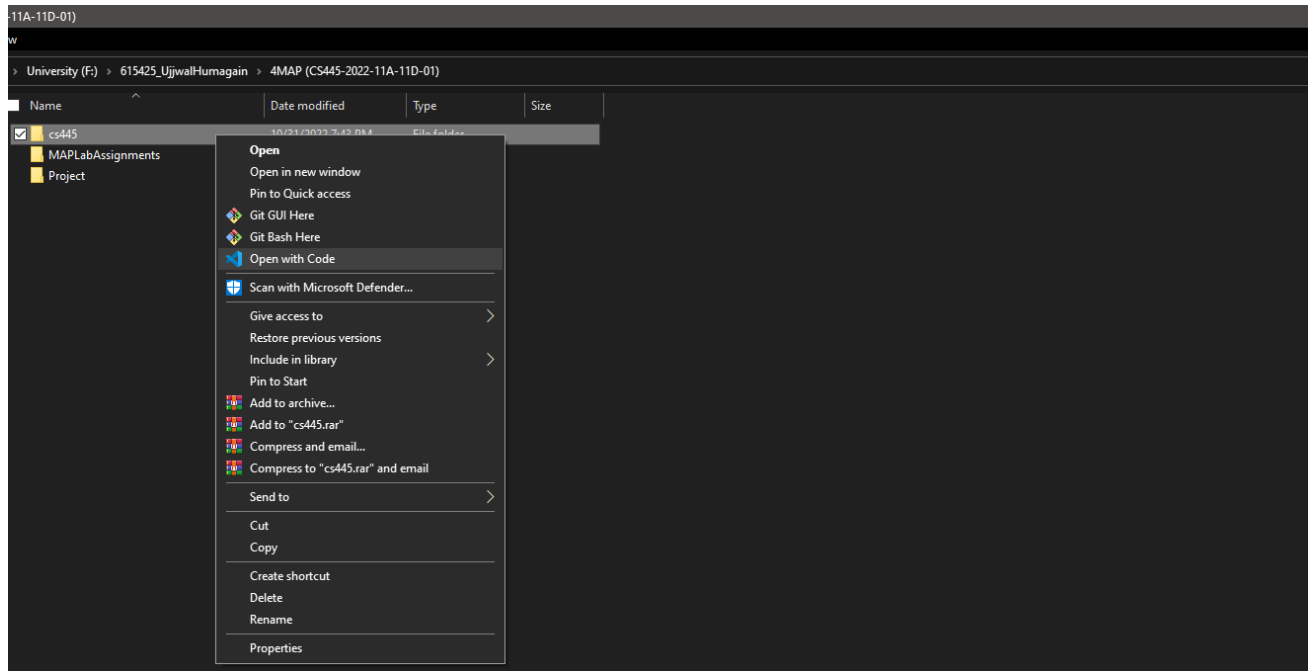


Practice with all the local repository commands:

Create a new folder:

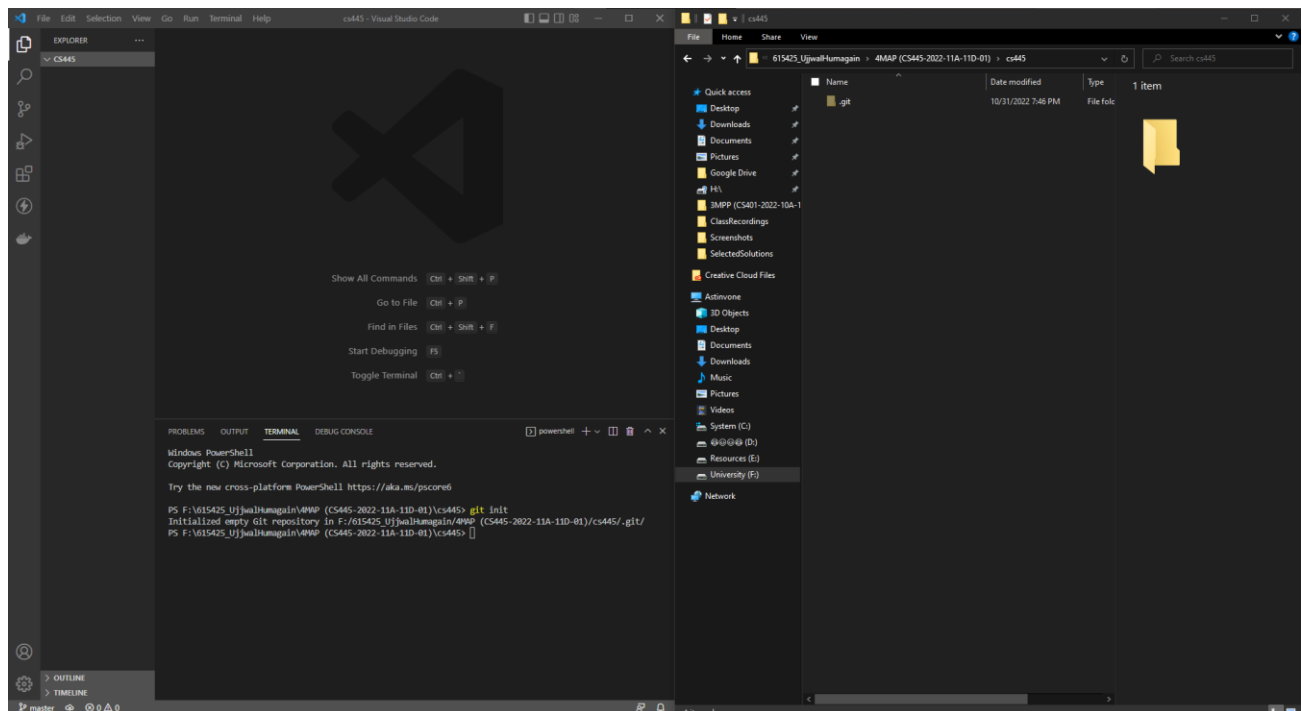
Name	Date modified	Type	Size
cs445	10/31/2022 7:43 PM	File folder	

Open the folder in VS Code:



1. git init

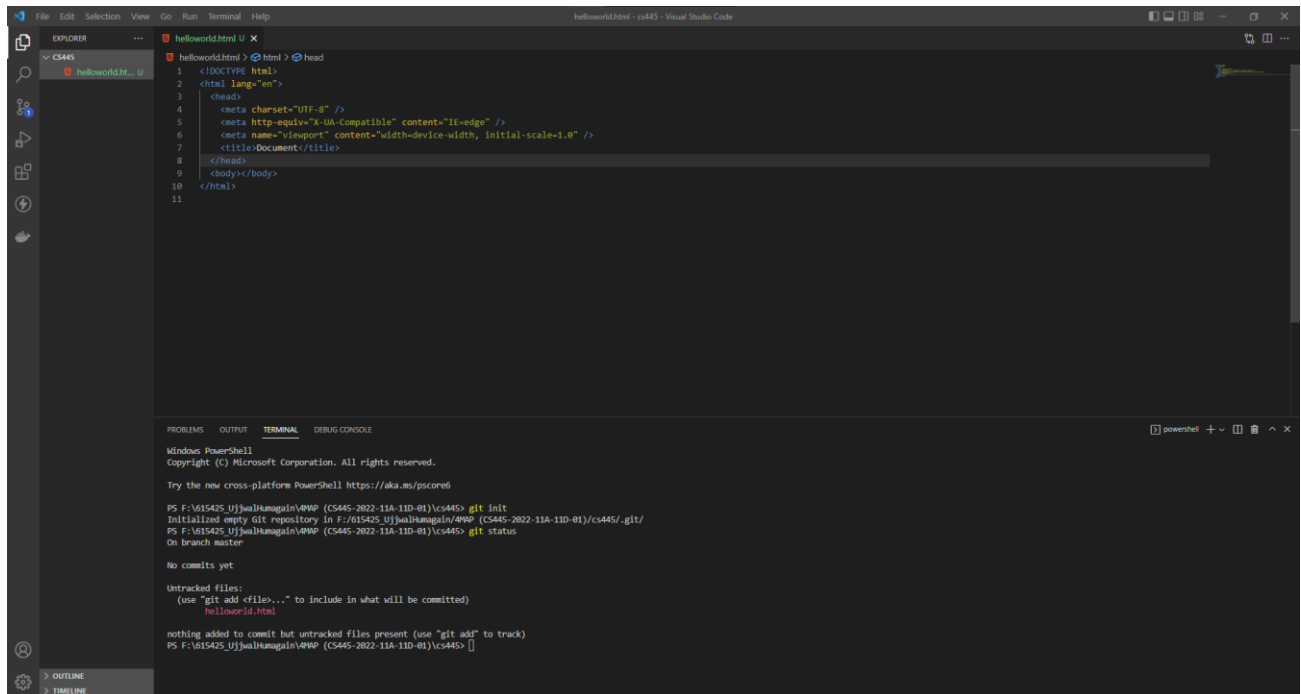
Initialize the folder as a git repository using 'git init' command



A new folder ‘.git’ is created inside the root folder.

2. git status

Now let’s add a new html file inside the working directory and run the ‘git status’ command so as to initiate the tracking of new files/folders.



The screenshot shows the Visual Studio Code interface with a file explorer on the left showing a folder named 'helloworldM...' containing a file 'helloworldM...'. The main editor shows the content of 'helloworld.html', which is a basic HTML document. The terminal at the bottom shows the following commands and output:

```
PS F:\615425_UjjwalKamgain\WWW (C5445-2022-11A-11D-01)\cs445> git init
Initialized empty Git repository in F:\615425_UjjwalKamgain\WWW (C5445-2022-11A-11D-01)\cs445/.git/
PS F:\615425_UjjwalKamgain\WWW (C5445-2022-11A-11D-01)\cs445> git status

On branch master

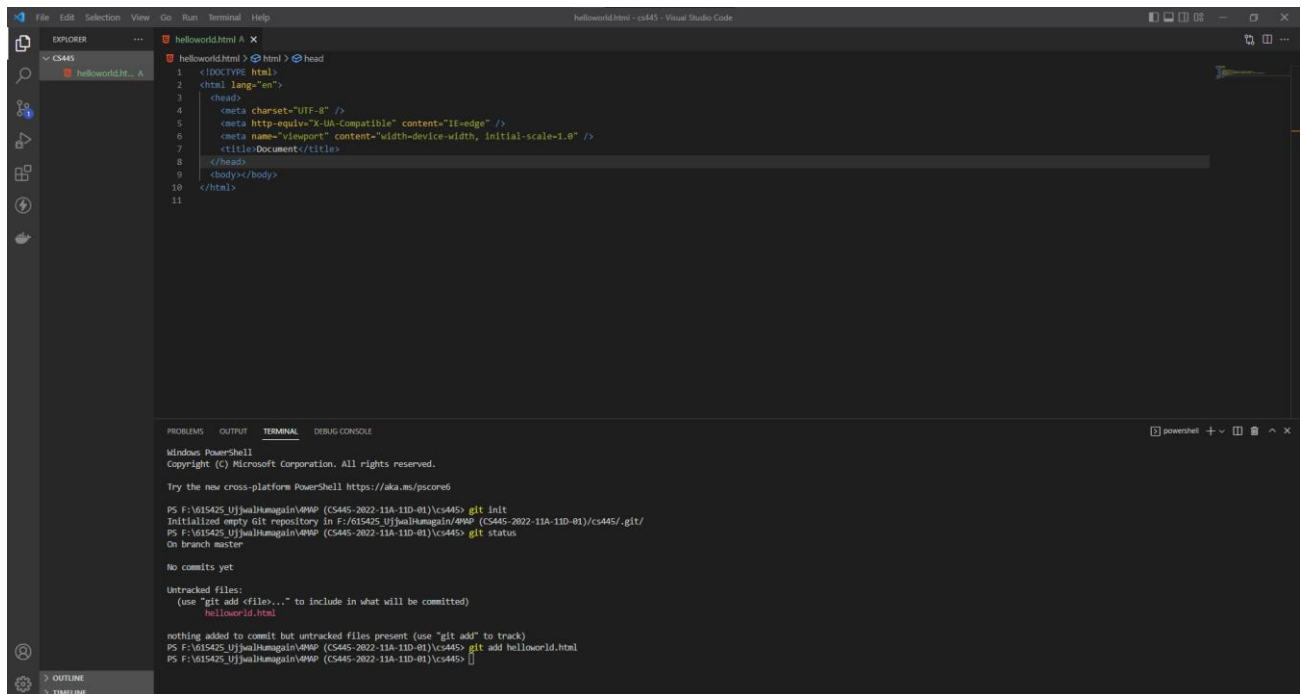
No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    helloworld.html

nothing added to commit but untracked files present (use "git add" to track)
PS F:\615425_UjjwalKamgain\WWW (C5445-2022-11A-11D-01)\cs445>
```

3. git add

Now in order to track the newly added file by adding it to the staging area we have to use ‘git add *filename OR **’ (Here * is as wildcard) command as;



The screenshot shows the Visual Studio Code interface with a file explorer on the left showing a folder named 'helloworldM...' containing a file 'helloworldM...'. The main editor shows the content of 'helloworld.html', which is a basic HTML document. The terminal at the bottom shows the following commands and output:

```
PS F:\615425_UjjwalKamgain\WWW (C5445-2022-11A-11D-01)\cs445> git init
Initialized empty Git repository in F:\615425_UjjwalKamgain\WWW (C5445-2022-11A-11D-01)\cs445/.git/
PS F:\615425_UjjwalKamgain\WWW (C5445-2022-11A-11D-01)\cs445> git status

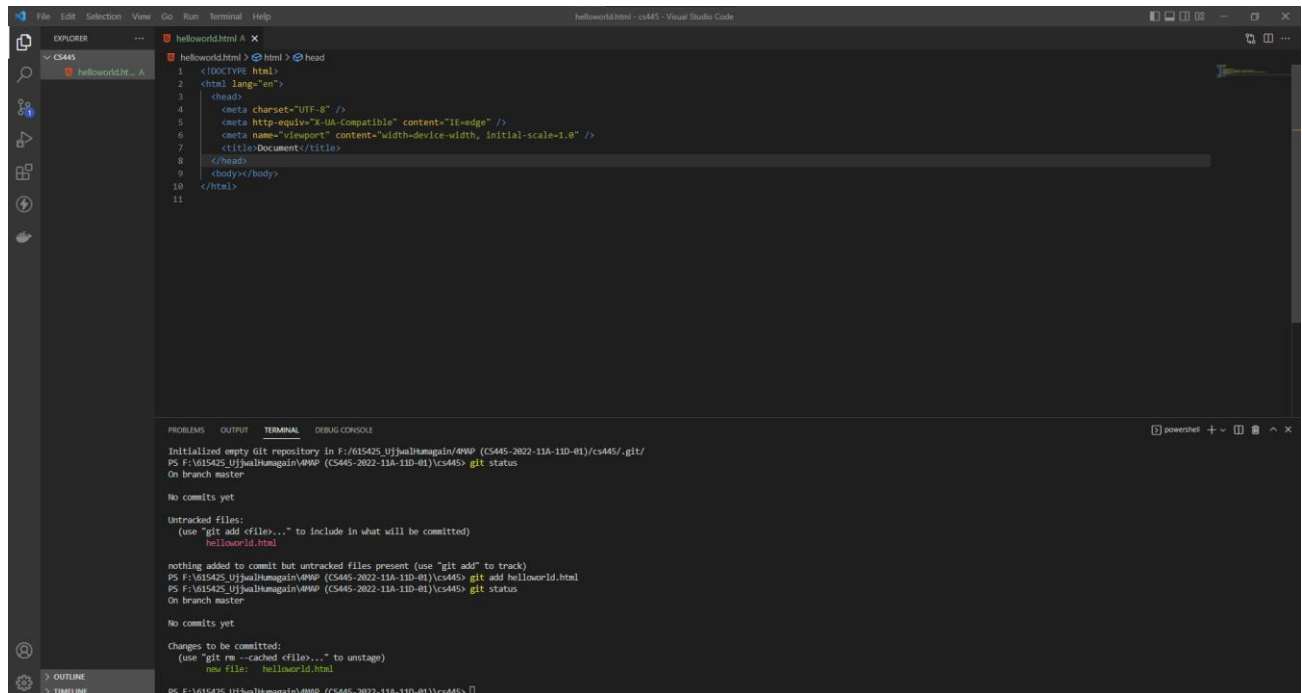
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    helloworld.html

nothing added to commit but untracked files present (use "git add" to track)
PS F:\615425_UjjwalKamgain\WWW (C5445-2022-11A-11D-01)\cs445> git add helloworld.html
PS F:\615425_UjjwalKamgain\WWW (C5445-2022-11A-11D-01)\cs445>
```

After this let us again see whether the file has been tracked or not using ‘git status’ command;



```
helloworld.html - cs445 - Visual Studio Code

EXPLORER
  CS445
    helloworld.html A
  helloworld.html A

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8" />
5   <meta http-equiv="X-UA-Compatible" content="IE=edge" />
6   <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7   <title>Document</title>
8 </head>
9 <body></body>
10 </html>
11

TERMINAL
  Initialized empty Git repository in F:/615425_UjjwalKamgain/WWP (CS445-2022-11A-11D-01)/cs445/.git/
  PS F:/615425_UjjwalKamgain/WWP (CS445-2022-11A-11D-01)/cs445> git status
  On branch master

  No commits yet

  Untracked files:
  (use "git add <file>..." to include in what will be committed)
    helloworld.html

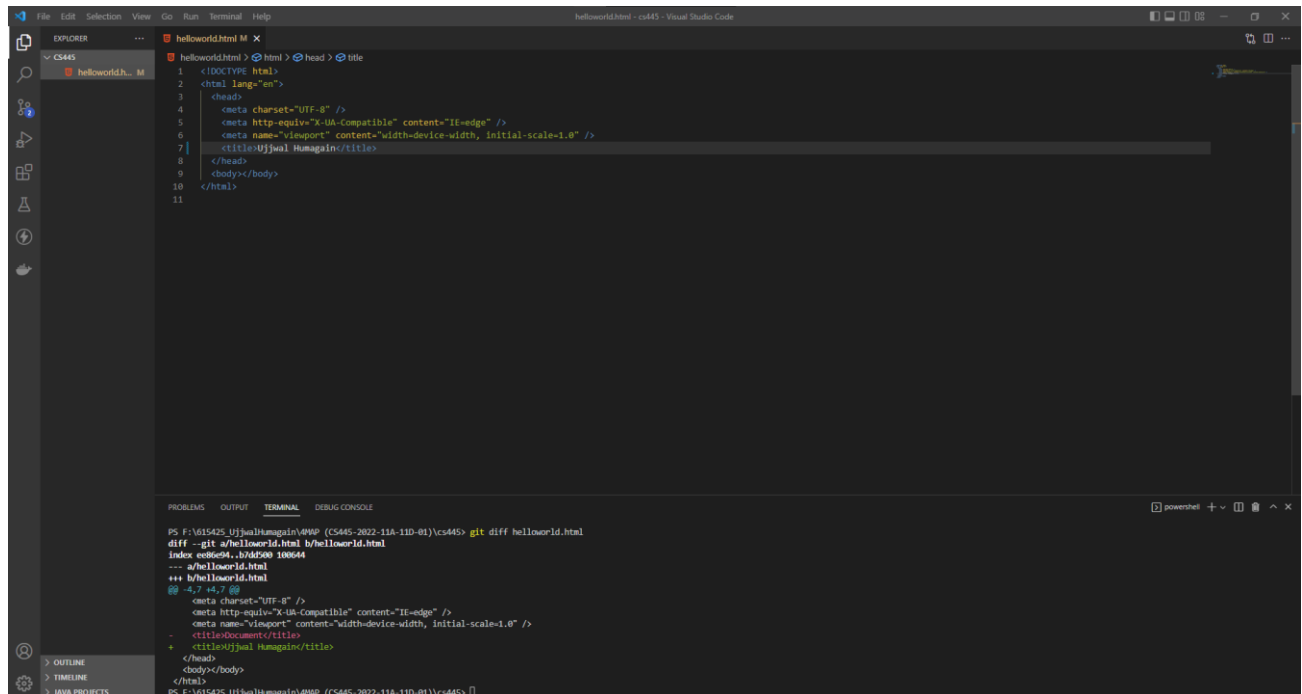
  nothing added to commit but untracked files present (use "git add" to track)
  PS F:/615425_UjjwalKamgain/WWP (CS445-2022-11A-11D-01)/cs445> git add helloworld.html
  PS F:/615425_UjjwalKamgain/WWP (CS445-2022-11A-11D-01)/cs445> git status
  On branch master

  No commits yet

  Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   helloworld.html
  PS F:/615425_UjjwalKamgain/WWP (CS445-2022-11A-11D-01)/cs445> |
```

4. git diff

Since, we have the file in staging area, so any changes made to it is tracked and we have to add it again. In order to see the changes we have made to the file there is a command ‘git diff *filename*’ which shows what changes are made to the specific file.

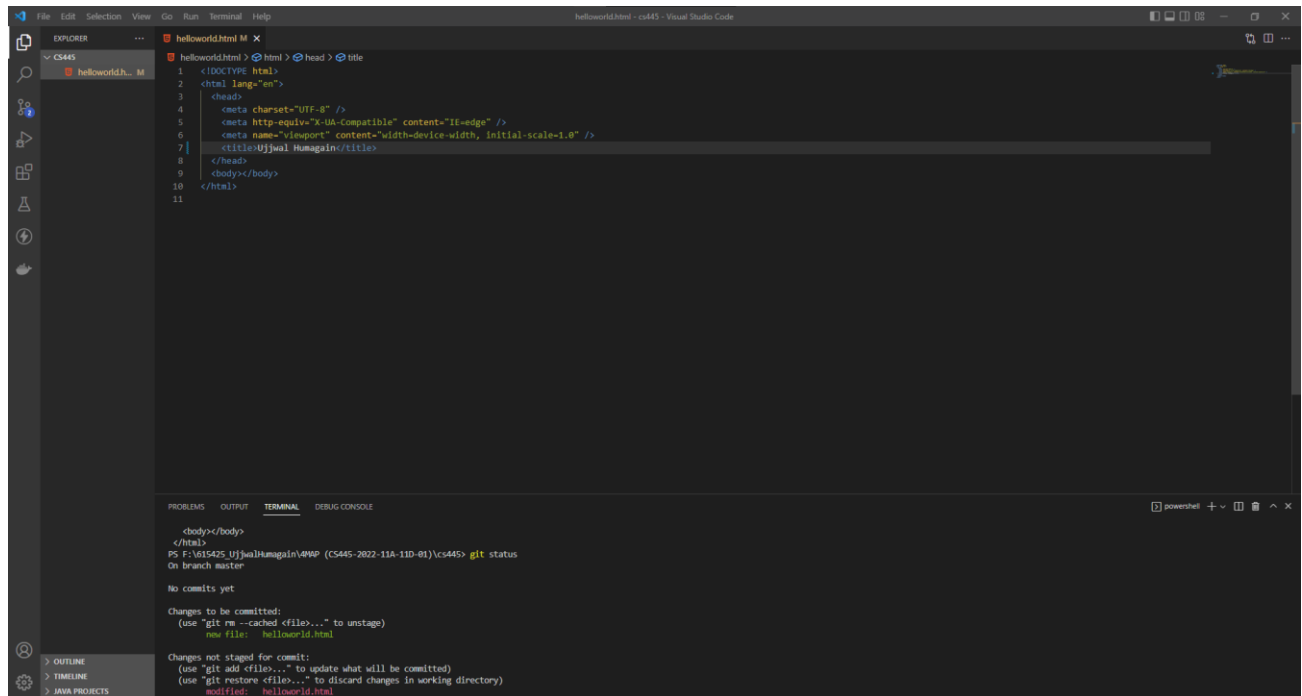


```
helloworld.html - cs445 - Visual Studio Code

EXPLORER
  CS445
    helloworld.html M
  helloworld.html M

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8" />
5   <meta http-equiv="X-UA-Compatible" content="IE=edge" />
6   <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7   <title>Ujjwal Kamgain</title>
8 </head>
9 <body></body>
10 </html>
11

TERMINAL
  PS F:/615425_UjjwalKamgain/WWP (CS445-2022-11A-11D-01)/cs445> git diff helloworld.html
  diff --git a/helloworld.html b/helloworld.html
  index ea86994..b7d5900 100644
  --- a/helloworld.html
  +++ b/helloworld.html
  @@ -4,7 +4,7 @@
  <meta charset="UTF-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  - <title>Document</title>
  + <title>Ujjwal Kamgain</title>
  </head>
  <body></body>
  </html>
  PS F:/615425_UjjwalKamgain/WWP (CS445-2022-11A-11D-01)/cs445> |
```



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8" />
5   <meta http-equiv="X-UA-Compatible" content="IE=edge" />
6   <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7   <title>Ujjwal Hamagain</title>
8 </head>
9 <body></body>
10 </html>
11
```

```
PS F:\615425-UjjwalHamagain\@WP (CS445-2022-11A-11D-81)\cs445> git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   helloworld.html

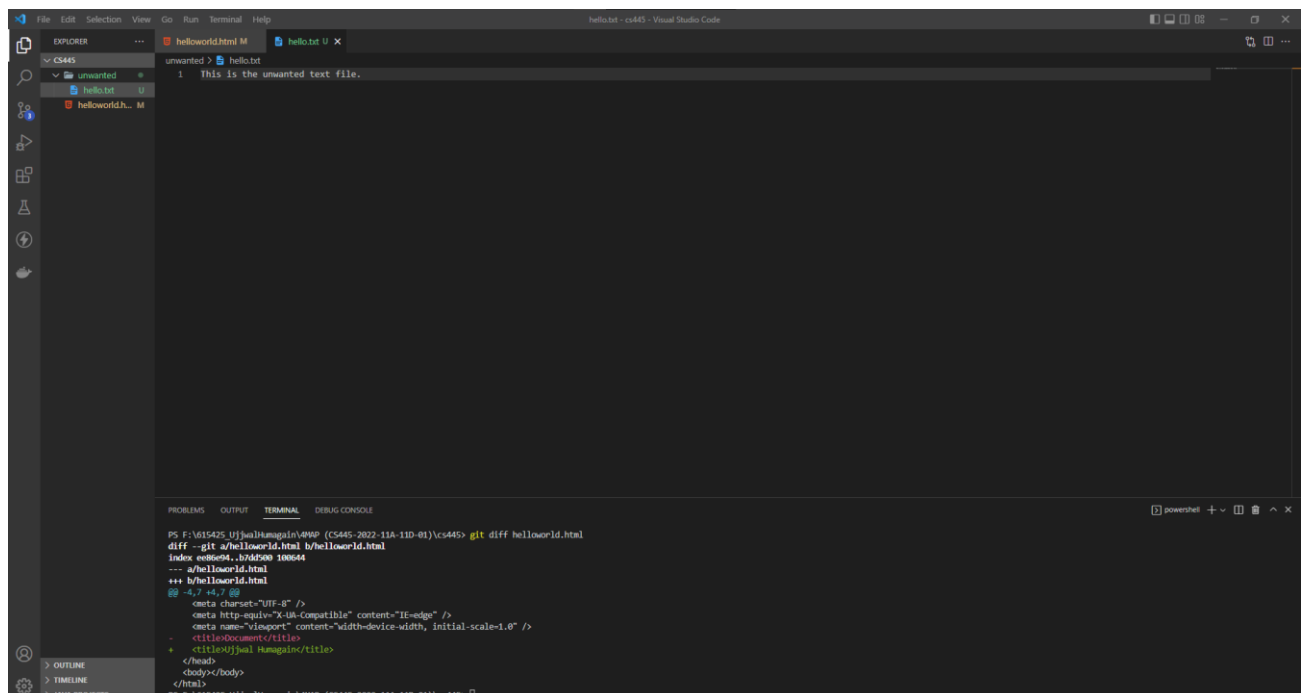
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   helloworld.html
```

We have to add the file again to stage changes made to it.

5. .gitignore

In most of the software projects there are different files and folders which are machine generated and build artifacts. So, in order to track the version of those projects by considering the development code only we have to ignore the generated files and folders. In order to do so, we have to create a new file ‘.gitignore’ inside the working directory and add path to the respective files and folders which are to be ignored and discarded from being tracked during the versioning of the project.

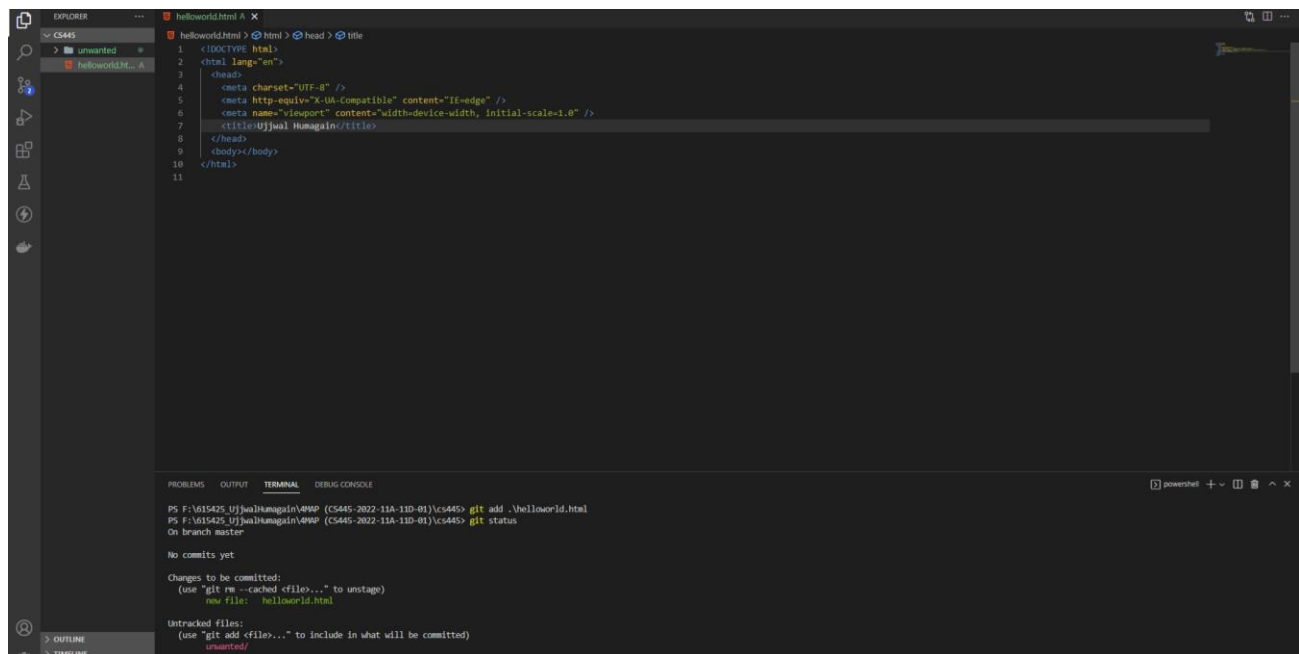
To demonstrate the working of .gitignore at first let us create a new folder and add a file inside it as;



```
1 This is the unwanted text file.
```

```
PS F:\615425-UjjwalHamagain\@WP (CS445-2022-11A-11D-81)\cs445> git diff helloworld.html
diff --git a/helloworld.html b/helloworld.html
index ee8694..b7d4590 100644
--- a/helloworld.html
+++ b/helloworld.html
@@ -4,7 +4,7 @@
   <meta charset="UTF-8" />
   <meta http-equiv="X-UA-Compatible" content="IE=edge" />
   <meta name="viewport" content="width=device-width, initial-scale=1.0" />
+  <title>Document</title>
+  <title>Ujjwal Hamagain</title>
   </head>
   <body></body>
   </html>
```

Since we created a new folder and file which are untracked at first as;



The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left shows a folder named 'CS445' containing an 'unwanted' folder and a file 'helloworld.html'. The main editor area displays the content of 'helloworld.html', which is a basic HTML5 document. The bottom panel shows the 'TERMINAL' with the following output:

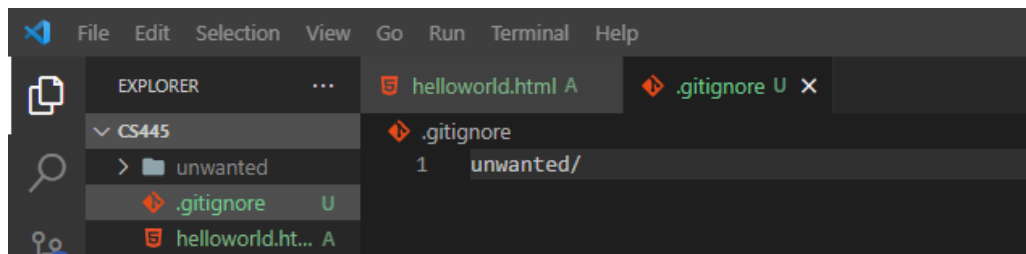
```
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git add .helloworld.html
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   helloworld.html

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    unwanted/
```

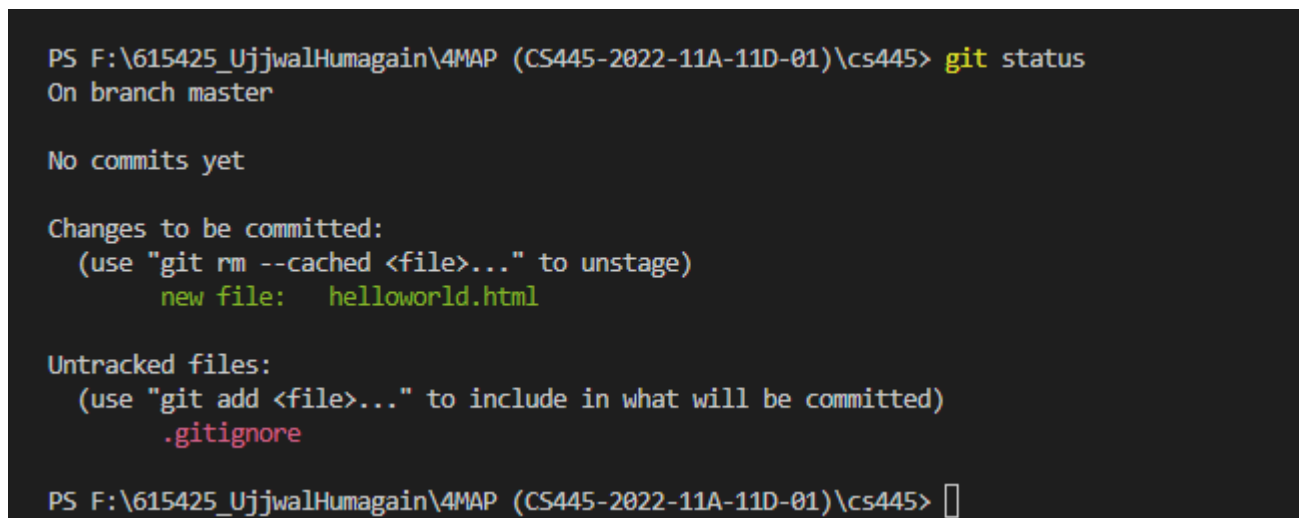
But we want to ignore the folder and files inside it so for this at first we have to add .gitignore in the working directory and add the path to the folder in it as;



The screenshot shows the Explorer sidebar in VS Code. The 'CS445' folder now contains a '.gitignore' file and the 'helloworld.html' file. The '.gitignore' file is highlighted, and its content is shown in the main editor area:

```
.gitignore
1  unwanted/
```

Again, let us do the 'git status'. This time the .gitignore file is untracked instead of the folder that we created previously.



The screenshot shows a terminal window with the following output:

```
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   helloworld.html

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    .gitignore

PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> █
```

Now, let us add the .gitignore file and move everything in the staging area as;

```

PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git add .gitignore
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   .gitignore
    new file:   helloworld.html

```

6. git commit

Since our main intention is to create a local repository and version the working directory so we have to create checkpoints for tracking every change made inside it. For this we have to use “git commit -m ‘commit message’ ” command. This saves the everything presented in the staging area to the repository. The commit message attached in this command explains what changes has been done in a single unit of codebase.

```

PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git commit -m "Initial Commit"
[master (root-commit) 39ee077] Initial Commit
 2 files changed, 11 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 helloworld.html
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> 

```

7. git log

In every project there are more than one commits so to see all the commits there is a command as ‘git log’ that displays every commit made in that repository

```

PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE

PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git commit -m "Initial Commit"
[master (root-commit) 39ee077] Initial Commit
 2 files changed, 11 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 helloworld.html
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git log
commit 39ee07704e1b486708384b28d7fc8decb1610ed (HEAD -> master)
Author: deStarxis <humagainuzjwol@gmail.com>
Date:   Mon Oct 31 20:26:29 2022 -0500

    Initial Commit
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> 

```

We can also format the git log using other optional command attributes as;

```

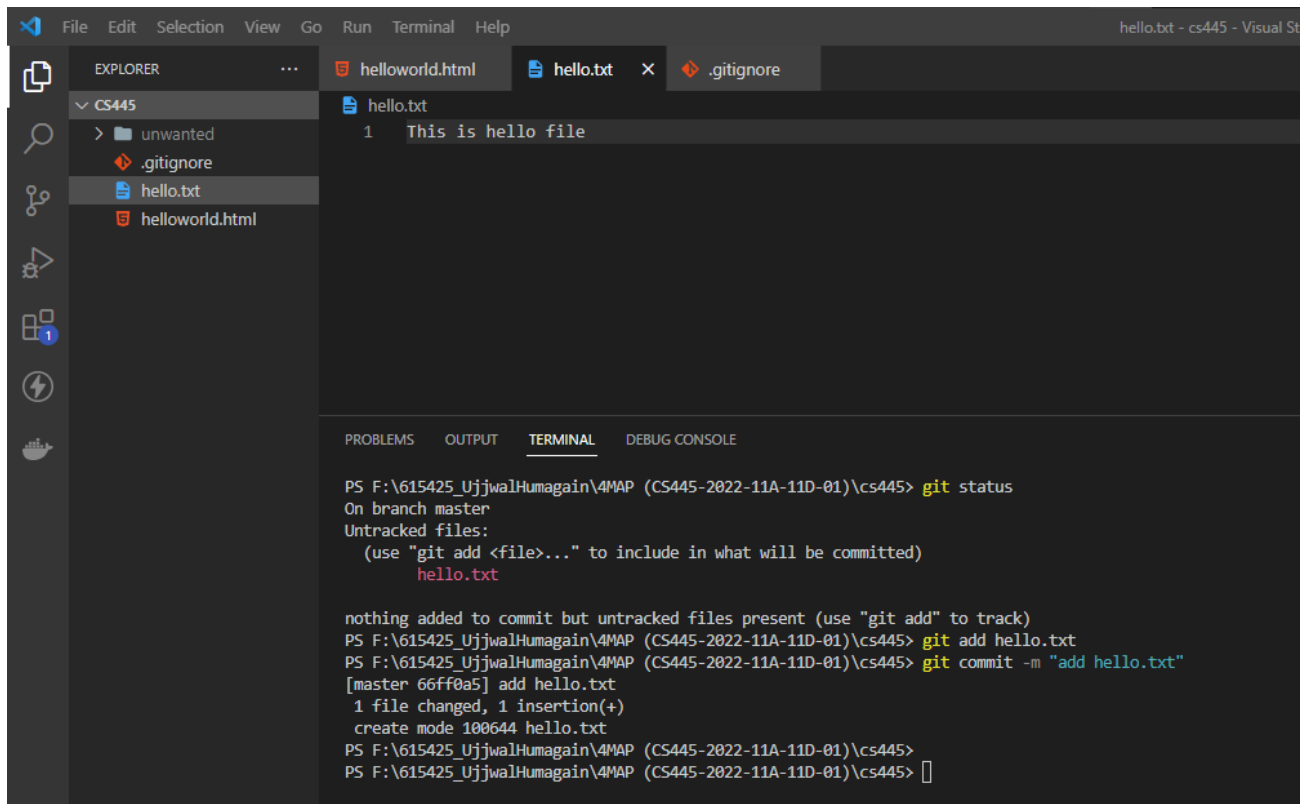
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git log
commit 39ee07704e1b486708384b28d7fc8decb1610ed (HEAD -> master)
Author: deStarxis <humagainuzjwol@gmail.com>
Date:   Mon Oct 31 20:26:29 2022 -0500

    Initial Commit
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git log --oneline
39ee077 (HEAD -> master) Initial Commit
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git log --all --pretty=format:"%h %cd %s (%an)" --since="7 days ago"
39ee077 Mon Oct 31 20:26:29 2022 -0500 Initial Commit (deStarxis)
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> 

```

8. git commit -amend

We can also update the previous commit by using `git commit -amend` command so that the changes done multiple times can be added under same commit. For the demonstration of this, at first let's add one commit and amend another commit to it as;



```
File Edit Selection View Go Run Terminal Help
hello.txt - cs445 - Visual Studio Code

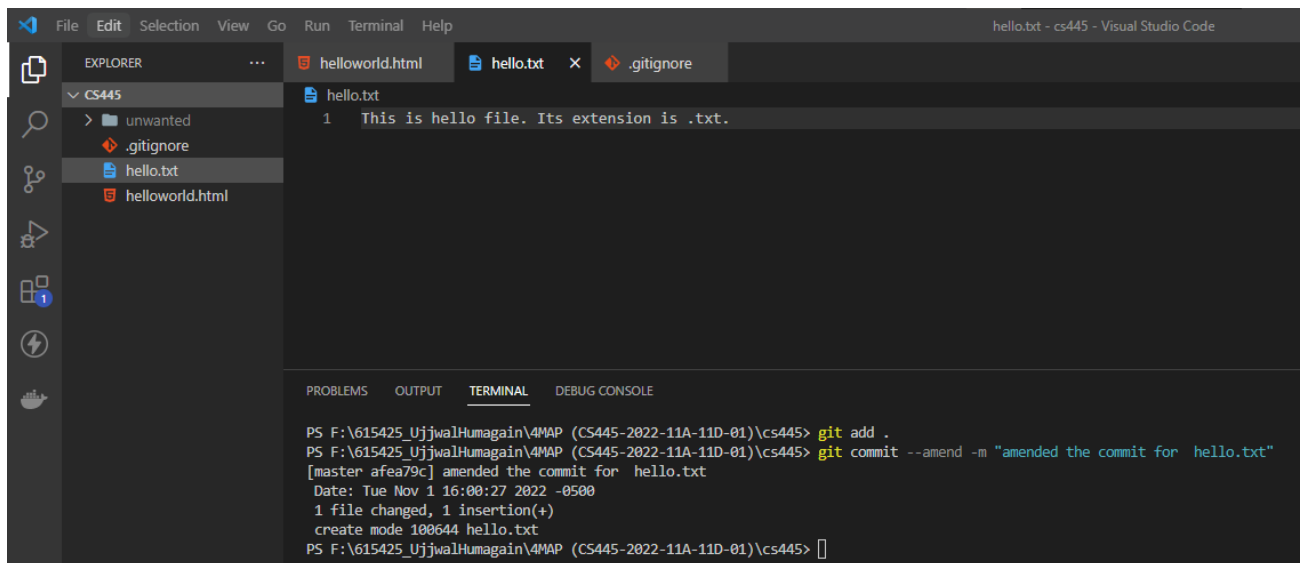
EXPLORER
CS445
  unwanted
  .gitignore
  hello.txt
  helloworld.html

hello.txt
1 This is hello file

TERMINAL
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    hello.txt

nothing added to commit but untracked files present (use "git add" to track)
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git add hello.txt
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git commit -m "add hello.txt"
[master 66ff0a5] add hello.txt
1 file changed, 1 insertion(+)
create mode 100644 hello.txt
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445>
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445>
```

Again, the `hello.txt` file will be modified once, and it will be again added to the repository. But this time instead of creating a new commit, the changes will be amended to the previous commit as;



```
File Edit Selection View Go Run Terminal Help
hello.txt - cs445 - Visual Studio Code

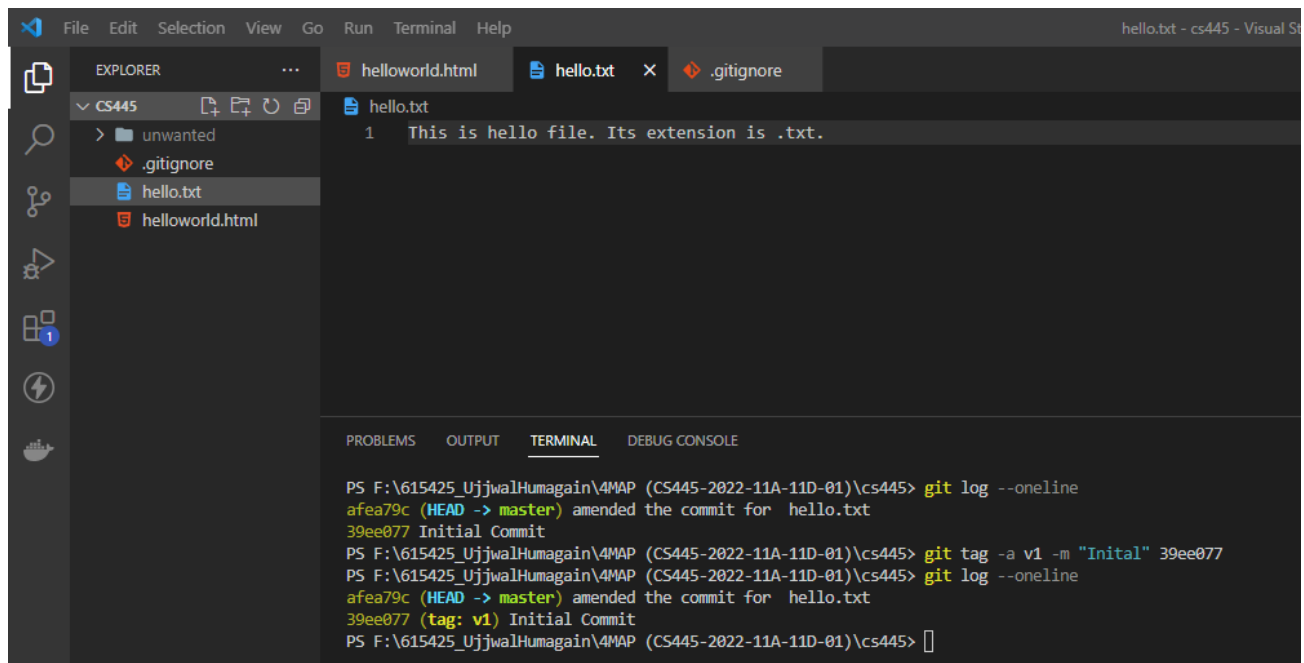
EXPLORER
CS445
  unwanted
  .gitignore
  hello.txt
  helloworld.html

hello.txt
1 This is hello file. Its extension is .txt.

TERMINAL
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git add .
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git commit --amend -m "amended the commit for hello.txt"
[master afea79c] amended the commit for hello.txt
Date: Tue Nov 1 16:00:27 2022 -0500
1 file changed, 1 insertion(+)
create mode 100644 hello.txt
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445>
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445>
```

9. git tag

We can also add marker to each of the commits so as to recognize them easily while performing checkout and reset. So, let us add tag to the first commit that we created in the repository as;



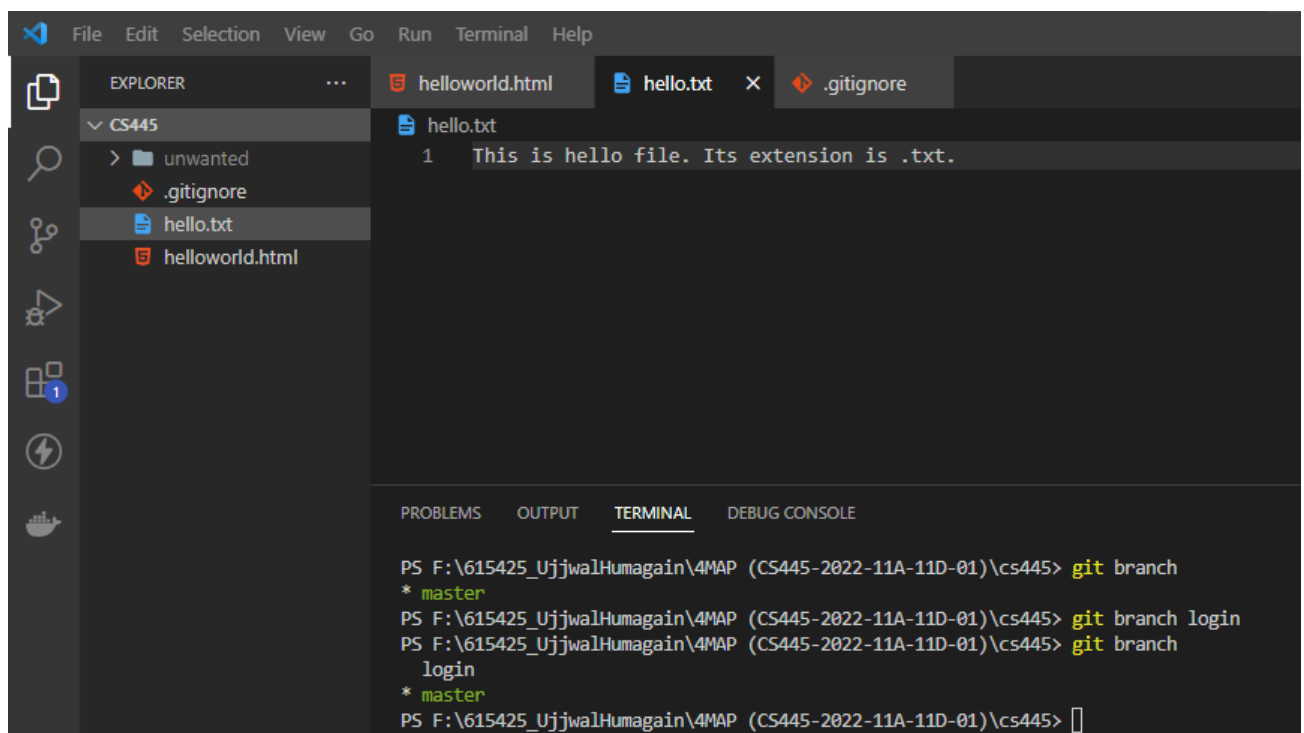
The screenshot shows the Visual Studio Code interface with the Explorer sidebar on the left displaying the file structure of a repository named 'CS445'. The files listed are 'unwanted', '.gitignore', 'hello.txt', and 'helloworld.html'. The main editor area shows the 'hello.txt' file with the content: '1 This is hello file. Its extension is .txt.'. The bottom panel shows the 'TERMINAL' tab with the following commands and output:

```
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git log --oneline
afea79c (HEAD -> master) amended the commit for hello.txt
39ee077 Initial Commit
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git tag -a v1 -m "Initial" 39ee077
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git log --oneline
afea79c (HEAD -> master) amended the commit for hello.txt
39ee077 (tag: v1) Initial Commit
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445>
```

Now with the help of this tag we can easily perform operations on that commit.

10. git branch

In repository a default branch is present initially, but we can have more based on the requirements and use case. Using 'git branch' command we can list all the branches present in the repository. We can also create a new branch using 'git branch <branchname>' command as;



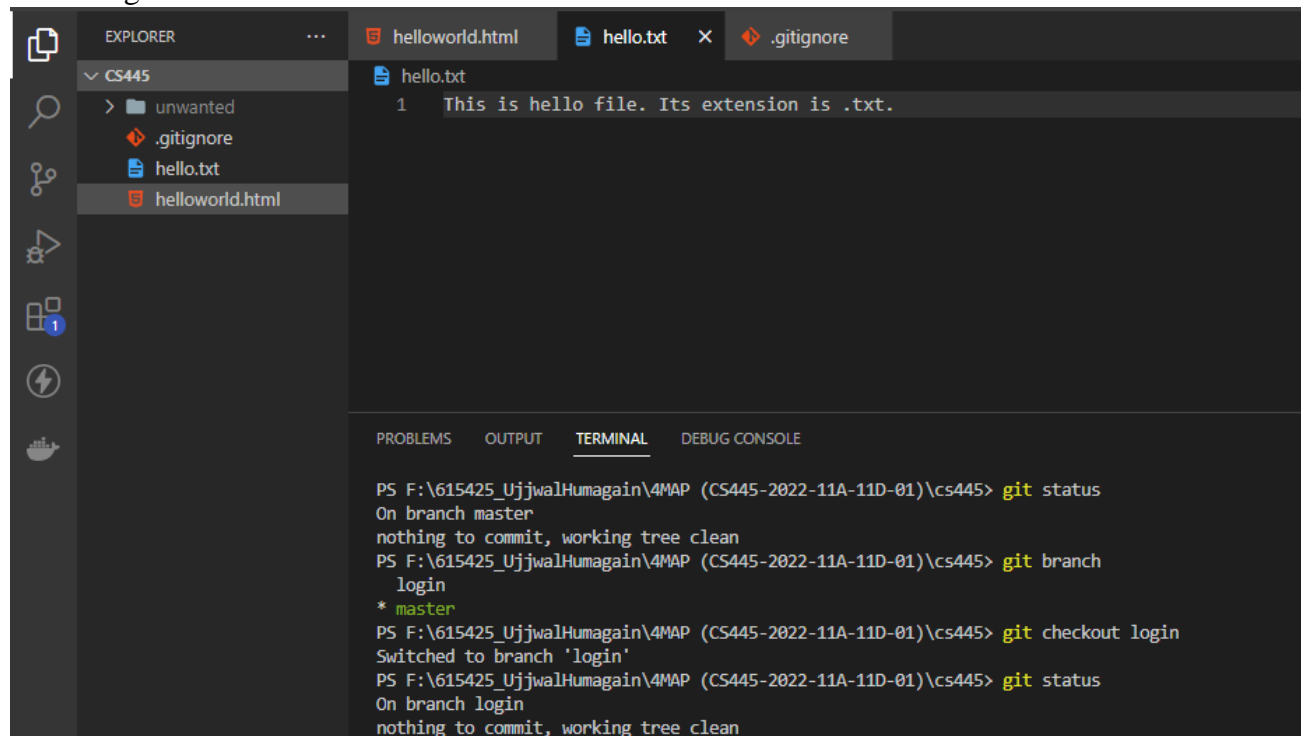
The screenshot shows the Visual Studio Code interface with the Explorer sidebar on the left displaying the file structure of a repository named 'CS445'. The files listed are 'unwanted', '.gitignore', 'hello.txt', and 'helloworld.html'. The main editor area shows the 'hello.txt' file with the content: '1 This is hello file. Its extension is .txt.'. The bottom panel shows the 'TERMINAL' tab with the following commands and output:

```
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git branch
* master
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git branch login
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git branch
* master
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445>
```


11. git checkout

We can also switch from one branch to another and from one commit to another using 'git checkout' command. Here, instead of checkout we can also use switch.

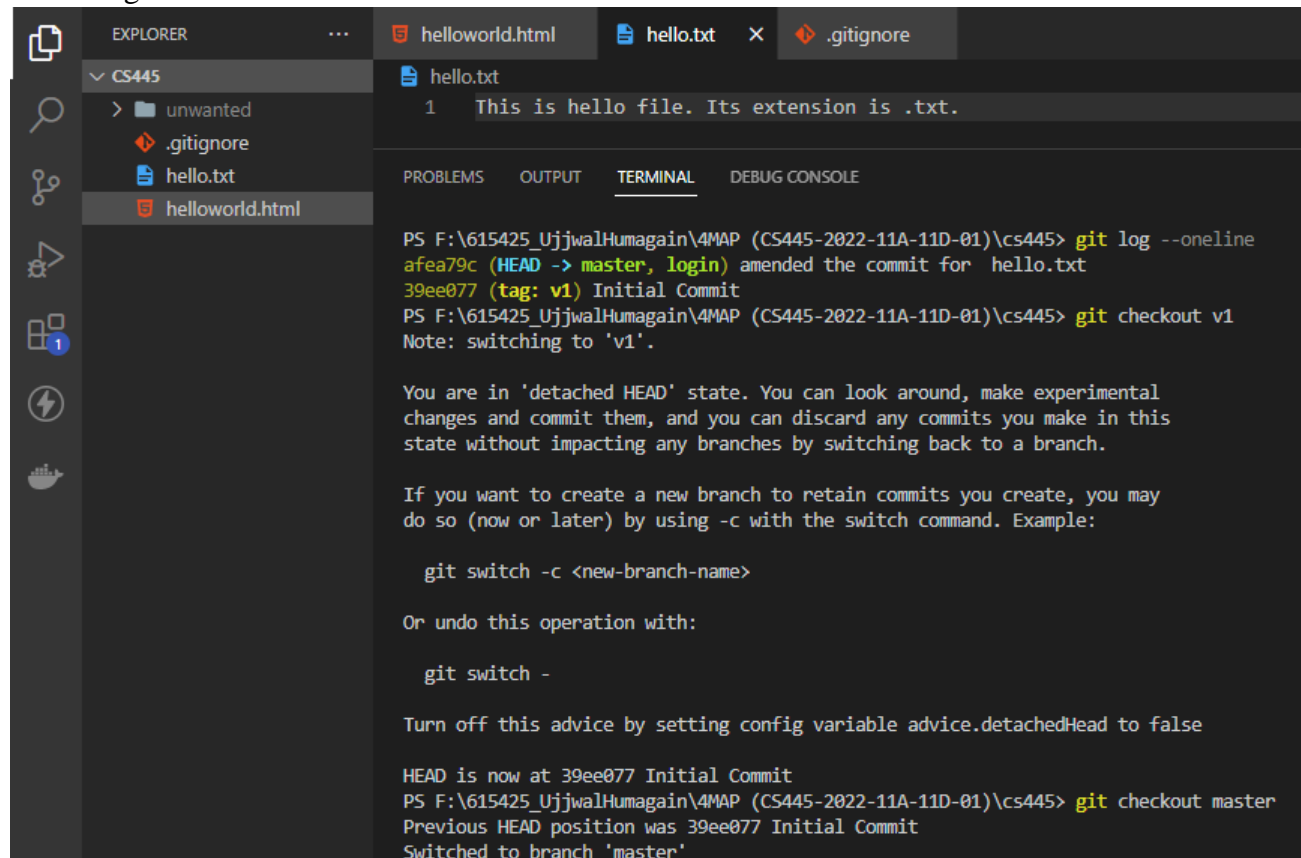
Switching branch:



The screenshot shows the VS Code interface with the Explorer sidebar on the left showing a project named 'CS445' with files 'unwanted', '.gitignore', 'hello.txt', and 'helloworld.html'. The main editor area shows 'hello.txt' with the content '1 This is hello file. Its extension is .txt.'. The bottom panel shows the TERMINAL with the following commands and output:

```
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git status
On branch master
nothing to commit, working tree clean
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git branch
login
* master
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git checkout login
Switched to branch 'login'
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git status
On branch login
nothing to commit, working tree clean
```

Switching commit:



The screenshot shows the VS Code interface with the Explorer sidebar on the left showing a project named 'CS445' with files 'unwanted', '.gitignore', 'hello.txt', and 'helloworld.html'. The main editor area shows 'hello.txt' with the content '1 This is hello file. Its extension is .txt.'. The bottom panel shows the TERMINAL with the following commands and output:

```
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git log --oneline
afea79c (HEAD -> master, login) amended the commit for hello.txt
39ee077 (tag: v1) Initial Commit
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git checkout v1
Note: switching to 'v1'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

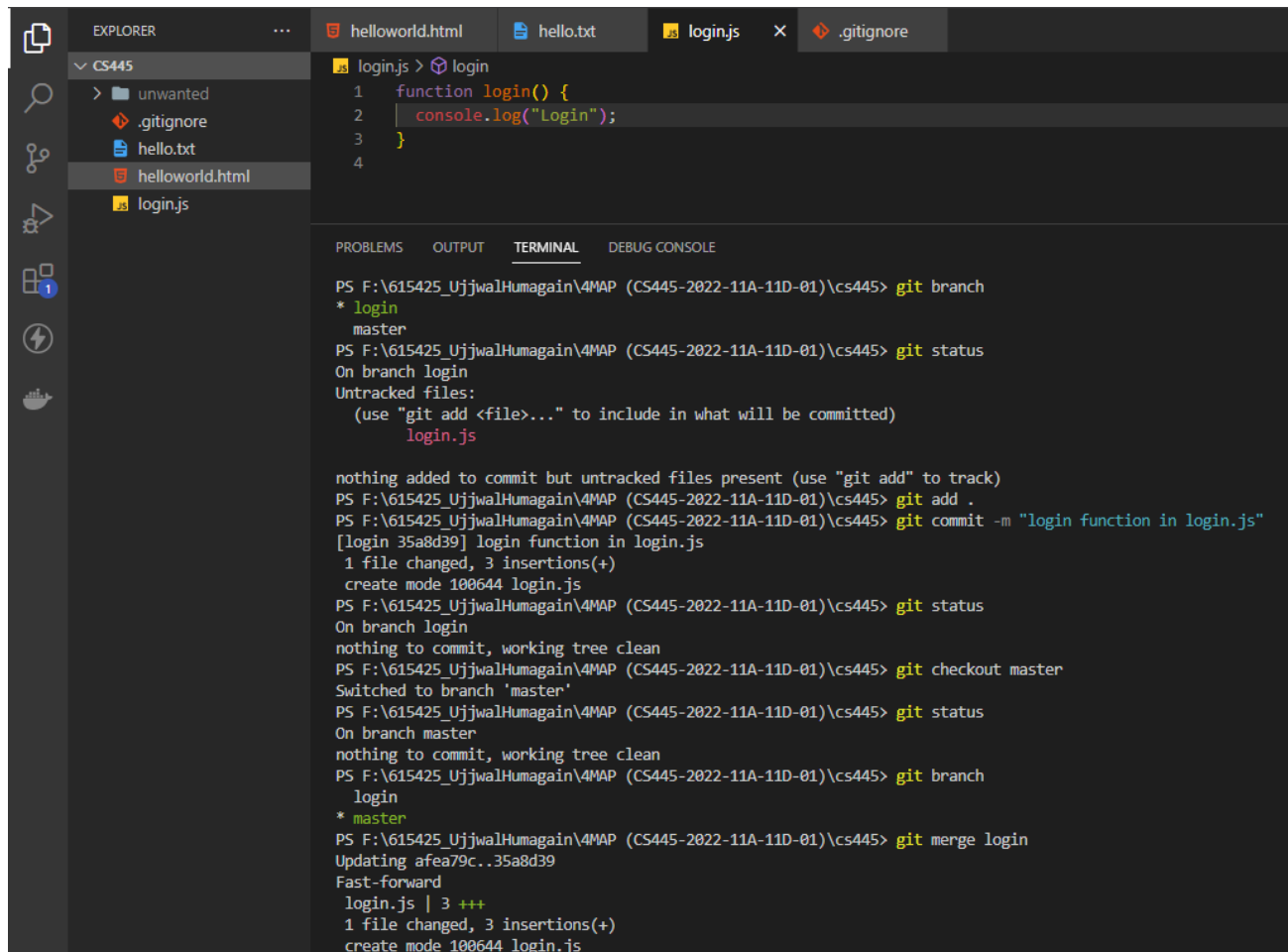
    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 39ee077 Initial Commit
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git checkout master
Previous HEAD position was 39ee077 Initial Commit
Switched to branch 'master'
```

12. git merge

This command is used to merge two branches. For example, we have two branches master and login. Here, in the login branch we added the code related to the login feature and committed it. But later we need that code in the master branch as well. For doing so at first we need to checkout to the master branch and add 'git merge login' command to merge the login branch in it as;



```
login.js > login
1  function login() {
2      console.log("Login");
3  }
4

PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE

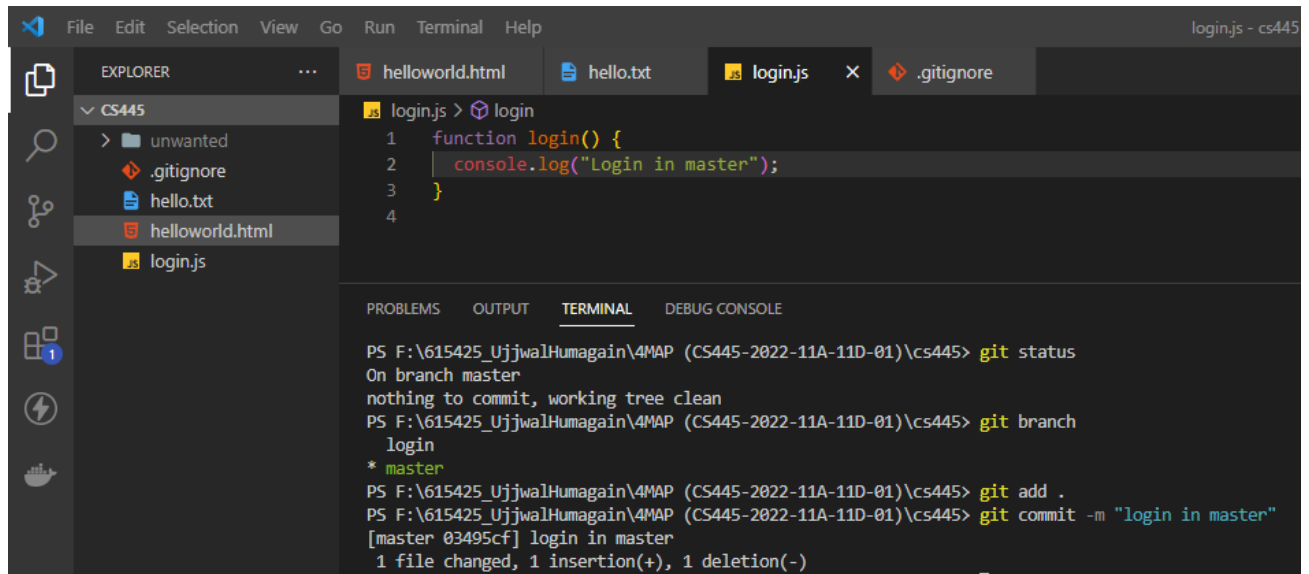
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git branch
* login
master
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git status
On branch login
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    login.js

nothing added to commit but untracked files present (use "git add" to track)
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git add .
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git commit -m "login function in login.js"
[login 35a8d39] login function in login.js
1 file changed, 3 insertions(+)
create mode 100644 login.js
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git status
On branch login
nothing to commit, working tree clean
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git checkout master
Switched to branch 'master'
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git status
On branch master
nothing to commit, working tree clean
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git branch
login
* master
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git merge login
Updating afea79c..35a8d39
Fast-forward
 login.js | 3 +++
1 file changed, 3 insertions(+)
create mode 100644 login.js
```

i. Create a conflict and solve the conflict

Sometimes during the merging of two branches conflict might occur because of the simultaneous changes done to the same file from different branches. In that case we need to fix the merging conflict manually by deciding the changes to keep and then commit the new changes and solve the conflict. Demonstration of this scenario is presented as;

a. Modified login.js in master branch



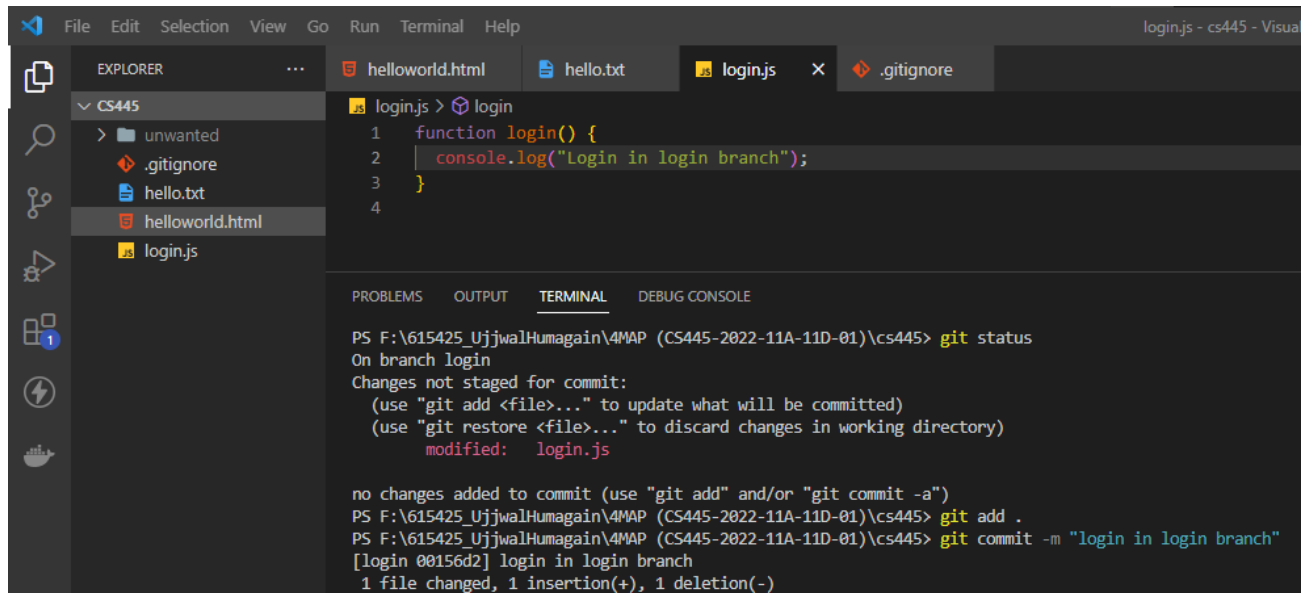
The screenshot shows the Visual Studio Code interface with the Explorer sidebar on the left displaying the file structure of a project named 'CS445'. The files listed are 'unwanted', '.gitignore', 'hello.txt', 'helloworld.html', and 'login.js'. The main editor area shows the 'login.js' file with the following code:

```
1 function login() {  
2   console.log("Login in master");  
3 }  
4
```

The TERMINAL panel at the bottom shows the following commands and output:

```
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git status  
On branch master  
nothing to commit, working tree clean  
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git branch  
login  
* master  
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git add .  
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git commit -m "login in master"  
[master 03495cf] login in master  
1 file changed, 1 insertion(+), 1 deletion(-)
```

b. Modified login.js in login branch



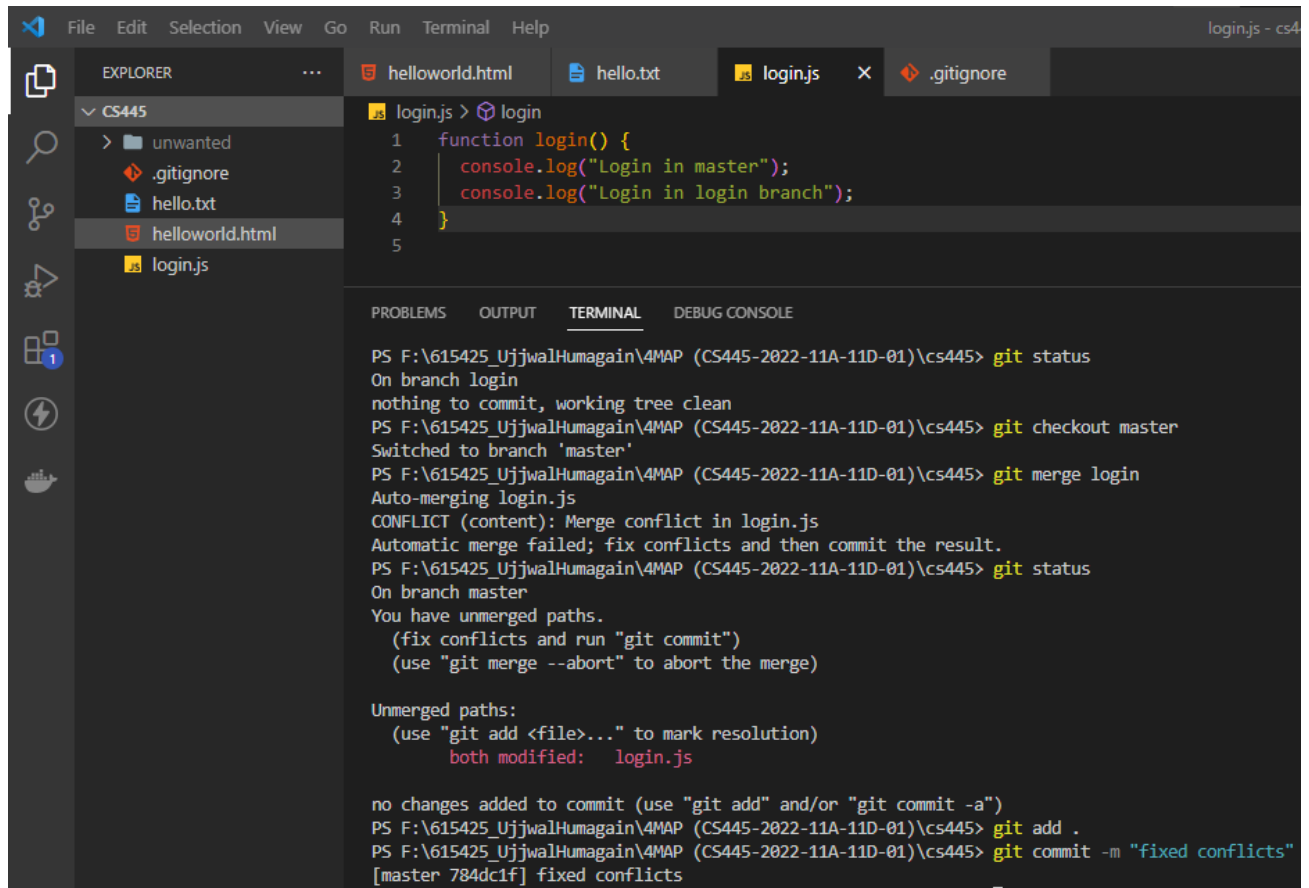
The screenshot shows the Visual Studio Code interface with the Explorer sidebar on the left displaying the file structure of a project named 'CS445'. The files listed are 'unwanted', '.gitignore', 'hello.txt', 'helloworld.html', and 'login.js'. The main editor area shows the 'login.js' file with the following code:

```
1 function login() {  
2   console.log("Login in login branch");  
3 }  
4
```

The TERMINAL panel at the bottom shows the following commands and output:

```
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git status  
On branch login  
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:   login.js  
  
no changes added to commit (use "git add" and/or "git commit -a")  
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git add .  
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git commit -m "login in login branch"  
[login 00156d2] login in login branch  
1 file changed, 1 insertion(+), 1 deletion(-)
```

c. Merging the login branch in master branch by fixing issue



```
login.js > login
1  function login() {
2      console.log("Login in master");
3      console.log("Login in login branch");
4  }
5

PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE

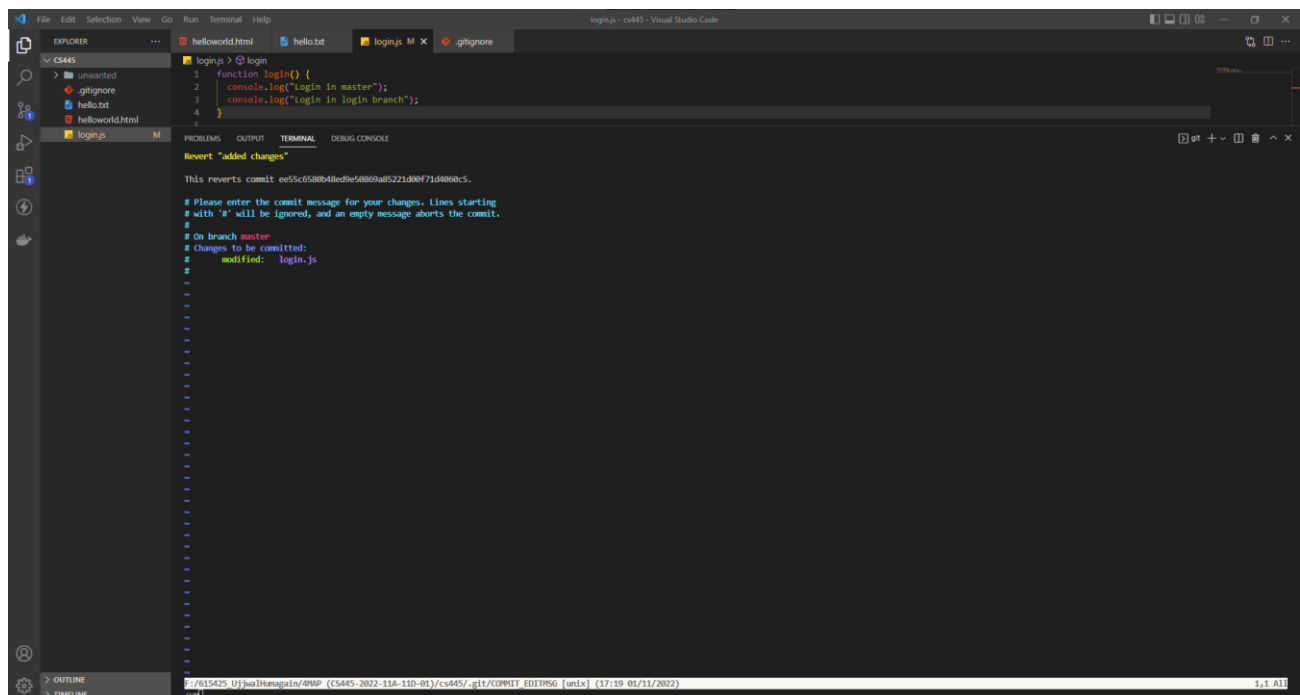
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git status
On branch login
nothing to commit, working tree clean
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git checkout master
Switched to branch 'master'
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git merge login
Auto-merging login.js
CONFLICT (content): Merge conflict in login.js
Automatic merge failed; fix conflicts and then commit the result.
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git status
On branch master
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)
    both modified:   login.js

no changes added to commit (use "git add" and/or "git commit -a")
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git add .
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git commit -m "fixed conflicts"
[master 784dc1f] fixed conflicts
```

13. git revert

We can also revert the changes that have been committed to a local repository. For this we need to specify the HEAD or the specific commit hash and the commit will be reverted as;



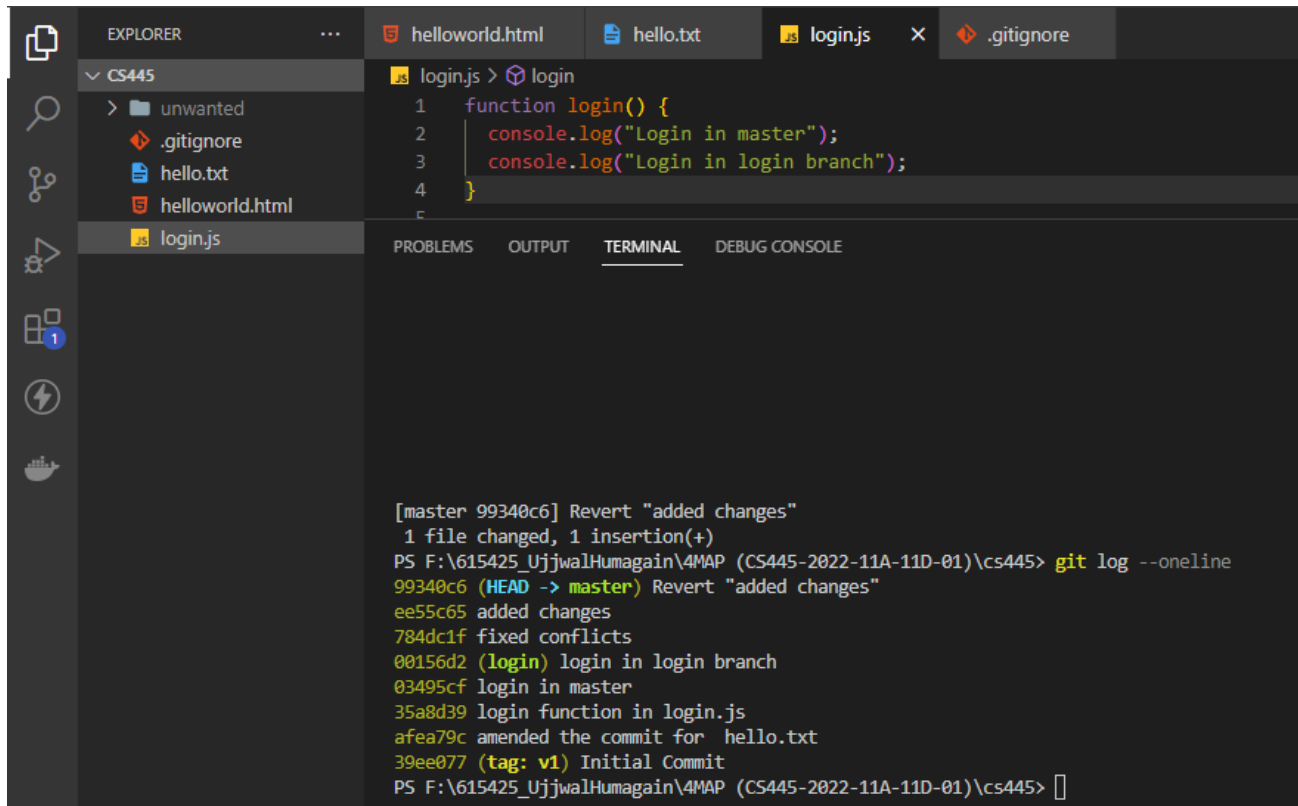
```
login.js > login
1  function login() {
2      console.log("Login in master");
3      console.log("Login in login branch");
4  }
5

PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE

Revert "added changes"

This reverts commit ee55c6588b48ed9e58869a8521d00f71d4060c5.

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
#
# changes to be committed:
#   modified:   login.js
#
#
```



```
login.js > login
1 function login() {
2   console.log("Login in master");
3   console.log("Login in login branch");
4 }

[master 99340c6] Revert "added changes"
1 file changed, 1 insertion(+)
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git log --oneline
99340c6 (HEAD -> master) Revert "added changes"
ee55c65 added changes
784dc1f fixed conflicts
00156d2 (login) login in login branch
03495cf login in master
35a8d39 login function in login.js
afea79c amended the commit for hello.txt
39ee077 (tag: v1) Initial Commit
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445>
```

Reverting will create a new commit and contains only those changes which were present before the reverted commit.

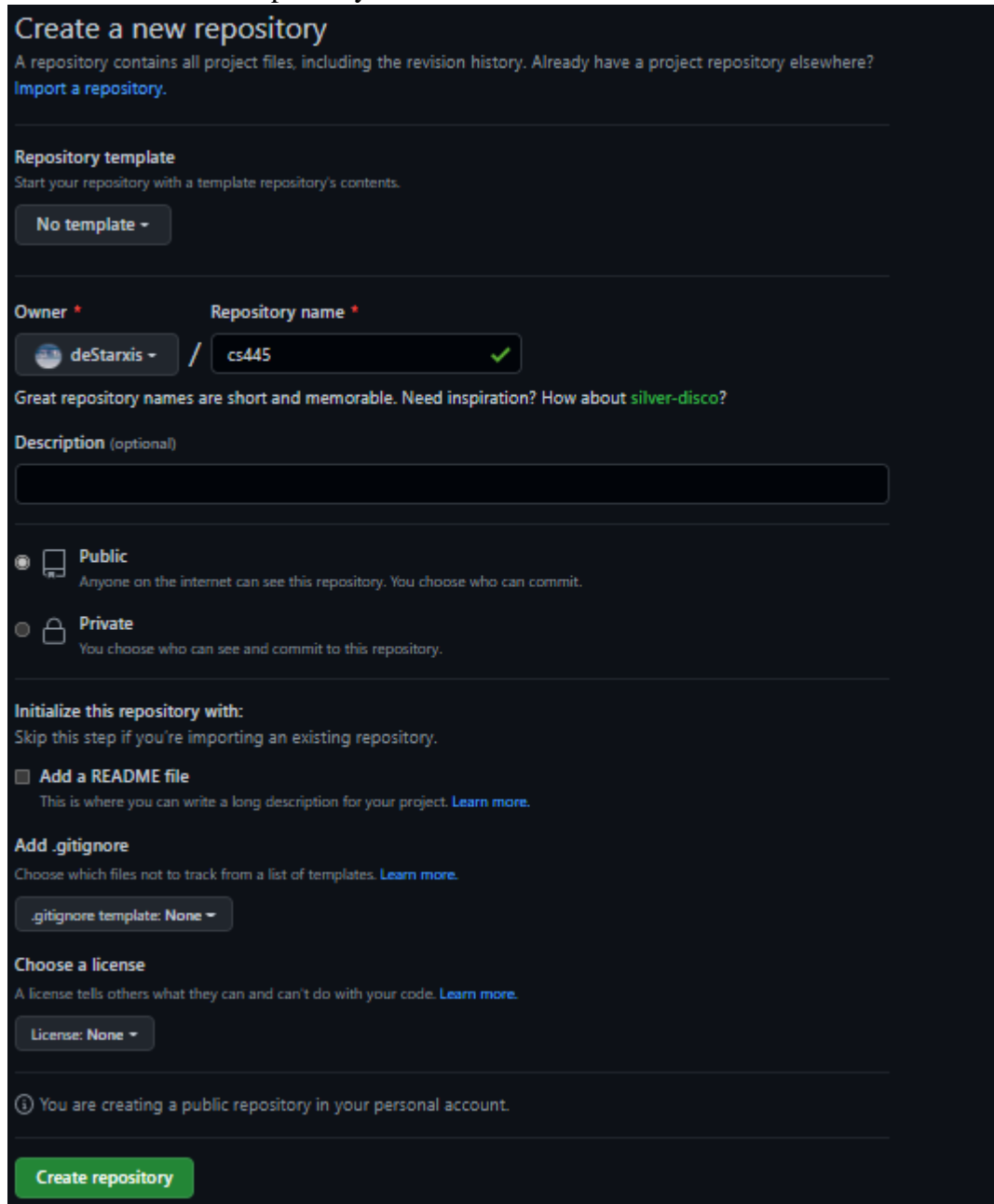
14. git reset

Similar to that of the revert we can also reset the commit history of the repository to the desired one by specifying the specific commit via 'git reset' command. Instead of creating a new commit like that of the 'git revert', the command resets the current branch to the specified commit and deletes the newer commits after that.

```
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git log --oneline
99340c6 (HEAD -> master) Revert "added changes"
ee55c65 added changes
784dc1f fixed conflicts
00156d2 (login) login in login branch
03495cf login in master
35a8d39 login function in login.js
afea79c amended the commit for hello.txt
39ee077 (tag: v1) Initial Commit
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git reset --hard ee55c65
HEAD is now at ee55c65 added changes
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445> git log --oneline
ee55c65 (HEAD -> master) added changes
784dc1f fixed conflicts
00156d2 (login) login in login branch
03495cf login in master
35a8d39 login function in login.js
afea79c amended the commit for hello.txt
39ee077 (tag: v1) Initial Commit
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\cs445>
```

Practice with all the remote repository commands:

1. Create a remote repository



Create a new repository
A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Repository template
Start your repository with a template repository's contents.
No template -

Owner * **Repository name ***
deStarxis / cs445 ✓

Great repository names are short and memorable. Need inspiration? How about [silver-disco?](#)

Description (optional)

☒ **Public**
Anyone on the internet can see this repository. You choose who can commit.

☐ **Private**
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☐ **Add a README file**
This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore
Choose which files not to track from a list of templates. [Learn more.](#)
.gitignore template: None -

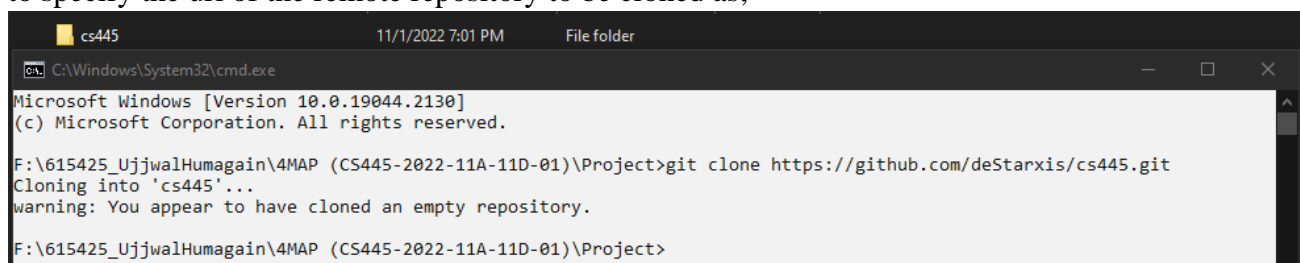
Choose a license
A license tells others what they can and can't do with your code. [Learn more.](#)
License: None -

You are creating a public repository in your personal account.

Create repository

2. git clone

This command is used to clone the remote repository in the local machine. In this command we have to specify the url of the remote repository to be cloned as;



```
cs445 11/1/2022 7:01 PM File folder
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19044.2130]
(c) Microsoft Corporation. All rights reserved.

F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project>git clone https://github.com/deStarxis/cs445.git
Cloning into 'cs445'...
warning: You appear to have cloned an empty repository.

F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project>
```

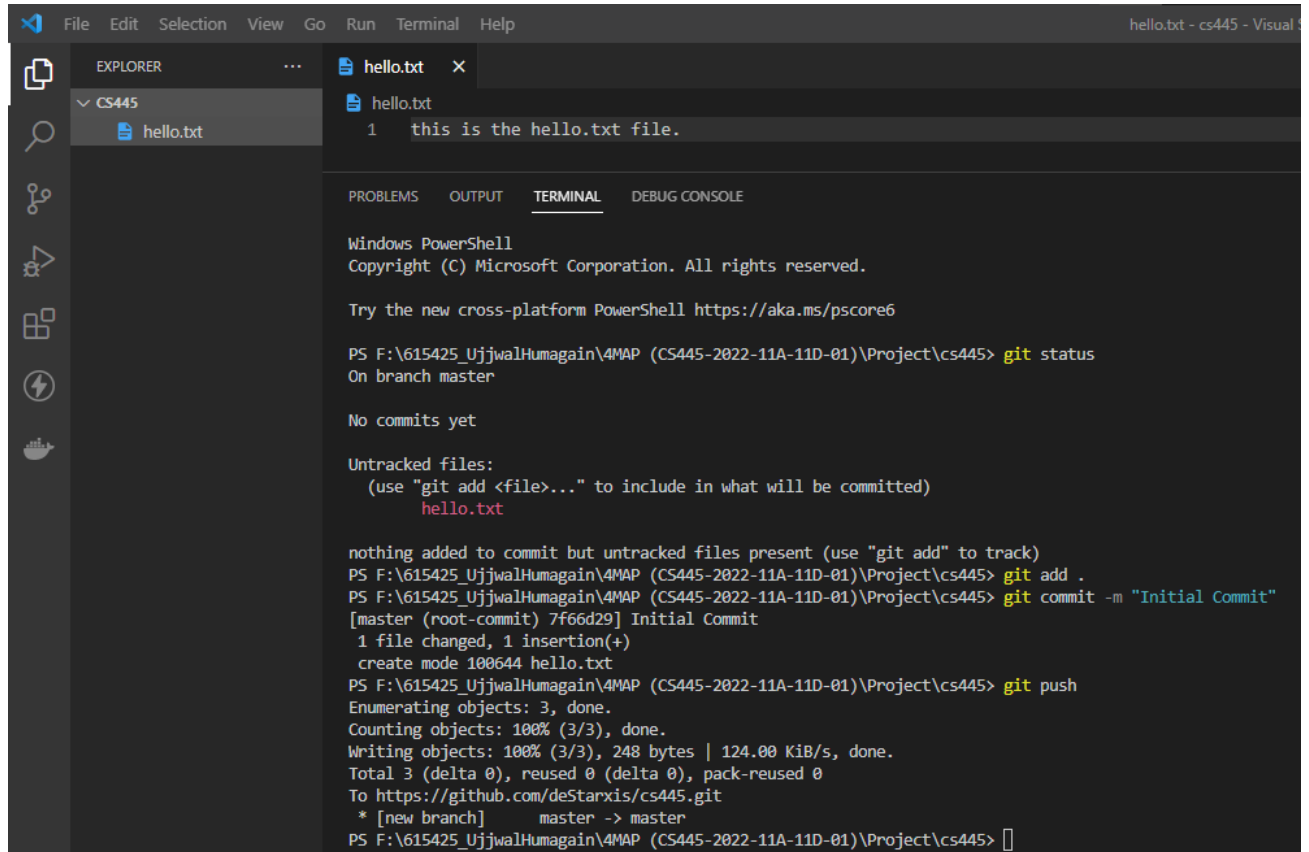
3. git remote

This command lists the current configured remote repository for the local repository.

```
F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445>git remote origin
```

4. git push

This command is used to add commits of the local repository to the remote repository.



The screenshot shows the Visual Studio Code interface with a terminal window open. The Explorer pane on the left shows a file named 'hello.txt' under a folder named 'CS445'. The main editor area shows the content of 'hello.txt' as '1 this is the hello.txt file.' The terminal window displays the output of several git commands:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445> git status
On branch master

No commits yet

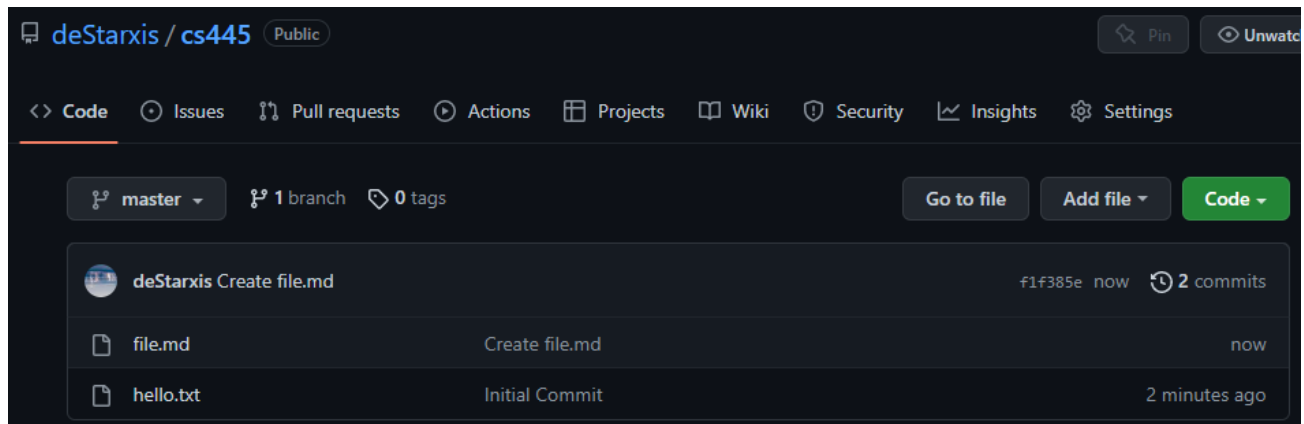
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    hello.txt

nothing added to commit but untracked files present (use "git add" to track)
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445> git add .
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445> git commit -m "Initial Commit"
[master (root-commit) 7f66d29] Initial Commit
 1 file changed, 1 insertion(+)
 create mode 100644 hello.txt
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445> git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 248 bytes | 124.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/deStarxis/cs445.git
 * [new branch]      master -> master
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445>
```

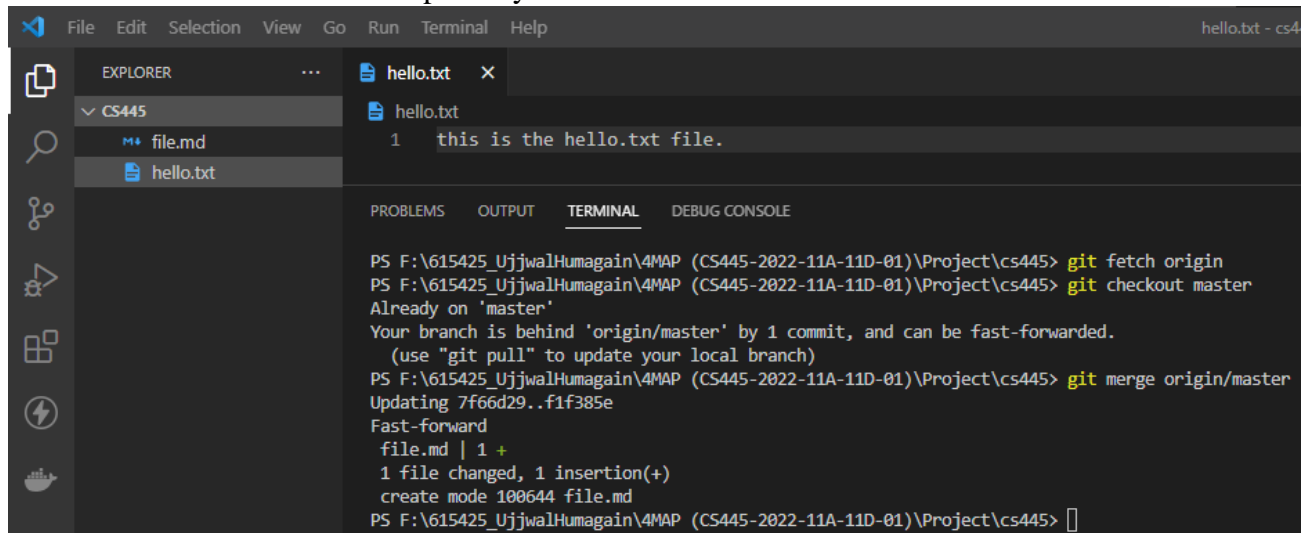
5. git fetch

This command is used to fetch the branches and their respective commits from the remote repository as;

Added new file to the remote repository which has to be fetched in the local repository using 'git fetch'.



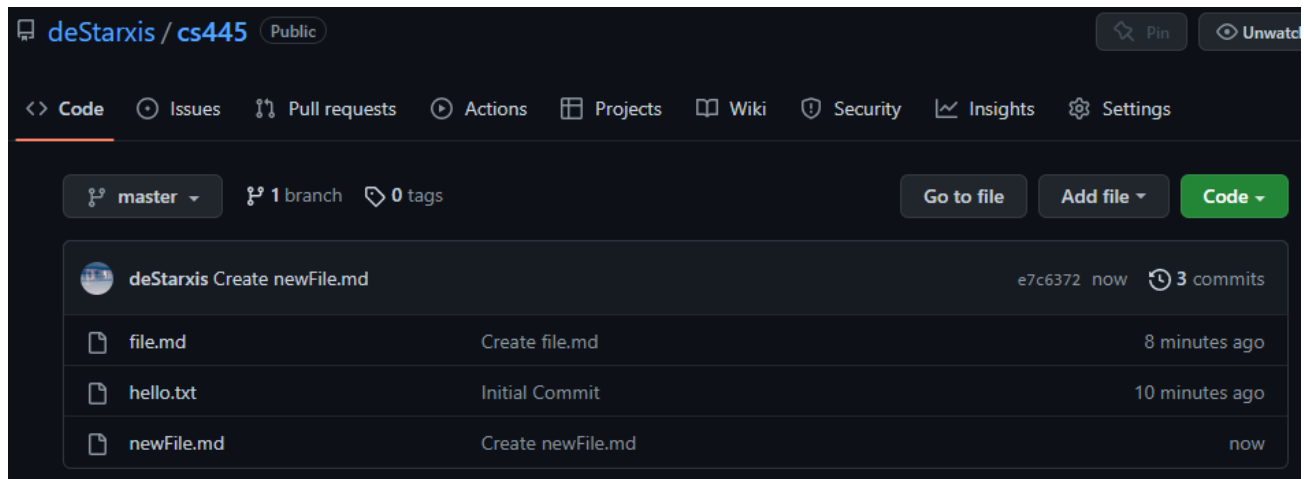
At first we need to fetch the changes from the remote and then we need to merge those changes to the default branch of the local repository.



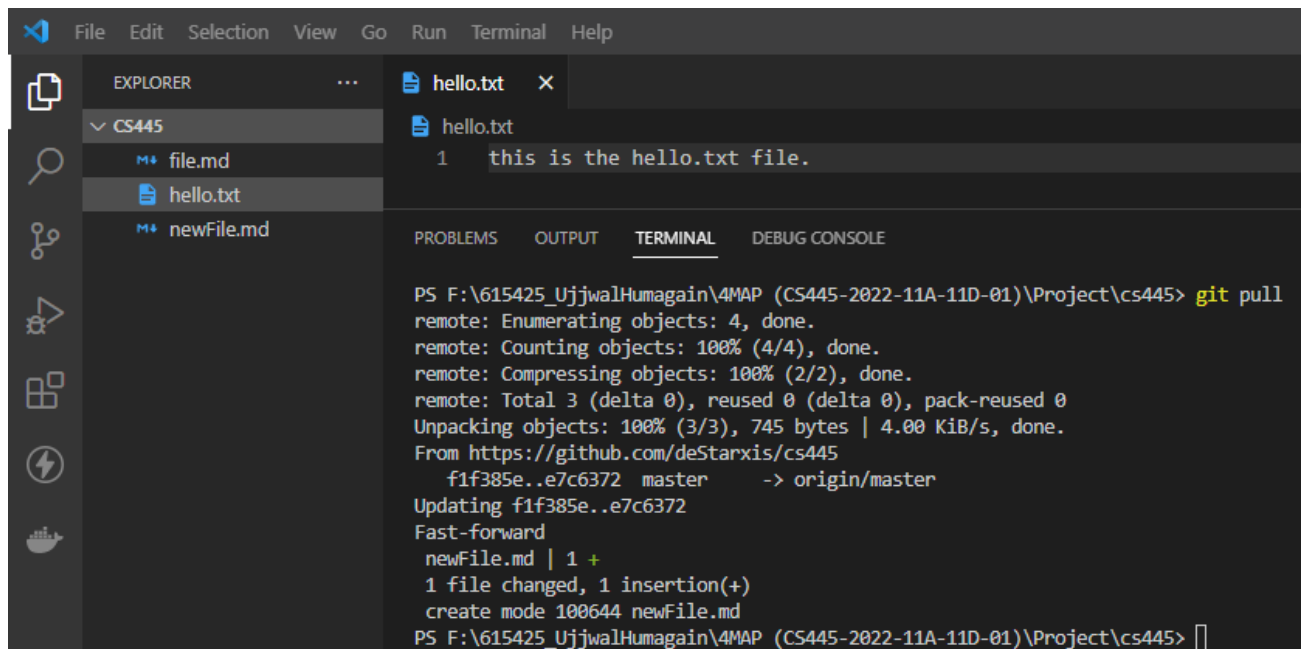
6. git pull

Despite using 'git fetch' and 'git merge' commands to get changes from remote repository to the local, we directly use the 'git pull' command.

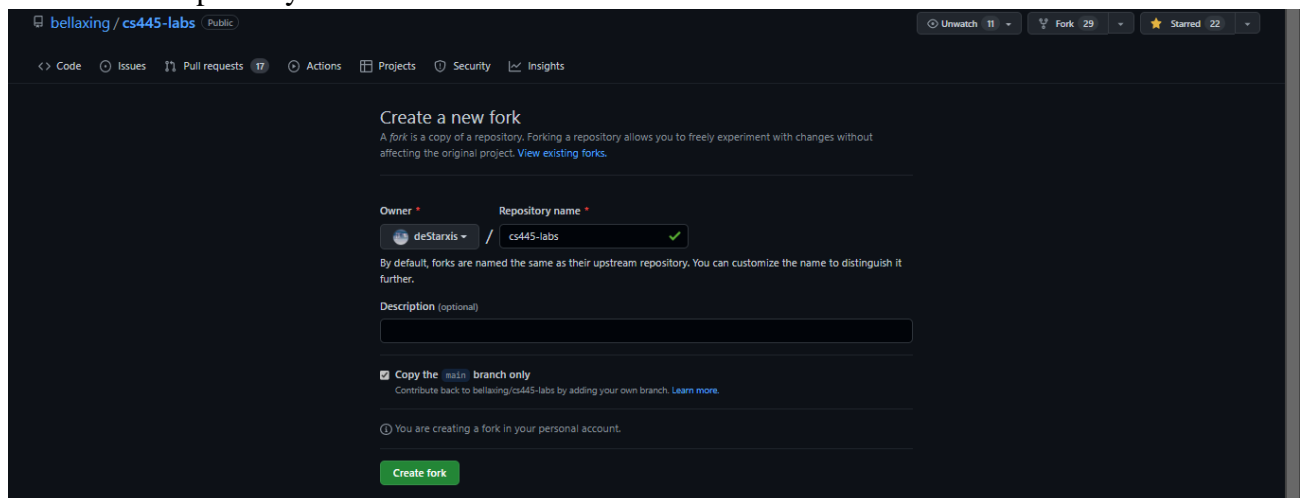
Added another file to the remote repository.



Using 'git pull' command to pull everything from remote repository to local.



1. Star the repository



4. Clone your forked repository into your local machine

```
cs445-labs 11/1/2022 7:29 PM File folder
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19044.2130]
(c) Microsoft Corporation. All rights reserved.

F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project>git clone https://github.com/deStarxis/cs445-labs.git
Cloning into 'cs445-labs'...
remote: Enumerating objects: 231, done.
remote: Counting objects: 100% (73/73), done.
remote: Compressing objects: 100% (40/40), done.
remote: Total 231 (delta 40), reused 26 (delta 25), pack-reused 158
Receiving objects: 100% (231/231), 108.18 KiB | 1.54 MiB/s, done.
Resolving deltas: 100% (84/84), done.

F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project>
```

5. Create a new topic branch using studentId-studentName format, For example: 610001-JohnSmith

```
File Edit Selection View Go Run Terminal Help README.md - cs445-labs - Visual Studio Code

EXPLORER
CS445-LABS
  Lab1
  Lab2
  Lab3
  Lab4
  Lab5
  Lab6
  Lab7
  Lab8
  Lab9
  Lab10
  .gitignore
  README.md

README.md x
README.md > # cs445-labs
1 # cs445-labs
2
3 1. [Lab 1 - Git](Lab1/lab1.md)
4 2. [Lab 2 - TypeScript](Lab3/lab3.md)
5 3. [Lab 3 - Design Pattern 1](Lab9/lab9.md)
6 4. [Lab 4 - Design Pattern 2](Lab10/lab10.md)
7 5. [Lab 5 - Modern Web Browser](Lab2/lab2.md)
8 6. [Lab 6 - JS DOM](Lab4/lab4.md)
9 8. [Lab 7 - Intro to Asynchronous programming & Promise](Lab5/lab5.md)
10 9. [Lab 8 - async&await](Lab6/lab6.md)
11 10. [Lab 9 - fetch](Lab7/lab7.md)
12 11. [Lab 10 - RXJS](Lab8/lab8.md)
13

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs> git branch 615425-UjjwalHumagain
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs>
```

6. Checkout your new branch

```
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs> git checkout 615425-UjjwalHumagain
Switched to branch '615425-UjjwalHumagain'
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs> git status
On branch 615425-UjjwalHumagain
nothing to commit, working tree clean
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs>
```

7. Add a new file with your name, example: hello.js

```
EXPLORER
CS445-LABS
  Lab1
  Lab2
  Lab3
  Lab4
  Lab5
  Lab6
  Lab7
  Lab8
  Lab9
  Lab10
  .gitignore
  README.md
  UjjwalHumagain.js A

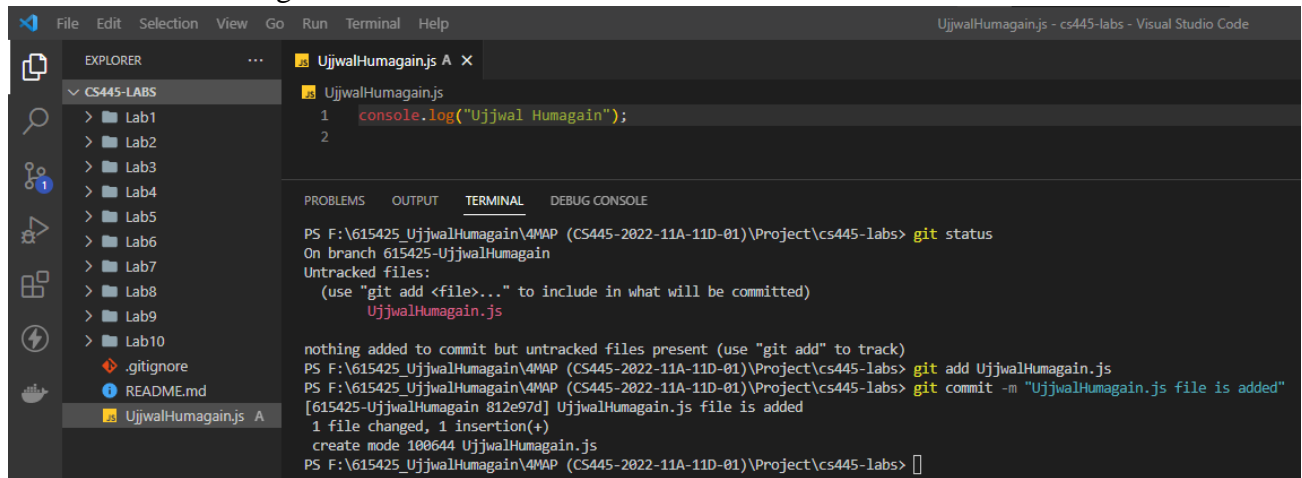
UjjwalHumagain.js A x
UjjwalHumagain.js
1 console.log("Ujjwal Humagain");
2

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs> git status
On branch 615425-UjjwalHumagain
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    UjjwalHumagain.js

nothing added to commit but untracked files present (use "git add" to track)
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs> git add UjjwalHumagain.js
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs>
```

8. Commit the changes



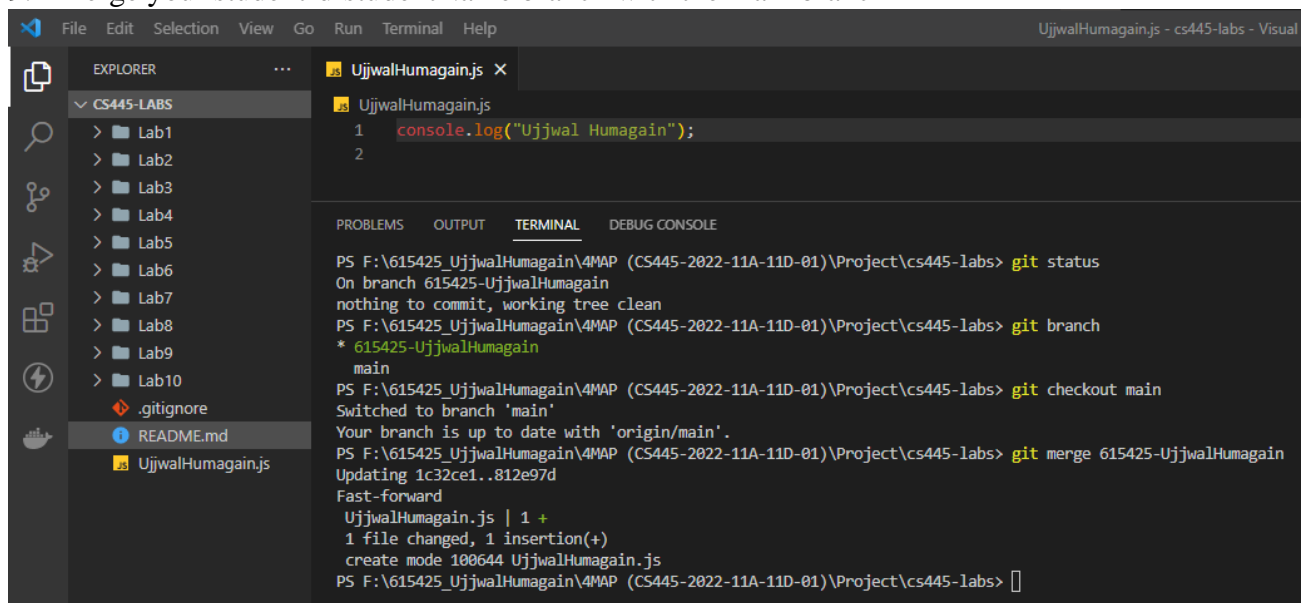
```
UjjwalHumagain.js A X
UjjwalHumagain.js
1 console.log("Ujjwal Humagain");
2

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs> git status
On branch 615425-UjjwalHumagain
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    UjjwalHumagain.js

nothing added to commit but untracked files present (use "git add" to track)
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs> git add UjjwalHumagain.js
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs> git commit -m "UjjwalHumagain.js file is added"
[615425-UjjwalHumagain 812e97d] UjjwalHumagain.js file is added
1 file changed, 1 insertion(+)
 create mode 100644 UjjwalHumagain.js
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs> 
```

9. Merge your studentId-studentName branch with the main branch

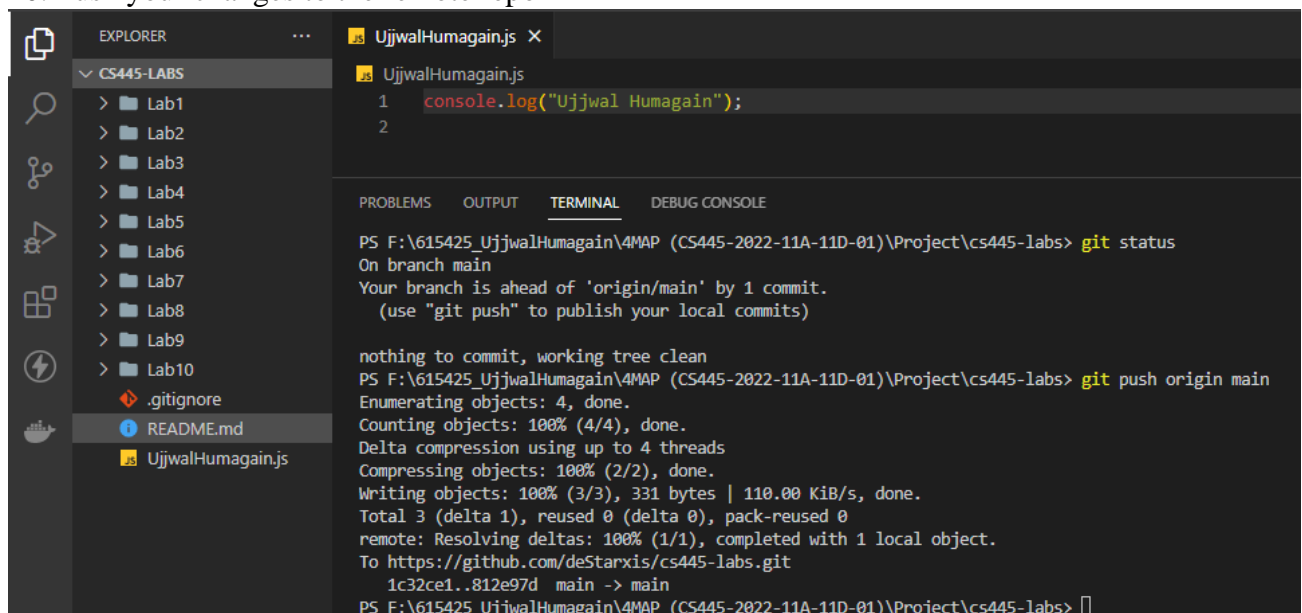


```
UjjwalHumagain.js X
UjjwalHumagain.js
1 console.log("Ujjwal Humagain");
2

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs> git status
On branch 615425-UjjwalHumagain
nothing to commit, working tree clean
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs> git branch
* 615425-UjjwalHumagain
main
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs> git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs> git merge 615425-UjjwalHumagain
Updating 1c32ce1..812e97d
Fast-forward
 UjjwalHumagain.js | 1 +
1 file changed, 1 insertion(+)
 create mode 100644 UjjwalHumagain.js
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs> 
```

10. Push your changes to the remote repo



```
UjjwalHumagain.js X
UjjwalHumagain.js
1 console.log("Ujjwal Humagain");
2

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs> git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs> git push origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 331 bytes | 110.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/deStarxis/cs445-labs.git
 1c32ce1..812e97d main -> main
PS F:\615425_UjjwalHumagain\4MAP (CS445-2022-11A-11D-01)\Project\cs445-labs> 
```

11. Create a new Pull Request to the upstream repo

This screenshot shows the 'Comparing changes' interface on GitHub for the repository `bellaxing/cs445-labs`. The base repository is `bellaxing/cs445-labs` with the `base: main` branch. The head repository is `deStarxis/cs445-labs` with the `compare: main` branch. A green checkmark indicates 'Able to merge. These branches can be automatically merged.' Below this, there is a 'Create pull request' button. The interface also shows '1 commit' and '1 file changed' by 1 contributor. A commit from `deStarxis` is listed: 'UjjwalHumagain.js file is added', committed 5 minutes ago. The diff shows a single file change: `UjjwalHumagain.js`, with a diff of `@@ -0,0 +1 @@` and the addition of `console.log("Ujjwal Humagain");`.

This screenshot shows the 'Open a pull request' interface on GitHub for the repository `deStarxis/cs445-labs`. The base repository is `bellaxing/cs445-labs` with the `base: main` branch. The head repository is `deStarxis/cs445-labs` with the `compare: main` branch. A green checkmark indicates 'Able to merge. These branches can be automatically merged.' Below this, there is a 'Create pull request' button. The interface also shows '1 commit' and '1 file changed' by 1 contributor. A commit from `deStarxis` is listed: 'UjjwalHumagain.js file is added', committed 6 minutes ago. The diff shows a single file change: `UjjwalHumagain.js`, with a diff of `@@ -0,0 +1 @@` and the addition of `console.log("Ujjwal Humagain");`. The page also includes a 'Write' section with a 'Preview' tab and a 'Create pull request' button.

The link to the forked repository is:
[deStarxis/cs445-labs \(github.com\)](https://github.com/deStarxis/cs445-labs)