Install GIT & make sure it is added into PATH.

Section 0 -Use GIT as local VCS. Steps to follow:

1. Create a directory ‘project\_dir’ & cd to ‘project\_dir’.

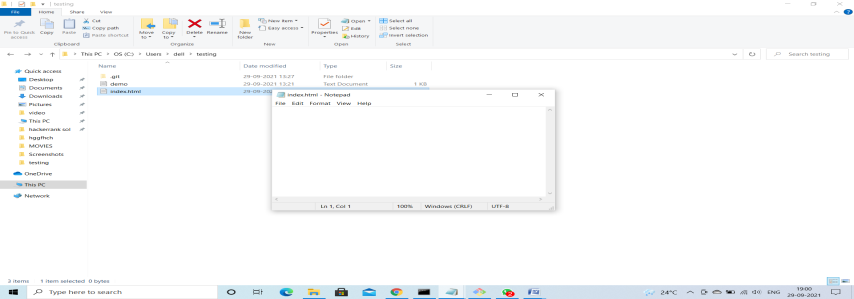
Ans . $ mkdir project\_dir & $ cd project\_dir

2.Initialize git version database.(git init)

Ans. $ git init

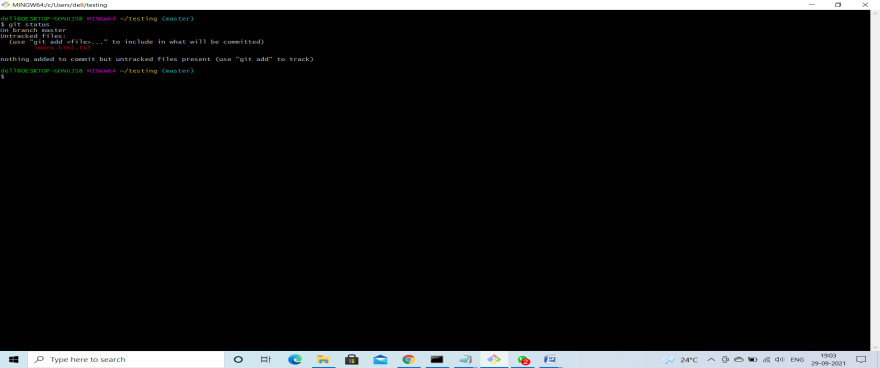
1. Create a new file index.html.

Ans.



1. Check the git status. You should find index.html as untracked file.

Ans. $ git status



1. Stage the index.html file

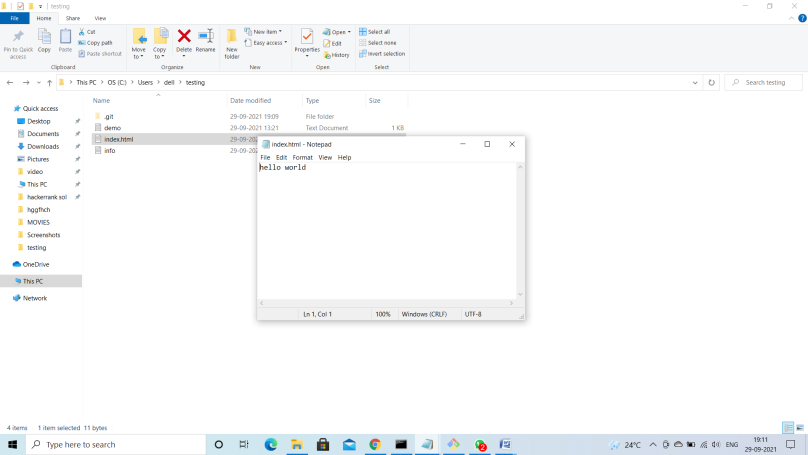
Ans. $ git add index.html

1. Commit index.html

Ans. $ git commit –m “committing index.html1”

1. Make few changes in index.html & create a new file info.txt file.

Ans.



1. Check git status. You should find index.html &info.txt as untracked files.

Ans. $ git rm --cached index.html

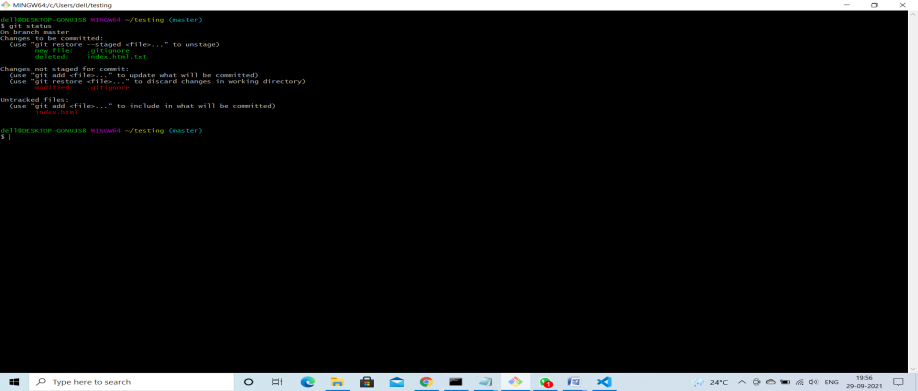
$ git status

1. Configure GIT to ignore all txt files.

Ans. $ touch .gitignore

1. Again check the git status. You should find only index.html as untracked file.

Ans. $ git status



1. State & commit index.html

Ans. $ git add index.html

$ git commit –m “committing index.html”

1. Log all your comments so far.

14Revert the change made in the previous step using git command

Ans. $ git revert HEAD

15. Stage index.html

Ans. $ git add index.html

16.Revert back the last stage.

Ans. $ git revert HEAD

17.Rename ‘add’ command to ‘my-add’.

Ans. $git config --global alias.my-add add

18.Using my\_add command Stage index.html again & commit the changes.

Ans. $ git my-add index.html

$ git commit –m “changed”

1. Revert the last commit.

Ans. $ git revert HEAD

*GIT Branching*

Objective: Commit HTML, CSS & JavaScript assignments into GIT.

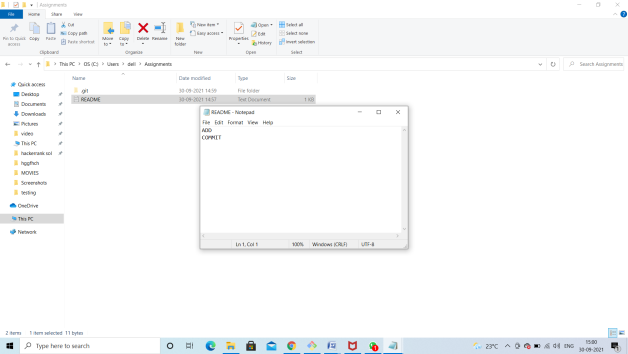
SECTION-1 (HTML assignments) - Steps to follow:

First take a backup of your assignments & projects. This is required because due to incorrect GIT operation you may lose your files.

Create an empty directory ‘Assignments’ & cd to ‘Assignments’.

1. Create a file README.txt inside ‘Assignments’ & write few lines about the contents of ‘Assignments’ folder.

Ans.



1. Commit README.txt file.

Ans.$ git commit –m “commiting README file”

1. Now create a new branch ‘html-assignments’.

Ans.$ git branch html-assignments

1. Switch to ‘html-assignments’ branch.

Ans. $ git checkout html-assignments

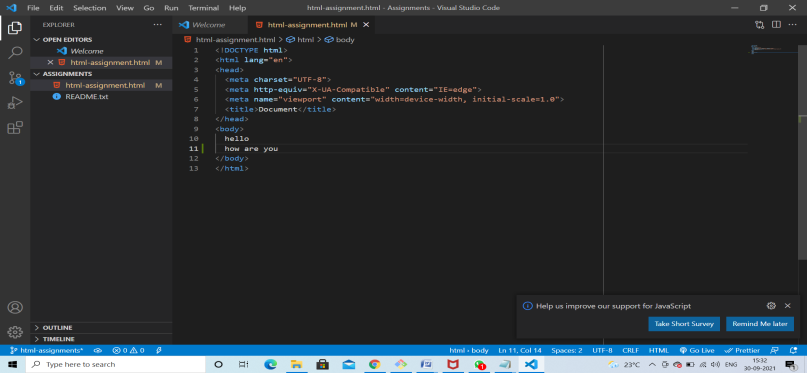
1. Copy all HTML assignments inside ‘Assignments’ folder.

Ans. Create a html assignment in assignment folder

1. Commit HTML assignments into ‘html-assignments’ branch.

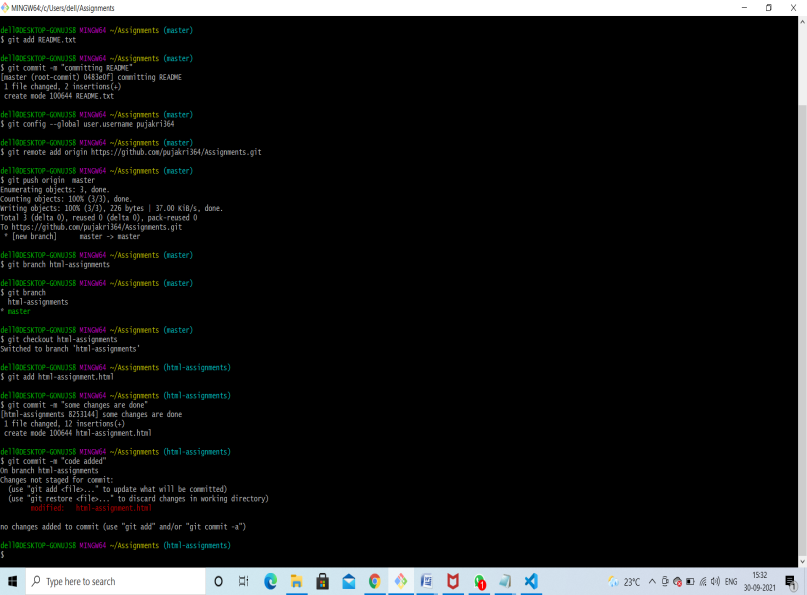
Ans. $ commit –m “some changes are made”

1. Make minor changes into few files belonging to ‘html-assignments’ branch.



1. Commit those changed files.

Ans. $ git commit –m “code added”



1. Switch to master branch.

Ans. $ git checkout master

1. Make minor changes into README.txt file & commit those changes into master.

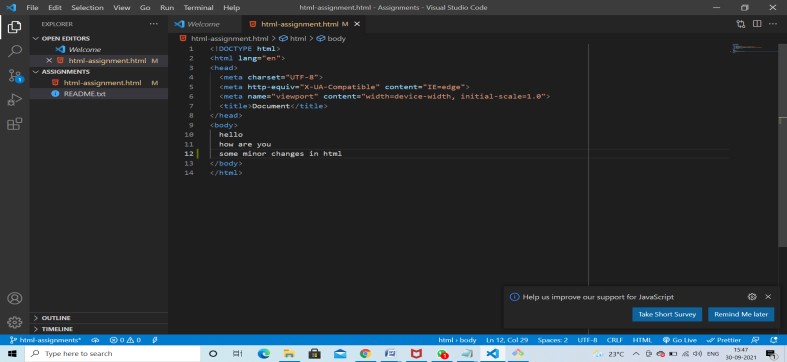
Ans.$ git commit –m “minor changes”

1. Again switch to ‘html-assignments’ branch.

Ans.$ git checkout html-assignments

1. Make minor changes into few files belonging to ‘html-assignments’ branch.

Ans.



1. Commit those changes.

Ans. $ git commit –m “some minor changes in html”

1. Switch to master.

Ans.$ git checkout master

1. Merge ‘html-assignments’ branch into master. Confirm all html assignments are shown in master.

Ans. $ git merge html-assignments

1. Finally delete the ‘html-assignments’ branch.

Ans. $ git branch –d html-assignments

SECTION-2 - (CSS assignments) Steps to follow:

1. Create a new branch ‘css-assignments’.

Ans.$ git branch css-assignments

1. Switch to ‘css-assignments’ branch.

Ans.$ git checkout css-assignments

1. Copy all CSS assignments inside ‘Assignments’ folder.

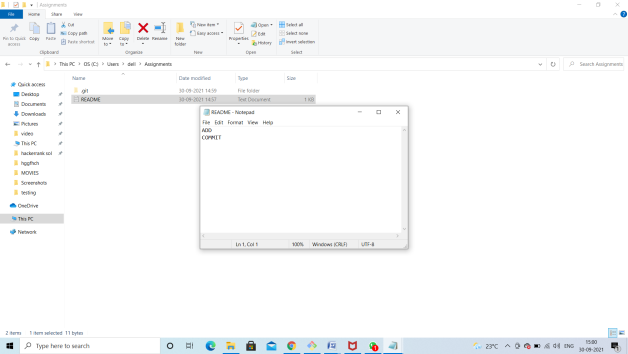
Ans. Create a css assignments in assignment folder

1. Commit CSS assignments into ‘css-assignments’ branch.

Ans. $ git commit –m “css”

1. Make minor changes into README.txt file on line 1 belonging to ‘css-assignments’ branch.

Ans.



1. Commit those changed files.

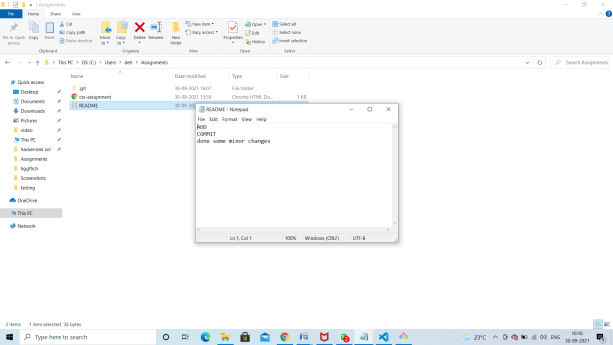
Ans. $ git commit –m “some changes made in readme.txt”

1. Switch to master branch.

Ans. $ git checkout master

1. Make minor changes into README.txt file on line 3 & commit those changes into master.

Ans.



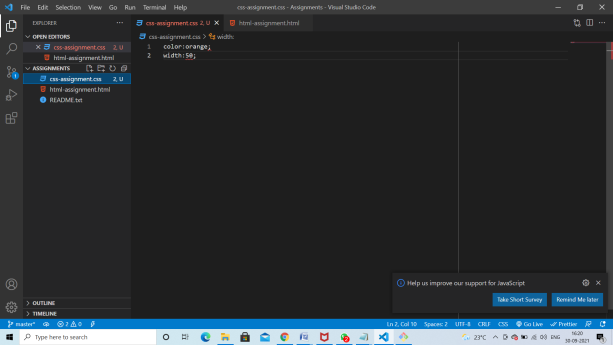
$ git commit –m “committing changes”

1. Again switch to ‘css-assignments’ branch.

Ans. $ git checkout css-assignments

1. Make minor changes into few files belonging to ‘css-assignments’ branch.

Ans.



1. Commit those changes.

Ans. $ git commit –m “changes in css assignment”

1. Switch to master.

Ans. $ git checkout master

1. Merge ‘css-assignments’ branch into master. Confirm all css assignments are shown in master.

Ans. $ git merge css-assignments

1. Finally delete the ‘css-assignments’ branch.

Ans. $ git branch –d css –assignments

SECTION-3 - (JavaScript assignments) Steps to follow:

1. Create a new branch ‘js-assignments’.

Ans. $ git branch js-assignments

1. Switch to ‘js-assignments’ branch.

Ans.$ git checkout js-assignments

1. Copy all JavaScript assignments inside ‘Assignments’ folder.

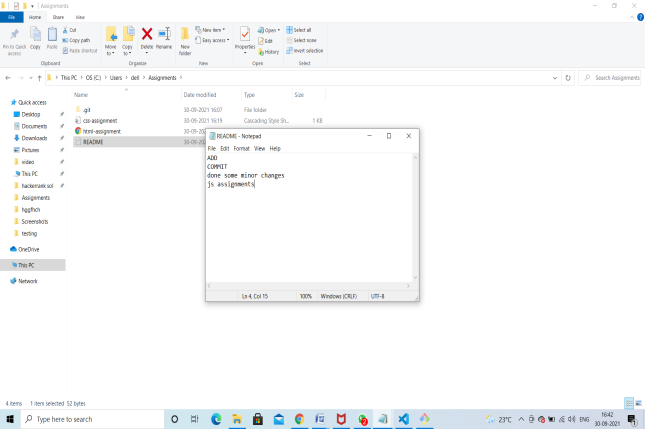
Ans. .Create a js-assignments in assignment folder

1. Commit JavaScript assignments into ‘js-assignments’ branch.

Ans.$ git commit –m “js-assignments”

1. Make minor changes into README.txt file on line 1 belonging to ‘js-assignments’ branch.

Ans.



1. Commit those changed files.

Ans. $ git commit –m “changes in readme.txt”

1. Switch to master branch.

Ans.$ git checkout master

1. Make minor changes into README.txt file on line 1 & commit those changes into master.

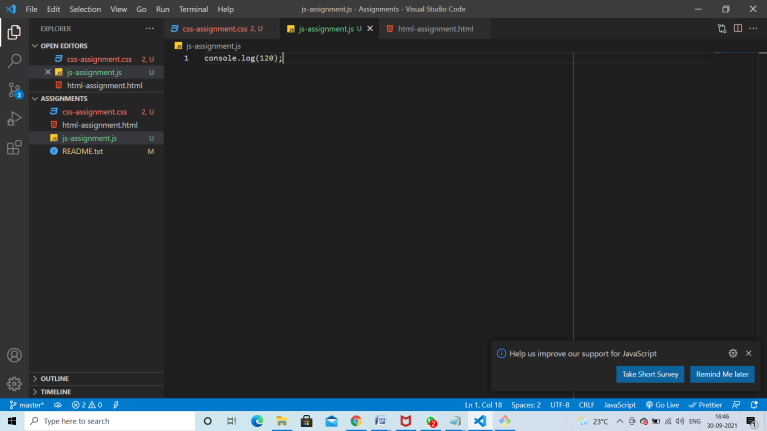
Ans. $ git commit –m “changes in line 1”

1. Again switch to ‘js-assignments’ branch.

Ans. $ git checkout js-assignments

1. Make minor changes into few files belonging to ‘js-assignments’ branch.

Ans.



1. Commit those changes.

Ans. $ git commit –m “minor changes in js”

1. Switch to master.

Ans . $ git checkout master

1. Merge ‘js-assignments’ branch into master. Confirm all JavaScript assignments are shown in master.

Ans. $ git merge js-assignment

1. Finally delete the ‘js-assignments’ branch.

Ans. . $ git branch –d js–assignments