TOPIC: GIT AND GIT HUB ASSIGNMENT

MALAVIKKA SOKKALINGAM

Install GIT & make sure it is added into PATH.
Use GIT as local VCS. Steps to follow:
1. Create a directory 'project_dir' & cd to 'project_dir'.
2. Initialize git version database. (git init)
3. Create a new file index.html.
4. Check the git status. You should find index.html as untracked file.
5. Stage the index.html file.
6. Commit index.html
7. Make few changes in index.html & create a new file info.txt file.
8. Check git status. You should find index.html & info.txt as untracked files.
9. Configure GIT to ignore all txt files.
10. Again check the git status. You should find only index.html as untracked file.
11. State & commit index.html
12. Log all your comments so far.
13. Make some changes in index.html.
14. Revert the change made in the previous step using git command.
15. Again change index.html.
16. Stage index.html
17. Revert back the last stage.
18. Rename 'add' command to 'my-add'.
19. Using my_add command Stage index.html again & commit the changes.
20. Revert the last commit.

```
MINGW64:/c/Users/Admin/project_dir

Admin@DESKTOP-S2142DE MINGW64 ~
$ mkdir project_dir

Admin@DESKTOP-S2142DE MINGW64 ~
$ cd project_dir

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir
$ git init
Initialized empty Git repository in C:/Users/Admin/project_dir/.git/

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ touch index.html

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git status
On branch master

No commits yet

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

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

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git add index.html

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git status
On branch master

No commits yet

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

```
Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git commit -m "committing index file"
Author identity unknown

*** Please tell me who you are.

Run

  git config --global user.email "you@example.com"
  git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'Admin@DESKTOP-S2142DE.(none)')

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git config --global user.email "malavikka2k@gmail.com"

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git config --global user.name "malavikka"

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git status
On branch master

No commits yet

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


Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git commit -m "committing index file"
[master (root-commit) 74ce87a] committing index file
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 index.html

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git status
On branch master
nothing to commit, working tree clean

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ vi index.html
```

```
Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ vi index.html

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ touch info.text

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git status
On branch master
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:   index.html

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

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

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ vi .gitignore

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git status
On branch master
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:   index.html

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

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

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git commit -a -m "committing updated index file"
warning: LF will be replaced by CRLF in index.html.
The file will have its original line endings in your working directory
[master a0b34ce] committing updated index file
 1 file changed, 1 insertion(+)
```

```
Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git commit -a -m "committing updated index file"
warning: LF will be replaced by CRLF in index.html.
The file will have its original line endings in your working directory
[master a0b34ce] committing updated index file
 1 file changed, 1 insertion(+)

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git log
commit a0b34cead1a15a8c456c7d4a60131f3f83d39614 (HEAD -> master)
Author: malavikka <malavikka2k@gmail.com>
Date:   Mon Feb 7 18:18:32 2022 +0530

    committing updated index file

commit 74ce87ae385821022e2dfd745c18703157b98ea5
Author: malavikka <malavikka2k@gmail.com>
Date:   Mon Feb 7 18:16:24 2022 +0530

    committing index file

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ vi index.html

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git status
On branch master
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:   index.html

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

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

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git restore index.html
```

```
Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git restore index.html

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .gitignore

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

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ vi index.html

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git status
On branch master
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:   index.html

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

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

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git add index.html

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   index.html

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

```
Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git add index.html

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   index.html

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


Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git reset
Unstaged changes after reset:
M       index.html

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ alias my_add= 'git add'
bash: alias: git add: not found

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ alias my_add='git add'

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ my_add index.html

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   index.html

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


Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git commit -m 'comitting after changing add to my_add'
[master fc8699f] comitting after changing add to my_add
 1 file changed, 1 insertion(+), 1 deletion(-)
```

```
Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git reset
Unstaged changes after reset:
M       index.html

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ alias my_add= 'git add'
bash: alias: git add: not found

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ alias my_add='git add'

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ my_add index.html

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   index.html

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


Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git commit -m 'comitting after changing add to my_add'
[master fc8699f] comitting after changing add to my_add
 1 file changed, 1 insertion(+), 1 deletion(-)

Admin@DESKTOP-S2142DE MINGW64 ~/project_dir (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .gitignore

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

GIT Branching

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

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

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

22. Create an empty directory 'Assignments' & cd to 'Assignments'.

23. Create a file README.txt inside 'Assignments' & write few lines about the contents of 'Assignments' folder.

24. Commit README.txt file.

25. Now create a new branch 'html-assignments'.

26. Switch to 'html-assignments' branch.

27. Copy all HTML assignments inside 'Assignments' folder.

28. Commit HTML assignments into 'html-assignments' branch.

29. Make minor changes into few files belonging to 'html-assignments' branch.

30. Commit those changed files.

31. Switch to master branch.

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

33. Again switch to 'html-assignments' branch.

34. Make minor changes into few files belonging to 'html-assignments' branch.

35. Commit those changes.

36. Switch to master.

37. Merge 'html-assignments' branch into master. Confirm all html assignments are shown in master.

38. Finally delete the 'html-assignments' branch.

```
Admin@DESKTOP-S2142DE MINGW64 ~
$ mkdir assignments

Admin@DESKTOP-S2142DE MINGW64 ~
$ cd assignments

Admin@DESKTOP-S2142DE MINGW64 ~/assignments
$ git init
Initialized empty Git repository in C:/Users/Admin/assignments/.git/

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ touch README.txt

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git add .

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git status
On branch master

No commits yet

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


Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git commit -m "committing README.txt"
[master (root-commit) 612a8d4] committing README.txt
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 README.txt

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git branch html-assignments

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git checkout html-assignments
Switched to branch 'html-assignments'
```

```
Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git checkout html-assignments
Switched to branch 'html-assignments'

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git status
On branch html-assignments
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        sample1.html

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

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git add sample1.html

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git status
On branch html-assignments
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   sample1.html


Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git commit -m "committing html file"
[html-assignments b56f760] committing html file
 1 file changed, 287 insertions(+)
 create mode 100644 sample1.html

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git status
On branch html-assignments
nothing to commit, working tree clean

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ vim sample1.html

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git status
On branch html-assignments
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:   sample1.html

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

```
Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git add sample1.html

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git commit -m "committing after change in html file"
[html-assignments bcccf4d] committing after change in html file
 1 file changed, 2 insertions(+), 2 deletions(-)

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git status
On branch html-assignments
nothing to commit, working tree clean

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ vim README.txt

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git commit -m "committing README file after change"
On branch html-assignments
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:   README.txt

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

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ vim sample1.html

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git status
On branch html-assignments
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:   README.txt

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

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git commit
On branch html-assignments
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:   README.txt

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

```
Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git commit
On branch html-assignments
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:   README.txt

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

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git checkout html-assignments
Already on 'html-assignments'
M       README.txt

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ vim sample1.html

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git status
On branch html-assignments
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:   README.txt

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

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git add sample1.html

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git commit -m "committing again after change"
On branch html-assignments
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:   README.txt

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

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git checkout master
Switched to branch 'master'
M       README.txt
```

```
Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git checkout html-assignments
Already on 'html-assignments'
M       README.txt

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ vim sample1.html

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git status
On branch html-assignments
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:   README.txt

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

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git add sample1.html

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git commit -m "committing again after change"
On branch html-assignments
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:   README.txt

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

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (html-assignments)
$ git checkout master
Switched to branch 'master'
M       README.txt

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git merge html-assignments
Updating 612a8d4..bcccf4d
Fast-forward
 sample1.html | 287 +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 1 file changed, 287 insertions(+)
 create mode 100644 sample1.html

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git branch -d html-assignments
Deleted branch html-assignments (was bcccf4d).
```

SECTION-2 - (CSS assignments) Steps to follow:
1. Create a new branch 'css-assignments'.
2. Switch to 'css-assignments' branch.
3. Copy all CSS assignments inside 'Assignments' folder.
4. Commit CSS assignments into 'css-assignments' branch.
5. Make minor changes into README.txt file on line 1 belonging to 'css-assignments' branch.
6. Commit those changed files.
7. Switch to master branch.
8. Make minor changes into README.txt file on line 3 & commit those changes into master.
9. Again switch to 'css-assignments' branch.
10. Make minor changes into few files belonging to 'css-assignments' branch.
11. Commit those changes.
12. Switch to master.
13. Merge 'css-assignments' branch into master. Confirm all css assignments are shown in master.
14. Finally delete the 'css-assignments' branch.

```
Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git branch css-assignments

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git checkout css-assignments
Switched to branch 'css-assignments'
M       README.txt

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ git status
On branch css-assignments
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:   README.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        sample1.css

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

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ git add sample1.css
warning: LF will be replaced by CRLF in sample1.css.
The file will have its original line endings in your working directory

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ git commit -m "committing README.txt"
[css-assignments 2d4c4fd] committing README.txt
 1 file changed, 7 insertions(+)
 create mode 100644 sample1.css

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ git checkout master
Switched to branch 'master'
M       README.txt

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git add README.txt
warning: LF will be replaced by CRLF in README.txt.
The file will have its original line endings in your working directory

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git commit -m "committing master README.txt"
[master 1f24fc1] committing master README.txt
 1 file changed, 1 insertion(+)
```

```
Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git commit -m "committing master README.txt"
[master 1f24fc1] committing master README.txt
 1 file changed, 1 insertion(+)

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git commit -m "committing master README.txt"
On branch master
nothing to commit, working tree clean

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git checkout css-assignments
Switched to branch 'css-assignments'

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ git add sample1.css

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ git commit -m "committing css file"
On branch css-assignments
nothing to commit, working tree clean

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ vim README.txt

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ git status
On branch css-assignments
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:   README.txt

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

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ git add README.txt
warning: LF will be replaced by CRLF in README.txt.
The file will have its original line endings in your working directory

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ git checkout css-assignments
Already on 'css-assignments'
M       README.txt

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ git add sample1.css
```

```
Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ git status
On branch css-assignments
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:   README.txt

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

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ git add README.txt
warning: LF will be replaced by CRLF in README.txt.
The file will have its original line endings in your working directory

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ git checkout css-assignments
Already on 'css-assignments'
M       README.txt

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ git add sample1.css

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ git commit -m "committing css file"
[css-assignments 8c7b8d1] committing css file
 1 file changed, 1 insertion(+)

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ git commit -m "committing css file"
On branch css-assignments
nothing to commit, working tree clean

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (css-assignments)
$ git checkout master
Switched to branch 'master'

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git merge css-assignments
Merge made by the 'ort' strategy.
 sample1.css | 7 +++++++
 1 file changed, 7 insertions(+)
 create mode 100644 sample1.css

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git branch -d css-assignments
Deleted branch css-assignments (was 8c7b8d1).
```

SECTION-3 - (JavaScript assignments) Steps to follow:
1. Create a new branch 'js-assignments'.
2. Switch to 'js-assignments' branch.
3. Copy all JavaScript assignments inside 'Assignments' folder.
4. Commit JavaScript assignments into 'js-assignments' branch.
5. Make minor changes into README.txt file on line 1 belonging to 'js-assignments' branch.
6. Commit those changed files.
7. Switch to master branch.
8. Make minor changes into README.txt file on line 1 & commit those changes into master.
9. Again switch to 'js-assignments' branch.
10. Make minor changes into few files belonging to 'js-assignments' branch.
11. Commit those changes.
12. Switch to master.
13. Merge 'js-assignments' branch into master. Confirm all JavaScript assignments are shown in master.
14. Finally delete the 'js-assignments' branch.

```
Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git branch js-assignments

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git checkout js-assignments
Switched to branch 'js-assignments'

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (js-assignments)
$ git status
On branch js-assignments
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        index.js

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

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (js-assignments)
$ git add index.js

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (js-assignments)
$ git commit -m "committing js file"
[js-assignments 763d35e] committing js file
 1 file changed, 2863 insertions(+)
 create mode 100644 index.js

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (js-assignments)
$ git add README.txt

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (js-assignments)
$ git commit -m "committing README.txt file"
On branch js-assignments
nothing to commit, working tree clean

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (js-assignments)
$ git checkout master
Switched to branch 'master'

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git add README.txt

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git commit -m "committing README file of master branch"
On branch master
nothing to commit, working tree clean

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git checkout js-assignments
Switched to branch 'js-assignments'
```

```
Admin@DESKTOP-S2142DE MINGW64 ~/assignments (js-assignments)
$ git add README.txt

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (js-assignments)
$ git commit -m "commit change"
On branch js-assignments
nothing to commit, working tree clean

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (js-assignments)
$ git checkout master
Switched to branch 'master'

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git add README.txt

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git commit -m "commit change"
On branch master
nothing to commit, working tree clean

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git checkout master
Already on 'master'

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git merge js-assignments
Updating f5a1f42..763d35e
Fast-forward
 index.js | 2863 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 1 file changed, 2863 insertions(+)
 create mode 100644 index.js

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git status
On branch master
nothing to commit, working tree clean

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git add .

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git commit -m "final done"
On branch master
nothing to commit, working tree clean

Admin@DESKTOP-S2142DE MINGW64 ~/assignments (master)
$ git branch -d js-assignments
Deleted branch js-assignments (was 763d35e).
```