Section 0 -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

```
nikes@DESKTOP-MS7QBRK MINGW64 ~ (master)
$ mkdir project_dir

nikes@DESKTOP-MS7QBRK MINGW64 ~ (master)
$ cd project_dir

nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ git init
Initialized empty Git repository in C:/Users/nikes/project_dir/.git/

nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ touch index.html

nikes@DESKTOP-MS7QBRK 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)

nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ git add index.html

nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ git commit -m "html file addeed"
[master (root-commit) ae5eebe] html file addeed
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 index.html

nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$
```

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.

```
create mode 100644 index.html

nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ touch info.txt

nikes@DESKTOP-MS7QBRK 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.txt

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

nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ touch .gitignore

nikes@DESKTOP-MS7QBRK 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
```

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

11. State & commit index.html

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

nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ git add index.html

nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ git commit -m "modified file"
[master b93aded] modified file
 1 file changed, 8 insertions(+)

nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$
```

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.

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ git log
commit b93aded4cb4f5512428ea4af199b51e6f29129db (HEAD -> master)
Author: nikesh <nikeshpuanati@gmail.com>
Date:    Thu Oct 21 16:56:49 2021 +0530

    modified file

commit ae5eebeeaf862428c5059956e40af84e773e0c64
Author: nikesh <nikeshpuanati@gmail.com>
Date:    Thu Oct 21 16:30:09 2021 +0530

    html file addeed
```

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ git checkout index.html
Updated 1 path from the index
```

16. Stage index.html

17. Revert back the last stage.

18. Rename 'add' command to my-add

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ git revert HEAD index.html
fatal: bad revision 'index.html'

nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ git reset HEAD index.html
Unstaged changes after reset:
M       index.html

nikes@DESKTOP-MS7QBRK 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")

nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ git config --global alias.my-add add
```

19. Using my_add command Stage index.html again & commit the changes.

20. Revert the last commit.

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ git my-add index.html

nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ git commit -m "html file"
[master dedb3ef] html file
 1 file changed, 1 insertion(+)

nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ git log
commit dedb3ef54278e6ae65687942f33f4120076b8e8a (HEAD -> master)
Author: nikesh <nikeshpuanati@gmail.com>
Date:   Thu Oct 21 17:20:03 2021 +0530

    html file

commit b93aded4cb4f5512428ea4af199b51e6f29129db
Author: nikesh <nikeshpuanati@gmail.com>
Date:   Thu Oct 21 16:56:49 2021 +0530

    modified file

commit ae5eebeeaf862428c5059956e40af84e773e0c64
Author: nikesh <nikeshpuanati@gmail.com>
Date:   Thu Oct 21 16:30:09 2021 +0530

    html file addeed

nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ git reset --soft HEAD^

nikes@DESKTOP-MS7QBRK MINGW64 ~/project_dir (master)
$ git log
commit b93aded4cb4f5512428ea4af199b51e6f29129db (HEAD -> master)
Author: nikesh <nikeshpuanati@gmail.com>
Date:   Thu Oct 21 16:56:49 2021 +0530

    modified file

commit ae5eebeeaf862428c5059956e40af84e773e0c64
Author: nikesh <nikeshpuanati@gmail.com>
Date:   Thu Oct 21 16:30:09 2021 +0530

    html file addeed
```

SECTION -1(HTML ASSIGNMENTS)

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.

```
nikes@DESKTOP-MS7QBRK MINGW64 ~ (master)
$ mkdir assignment

nikes@DESKTOP-MS7QBRK MINGW64 ~ (master)
$ cd assignment

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (master)
$ git init
Initialized empty Git repository in C:/Users/nikes/assignment/.git/

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (master)
$ touch README.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (master)
$ git status
On branch master

No commits yet

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

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

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (master)
$ git add README.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (master)
$ git commit -m " added a file"
[master (root-commit) efa89c8]  added a file
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 README.txt
```

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

26.Switch to 'html-assignments' branch.

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (master)
$ git branch html-assignments

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (master)
$ git checkout html-assignments
Switched to branch 'html-assignments'
```

 27Copy all HTML assignments inside assignments folder.

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

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (html-assignments)
$ touch create.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (html-assignments)
$ git status
On branch html-assignments
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   create.txt


nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (html-assignments)
$ git add create.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (html-assignments)
$ git status
On branch html-assignments
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   create.txt


nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (html-assignments)
$ touch create1.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (html-assignments)
$ git add create1.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (html-assignments)
$ git status
On branch html-assignments
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   create.txt
        new file:   create1.txt


nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (html-assignments)
$ git commit -m "added files"
[html-assignments b9ce9c6] added files
 2 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 create.txt
 create mode 100644 create1.txt
```

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

30. commit those changes.

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (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:   create.txt

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

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (html-assignments)
$ git add create.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (html-assignments)
$ git commit -m "modified file"
[html-assignments 0988531] modified file
 1 file changed, 1 insertion(+)
```

31.Switch to master branch.

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

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (html-assignments)
$ git checkout master
Switched to branch 'master'

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (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:   README.txt

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

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (master)
$ git add README.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (master)
$ git commit -m"modified file"
[master b22bf75] modified file
 1 file changed, 1 insertion(+)
```

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

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

35.Commit those changes.

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (master)
$ git checkout html-assignments
Switched to branch 'html-assignments'

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (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:   create1.txt

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

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (html-assignments)
$ git add create1.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (html-assignments)
$ git commit -m "modified file"
[html-assignments 24d1a03] modified file
 1 file changed, 1 insertion(+)
```

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.

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (html-assignments)
$ git checkout master
Switched to branch 'master'
```

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (master)
$ git merge html-assignments -m "merging braches"
Merge made by the 'recursive' strategy.
 create.txt  | 1 +
 create1.txt | 1 +
 2 files changed, 2 insertions(+)
 create mode 100644 create.txt
 create mode 100644 create1.txt
```

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (master)
$ git branch --merged
  html-assignments
* master
```

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment (master)
$ git branch -d html-assignments
Deleted branch html-assignments (was 24d1a03).
```

SECTION-2

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.

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (master)
$ git branch css-assignments

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (master)
$ git checkout css-assignments
Switched to branch 'css-assignments'

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (css-assignments)
$ touch demo.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (css-assignments)
$ git status
On branch css-assignments
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        demo.txt

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

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (css-assignments)
$ git add demo.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (css-assignments)
$ git status
On branch css-assignments
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   demo.txt


nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (css-assignments)
$ touch demo1.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (css-assignments)
$ git add demo1.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (css-assignments)
$ git status
On branch css-assignments
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   demo.txt
        new file:   demo1.txt


nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (css-assignments)
$ git commit -m "added files"
[css-assignments 13e5818] added files
 2 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 demo.txt
 create mode 100644 demo1.txt
```

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

6.Commit those changed files.

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (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")

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (css-assignments)
$ git add README.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (css-assignments)
$ git commit -m "modified readme "
[css-assignments 73d88d3] modified readme
 1 file changed, 1 insertion(+)
```

7.Switch to master branch.

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

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (css-assignments)
$ git checkout master
Switched to branch 'master'

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (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:   README.txt

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

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (master)
$ git add README.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (master)
$ git commit -m "modified file"
[master 023755f] modified file
 1 file changed, 4 insertions(+)
```

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.

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (master)
$ git checkout css-assignments
Switched to branch 'css-assignments'

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (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:   demo.txt

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

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (css-assignments)
$ git add demo.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (css-assignments)
$ git commit -m "added amodified file"
[css-assignments 5600f0c] added amodified file
 1 file changed, 1 insertion(+)

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (css-assignments)
$ git checkout master
Switched to branch 'master'
```

13. Merge 'css-assignments' branch into master. Confirm all css- assignments are shown in master.

14. Finally delete the 'css-assignments' branch

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (master)
$ git merge css-assignments -m "merged"
Already up to date.

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment1 (master)
$ git branch -d css-assignments
Deleted branch css-assignments (was 5600f0c).
```

SECTION-3

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.

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (master)
$ git branch js-assignments

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (master)
$ git checkout js-assignments
Switched to branch 'js-assignments'

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (js-assignments)
$ touch demo.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (js-assignments)
$ git status
On branch js-assignments
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        demo.txt

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

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (js-assignments)
$ git add demo.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (js-assignments)
$ git status
On branch js-assignments
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   demo.txt


nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (js-assignments)
$ touch demo1.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (js-assignments)
$ git add demo1.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (js-assignments)
$ git status
On branch js-assignments
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   demo.txt
        new file:   demo1.txt


nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (js-assignments)
$ git commit -m"adding files"
[js-assignments a275bed] adding files
 2 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 demo.txt
 create mode 100644 demo1.txt
```

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

6.Commit those changed files.

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (js-assignments)
$ git status
On branch js-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")

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (js-assignments)
$ git add README.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (js-assignments)
$ git commit -m"modified file"
[js-assignments 4ba82b2] modified file
 1 file changed, 1 insertion(+)
```

7.Switch to master branch.

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

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (js-assignments)
$ git checkout master
Switched to branch 'master'

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (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:   README.txt

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

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (master)
$ git add README.txt
```

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (master)
$ git commit -m"added a modified file"
[master 5789592] added a modified file
 1 file changed, 1 insertion(+)
```

9.Again switch to 'js-assignments' branch.

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (master)
$ git checkout js-assignments
Switched to branch 'js-assignments'
```

10.Make minor changes into few files belonging to 'js-assignments' branch.

11.Commit those changes.

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (js-assignments)
$ git status
On branch js-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:   demo.txt

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

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (js-assignments)
$ git add demo.txt

nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (js-assignments)
$ git commit -m "added a file modification"
[js-assignments b46579a] added a file modification
 1 file changed, 1 insertion(+)
```

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.

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (master|MERGING)
$ git status
On branch master
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)
```

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (master)
$ git merge js-assignments
Already up to date.
```

```
nikes@DESKTOP-MS7QBRK MINGW64 ~/assignment2 (master)
$ git branch -d js-assignments
Deleted branch js-assignments (was b46579a).
```