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.

```
Hp@DESKTOP-HBODGST MINGW64 ~
$ mkdir project_dir

Hp@DESKTOP-HBODGST MINGW64 ~
$ cd project_dir

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

Hp@DESKTOP-HBODGST MINGW64 ~/project_dir (master)
$ touch index.html

Hp@DESKTOP-HBODGST 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)

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

Hp@DESKTOP-HBODGST 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


Hp@DESKTOP-HBODGST MINGW64 ~/project_dir (master)
$ git commit -m "committing index file"
[master (root-commit) 4e7e824] committing index file
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 index.html


Hp@DESKTOP-HBODGST MINGW64 ~/project_dir (master)
$ touch info.txt


Hp@DESKTOP-HBODGST 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")
```

```
Hp@DESKTOP-HBODGST MINGW64 ~/project_dir (master)
$ touch .gitignore

Hp@DESKTOP-HBODGST 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
        info.txt

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

Hp@DESKTOP-HBODGST 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")

Hp@DESKTOP-HBODGST MINGW64 ~/project_dir (master)
$ git add .gitignore

Hp@DESKTOP-HBODGST MINGW64 ~/project_dir (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   .gitignore

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


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

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


Hp@DESKTOP-HBODGST MINGW64 ~/project_dir (master)
$ git commit -m "intex2"
[master c3232f8] intex2
 2 files changed, 3 insertions(+)
 create mode 100644 .gitignore
```

```
Hp@DESKTOP-HBODGST MINGW64 ~/project_dir (master)
$ git log
commit c3232f8b40b75eb74c59a187fd566a1e0085c39c (HEAD -> master)
Author: nishanth <dnishanth2000@gmail.com>
Date:   Sun Jan 10 13:54:29 2021 +0530

    intex2

commit 4e7e8240dadb5c1a3174b3dbe1e93103005da8bf
Author: nishanth <dnishanth2000@gmail.com>
Date:   Sun Jan 10 13:19:52 2021 +0530

    committing index file

Hp@DESKTOP-HBODGST 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

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

Hp@DESKTOP-HBODGST MINGW64 ~/project_dir (master)
$ git restore index.html

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

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

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


Hp@DESKTOP-HBODGST MINGW64 ~/project_dir (master)
$ git commit -m "commit after editing"
[master 0c1a518] commit after editing
 1 file changed, 1 insertion(+)

Hp@DESKTOP-HBODGST MINGW64 ~/project_dir (master)
$ git reset

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


Hp@DESKTOP-HBODGST MINGW64 ~/project_dir (master)
$ git reset HEAD~1
Unstaged changes after reset:
M       index.html

Hp@DESKTOP-HBODGST 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

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

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

Hp@DESKTOP-HBODGST 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

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


Hp@DESKTOP-HBODGST MINGW64 ~/project_dir (master)
$ git my-add index.html

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

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.

```
Hp@DESKTOP-HBODGST MINGW64 ~
$ mkdir Assignments

Hp@DESKTOP-HBODGST MINGW64 ~
$ cd Assignments

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments
$ git init
Initialized empty Git repository in C:/Users/Hp/Assignments/.git/

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ touch README.txt

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git add README.txt

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git commit -m " Committing README text file"
[master (root-commit) 20db6be]  Committing README text file
 1 file changed, 9 insertions(+)
 create mode 100644 README.txt
Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git branch html-assignments

MINGW64 ~/Assignments (master)
$ git checkout html-assignments
Switched to branch 'html-assignments'

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (html-assignments)
$ git status
On branch html-assignments
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) Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (html-assignments)
$ git add .
```

```
Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (html-assignments)
$ git commit -m "committing HTML assignments into html-assignments branch"
[html-assignments 78eb2c9] committing HTML assignments into html-assignments branch
 1 file changed, 2 insertions(+)
 create mode 100644 index.html

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

Hp@DESKTOP-HBODGST 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:    index.html

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

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (html-assignments)
$ git add .

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (html-assignments)
$ git commit -m "committing minorly changed html file into html-assignments branch"
[html-assignments e098278] committing minorly changed html file into html-assignments
branch
 1 file changed, 2 insertions(+), 1 deletion(-)

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (html-assignments)
$ git checkout master
Switched to branch 'master'
```

```
Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (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")

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git add .

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git commit -m "  commiting minorly changed text file into master
branch" [master 0667eb5]  commiting minorly changed text file into master
branch
 1 file changed, 3 insertions(+), 1 deletion(-)

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git checkout html-assignments
Switched to branch 'html-assignments'

Hp@DESKTOP-HBODGST 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:    index.html

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

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (html-assignments)
$ git add .

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (html-assignments)
$ git commit -m "making final commit of html file into html-assignments branch"
[html-assignments ace842f] making final commit of html file into html-assignments
branch
 1 file changed, 1 insertion(+)

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (html-assignments)
$ git checkout master
Switched to branch 'master'
Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git merge html-assignments
Merge made by the 'recursive' strategy.
 index.html | 4 ++++
 1 file changed, 4 insertions(+)
 create mode 100644 index.html

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git branch -d html-assignments
Deleted branch html-assignments (was ace842f).
```

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.

```
Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git branch css-assignments

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git checkout css-assignments
Switched to branch 'css-assignments'

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (css-assignments)
$ touch file.css

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (css-assignments)
$ git add .

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (css-assignments)
$ git commit -m "committing css assignments into css-assignments branch"
[css-assignments f7c7971] committing css assignments into css-assignments branch
 1 file changed, 1 insertion(+)
 create mode 100644 file.css

Hp@DESKTOP-HBODGST 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")

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (css-assignments)
$ git add .

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (css-assignments)
$ git commit -m "committing modified readme text of css-assignments branch"
[css-assignments accb550] committing modified readme text of css-assignments branch
 1 file changed, 1 insertion(+), 1 deletion(-)

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (css-assignments)
$ git checkout master
Switched to branch 'master'

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (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")

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git add .

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git commit -m "committing modified readme text of master branch"
[master de67f63] committing modified readme text of master branch
 1 file changed, 1 insertion(+), 1 deletion(-)

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git checkout css-assignments
Switched to branch 'css-assignments'

Hp@DESKTOP-HBODGST 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:    file.css

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

```
Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (css-assignments)
$ git add .
g
Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (css-assignments)
$ git commit -m "final commit of css-assignments branch"
[css-assignments 0f04977] final commit of css-assignments branch
 1 file changed, 3 insertions(+), 1 deletion(-)

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (css-assignments)
$ git checkout master
Switched to branch 'master'

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git merge css-
assignments Auto-merging
README.txt
Merge made by the 'recursive' strategy.
 README.txt | 2 +-
 file.css   | 3 +++
 2 files changed, 4 insertions(+), 1 deletion(-)
 create mode 100644 file.css

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git branch -d css-assignments
Deleted branch css-assignments (was 0f04977).
```

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

```
Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git branch js-assignments

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git checkout js-assignments
Switched to branch 'js-assignments'
M       README.txt

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (js-assignments)
$ touch jsfile.js

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (js-assignments)
$ git add .

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (js-assignments)
$ git commit -m "committing js assignment to js-assignments branch"
[js-assignments 6f50748] committing js assignment to js-assignments branch
 2 files changed, 2 insertions(+), 1 deletion(-)
 create mode 100644 jsfile.js

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (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")

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (js-assignments)
$ git add .

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (js-assignments)
$ git commit -m "committing modified README text file of js-assignments branch"
[js-assignments 634d0c0] committing modified README text file of js-assignments branch
 1 file changed, 1 insertion(+), 1 deletion(-)

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (js-assignments)
$ git checkout master
Switched to branch 'master'

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (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")

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git add .

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git commit -m "committing modified README text file of master branch"
[master 7edce3a] committing modified README text file of master branch
 1 file changed, 2 insertions(+)

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git checkout js-assignments
Switched to branch 'js-assignments'

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (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:    jsfile.js

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

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (js-assignments)
$ git add .

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (js-assignments)
$ git commit -m "Final commit of js-assignments"
[js-assignments 73ab364] Final commit of js-assignments
 1 file changed, 1 insertion(+)

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (js-assignments)
$ git checkout master
Switched to branch 'master'

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git merge js-assignments
Auto-merging README.txt
Merge made by the 'recursive' strategy.
 README.txt | 2 +-
 jsfile.js  | 2 ++
 2 files changed, 3 insertions(+), 1 deletion(-)
 create mode 100644 jsfile.js

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git branch -d js-assignments
Deleted branch js-assignments (was 73ab364).
```

Objective: Pushing source code into GITHUB & collaborate team

members. SECTION-1 (Pushing assignments to remote repository) -

Steps to follow:

39. Create a github account if you do not have already.
40. Login on into github account.
41. Create new public repository ‘freshersbatch-oct16’.
42. Commit & push any sample file to this repository under ‘Assignments’ directory.

```
Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git config --global user.name Nishanth

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git config --global user.email dnishanth2000@gmail.com
```

```
Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git remote add origin https://github.com/Nishanth /freshersbatch-oct16.git

Hp@DESKTOP-HBODGST MINGW64 ~/Assignments (master)
$ git push -u origin master
Enumerating objects: 51, done.
Counting objects: 100% (51/51),
done.
Delta compression using up to 4
threads Compressing objects: 100%
(46/46), done.
Writing objects: 100% (51/51), 5.63 KiB | 412.00 KiB/s,
done. Total 51 (delta 11), reused 0 (delta 0), pack-reused
0 remote: Resolving deltas: 100% (11/11), done.
To https://github.com/Nishanth/freshersbatch-oct16.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
```