Git Assignment by Abhishek Sharma

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’.

405an@DESKTOP-UTDBI3K MINGW64 ~

$ mkdir project\_dir

405an@DESKTOP-UTDBI3K MINGW64 ~

$ cd project\_dir

2. Initialize git version database.

405an@DESKTOP-UTDBI3K MINGW64 ~/project\_dir

$ git init

Initialized empty Git repository in C:/Users/405an/project\_dir/.git/

3. Create a new file index.html.

405an@DESKTOP-UTDBI3K MINGW64 ~/project\_dir (master)

$ touch index.html

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

405an@DESKTOP-UTDBI3K 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)

5. Stage the index.html file.

405an@DESKTOP-UTDBI3K MINGW64 ~/project\_dir (master)

$ git add index.html

6. Commit index.html

405an@DESKTOP-UTDBI3K MINGW64 ~/project\_dir (master)

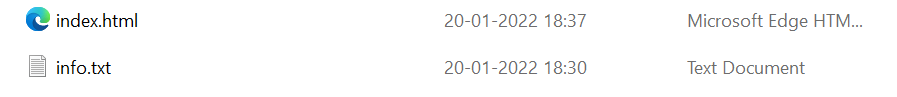
$ git commit -m "index.html is commited"

[master (root-commit) c16714f] index.html is commited

1 file changed, 0 insertions(+), 0 deletions(-)

create mode 100644 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.

405an@DESKTOP-UTDBI3K 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")

9. Configure GIT to ignore all txt files.

405an@DESKTOP-UTDBI3K MINGW64 ~/project\_dir (master)

$ git add \*.txt

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

405an@DESKTOP-UTDBI3K MINGW64 ~/project\_dir (master)

$ git status

On branch master

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

new file: info.txt

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

11. State & commit index.html

405an@DESKTOP-UTDBI3K MINGW64 ~/project\_dir (master)

$ git add index.html

405an@DESKTOP-UTDBI3K MINGW64 ~/project\_dir (master)

$ git commit -m " this is the 2nd commit of index.html"

[master 31713b7] this is the 2nd commit of index.html

2 files changed, 12 insertions(+)

create mode 100644 info.txt

12. Log all your comments so far

405an@DESKTOP-UTDBI3K MINGW64 ~/project\_dir (master)

$ git log

commit 31713b7e78a79cd60b1f24a7bd27b352a514e142 (HEAD -> master)

Author: Abhishek <405ansh@gmail.com>

Date: Thu Jan 20 18:33:27 2022 +0530

this is the 2nd commit of index.html

commit c16714f775b80e28e23f58bde051e30baca8316b

Author: Abhishek <405ansh@gmail.com>

Date: Thu Jan 20 18:28:26 2022 +0530

index.html is commited

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

405an@DESKTOP-UTDBI3K MINGW64 ~/project\_dir (master)

$ git checkout index.html

Updated 1 path from the index

16. Stage index.html

405an@DESKTOP-UTDBI3K MINGW64 ~/project\_dir (master)

$ git add index.html

17. Revert back the last stage.

405an@DESKTOP-UTDBI3K MINGW64 ~/project\_dir (master)

$ git restore --staged index.html

405an@DESKTOP-UTDBI3K 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

18. Rename ‘add’ command to ‘my-add’.

405an@DESKTOP-UTDBI3K MINGW64 ~/project\_dir (master)

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

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

405an@DESKTOP-UTDBI3K MINGW64 ~/project\_dir (master)

$ git my-add index.html

405an@DESKTOP-UTDBI3K MINGW64 ~/project\_dir (master)

$ git commit -m "commited with new alias"

[master 361e3fd] commited with new alias

2 files changed, 4 insertions(+), 2 deletions(-)

create mode 100644 .gitignore

20. Revert the last commit.

405an@DESKTOP-UTDBI3K MINGW64 ~/project\_dir (master)

$ git reset HEAD~

Unstaged changes after reset:

M index.html

**GIT Branching**

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

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

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

405an@DESKTOP-UTDBI3K MINGW64 ~

$ mkdir Assignments

405an@DESKTOP-UTDBI3K MINGW64 ~

$ cd Assignments

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments

$ git init

Initialized empty Git repository in C:/Users/405an/Assignments/.git/

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments

$ touch README.txt

24. Commit README.txt file.

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git add README.txt

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git commit -m "readme.txt is commited"

[master (root-commit) 3e9d15e] readme.txt is commited

1 file changed, 3 insertions(+)

create mode 100644 README.txt

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git branch html-assignments

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git status

On branch master

nothing to commit, working tree clean

26. Switch to ‘html-assignments’ branch.

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git checkout html-assignments

Switched to branch 'html-assignments'

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (html-assignments)

$ git status

On branch html-assignments

nothing to commit, working tree clean

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (html-assignments)

$ git commit -m "commited all copied file into Assignments"

On branch html-assignments

Untracked files:

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

app.js

main.css

website.html

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

30. Commit those changed files

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (html-assignments)

$ git add .

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (html-assignments)

$ git commit -m "commited all copied files into Assignment"

[html-assignments 93625dc] commited all copied files into Assignment

3 files changed, 28 insertions(+)

create mode 100644 app.js

create mode 100644 main.css

create mode 100644 website.html

31. Switch to master branch.

@DESKTOP-UTDBI3K MINGW64 ~/Assignments (html-assignments)

$ git checkout master

Switched to branch 'master'

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git add .

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git commit -m "change is commited in master branch"

[master 5955204] change is commited in master branch

1 file changed, 1 insertion(+)

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git checkout html-assignments

Switched to branch 'html-assignments'

35. Commit those changes.

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (html-assignments)

$ git add .

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (html-assignments)

$ git commit -m "changes are commited in a branch"

[html-assignments 1f2221a] changes are commited in a branch

1 file changed, 1 insertion(+), 1 deletion(-)

36. Switch to master.

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (html-assignments)

$ git checkout master

Switched to branch 'master'

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git merge html-assignments

Merge made by the 'ort' strategy.

app.js | 2 ++

main.css | 11 +++++++++++

website.html | 15 +++++++++++++++

3 files changed, 28 insertions(+)

create mode 100644 app.js

create mode 100644 main.css

create mode 100644 website.html

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git branch -d html-assignments

Deleted branch html-assignments (was 1f2221a).

SECTION-2 - (CSS assignments) Steps to follow

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git branch css-assignment

2. Switch to ‘css-assignments’ branch.

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git checkout css-assignment

Switched to branch 'css-assignment'

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (css-assignment)

$ git add .

6. Commit those changed files.

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (css-assignment)

$ git commit -m "changes are commited"

[css-assignment 1d83113] changes are commited

1 file changed, 1 insertion(+), 1 deletion(-)

7. Switch to master branch.

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (css-assignment)

$ git checkout master

Switched to branch 'master'

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git add .

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git commit -m "changes on readme.txt in master branch"

[master 39e39bc] changes on readme.txt in master branch

1 file changed, 1 insertion(+), 1 deletion(-)

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git checkout css-assignment

Switched to branch 'css-assignment'

11. Commit those changes.405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (css-assignment)

$ git add .

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (css-assignment)

$ git commit -m "the minor changes in css branch"

[css-assignment d952473] the minor changes in css branch

1 file changed, 1 insertion(+)

12. Switch to master.

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (css-assignment)

$ git checkout master

Switched to branch 'master'

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git merge css-assignment

Auto-merging README.txt

Merge made by the 'ort' strategy.

README.txt | 2 +-

main.css | 1 +

2 files changed, 2 insertions(+), 1 deletion(-)

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git branch -d css-assignment

Deleted branch css-assignment (was d952473).

SECTION-3 - (JavaScript assignments) Steps to follow

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git branch js-assignment

2. Switch to ‘js-assignments’ branch.

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git checkout js-assignment

Switched to branch 'js-assignment'

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (js-assignment)

$ git add .

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (js-assignment)

$ git commit -m "the js file is added"

On branch js-assignment

nothing to commit, working tree clean

6. Commit those changed files.

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (js-assignment)

$ git add .

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (js-assignment)

$ git commit -m "changes are done in README.txt"

[js-assignment b324d74] changes are done in README.txt

1 file changed, 1 insertion(+), 1 deletion(-)

7. Switch to master branch.

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (js-assignment)

$ git checkout master

Switched to branch 'master'

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git add .

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git commit -m "the changes are done in master branch"

[master 326776a] the changes are done in master branch

1 file changed, 1 insertion(+), 1 deletion(-)

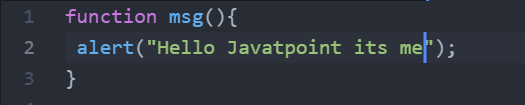
9. Again switch to ‘js-assignments’ branch.

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git checkout js-assignment

Switched to branch 'js-assignment'

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



11. Commit those changes.

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (js-assignment)

$ git add .

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (js-assignment)

$ git commit -m "the changes in js file"

[js-assignment 39a8ec1] the changes in js file

1 file changed, 2 insertions(+), 2 deletions(-)

12. Switch to master.

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (js-assignment)

$ git checkout master

Switched to branch 'master'

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git merge js-assignment

Auto-merging README.txt

CONFLICT (content): Merge conflict in README.txt

Automatic merge failed; fix conflicts and then commit the result.

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master|MERGING)

$ git merge js-assignment

error: Merging is not possible because you have unmerged files.

hint: Fix them up in the work tree, and then use 'git add/rm <file>'

hint: as appropriate to mark resolution and make a commit.

fatal: Exiting because of an unresolved conflict.

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git merge js-assignment

Already up to date.

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

405an@DESKTOP-UTDBI3K MINGW64 ~/Assignments (master)

$ git branch -d js-assignment

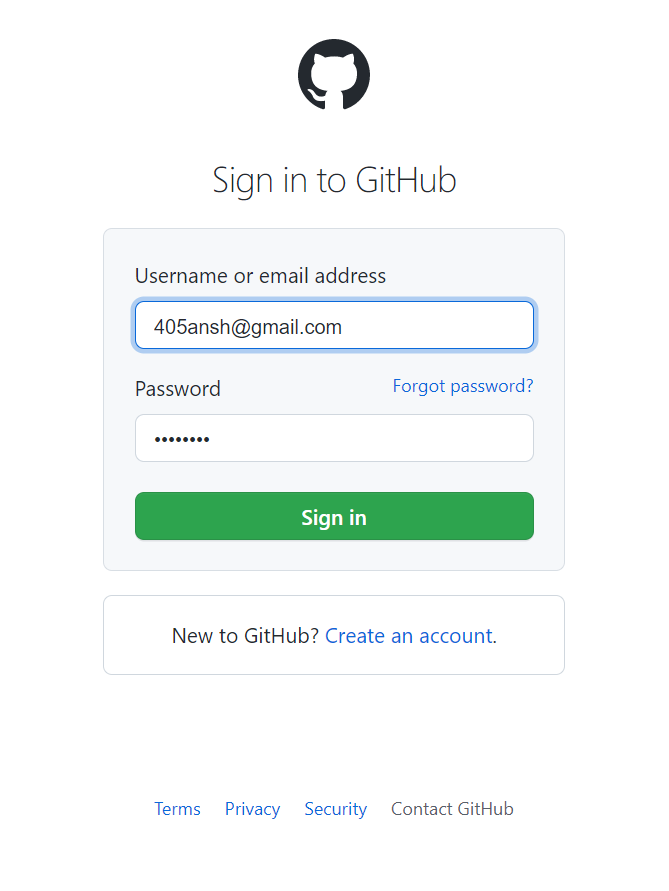
Deleted branch js-assignment (was 39a8ec1).

GIT Remoting

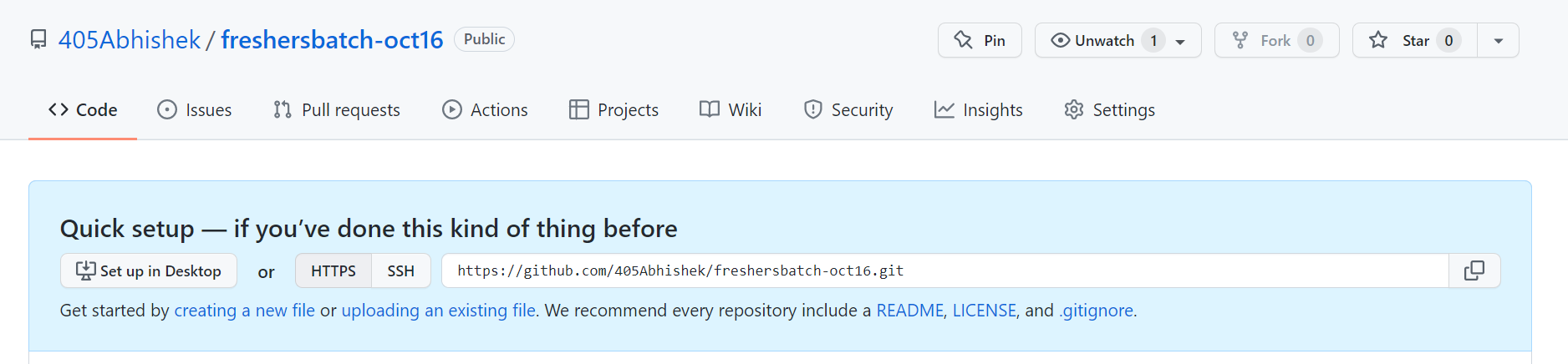
Objective: Pushing source code into GITHUB & collaborate team members.

SECTION-1 (Pushing assignments to remote repository) - Steps to follow

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.

