Install GIT & make sure it is added into PATH.

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

Q1. Create a directory ‘project\_dir’ & cd to ‘project\_dir’.

roneve@LIN24009020 MINGW64 ~

$ mkdir project\_dir

roneve@LIN24009020 MINGW64 ~

$ cd project\_dir

Q2. Initialize git version database. (git init)

roneve@LIN24009020 MINGW64 ~ /project\_dir

$ git init

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

Q3. Create a new file index.html.

Created index.html in the following directory (C:/Users/RONEVE/project\_dir/.git/)

Q4. Check the git status. You should find index.html as untracked file.

roneve@LIN24009020 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)

Q5. Stage the index.html file.

roneve@LIN24009020 MINGW64 -/project\_dir (master)

$ git add index.html

Q6. Commit index.html

roneve@LIN24009020 MINGW64 -/project\_dir (master)

$ git commit -m "Commiting"

[master (root-commit) b6e4aca] Commiting

Committer: Neve <rohan-jayvant.neve@capgemini.com>

Your name and email address were configured automatically based

on your username and hostname. Please check that they are accurate.

You can suppress this message by setting them explicitly. Run the

following command and follow the instructions in your editor to edit

your configuration file:

git config --global --edit

After doing this, you may fix the identity used for this commit with:

git commit --amend --reset-author

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

create mode 100644 index.html

Q7. Make few changes in index.html & create a new file info.txt file.

Changes were made and info.txt file was created.

Q8. Check git status. You should find index.html & info.txt as untracked files.

roneve@LIN24009020 MINGW64 -/project\_dir

$ 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")

Q9. Configure GIT to ignore all txt files.

roneve@LIN24009020 MINGW64 -/project\_dir

$ touch .gitignore

Change the .gitignore file with “\*.txt”, so that it ignore all txt files.

roneve@LIN24009020 MINGW64 -/project\_dir

$ git add .gitignore

Q10. Again check the git status. You should find only index.html as untracked file.

roneve@LIN24009020 MINGW64 -/project\_dir

$ 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

Q11. State & commit index.html

roneve@LIN24009020 MINGW64 -/project\_dir

$ git add index.html

roneve@LIN24009020 MINGW64 -/project\_dir

$ git commit -m "Committing"

[master b07a0a4] Committing

2 files changed, 3 insertions(+)

create mode 100644 .gitignore

Q12. Log all your comments so far.

roneve@LIN24009020 MINGW64 -/project\_dir

$ git log -p

commit 73ba595dc0c97bef7135ea201bcaa6cace47c5ad (HEAD -> master)

Author: Neve <rohan-jayvant.neve@capgemini.com>

Date: Thu Jan 6 21:14:43 2022 +0530

Committing

diff --git a/.gitignore b/.gitignore

new file mode 100644

index 0000000..2211df6

--- /dev/null

+++ b/.gitignore

@@ -0,0 +1 @@

+\*.txt

diff --git a/index.html b/index.html

index e69de29..8c7fe21 100644

--- a/index.html

+++ b/index.html

@@ -0,0 +1,2 @@

+<html>

+</html>

\ No newline at end of file

commit b07a0a4b00116f09ca150ddb105 e14b04cd8dcfa (HEAD -> master)

:

If you want to just view the commits “git log” is the commandline.

Q13. Make some changes in index.html.

Made some changes to index.html

Q14. Revert the change made in the previous step using git command.

roneve@LIN24009020 MINGW64 -/project\_dir

$ git revert 821b9f1

[master 712a14d] Revert "Added h1 tag"

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

roneve@LIN24009020 MINGW64 -/project\_dir

$ git log --oneline

712a14d (HEAD -> master) Revert "Added h1 tag"

73ba595 Added h1 tag

554a8b6 Commiting

c43d2b0 Committing

b6e4aca Commiting

Q15. Again change index.html.

Made some changes to index.html

roneve@LIN24009020 MINGW64 -/project\_dir

$ 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")

Q16. Stage index.html

roneve@LIN24009020 MINGW64 -/project\_dir

$ git add index.html

roneve@LIN24009020 MINGW64 -/project\_dir

$ git commit -m "Added h2"

[master d3bf6d7] Added h2

1 file changed, 3 insertions(+), 1 deletion(-)

Q17. Revert back the last stage.

roneve@LIN24009020 MINGW64 -/project\_dir

$ git log --oneline

d3bf6d7 (HEAD -> master) Added h2

712a14d Revert "Added h1 tag"

821b9f1 Added h1 tag

554a8b6 Commiting

c43d2b0 Committing

b6e4aca Commiting

subsatpa@LIN24009587 MINGW64 ~/project\_dir (master)

$ git revert d3bf6d7

[master 8271e04] Revert "Added h2"

1 file changed, 1 insertion(+), 3 deletions(-)

roneve@LIN24009020 MINGW64 -/project\_dir

$ git log --oneline

8271e04 (HEAD -> master) Revert "Added h2"

d3bf6d7 Added h2

712a14d Revert "Added h1 tag"

821b9f1 Added h1 tag

554a8b6 Commiting

c43d2b0 Committing

b6e4aca Commiting

Reverted the last stage. (“Added h2”)

Q18. Rename ‘add’ command to ‘my-add’.

roneve@LIN24009020 MINGW64 -/project\_dir

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

Q19. Using my\_add command Stage index.html again & commit the changes.

roneve@LIN24009020 MINGW64 -/project\_dir

$ git my-add index.html

roneve@LIN24009020 MINGW64 -/project\_dir

$ git commit -m "Adding head tag"

[master 19a930f] Adding head tag

1 file changed, 3 insertions(+), 1 deletion(-)

Q20. Revert the last commit.

roneve@LIN24009020 MINGW64 -/project\_dir

$ git revert c6c5563

[master 6be8e3f] Revert "Adding head tag"

1 file changed, 1 insertion(+), 3 deletions(-)

roneve@LIN24009020 MINGW64 -/project\_dir (master)

$ git log --oneline

6be8e3f (HEAD -> master) Revert "Adding head tag"

c6c5563 Adding head tag

8271e04 Revert "Added h2"

d3bf6d7 Added h2

712a14d Revert "Added h1 tag"

821b9f1 Added h1 tag

554a8b6 Commiting

c43d2b0 Committing

b6e4aca Commiting

*GIT Branching*

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

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

Q21. First take a backup of your assignments & projects. This is required because due to incorrect GIT operation you may lose your files.

Local device backup taken for the directories and files available.

Q22. Create an empty directory ‘Assignments’ & cd to ‘Assignments’.

roneve@LIN24009020 MINGW64 ~

$ mkdir assignments

roneve@LIN24009020 MINGW64 ~

$ cd assignments

roneve@LIN24009020 MINGW64 ~/assignments

Q23. Create a file README.txt inside ‘Assignments’ & write few lines about the contents of ‘Assignments’ folder.

Created and added a short description in the file (“README.txt”)

Q24. Commit README.txt file.

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git add README.txt

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git commit -m "Added README.txt file"

[master (root-commit) 93f199b] Added README.txt file

1 file changed, 1 insertion(+)

create mode 100644 README.txt

Q25. Now create a new branch ‘html-assignments’.

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git branch html-assignment

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git branch

html-assignment

\* master

Q26. Switch to ‘html-assignments’ branch.

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git checkout html-assignment

Switched to branch 'html-assignment'

roneve@LIN24009020 MINGW64 ~/assignments (html-assignment)

$ git branch

\* html-assignment

master

Q27. Copy all HTML assignments inside ‘Assignments’ folder.

Copied all available assignments into the assignments folder.

Q28. Commit HTML assignments into ‘html-assignments’ branch.

roneve@LIN24009020 MINGW64 ~/assignments (html-assignment)

$ git add .

roneve@LIN24009020 MINGW64 ~/assignments (html-assignment)

$ git commit -m "HTML files committed"

[html-assignment 3befe37] HTML files committed

2 files changed, 0 insertions(+), 0 deletions(-)

create mode 100644 html1.html

create mode 100644 html2.html

Q29. Make minor changes into few files belonging to ‘html-assignments’ branch.

Initiated minor changes into few html files in the assignments folder in the html-assignments branch.

Q30. Commit those changed files.

roneve@LIN24009020 MINGW64 ~/assignments (html-assignment)

$ git status

On branch html-assignment

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: html1.html

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

roneve@LIN24009020 MINGW64 ~/assignments (html-assignment)

$ git add html1.html

roneve@LIN24009020 MINGW64 ~/assignments (html-assignment)

$ git commit -m "Updated html1"

[html-assignment 49efafd] Updated html1

1 file changed, 1 insertion(+)

Q31. Switch to master branch.

roneve@LIN24009020 MINGW64 ~/assignments (html-assignment)

$ git checkout master

Switched to branch 'master'

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git branch

html-assignment

\* master

Q32. Make minor changes into README.txt file & commit those changes into master.

Minor changes made to the README.txt

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git add README.txt

g

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git commit -m "README.txt Updated"

[master d192930] README.txt Updated

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

Q33. Again switch to ‘html-assignments’ branch.

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git checkout html-assignment

Switched to branch 'html-assignment'

roneve@LIN24009020 MINGW64 ~/assignments (html-assignment)

$ git branch

\* html-assignment

Master

Q34. Make minor changes into few files belonging to ‘html-assignments’ branch.

Changed few code lines in an html file in the assignments folder in the html-assignment branch.

Q35. Commit those changes.

roneve@LIN24009020 MINGW64 ~/assignments (html-assignment)

$ git status

On branch html-assignment

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: html2.html

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

roneve@LIN24009020 MINGW64 ~/assignments (html-assignment)

$ git add html2.html

subsatpa@LIN24009587 MINGW64 ~/assignments (html-assignment)

$ git commit -m "Changed html2"

[html-assignment 178eaa5] Changed html2

1 file changed, 1 insertion(+)

Q36. Switch to master.

roneve@LIN24009020 MINGW64 ~/assignments (html-assignment)

$ git checkout master

Switched to branch 'master'

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git branch

html-assignment

\* master

Q37. Merge ‘html-assignments’ branch into master. Confirm all html assignments are shown in master.

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git merge html-assignment

Changes to be committed:

new file: html1.html

new file: html2.html

subsatpa@LIN24009587 MINGW64 ~/assignments (master|MERGING)

$ git commit -m "All Changes Committed"

[master dd72d74] All Changes Committed

Q38. Finally delete the ‘html-assignments’ branch.

roneve@LIN24009020 MINGW64 ~/assignments ((178eaa5...))

$ git branch -d html-assignment

Deleted branch html-assignment (was 178eaa5).

roneve@LIN24009020 MINGW64 ~/assignments ((178eaa5...))

$ git branch

\* (HEAD detached at 178eaa5)

master

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

Q39. Create a new branch ‘css-assignments’.

roneve@LIN24009020 MINGW64 ~/assignments ((178eaa5...))

$ git branch css-assignment

roneve@LIN24009020 MINGW64 ~/assignments ((178eaa5...))

$ git branch

\* (HEAD detached at 178eaa5)

css-assignment

master

Q40. Switch to ‘css-assignments’ branch.

roneve@LIN24009020 MINGW64 ~/assignments ((178eaa5...))

$ git checkout css-assignment

Switched to branch 'css-assignment'

roneve@LIN24009020 MINGW64 ~/assignments (css-assignment)

$ git branch

\* css-assignment

Master

Q41. Copy all CSS assignments inside ‘Assignments’ folder.

Copied few css code files in assignment folder for css-assignment branch.

Q42. Commit CSS assignments into ‘css-assignments’ branch.

roneve@LIN24009020 MINGW64 ~/assignments (css-assignment)

$ git status

On branch css-assignment

Untracked files:

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

css1.css

css2.css

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

roneve@LIN24009020 MINGW64 ~/assignments (css-assignment)

$ git add .

roneve@LIN24009020 MINGW64 ~/assignments (css-assignment)

$ git commit -m "Commit All"

[css-assignment 7704c5e] Commit All

2 files changed, 0 insertions(+), 0 deletions(-)

create mode 100644 css1.css

create mode 100644 css2.css

Q43. Make minor changes into README.txt file on line 1 belonging to ‘css-assignments’ branch.

Made some minor changes to the file README.txt belonging to css-assignments branch. (Conflicts have to arise while merging as we are writing in the same file)

Q.44 Commit those changed files.

roneve@LIN24009020 MINGW64 ~/assignments (css-assignment)

$ git status

On branch css-assignment

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")

roneve@LIN24009020 MINGW64 ~/assignments (css-assignment)

$ git add README.txt

roneve@LIN24009020 MINGW64 ~/assignments (css-assignment)

$ git commit -m "README.txt Updated"

[css-assignment 78c1a70] README.txt Updated

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

Q45. Switch to master branch.

roneve@LIN24009020 MINGW64 ~/assignments (css-assignment)

$ git checkout master

Switched to branch 'master'

Q46. Make minor changes into README.txt file on line 3 & commit those changes into master.

(Conflicts have to arise while merging as we are writing in the same file)

roneve@LIN24009020 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")

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git add README.txt

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git commit -m "Updated README.txt"

[master 52465b3] Updated README.txt

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

Q47. Again switch to ‘css-assignments’ branch.

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git checkout css-assignment

Switched to branch 'css-assignment'

roneve@LIN24009020 MINGW64 ~/assignments (css-assignment)

$ git branch

\* css-assignment

master

Q48. Make minor changes into few files belonging to ‘css-assignments’ branch.

Changes are being made to the css files in the css assignment branch.

Q49. Commit those changes.

roneve@LIN24009020 MINGW64 ~/assignments (css-assignment)

$ git status

On branch css-assignment

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: css1.css

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

roneve@LIN24009020 MINGW64 ~/assignments (css-assignment)

$ git add css1.css

subsatpa@LIN24009587 MINGW64 ~/assignments (css-assignment)

$ git commit -m "Updated CSS File"

[css-assignment 4b39324] Updated CSS File

1. file changed, 1 insertion(+)

Q50. Switch to master.

roneve@LIN24009020 MINGW64 ~/assignments (css-assignment)

$ git checkout master

Switched to branch 'master'

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git branch

css-assignment

\* master

Q51. Merge ‘css-assignments’ branch into master. Confirm all css assignments are shown in master.

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git merge css-assignment

Changes to be committed:

new file: css1.css

new file: css2.css

roneve@LIN24009020 MINGW64 ~/assignments (master|MERGING)

$ git commit -m "All Changes Committed"

[master dd72d74] All Changes Committed

Q52. Finally delete the ‘css-assignments’ branch.

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git branch -D css-assignment

Deleted branch css-assignment (was 4b39324).

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git branch

\* master

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

Create a new branch ‘js-assignments’.

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git branch js-assignments

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git branch

js-assignments

\* master

Switch to ‘js-assignments’ branch.

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git checkout js-assignments

Switched to branch 'js-assignments'

roneve@LIN24009020 MINGW64 ~/assignments (js-assignments)

$ git branch

\* js-assignments

Master

Copy all JavaScript assignments inside ‘Assignments’ folder.

Copied and added some js files in the assignment folder in the js-assignments branch.

Commit JavaScript assignments into ‘js-assignments’ branch.

roneve@LIN24009020 MINGW64 ~/assignments (js-assignments)

$ git status

On branch js-assignments

Untracked files:

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

js1.js

js2.js

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

roneve@LIN24009020 MINGW64 ~/assignments (js-assignments)

$ git add js1.js

roneve@LIN24009020 MINGW64 ~/assignments (js-assignments)

$ git add js2.js

roneve@LIN24009020 MINGW64 ~/assignments (js-assignments)

$ git commit -m "Added All"

[js-assignments bfb19c7] Added All

2 files changed, 0 insertions(+), 0 deletions(-)

create mode 100644 js1.js

create mode 100644 js2.js

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

Made some minor changes to the file README.txt belonging to css-assignments branch (Conflicts have to arise while merging as we are writing in the same file)

Commit those changed files.

roneve@LIN24009020 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")

roneve@LIN24009020 MINGW64 ~/assignments (js-assignments)

$ git add README.txt

roneve@LIN24009020 MINGW64 ~/assignments (js-assignments)

$ git commit -m "README.txt Updated"

[js-assignments 78c1a70] README.txt Updated

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

Switch to master branch.

roneve@LIN24009020 MINGW64 ~/assignments (js-assignments)

$ git checkout master

Switched to branch 'master'

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git branch

js-assignments

\* master

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

Changes are made manually, (Conflicts have to arise while merging as we are writing in the same file)

roneve@LIN24009020 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")

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git add README.txt

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git commit -m "Added All"

[master a2e63e4] Added All

1 file changed, 1 insertion(+), 5 deletions(-)

Again switch to ‘js-assignments’ branch.

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git checkout js-assignments

Switched to branch 'js-assignments'

roneve@LIN24009020 MINGW64 ~/assignments (js-assignments)

$ git branch

\* js-assignments

master

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

Made some few changes in the code manually in the js-assignments branch.

Commit those changes.

roneve@LIN24009020 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: js1.js

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

roneve@LIN24009020 MINGW64 ~/assignments (js-assignments)

$ git add js1.js

roneve@LIN24009020 MINGW64 ~/assignments (js-assignments)

$ git commit -m "Updated JS File"

[js-assignments bc3fc3f] Updated JS File

1 file changed, 1 insertion(+)

Switch to master.

roneve@LIN24009020 MINGW64 ~/assignments (js-assignments)

$ git checkout master

Switched to branch 'master'

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git branch

js-assignments

\* master

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

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git merge js-assignments

On branch master

All conflicts fixed but you are still merging.

(use "git commit" to conclude merge)

Changes to be committed:

new file: js1.js

new file: js2.js

roneve@LIN24009020 MINGW64 ~/assignments (master|MERGING)

$ git commit -m "All Changes Committed"

[master dd72d74] All Changes Committed

Finally delete the ‘js-assignments’ branch.

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git branch -D js-assignments

Deleted branch js-assignments (was bc3fc3f).

roneve@LIN24009020 MINGW64 ~/assignments (master)

$ git branch

\* master