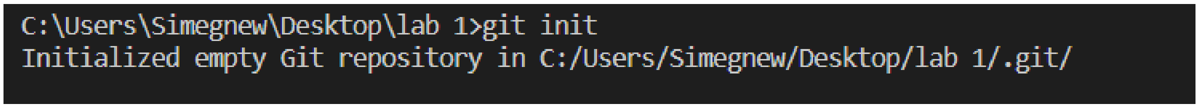
Lab Solutions

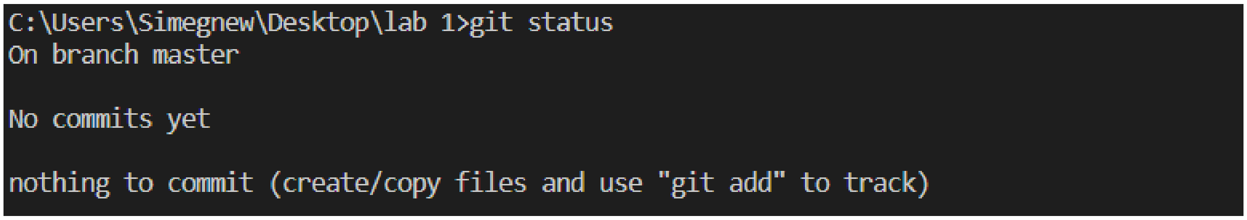
Part 1

1, git init (used to generate. git hidden folder)



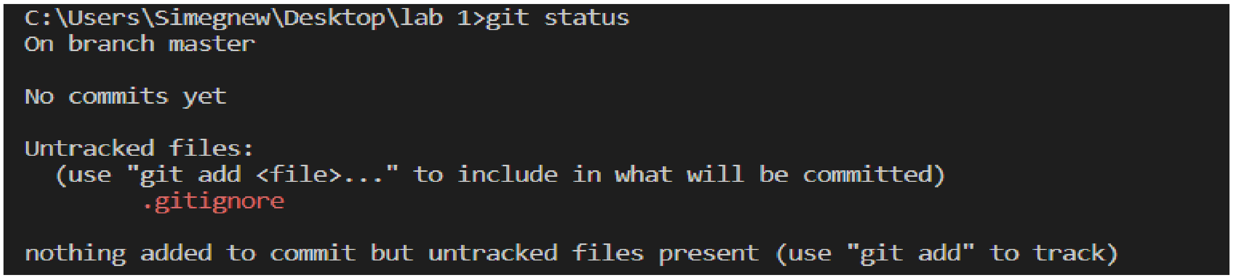
```
C:\Users\Simegnew\Desktop\lab 1>git init
Initialized empty Git repository in C:/Users/Simegnew/Desktop/lab 1/.git/
```

2, git status (used to check all changes directory)



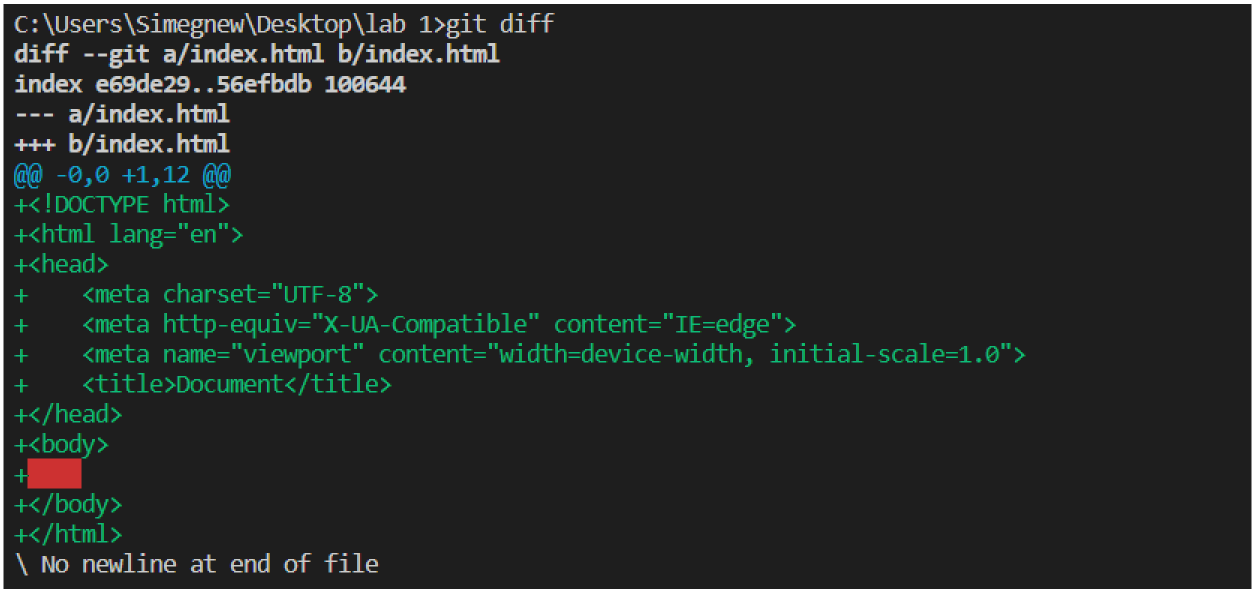
```
C:\Users\Simegnew\Desktop\lab 1>git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
```

Or



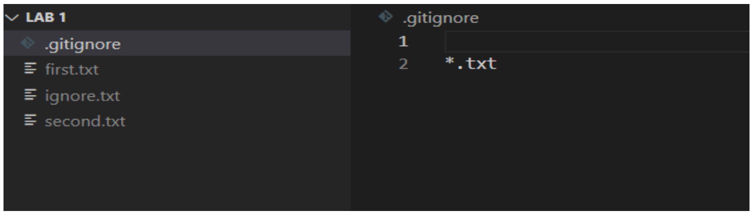
```
C:\Users\Simegnew\Desktop\lab 1>git status
On branch master

No commits yet

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)
```

3, git diff (used to show the difference between before some change and after change)

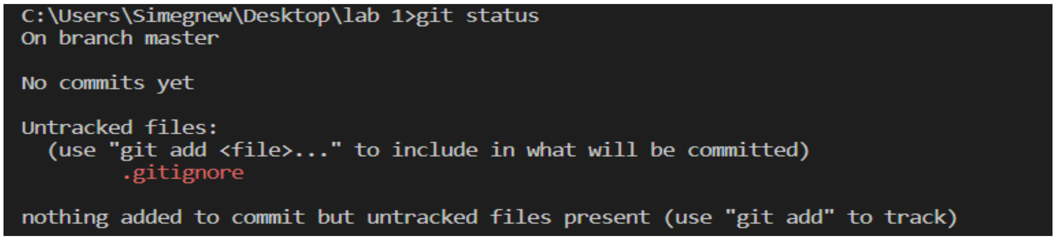


```
C:\Users\Simegnew\Desktop\lab 1>git diff
diff --git a/index.html b/index.html
index e69de29..56efbdb 100644
--- a/index.html
+++ b/index.html
@@ -0,0 +1,12 @@
+<!DOCTYPE html>
+<html lang="en">
+<head>
+    <meta charset="UTF-8">
+    <meta http-equiv="X-UA-Compatible" content="IE=edge">
+    <meta name="viewport" content="width=device-width, initial-scale=1.0">
+    <title>Document</title>
+</head>
+<body>
+    
+</body>
+</html>
\ No newline at end of file
```

4, Create `.gitignore` file, and notice how it works(used to add the file you don't went to add)

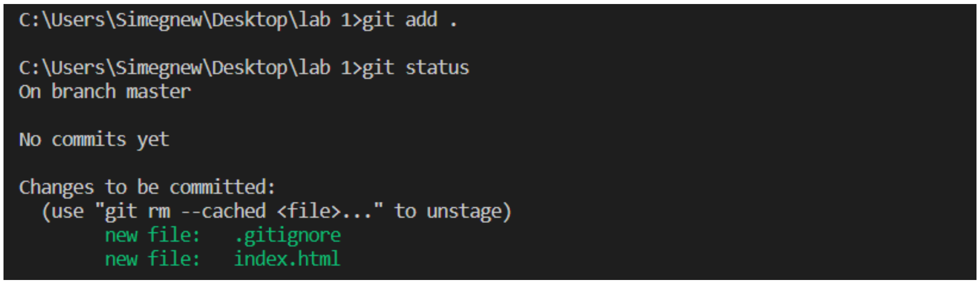


```
1
2    *.txt
```

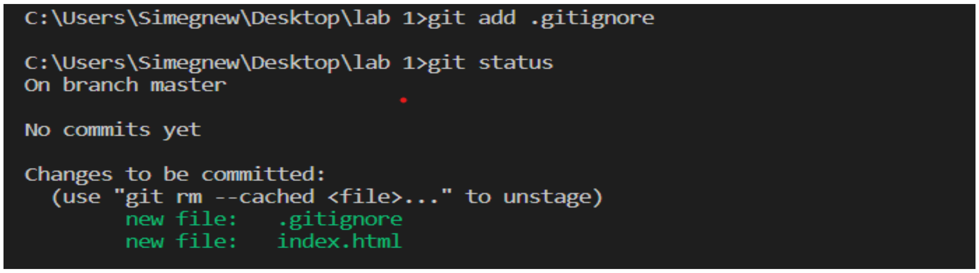


```
C:\Users\Simegnew\Desktop\lab 1>git status
On branch master

No commits yet

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)
```

5,git add (add file from working area to staging area)

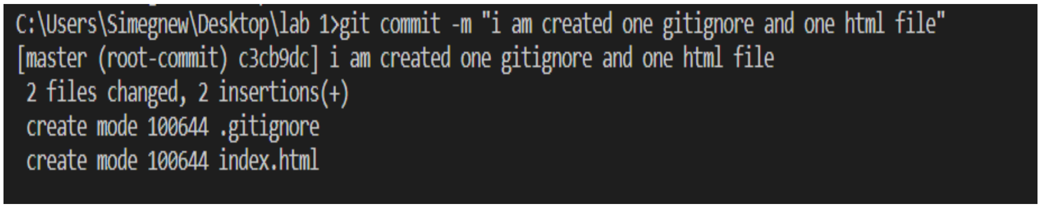


```
C:\Users\Simegnew\Desktop\lab 1>git add .

C:\Users\Simegnew\Desktop\lab 1>git status
On branch master

No commits yet

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



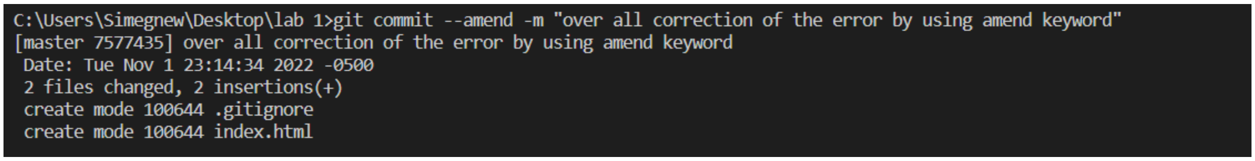
```
C:\Users\Simegnew\Desktop\lab 1>git add .gitignore

C:\Users\Simegnew\Desktop\lab 1>git status
On branch master

No commits yet

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

6, git commit (commit file from the staging to the repository)



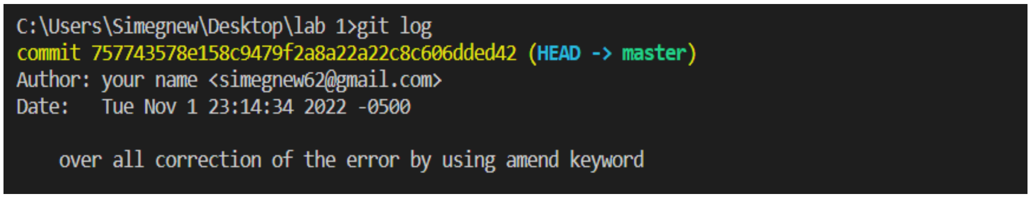
```
C:\Users\Simegnew\Desktop\lab 1>git commit -m "i am created one gitignore and one html file"
[master (root-commit) c3cb9dc] i am created one gitignore and one html file
 2 files changed, 2 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 index.html
```

7, git commit -amend (used to updating the last commit)

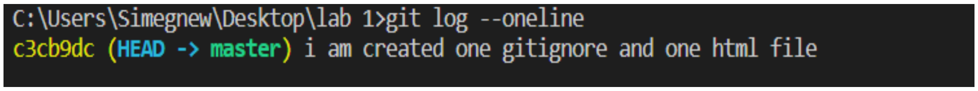


```
C:\Users\Simegnew\Desktop\lab 1>git commit --amend -m "over all correction of the error by using amend keyword"
[master 7577435] over all correction of the error by using amend keyword
 Date: Tue Nov 1 23:14:34 2022 -0500
 2 files changed, 2 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 index.html
```

8, git log (used to show all action and change in local repository)

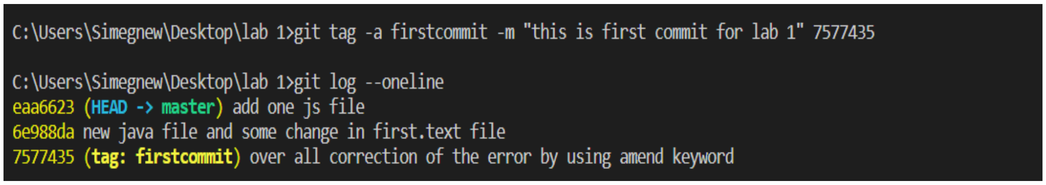


```
C:\Users\Simegnew\Desktop\lab 1>git log
commit 757743578e158c9479f2a8a22a22c8c606dded42 (HEAD -> master)
Author: your name <simegnew62@gmail.com>
Date:   Tue Nov 1 23:14:34 2022 -0500

    over all correction of the error by using amend keyword
```



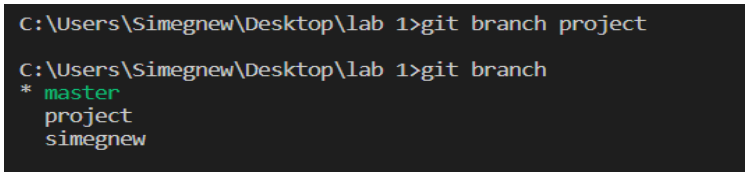
```
C:\Users\Simegnew\Desktop\lab 1>git log --oneline
c3cb9dc (HEAD -> master) i am created one gitignore and one html file
```

9, git tag(used to give tag and used this tag for checkout simply by name)



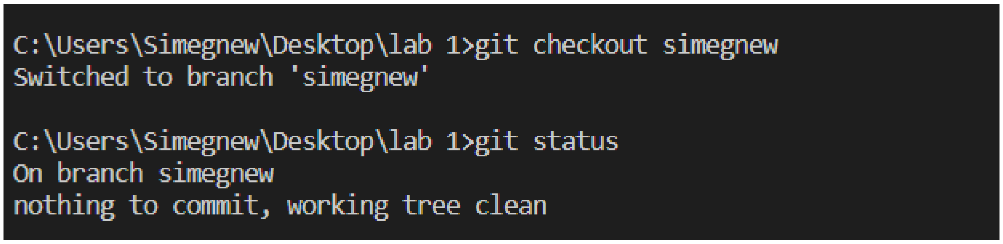
```
C:\Users\Simegnew\Desktop\lab 1>git tag -a firstcommit -m "this is first commit for lab 1" 7577435

C:\Users\Simegnew\Desktop\lab 1>git log --oneline
eaa6623 (HEAD -> master) add one js file
6e988da new java file and some change in first.text file
7577435 (tag: firstcommit) over all correction of the error by using amend keyword
```

10, git branch (use to show the status of repository in which branch are you work)



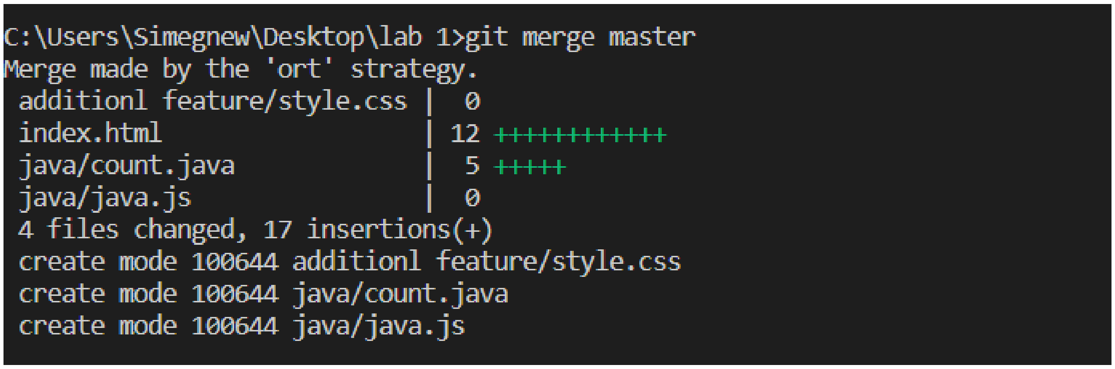
```
C:\Users\Simegnew\Desktop\lab 1>git branch project

C:\Users\Simegnew\Desktop\lab 1>git branch
* master
  project
  simegnew
```

11, git checkout (used to switch between different version and branches)



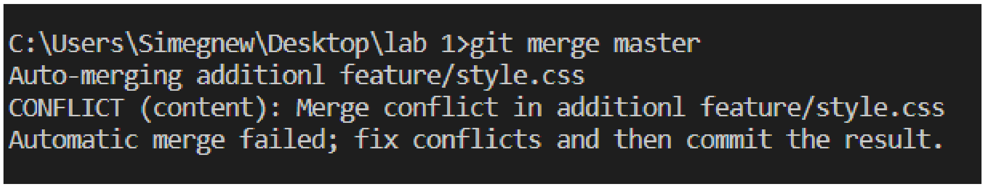
```
C:\Users\Simegnew\Desktop\lab 1>git checkout simegnew
Switched to branch 'simegnew'

C:\Users\Simegnew\Desktop\lab 1>git status
On branch simegnew
nothing to commit, working tree clean
```

12, git merge (without conflict used to merge two branch code in one)

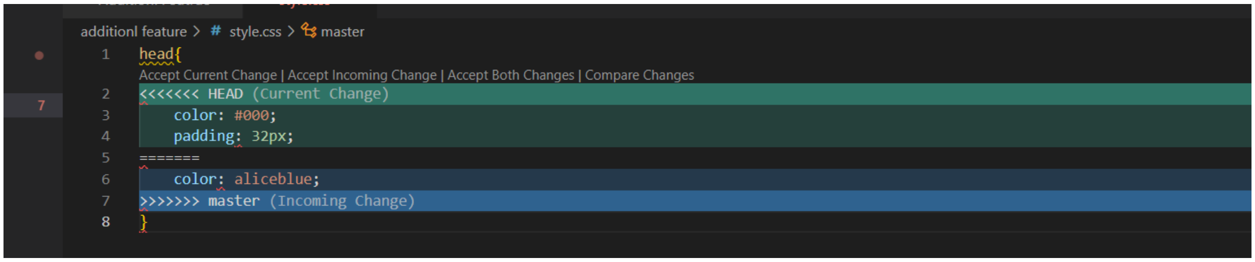


```
C:\Users\Simegnew\Desktop\lab 1>git merge master
Merge made by the 'ort' strategy.
 additionl feature/style.css |  0
 index.html                  | 12 ++++++++++++
 java/count.java             |  5 +++++
 java/java.js                |  0
 4 files changed, 17 insertions(+)
 create mode 100644 additionl feature/style.css
 create mode 100644 java/count.java
 create mode 100644 java/java.js
```

i. Create a conflict and solve the conflict (conflict happen when the exact same line is changed in separate branches.)



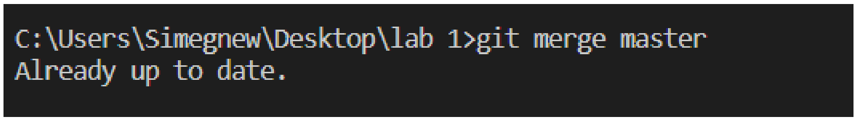
```
C:\Users\Simegnew\Desktop\lab 1>git merge master
Auto-merging additionl feature/style.css
CONFLICT (content): Merge conflict in additionl feature/style.css
Automatic merge failed; fix conflicts and then commit the result.
```



additionl feature > # style.css > master
Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes

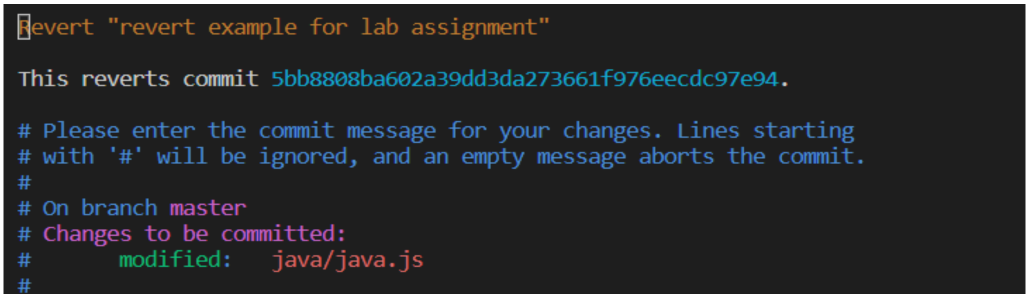
```
head{
<<<<<<< HEAD (Current Change)
    color: #000;
    padding: 32px;
=======
    color: aliceblue;
>>>>>>> master (Incoming Change)
}
```

Solve conflict (remove manually special characters after that if you went to approve the change just save next add and commit and finally merge)

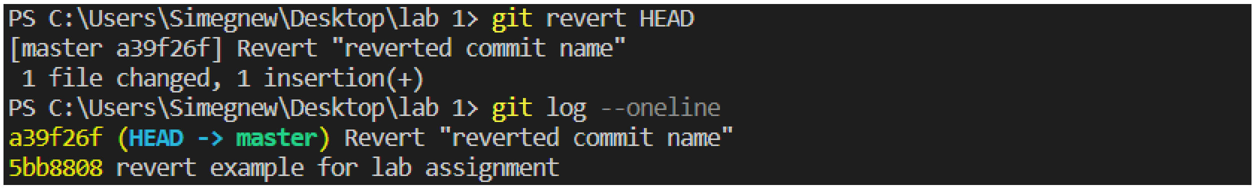


```
C:\Users\Simegnew\Desktop\lab 1>git merge master
Already up to date.
```

13, Undo changes from a certain commit `git revert( used to update the latest commit)`

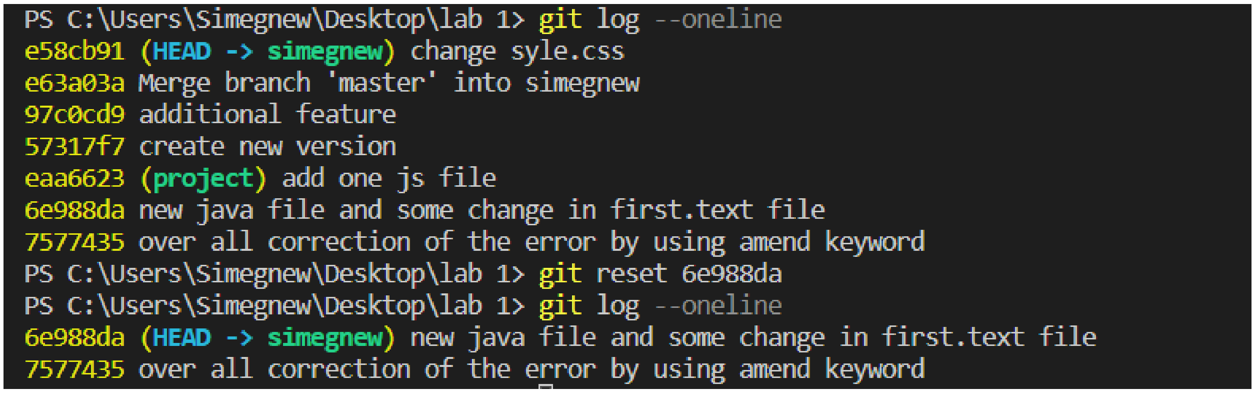


```
Revert "revert example for lab assignment"

This reverts commit 5bb8808ba602a39dd3da273661f976eecdc97e94.

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# Changes to be committed:
#       modified:   java/java.js
#
```



```
PS C:\Users\Simegnew\Desktop\lab 1> git revert HEAD
[master a39f26f] Revert "reverted commit name"
 1 file changed, 1 insertion(+)
PS C:\Users\Simegnew\Desktop\lab 1> git log --oneline
a39f26f (HEAD -> master) Revert "reverted commit name"
5bb8808 revert example for lab assignment
```

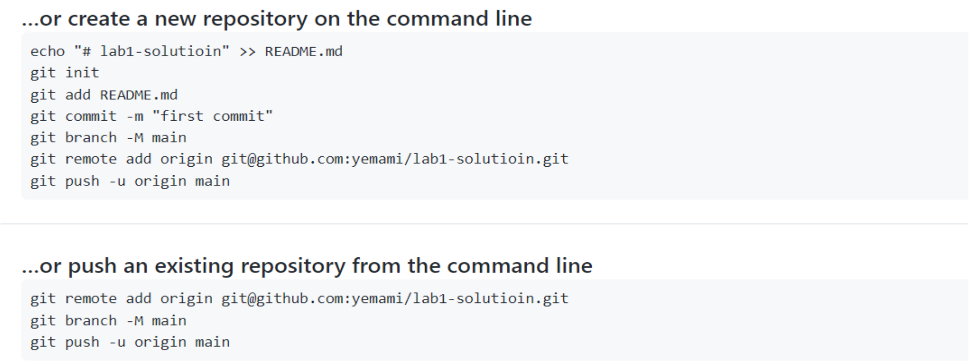
14, Erase commits with `git reset (use to remove the previous commit)`



```
PS C:\Users\Simegnew\Desktop\lab 1> git log --oneline
e58cb91 (HEAD -> simegnew) change syle.css
e63a03a Merge branch 'master' into simegnew
97c0cd9 additional feature
57317f7 create new version
eaa6623 (project) add one js file
6e988da new java file and some change in first.text file
7577435 over all correction of the error by using amend keyword
PS C:\Users\Simegnew\Desktop\lab 1> git reset 6e988da
PS C:\Users\Simegnew\Desktop\lab 1> git log --oneline
6e988da (HEAD -> simegnew) new java file and some change in first.text file
7577435 over all correction of the error by using amend keyword
```

Part 2

1, Create a remote repository



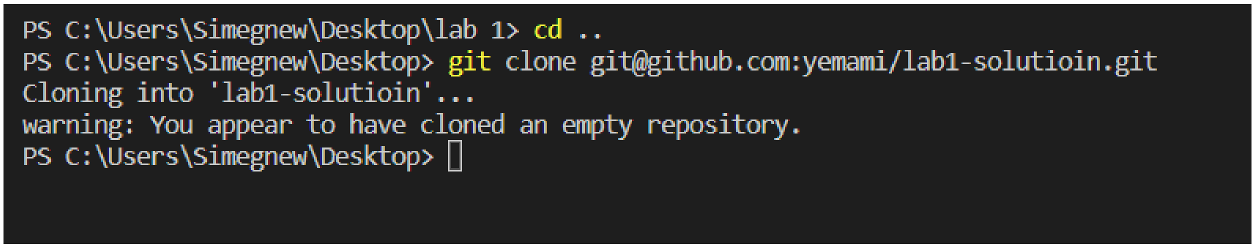
...or create a new repository on the command line

```
echo "# lab1-solutioin" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin git@github.com:yemami/lab1-solutioin.git
git push -u origin main
```

...or push an existing repository from the command line

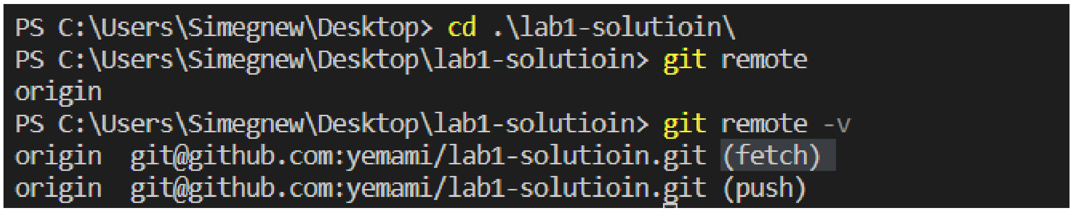
```
git remote add origin git@github.com:yemami/lab1-solutioin.git
git branch -M main
git push -u origin main
```

2, git clone (used to copy the repository to locally)



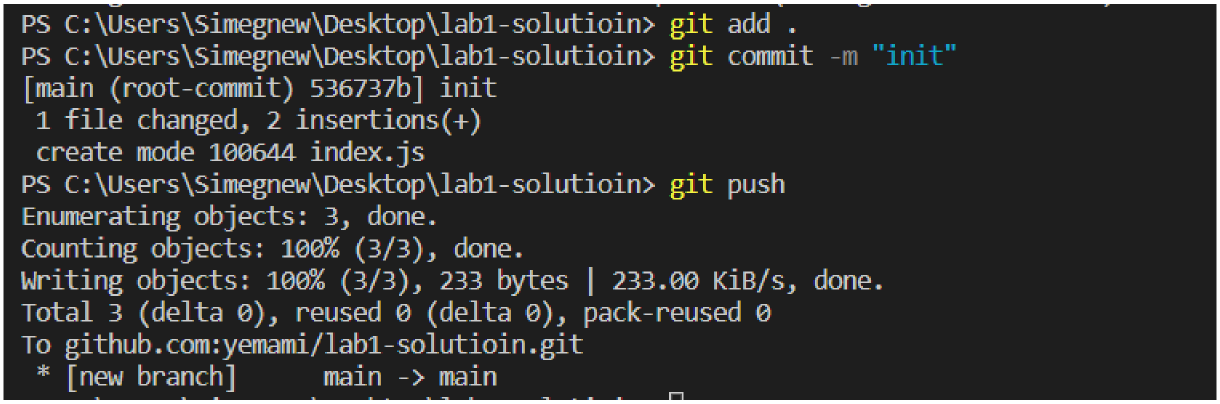
```
PS C:\Users\Simegnew\Desktop\lab 1> cd ..
PS C:\Users\Simegnew\Desktop> git clone git@github.com:yemami/lab1-solutioin.git
Cloning into 'lab1-solutioin'...
warning: You appear to have cloned an empty repository.
PS C:\Users\Simegnew\Desktop> 
```

3, git remote



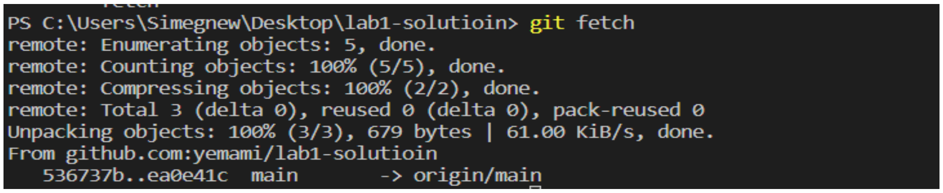
```
PS C:\Users\Simegnew\Desktop> cd .\lab1-solutioin\
PS C:\Users\Simegnew\Desktop\lab1-solutioin> git remote
origin
PS C:\Users\Simegnew\Desktop\lab1-solutioin> git remote -v
origin  git@github.com:yemami/lab1-solutioin.git (fetch)
origin  git@github.com:yemami/lab1-solutioin.git (push)
```

4, git push



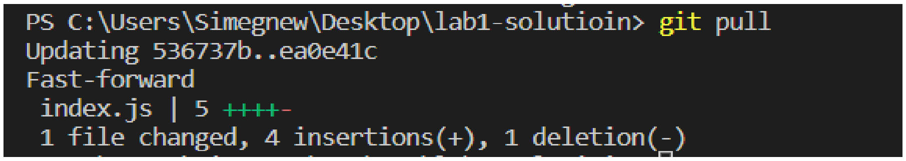
```
PS C:\Users\Simegnew\Desktop\lab1-solutioin> git add .
PS C:\Users\Simegnew\Desktop\lab1-solutioin> git commit -m "init"
[main (root-commit) 536737b] init
 1 file changed, 2 insertions(+)
 create mode 100644 index.js
PS C:\Users\Simegnew\Desktop\lab1-solutioin> git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 233 bytes | 233.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:yemami/lab1-solutioin.git
 * [new branch]      main -> main
```

5, git fetch



```
PS C:\Users\Simegnew\Desktop\lab1-solutioin> git fetch
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 679 bytes | 61.00 KiB/s, done.
From github.com:yemami/lab1-solutioin
   536737b..ea0e41c  main       -> origin/main
```

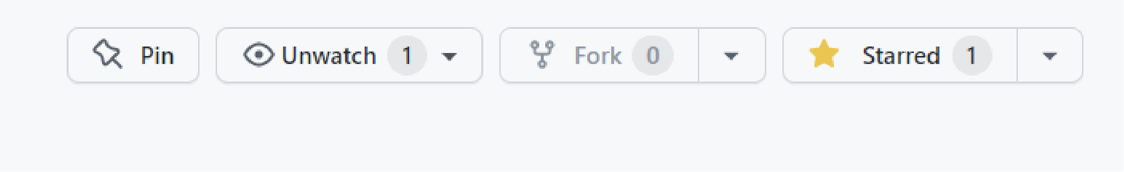
6, git pull (the result of git fetch plus git merge)



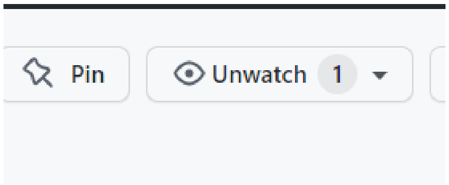
```
PS C:\Users\Simegnew\Desktop\lab1-solutioin> git pull
Updating 536737b..ea0e41c
Fast-forward
 index.js | 5 ++++-
 1 file changed, 4 insertions(+), 1 deletion(-)
```

Part 3

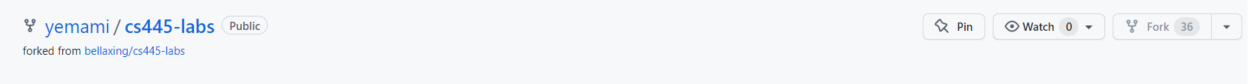
1, Star the repository (when like repository)



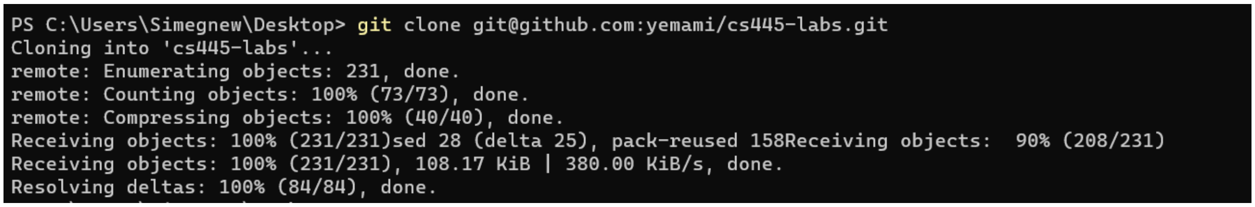
2, Watch the repository for any changes (when any change in the repository got notification message)



3, Fork the repository (use to copy repository)

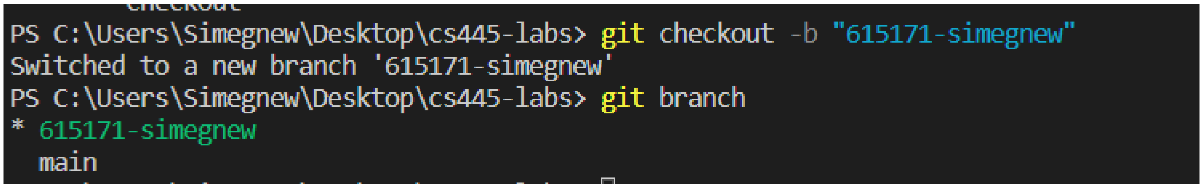

yemami / cs445-labs Public
forked from bellaxing/cs445-labs

4, Clone your forked repository into your local machine



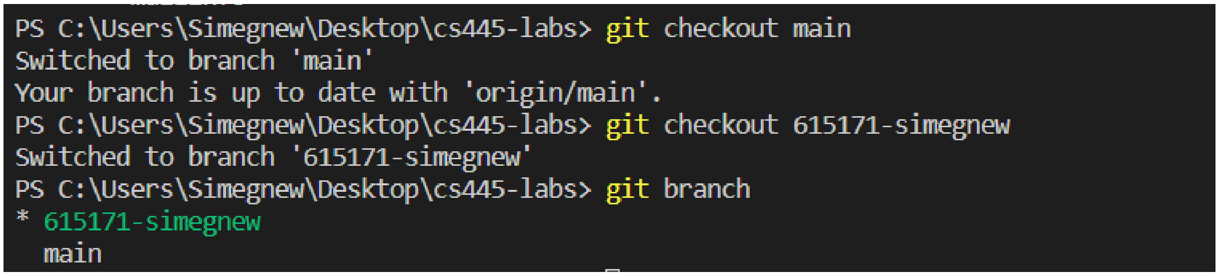
```
PS C:\Users\Simegnew\Desktop> git clone git@github.com:yemami/cs445-labs.git
Cloning into 'cs445-labs'...
remote: Enumerating objects: 231, done.
remote: Counting objects: 100% (73/73), done.
remote: Compressing objects: 100% (40/40), done.
Receiving objects: 100% (231/231)sed 28 (delta 25), pack-reused 158Receiving objects:  90% (208/231)
Receiving objects: 100% (231/231), 108.17 KiB | 380.00 KiB/s, done.
Resolving deltas: 100% (84/84), done.
```

5, Create a new topic branch using `studentId-studentName` format, For example: 610001-JohnSmith



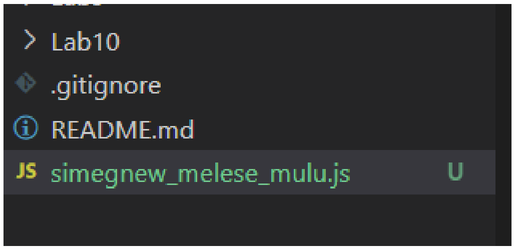
```
PS C:\Users\Simegnew\Desktop\cs445-labs> git checkout -b "615171-simegnew"
Switched to a new branch '615171-simegnew'
PS C:\Users\Simegnew\Desktop\cs445-labs> git branch
* 615171-simegnew
  main
```

6, Checkout your new branch



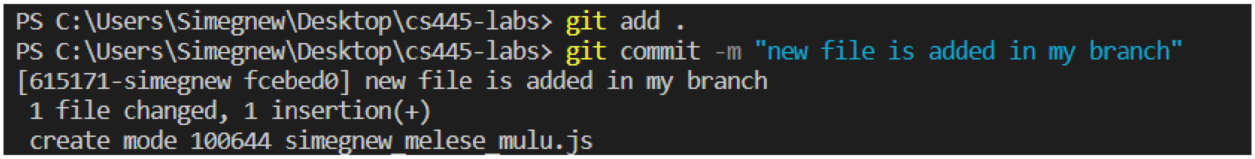
```
PS C:\Users\Simegnew\Desktop\cs445-labs> git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
PS C:\Users\Simegnew\Desktop\cs445-labs> git checkout 615171-simegnew
Switched to branch '615171-simegnew'
PS C:\Users\Simegnew\Desktop\cs445-labs> git branch
* 615171-simegnew
  main
```

7, Add a new file with your name, example: `hello.js`



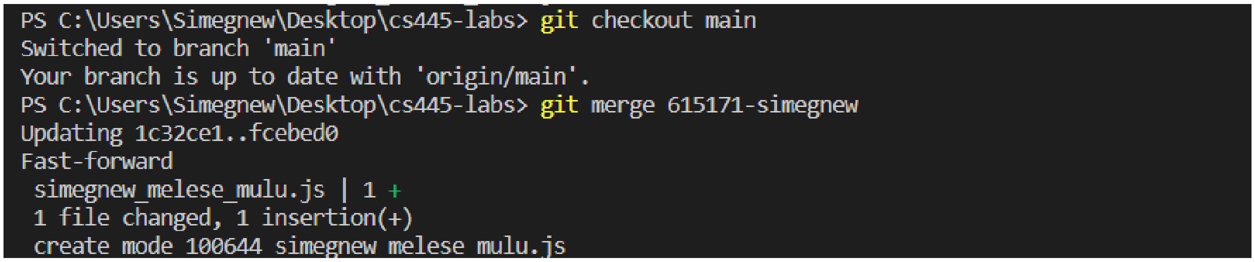
```
> Lab10
  .gitignore
  README.md
  simegnew_melese_mulu.js    U
```

8, Commit the changes



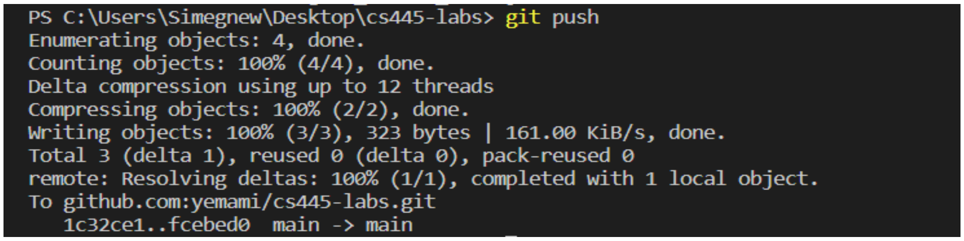
```
PS C:\Users\Simegnew\Desktop\cs445-labs> git add .
PS C:\Users\Simegnew\Desktop\cs445-labs> git commit -m "new file is added in my branch"
[615171-simegnew fcebed0] new file is added in my branch
 1 file changed, 1 insertion(+)
 create mode 100644 simegnew_melese_mulu.js
```

9, Merge your `studentId-studentName` branch with the main branch



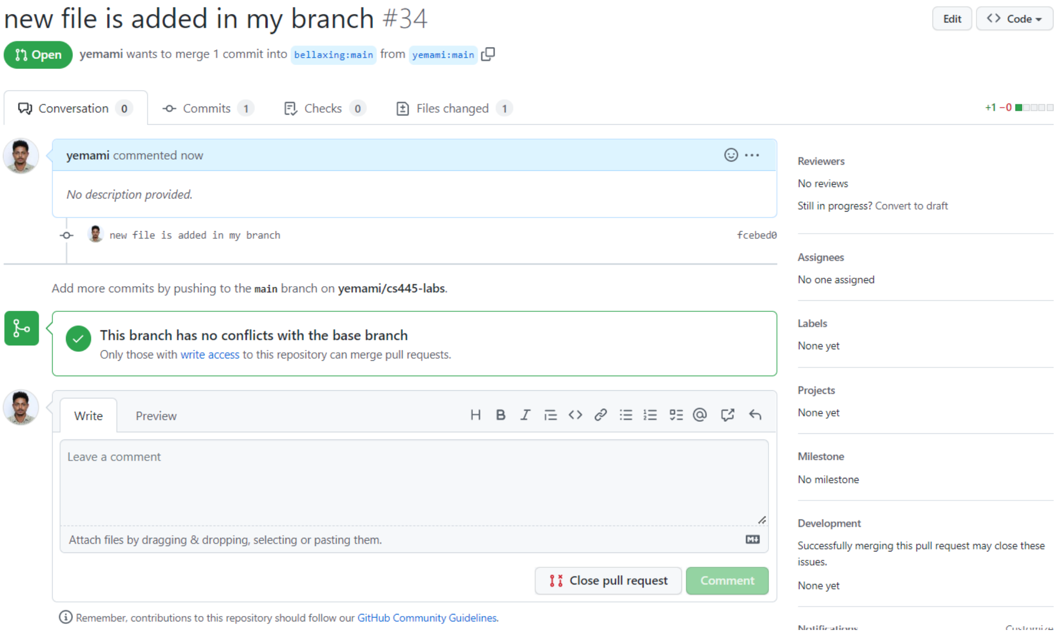
```
PS C:\Users\Simegnew\Desktop\cs445-labs> git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
PS C:\Users\Simegnew\Desktop\cs445-labs> git merge 615171-simegnew
Updating 1c32ce1..fcebed0
Fast-forward
 simegnew_melese_mulu.js | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 simegnew_melese_mulu.js
```

10, Push your changes to the remote repo



```
PS C:\Users\Simegnew\Desktop\cs445-labs> git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 323 bytes | 161.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:yemami/cs445-labs.git
   1c32ce1..fcebed0  main -> main
```

11, Create a new Pull Request to the upstream repo



new file is added in my branch #34

Open yemami wants to merge 1 commit into `bellaxing:main` from `yemami:main`

yemami commented now

No description provided.

`new file is added in my branch` fcebed0

Add more commits by pushing to the `main` branch on yemami/cs445-labs.

This branch has no conflicts with the base branch

Only those with write access to this repository can merge pull requests.

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

Remember, contributions to this repository should follow our GitHub Community Guidelines.

Reviewers

No reviews

Still in progress? Convert to draft

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request may close these issues.

None yet

Notifications Customize