

Lab1 Solutions

1. git init

```
mahlethonelign@Mahlets-MacBook-Pro lab1 % git init
hint: Using 'master' as the name for the initial branch. This is the default branch name
hint: 'master' is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /Users/mahlethonelign/Desktop/lab1/.git/
```

2. git status

```
mahlethonelign@Mahlets-MacBook-Pro lab1 % git status
On branch master

No commits yet

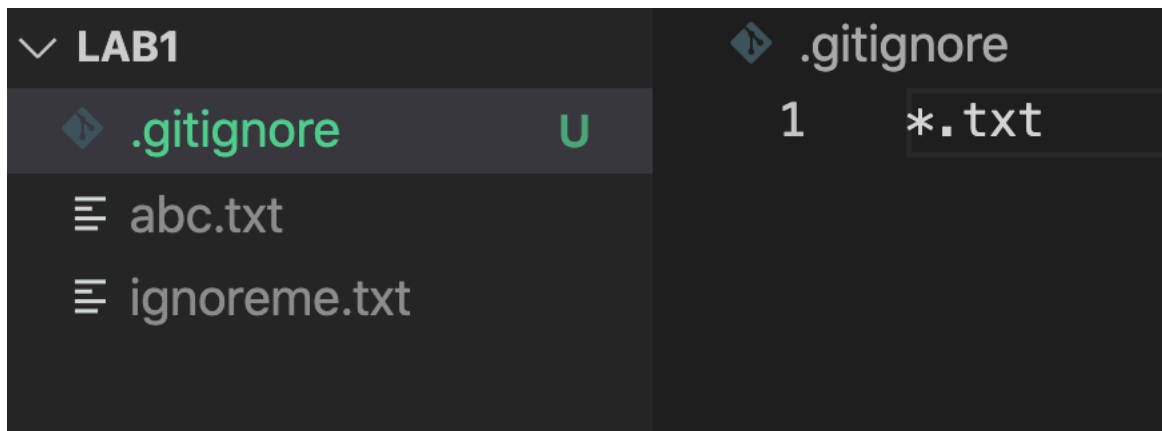
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .gitignore

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

3. git diff

```
mahlethonelign@Mahlets-MacBook-Pro lab1 % git diff .gitignore
diff --git a/.gitignore b/.gitignore
index 314f02b..b92c542 100644
--- a/.gitignore
+++ b/.gitignore
@@ -1,2 @@
-*.txt
\ No newline at end of file
+*.txt
+java
\ No newline at end of file
```

4. Create .gitignore file, and notice how it works



5. git add

```
mahlethonelign@Mahlets-MacBook-Pro lab1 % git add .
mahlethonelign@Mahlets-MacBook-Pro lab1 % git status
On branch master

No commits yet

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

6. git commit

```
mahlethonelign@Mahlets-MacBook-Pro lab1 % git commit -m "added gitignore file"
[master (root-commit) 16308f2] added gitignore file
 1 file changed, 1 insertion(+)
 create mode 100644 .gitignore
mahlethonelign@Mahlets-MacBook-Pro lab1 % git status
On branch master
nothing to commit, working tree clean
```

7. git commit --amend (updating the last commit)

```
added gitignore file
mahlethonelign@Mahlets-MacBook-Pro lab1 % git commit --amend -m "amended gitignore init commit"
[master 67f6ddf] amended gitignore init commit
Date: Mon Oct 25 23:46:25 2021 -0500
 1 file changed, 1 insertion(+)
 create mode 100644 .gitignore
```

8. git log

```
mahlethonelign@Mahlets-MacBook-Pro lab1 % git log
commit 67f6ddf83b45c03d2a8eac8b49267a9f4a8cfa10 (HEAD -> master)
Author: Mahlet Honelign <mhoneign@miu.edu>
Date: Mon Oct 25 23:46:25 2021 -0500

    amended gitignore init commit
```

9. git tag

```
mahlethonelign@Mahlets-MacBook-Pro lab1 % git tag -a firstcommit -m "first commit" 5d76dea
mahlethonelign@Mahlets-MacBook-Pro lab1 % git log --oneline
d8bfb47 (HEAD -> master) hello.js 3
8f2de8d hello.js init commit
5d76dea (tag: firstcommit) hello.js init commit
f76b797 ignore java folder
67f6ddf amended gitignore init commit
```

10.git branch

```
mahlethoneign@Mahlets-MacBook-Pro lab1 % git branch new-branch
mahlethoneign@Mahlets-MacBook-Pro lab1 % git branch
* master
  new-branch
  shoppingcart
  user-manager
```

11.git checkout

```
mahlethoneign@Mahlets-MacBook-Pro lab1 % git checkout shoppingcart
Switched to branch 'shoppingcart'
mahlethoneign@Mahlets-MacBook-Pro lab1 % git checkout master
Switched to branch 'master'
```

12.git merge

```
mahlethoneign@Mahlets-MacBook-Pro lab1 % git merge new-branch
Updating 88e74e4..5dd9356
Fast-forward
 newfile.js | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 newfile.js
```

i. Create a conflict and solve the conflict

```
mahlethoneign@Mahlets-MacBook-Pro lab1 % git checkout master
Switched to branch 'master'
mahlethoneign@Mahlets-MacBook-Pro lab1 % git merge shoppingcart

Auto-merging shoppingcart.js
CONFLICT (content): Merge conflict in shoppingcart.js
Automatic merge failed; fix conflicts and then commit the result
.
mahlethoneign@Mahlets-MacBook-Pro lab1 % git add .
mahlethoneign@Mahlets-MacBook-Pro lab1 % git commit -m "solved
conflict"
[master 38f3539] solved conflict
mahlethoneign@Mahlets-MacBook-Pro lab1 % git merge shoppingcart

Already up to date.
```

13. Undo changes from a certain commit `git revert`

```
mahlethonelign@Mahlets-MacBook-Pro lab1 % git log --oneline
a614ecb (HEAD -> master) revert 2
bce7a84 new commit
ab88e56 to be reverted
38f3539 solved conflict
1e4d42b (shoppingcart) change commit
2dc55c8 changes on master
5dd9356 (new-branch) new file
88e74e4 user commit
b52f6d2 (user-manager) user commit
4fcc83a another commit
aa73e2b init commit
a325a3a new commit
d8bfb47 hello.js 3
8f2de8d hello.js init commit
5d76dea (tag: firstcommit) hello.js init commit
f76b797 ignore java folder
67f6ddf amended gitignore init commit
mahlethonelign@Mahlets-MacBook-Pro lab1 % git revert a614ecb
```

14. Erase commits with `git reset`

```
mahlethonelign@Mahlets-MacBook-Pro lab1 % git log --oneline
1e4d42b (HEAD -> master, shoppingcart) change commit
aa73e2b init commit
d8bfb47 hello.js 3
8f2de8d hello.js init commit
5d76dea (tag: firstcommit) hello.js init commit
f76b797 ignore java folder
67f6ddf amended gitignore init commit
mahlethonelign@Mahlets-MacBook-Pro lab1 % git reset 8f2de8d
Unstaged changes after reset:
M      hello.js
mahlethonelign@Mahlets-MacBook-Pro lab1 % git log --oneline
8f2de8d (HEAD -> master) hello.js init commit
5d76dea (tag: firstcommit) hello.js init commit
f76b797 ignore java folder
67f6ddf amended gitignore init commit
```


Practice with all the remote repository commands from the slides:

1. Create a remote repository

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Owner *

 amayah2021 ▾

Repository name *

/

Great repository names are short and memorable. Need inspiration? How about **super-doodle**?

Description (optional)



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☐ **Add a README file**

This is where you can write a long description for your project. [Learn more](#).

☐ **Add .gitignore**

Choose which files not to track from a list of templates. [Learn more](#).

☐ **Choose a license**

A license tells others what they can and can't do with your code. [Learn more](#).

Create repository

2. git clone

```
mahlethonelign@Mahlets-MacBook-Pro CS445-labs % git clone https://github.com/amayah2021/CS445-labs.git
Cloning into 'CS445-labs'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (4/4), done.
```

3. git remote

```
mahlethonelign@Mahlets-MacBook-Pro CS445-labs % git remote origin
```

4. git push

```
mahlethonelign@Mahlets-MacBook-Pro CS445-labs % git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 284 bytes | 284.00 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 https://github.com/amayah2021/CS445-labs.git
    b429741..7a8a7db  main -> main
```

5. git fetch

```
mahlethonelign@Mahlets-MacBook-Pro CS445-labs % 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), 712 bytes | 237.00 KiB/s, done.
From https://github.com/amayah2021/CS445-labs
    7a8a7db..fbc0d6a  main      -> origin/main
```

6. git pull

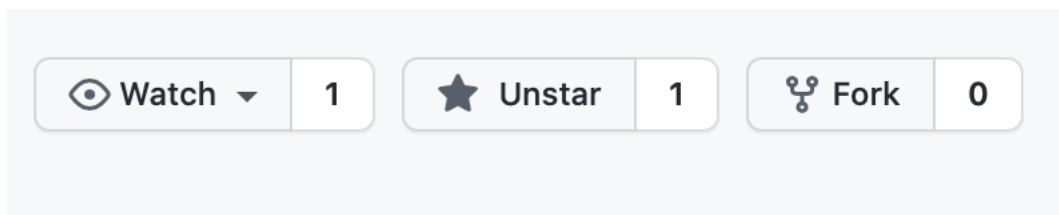
```
mahlethonelign@Mahlets-MacBook-Pro CS445-labs % git pull
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 715 bytes | 357.00 KiB/s, done.
From https://github.com/amayah2021/CS445-labs
   fbc0d6a..2f18116  main       -> origin/main
Updating fbc0d6a..2f18116
Fast-forward
 README.md | 1 +
 1 file changed, 1 insertion(+)
```

Practice with all the third party repositories commands from the slides:

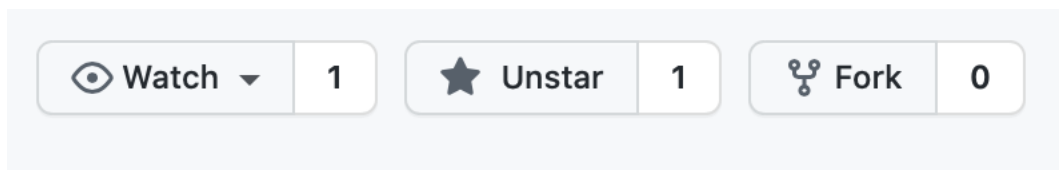
Assume we are working with the following

upstream: <https://github.com/bellaxing/cs445-labs>

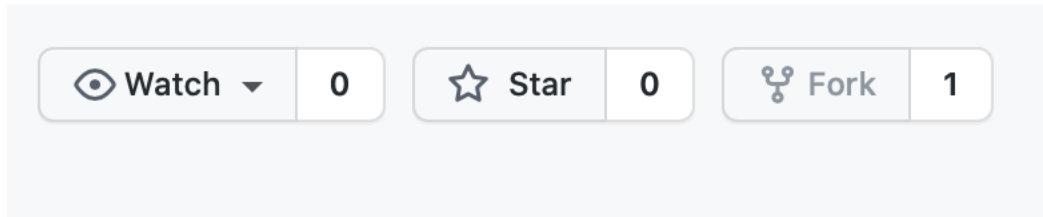
1. Star the repository



2. Watch the repository for any changes



3. Fork the repository



4. Clone your forked repository into your local machine

<https://github.com/mahlethon/CS445-labs.git>

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

```
mahlethonelign@Mahlets-MacBook-Pro mahlethon.github.io % git branch 112174-Honelign
mahlethonelign@Mahlets-MacBook-Pro mahlethon.github.io % git branch
112174-Honelign
* main
```

6. Checkout your new branch

```
mahlethonelign@Mahlets-MacBook-Pro mahlethon.github.io % git checkout 112174-Honelign
Switched to branch '112174-Honelign'
```

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

A dark rectangular box containing the text 'JS mahlet.js' in a light-colored font. The 'JS' is in a larger, bold font and the 'mahlet.js' is in a smaller font.

8. Commit the changes

```
mahlethonelign@Mahlets-MacBook-Pro mahlethon.github.io % git add .
mahlethonelign@Mahlets-MacBook-Pro mahlethon.github.io % git commit -m "changes in 112174-Honelign"
[112174-Honelign 4eee337] changes in 112174-Honelign
1 file changed, 1 insertion(+)
create mode 100644 mahlet.js
```

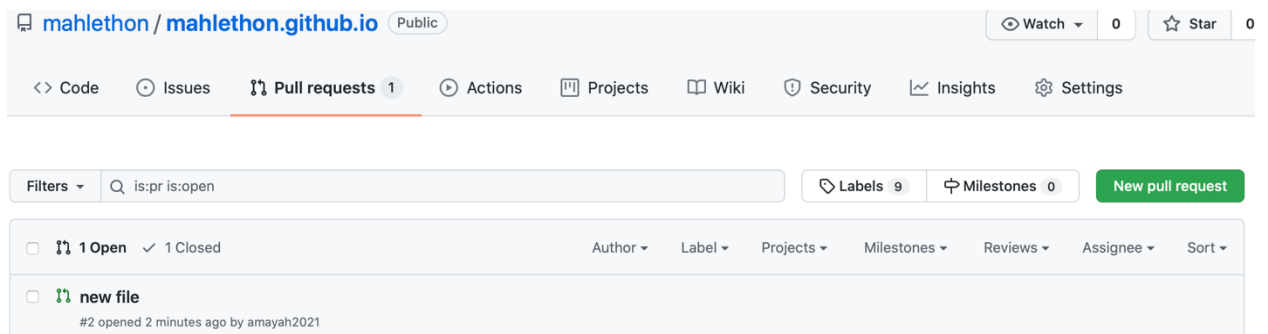
9. Merge your studentId-studentName branch with the main branch

```
mahlethonelign@Mahlets-MacBook-Pro mahlethon.github.io % git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
mahlethonelign@Mahlets-MacBook-Pro mahlethon.github.io % git merge 112174-Honelign
Updating 278fa0c..4eee337
Fast-forward
 mahlet.js | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 mahlet.js
```

10. Push your changes to the remote repo

```
mahlethonelign@Mahlets-MacBook-Pro mahlethon.github.io % 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), 294 bytes | 294.00 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 https://github.com/amayah2021/mahlethon.github.io.git
 278fa0c..4eee337  main -> main
```

11. Create a new Pull Request to the upstream repo



The screenshot shows the GitHub interface for the repository `mahlethon / mahlethon.github.io`. The `Pull requests` tab is selected, showing 1 pull request. The pull request is titled `new file` and was opened 2 minutes ago by `amayah2021`. The interface includes filters, search, and sorting options.

Repository: `mahlethon / mahlethon.github.io` (Public)

Navigation: `<> Code`, `Issues`, **`Pull requests 1`**, `Actions`, `Projects`, `Wiki`, `Security`, `Insights`, `Settings`

Filters: `is:pr is:open` | `Labels 9` | `Milestones 0` | [New pull request](#)

1 Open | 1 Closed

new file
#2 opened 2 minutes ago by amayah2021

What you need to submit:

Copy and paste your github link of this lab to Sakai.

The below is Optional homework if you're interested in. This is self study part, not going to lecture. If you have questions, please ask TA.

1. Create a remote repository
2. Create README.md
3. Add at least 5 markdown syntax (For example: Headers, Bold, Code, Unordered List, Links) [Cheatsheet](#)