

1. git init

```
Abels-MacBook-Pro:Github Asy abel$ git init
Reinitialized existing Git repository in /Users/abel/Desktop/Github Asy/.git/
Abels-MacBook-Pro:Github Asy abel$
```

2. git status

```
Abels-MacBook-Pro:Github Asy abel$ git status
On branch master
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    a.js

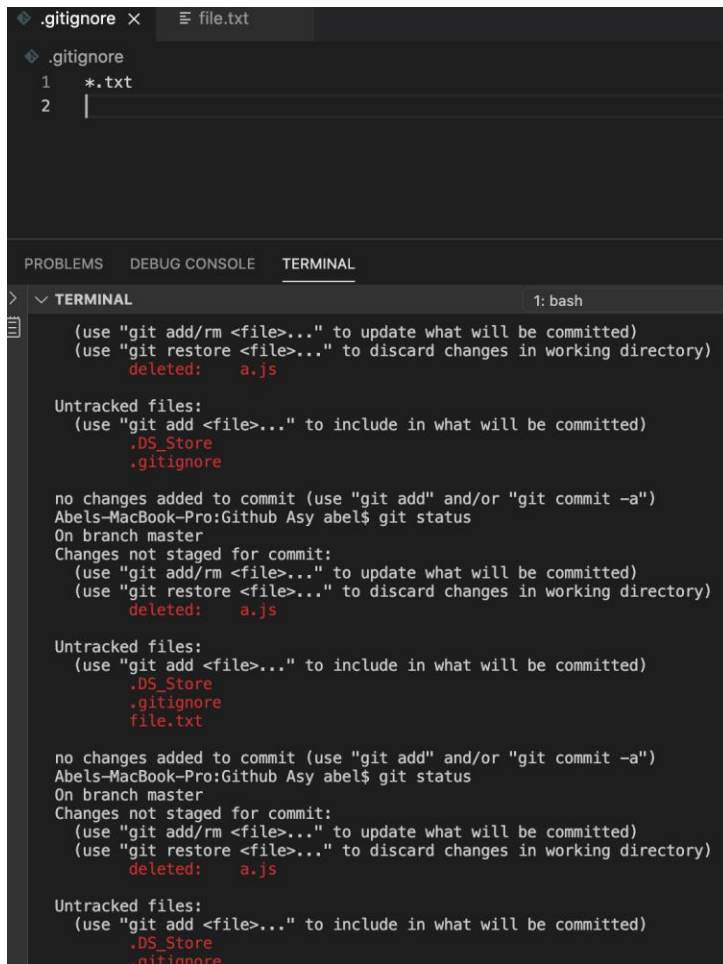
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .DS_Store
        .gitignore

no changes added to commit (use "git add" and/or "git commit -a")
Abels-MacBook-Pro:Github Asy abel$
```

3. git diff

```
Abels-MacBook-Pro:Github Asy abel$ git diff .gitignore
diff --git a/.gitignore b/.gitignore
index 2211df6..c6ab4a3 100644
--- a/.gitignore
+++ b/.gitignore
@@ -1,2 @@
 *.txt
+abel
\ No newline at end of file
```

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



The screenshot shows a code editor with a dark theme. At the top, there are two tabs: ".gitignore" and "file.txt". The ".gitignore" tab is active, showing a file with two lines: "1 *.txt" and "2 |". Below the editor, there is a panel with three tabs: "PROBLEMS", "DEBUG CONSOLE", and "TERMINAL". The "TERMINAL" tab is active, showing a bash shell session. The terminal output includes instructions for using "git add/rm" and "git restore" to update or discard changes, followed by the output of "git status" which shows untracked files (.DS_Store, .gitignore) and a deleted file (a.js).

```
.gitignore
1 *.txt
2 |

PROBLEMS  DEBUG CONSOLE  TERMINAL
> 1: bash

(use "git add/rm <file>..." to update what will be committed)
(use "git restore <file>..." to discard changes in working directory)
deleted:    a.js

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

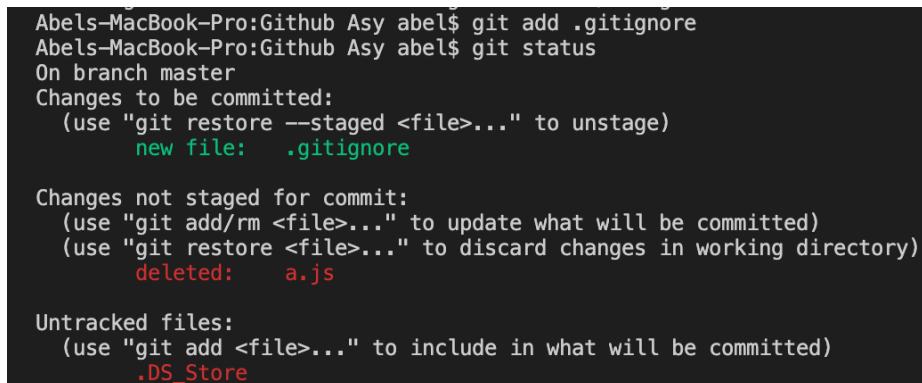
no changes added to commit (use "git add" and/or "git commit -a")
Abels-MacBook-Pro:Github Asy abel$ git status
On branch master
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
deleted:    a.js

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

no changes added to commit (use "git add" and/or "git commit -a")
Abels-MacBook-Pro:Github Asy abel$ git status
On branch master
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
deleted:    a.js

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

5. git add



The screenshot shows a terminal window with a dark background. The terminal output shows the execution of "git add .gitignore" and "git status". The output indicates that the file ".gitignore" is now staged for commit, and the file "a.js" is still deleted. The terminal also shows untracked files (.DS_Store) and instructions for using "git add/rm" and "git restore" to update or discard changes.

```
Abels-MacBook-Pro:Github Asy abel$ git add .gitignore
Abels-MacBook-Pro:Github Asy abel$ 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/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
deleted:    a.js

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

6. git commit

```
Abels-MacBook-Pro:Github Asy abel$ git commit -m ".gitignore file"
[master 50b74fe] .gitignore file
1 file changed, 1 insertion(+)
create mode 100644 .gitignore
```

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

```
Abels-MacBook-Pro:Github Asy abel$ git log --oneline
50b74fe (HEAD -> master) .gitignore file
3fa6279 check
f74be4d a
Abels-MacBook-Pro:Github Asy abel$ git commit --amend -m "amended"
[master 6cb9e91] amended
Date: Tue Oct 26 17:13:16 2021 -0500
1 file changed, 1 insertion(+)
create mode 100644 .gitignore
Abels-MacBook-Pro:Github Asy abel$ git log --oneline
6cb9e91 (HEAD -> master) amended
3fa6279 check
f74be4d a
Abels-MacBook-Pro:Github Asy abel$
```

8. git log

```
Abels-MacBook-Pro:Github Asy abel$ git log
commit 6cb9e91921b141cdeab864b0798b4e6d962aa941 (HEAD -> master)
Author: abelmicael <amicael@miu.edu>
Date: Tue Oct 26 17:13:16 2021 -0500

    amended

commit 3fa62796457e73f36b02fa73499b33f9ddd517f3
Author: abelmicael <amicael@miu.edu>
Date: Mon Oct 25 12:00:31 2021 -0500

    check

commit f74be4d08b817dd378f450cf2aea626be2879c5d
Author: abelmicael <amicael@miu.edu>
Date: Mon Oct 25 11:41:02 2021 -0500

    a
```

9. git tag

```

Abels-MacBook-Pro:Github Asy abel$ git log --oneline
838eeba (HEAD -> master) java.js init commit
b5d7ea1 (tag: abel) hello java
6cb9e91 amended
3fa6279 check
f74be4d a
Abels-MacBook-Pro:Github Asy abel$ 

```

10. git branch

```

no changes added to commit (use "git add" and/or "git commit -a")
Abels-MacBook-Pro:Github Asy abel$ git add .
Abels-MacBook-Pro:Github Asy abel$ git log --oneline
838eeba (HEAD -> loveJS, master, loveJava) java.js init commit
b5d7ea1 (tag: abel) hello java
6cb9e91 amended
3fa6279 check
f74be4d a
Abels-MacBook-Pro:Github Asy abel$ git add .
Abels-MacBook-Pro:Github Asy abel$ git commit -m "commit loveJS"
[loveJS 5e13991] commit loveJS
3 files changed, 1 insertion(+), 1 deletion(-)
create mode 100644 .DS_Store
create mode 100644 JS.js
delete mode 100644 a.js
Abels-MacBook-Pro:Github Asy abel$ git log --oneline
5e13991 (HEAD -> loveJS) commit loveJS
838eeba (master, loveJava) java.js init commit
b5d7ea1 (tag: abel) hello java
6cb9e91 amended
3fa6279 check
f74be4d a
Abels-MacBook-Pro:Github Asy abel$ 

```

11. git checkout

```

Abels-MacBook-Pro:Github Asy abel$ git checkout
Abels-MacBook-Pro:Github Asy abel$ git checkout 6cb9e91
Note: switching to '6cb9e91'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 6cb9e91 amended
Abels-MacBook-Pro:Github Asy abel$ git checkout master
Previous HEAD position was 6cb9e91 amended
Switched to branch 'master'

```

12. git merge

i. Create a conflict and solve the conflict

```

Abels-MacBook-Pro:Github Asy abel$ git checkout master
Switched to branch 'master'
Abels-MacBook-Pro:Github Asy abel$ git log --oneline
f2fb6b0 (HEAD -> master) new
838eeba java.js init commit
b5d7ea1 (tag: abel) hello java
6cb9e91 amended
3fa6279 check
f74be4d a
Abels-MacBook-Pro:Github Asy abel$ git merge loveJS
Auto-merging java.js
CONFLICT (content): Merge conflict in java.js
Removing a.js
Automatic merge failed; fix conflicts and then commit the result.
Abels-MacBook-Pro:Github Asy abel$ git add

```

```

Abels-MacBook-Pro:Github Asy abel$ git add .
Abels-MacBook-Pro:Github Asy abel$ git commit -m "new merge from loveJs"
[master f426f16] new merge from loveJs
Abels-MacBook-Pro:Github Asy abel$ git log --oneline
f426f16 (HEAD -> master) new merge from loveJs
7f25d31 (loveJS) ---
f2fb6b0 new
5e13991 commit loveJS
838eeba java.js init commit
b5d7ea1 (tag: abel) hello java
6cb9e91 amended
3fa6279 check
f74be4d a
Abels-MacBook-Pro:Github Asy abel$

```

13. Undo changes from a certain commit git revert

```

git: 'revert3fa6279' is not a git command. See 'git --help'.
Abels-MacBook-Pro:Github Asy abel$ git revert 3fa6279

```

14. Erase commits with git reset

```

Abels-MacBook-Pro:Github Asy abel$ git log --oneline
f426f16 (HEAD -> master) new merge from loveJs
7f25d31 (loveJS) ---
f2fb6b0 new
5e13991 commit loveJS
838eeba java.js init commit
b5d7ea1 (tag: abel) hello java
6cb9e91 amended
3fa6279 check
f74be4d a
Abels-MacBook-Pro:Github Asy abel$ git reset 5e13991
Unstaged changes after reset:
M      .DS_Store
M      java.js
Abels-MacBook-Pro:Github Asy abel$ git log --oneline
5e13991 (HEAD -> master) commit loveJS
838eeba java.js init commit
b5d7ea1 (tag: abel) hello java
6cb9e91 amended
3fa6279 check
f74be4d a
Abels-MacBook-Pro:Github Asy abel$

```

Create a remote repository

15. git clone

```
Abels-MacBook-Pro:git clone cs445 abel$ git clone https://github.com/abelmichael/CS445-course
Cloning into 'CS445-course'...
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
Unpacking objects: 100% (4/4), done.
Abels-MacBook-Pro:git clone cs445 abel$
```

16. git remote

```
The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
Abels-MacBook-Pro:CS445-course abel$ git remote
origin
Abels-MacBook-Pro:CS445-course abel$
```

17. git push

```
Abels-MacBook-Pro:CS445-course abel$ git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 274 bytes | 274.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/abelmichael/CS445-course
057d4d2..2025601 main -> main
Abels-MacBook-Pro:CS445-course abel$ 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 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/abelmichael/CS445-course
2025601..f17c57b main -> origin/main
Abels-MacBook-Pro:CS445-course abel$ git merge
Updating 2025601..f17c57b
Fast-forward
 hello.js | 3 +--
1 file changed, 1 insertion(+), 2 deletions(-)
Abels-MacBook-Pro:CS445-course abel$ git merge
```

18. git fetch

```
Abels-MacBook-Pro:CS445-course abel$ 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 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/abelmichael/CS445-course
2025601..f17c57b main -> origin/main
Abels-MacBook-Pro:CS445-course abel$ git merge
```

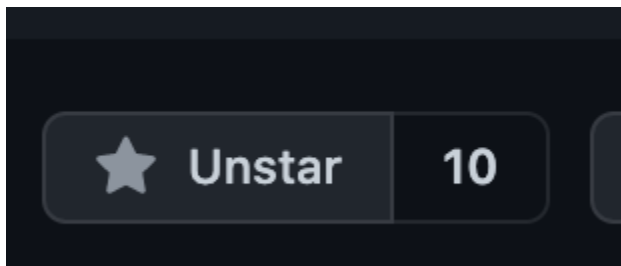
19. git pull

```
Abels-MacBook-Pro:CS445-course abel$ 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 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/abelmichael/CS445-course
   f17c57b..70e0321  main       -> origin/main
Updating f17c57b..70e0321
Fast-forward
 README.md | 6 +++++-
 1 file changed, 5 insertions(+), 1 deletion(-)
```

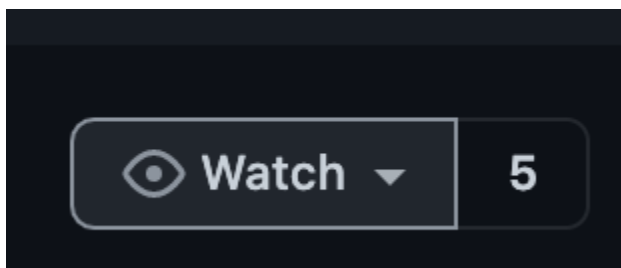
Practice with all the third party repositories commands from the slides:

Assume we are working with the following upstream: <https://github.com/bellaxing/cs445-labs>

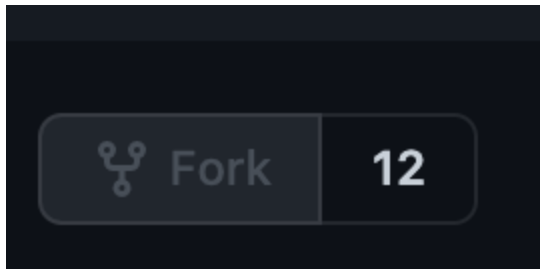
20. Star the repository



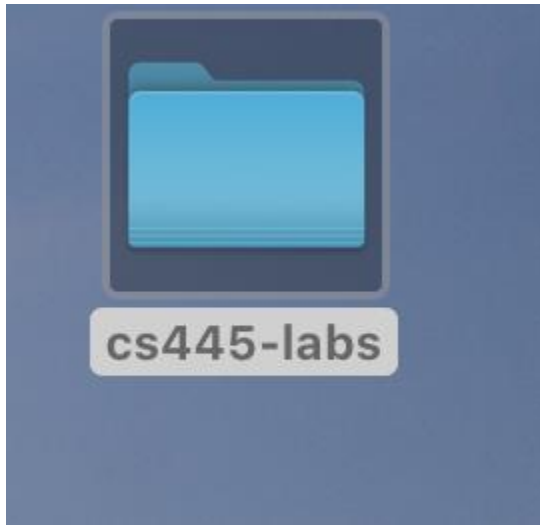
21. Watch the repository for any changes



22. Fork the repository



23. Clone your forked repository into your local machine



Cloned to desktop

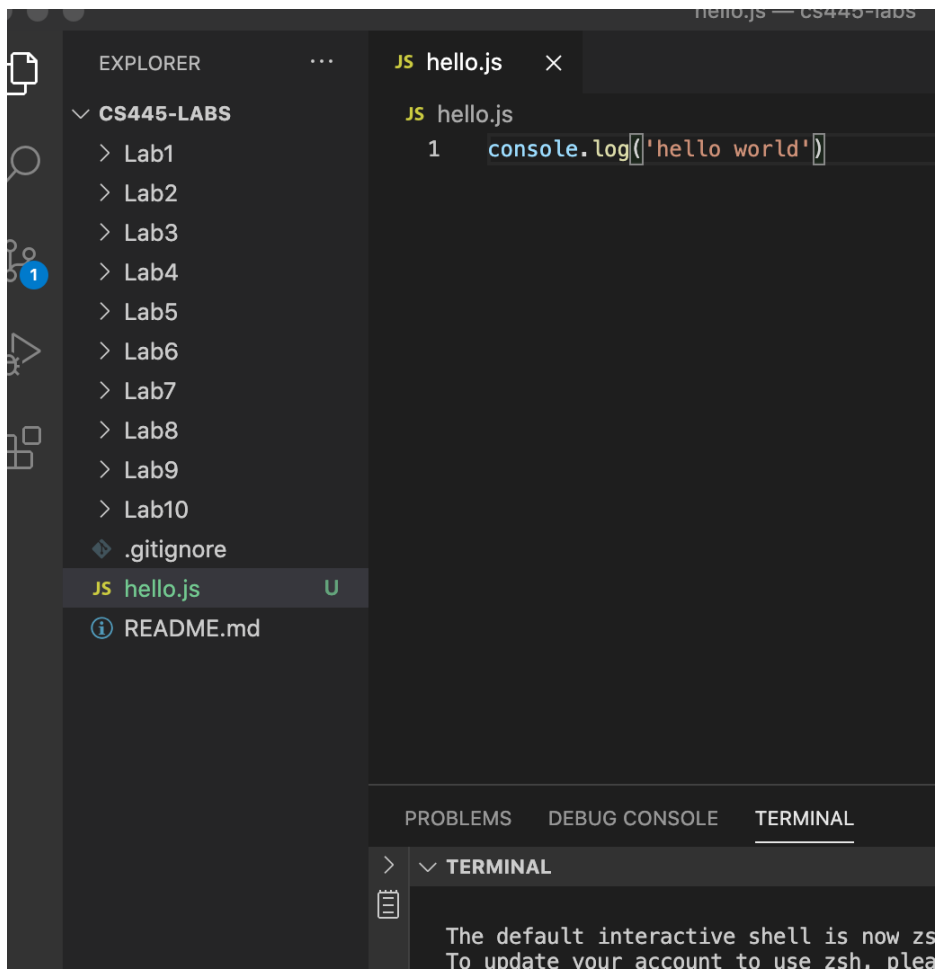
24. Create a new topic branch using studentId-studentName format, For example: 610001-JohnSmith

```
Abels-MacBook-Pro:cs445-labs abel$ git branch 610001-JohnSmith
```

25. Checkout your new branch

```
Abels-MacBook-Pro:cs445-labs abel$ git checkout 610001-JohnSmith  
Switched to branch '610001-JohnSmith'
```

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



27. Commit the changes

```
Abels-MacBook-Pro:cs445-labs abel$ git add .
Abels-MacBook-Pro:cs445-labs abel$ git commit -m "hello world"
[610001-JohnSmith eff0b53] hello world
1 file changed, 1 insertion(+)
create mode 100644 hello.js
```

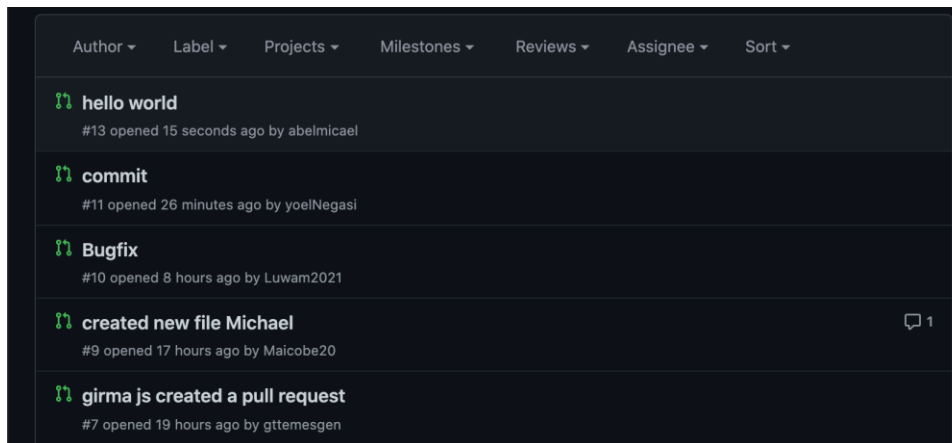
28. Merge your studentId-studentName branch with the main branch

```
Abels-MacBook-Pro:cs445-labs abel$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
Abels-MacBook-Pro:cs445-labs abel$ git merge 610001-JohnSmith
Updating fc075c5..eff0b53
Fast-forward
 hello.js | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 hello.js
```

29. Push your changes to the remote repo

```
Abels-MacBook-Pro:cs445-labs abel$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 290 bytes | 290.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/abelmichael/cs445-labs.git
 fc075c5..eff0b53  main -> main
```

30. Create a new Pull Request to the upstream repo



A screenshot of the GitHub Pull Request (PR) list interface. At the top, there are filters for Author, Label, Projects, Milestones, Reviews, Assignee, and Sort. Below the filters, a list of PRs is shown, each with a green checkmark icon, a title, a number, and the time since it was opened by the author.

Author	Label	Projects	Milestones	Reviews	Assignee	Sort
hello world	#13	opened 15 seconds ago	by abelmichael			
commit	#11	opened 26 minutes ago	by yoelNegasi			
Bugfix	#10	opened 8 hours ago	by Luwam2021			
created new file Michael	#9	opened 17 hours ago	by Maicobe20			1
girma js created a pull request	#7	opened 19 hours ago	by gttemesgen			