EECS 367 Lab

Git-ing started with Git

Administrative

Welcome to AutoRob!

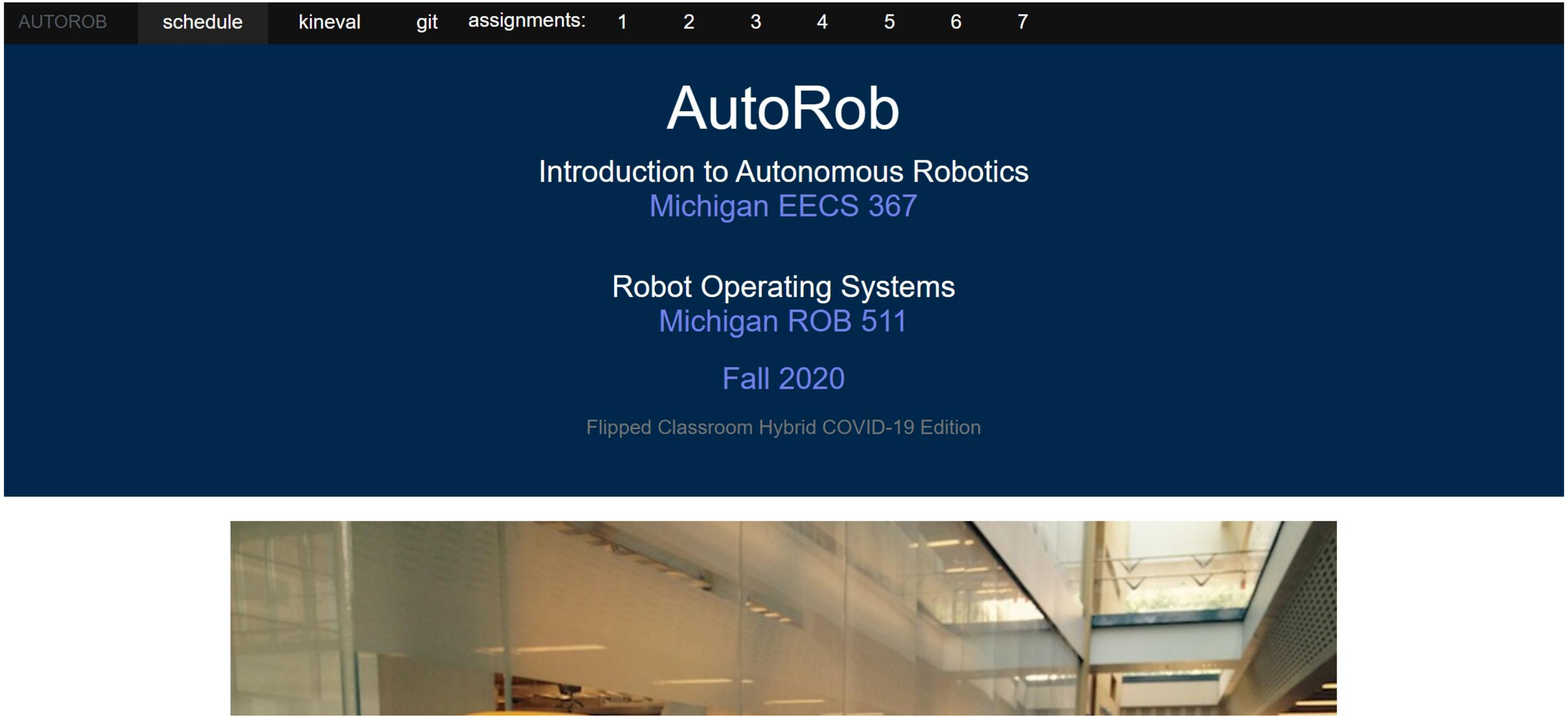
Lots of information on the course website

- https://autorob.org

Assignment #1: Path Planning

- Due 11:59pm, Wednesday, September 16

Administrative



AutoRob

Introduction to Autonomous Robotics
Michigan EECS 367

Robot Operating Systems
Michigan ROB 511

Fall 2020

Flipped Classroom Hybrid COVID-19 Edition

Lab Takeaways

1. Git initialization
2. Clone KinEval stencil
3. Make a change, commit, and push
4. Practice with HTML
5. Validate your changes

* What we won’t cover today: branching!

What is Git?

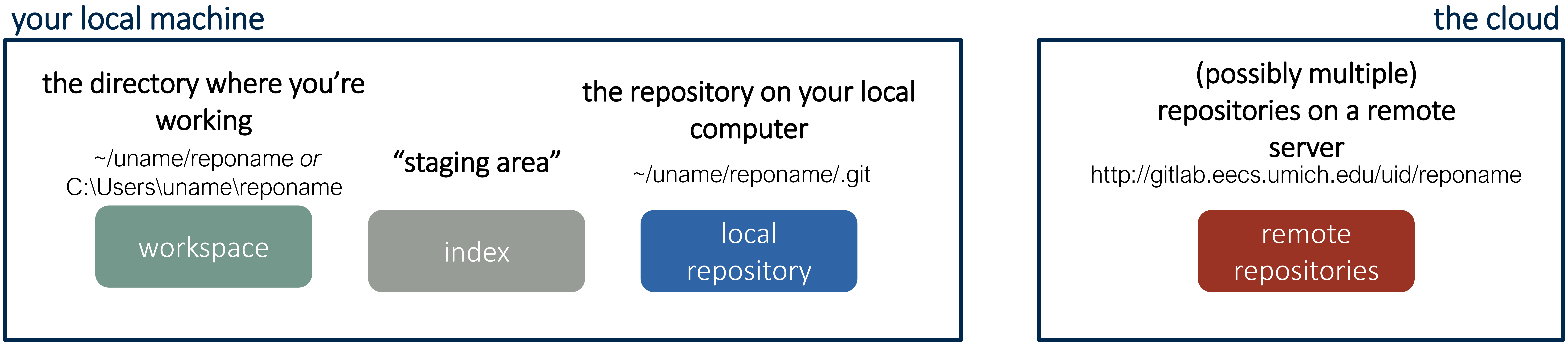
Git is **version control** software, meaning that it tracks changes to your files (code, papers, …) as you work on them over time

Similar to “track changes” feature in document writing programs, except you must choose which versions to include in the tracking—it is not automatic

Widely used in academia and industry

The only way for you to submit your homework for this course!

Git Process Overview



your local machine

the directory where you're working

~/uname/reponame *or* C:\Users\uname\reponame

workspace

"staging area"

index

the repository on your local computer

~/uname/reponame/.git

local repository

the cloud

(possibly multiple) repositories on a remote server

http://gitlab.eecs.umich.edu/uid/reponame

remote repositories

Git Process Overview

your local machine

the cloud

the directory where you're working

~/uname/reponame *or* C:\Users\uname\reponame

"staging area"

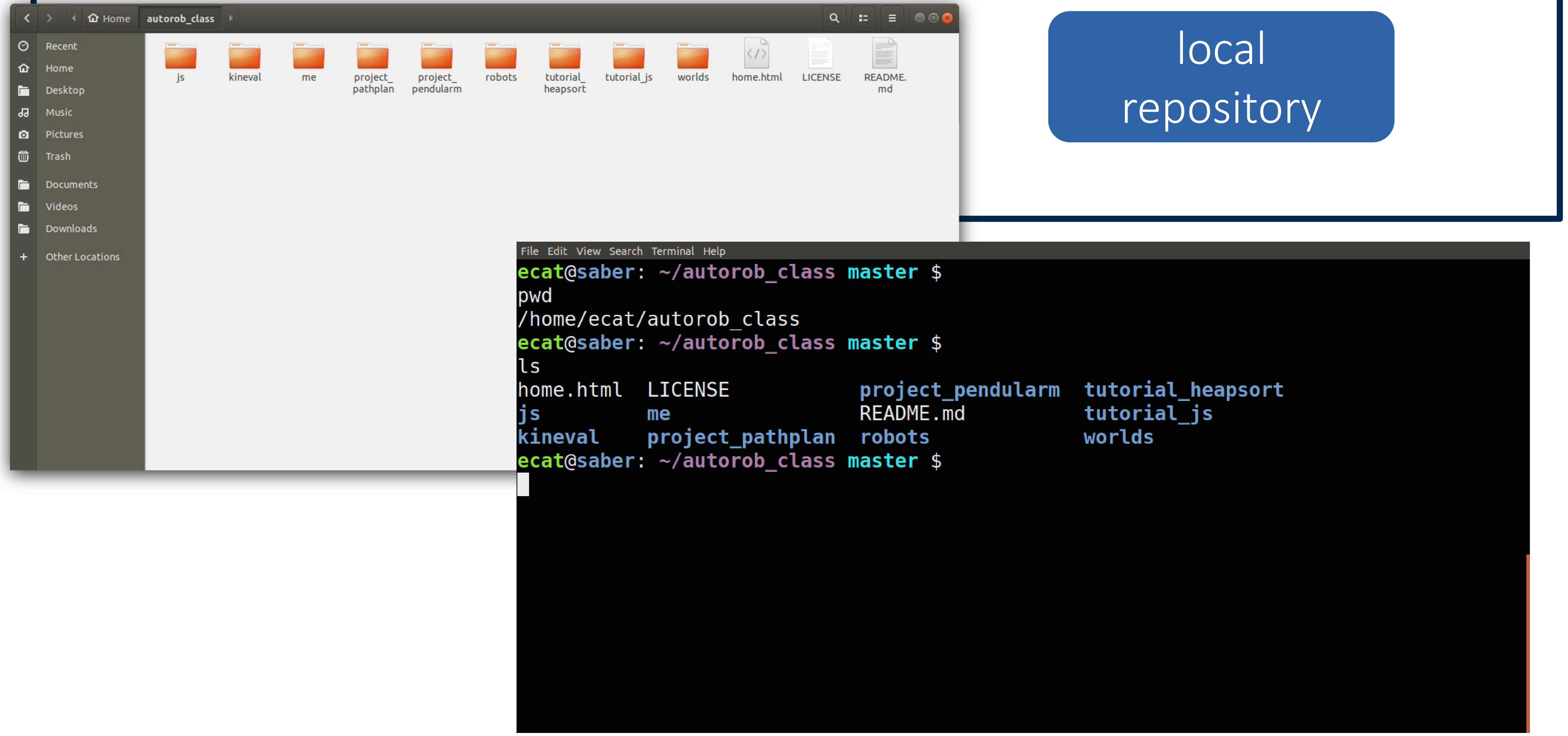
the repository on your local computer

~/uname/reponame/.git

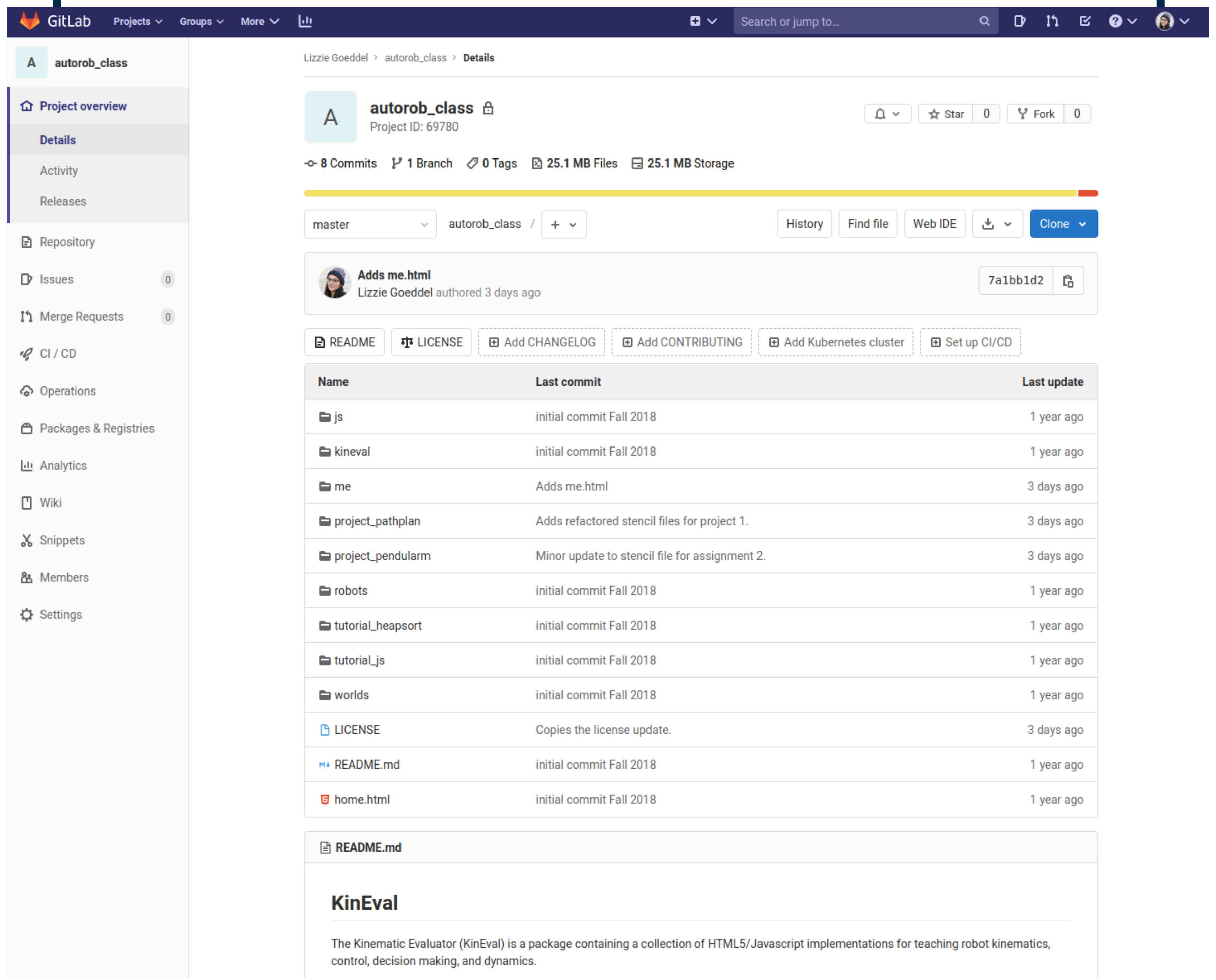
local repository

(possibly multiple) repositories on a remote server

http://gitlab.eecs.umich.edu/uid/reponame



```
ecat@saber: ~/autorob_class master $
pwd
/home/ecat/autorob_class
ecat@saber: ~/autorob_class master $
ls
home.html  LICENSE          project_pendularm  tutorial_heapsort
js         me               README.md          tutorial_js
kineval    project_pathplan robots             worlds
ecat@saber: ~/autorob_class master $
```



autorob_class

Project ID: 69780

8 Commits 1 Branch 0 Tags 25.1 MB Files 25.1 MB Storage

Adds me.html

Lizzie Goeddel authored 3 days ago

7a1bb1d2

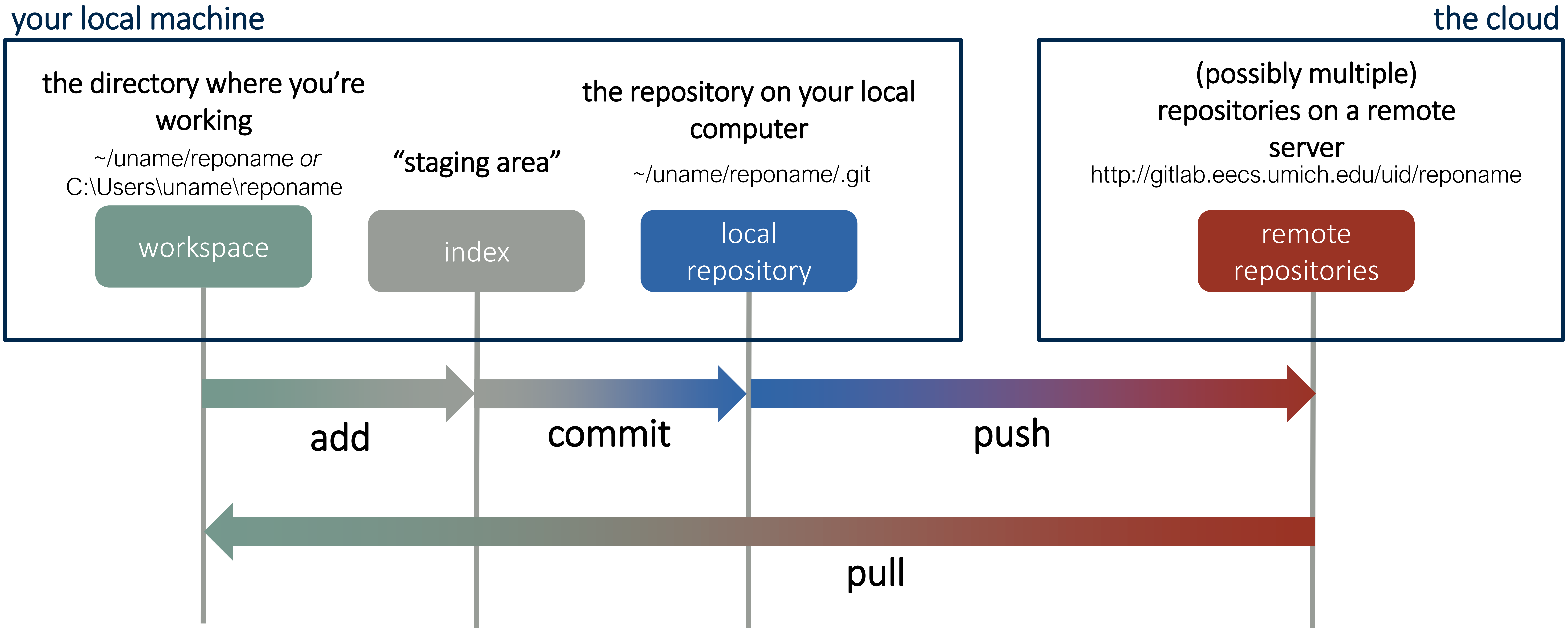
Name	Last commit	Last update
js	initial commit Fall 2018	1 year ago
kineval	initial commit Fall 2018	1 year ago
me	Adds me.html	3 days ago
project_pathplan	Adds refactored stencil files for project 1.	3 days ago
project_pendularm	Minor update to stencil file for assignment 2.	3 days ago
robots	initial commit Fall 2018	1 year ago
tutorial_heapsort	initial commit Fall 2018	1 year ago
tutorial_js	initial commit Fall 2018	1 year ago
worlds	initial commit Fall 2018	1 year ago
LICENSE	Copies the license update.	3 days ago
README.md	initial commit Fall 2018	1 year ago
home.html	initial commit Fall 2018	1 year ago

README.md

KinEval

The Kinematic Evaluator (KinEval) is a package containing a collection of HTML5/Javascript implementations for teaching robot kinematics, control, decision making, and dynamics.

Git Process Overview



your local machine

the cloud

the directory where you're working

~/uname/reponame *or* C:\Users\uname\reponame

"staging area"

the repository on your local computer

~/uname/reponame/.git

(possibly multiple) repositories on a remote server

http://gitlab.eecs.umich.edu/uid/reponame

Step 1: Git Initialization

1. Install Git
   https://git-scm.com/book/en/v2/Getting-Started-Installing-Git

2. Create an empty private repository on GitHub, Bitbucket, or EECS GitLab

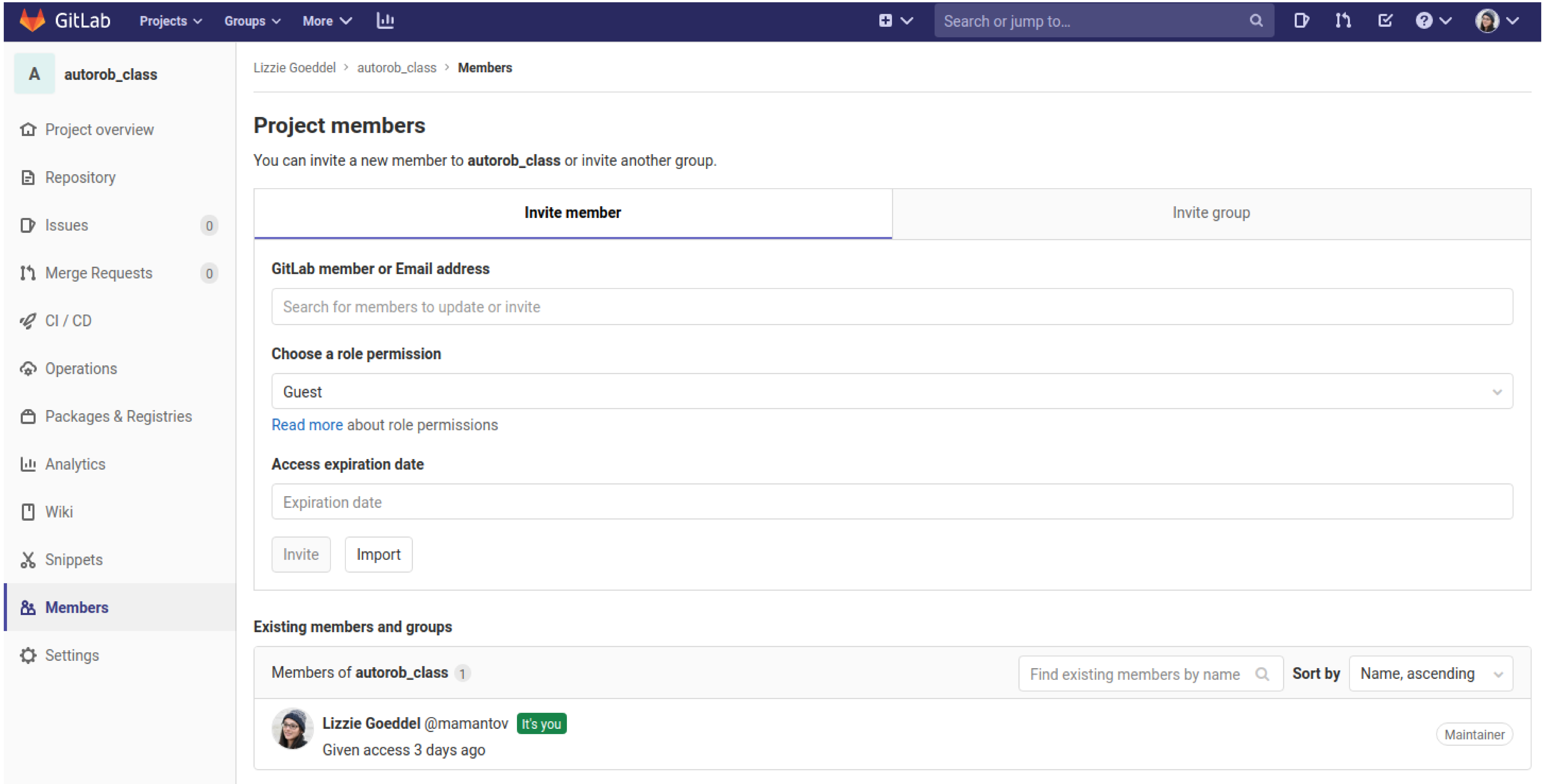
3. Make sure you have added the course staff as collaborators on your remote repository:

   GitHub: ohseejay, emgoeddel, zhezhou1993, cxt98

   Bitbucket: ohseejay, emgoeddel, zhezhou, cxt98

   EECS GitLab: ocj, mamantov, zhezhou, cxt

Step 1: Git Initialization



Project members

You can invite a new member to **autorob_class** or invite another group.

GitLab member or Email address

Search for members to update or invite

Choose a role permission

Guest

Read more about role permissions

Access expiration date

Expiration date

Existing members and groups

Members of **autorob_class** 1

Find existing members by name

Sort by Name, ascending

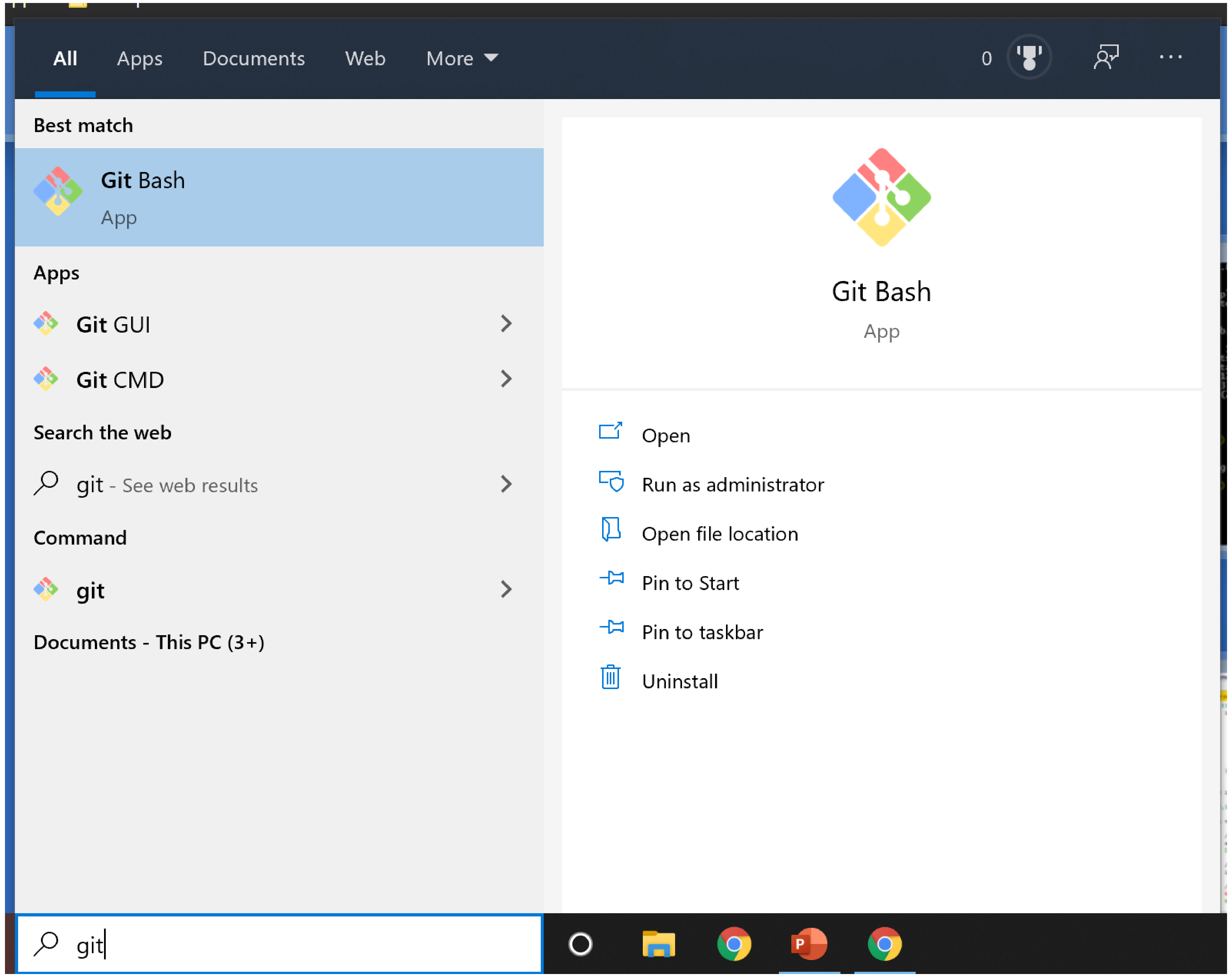
Lizzie Goeddel @mamantov It's you

Given access 3 days ago

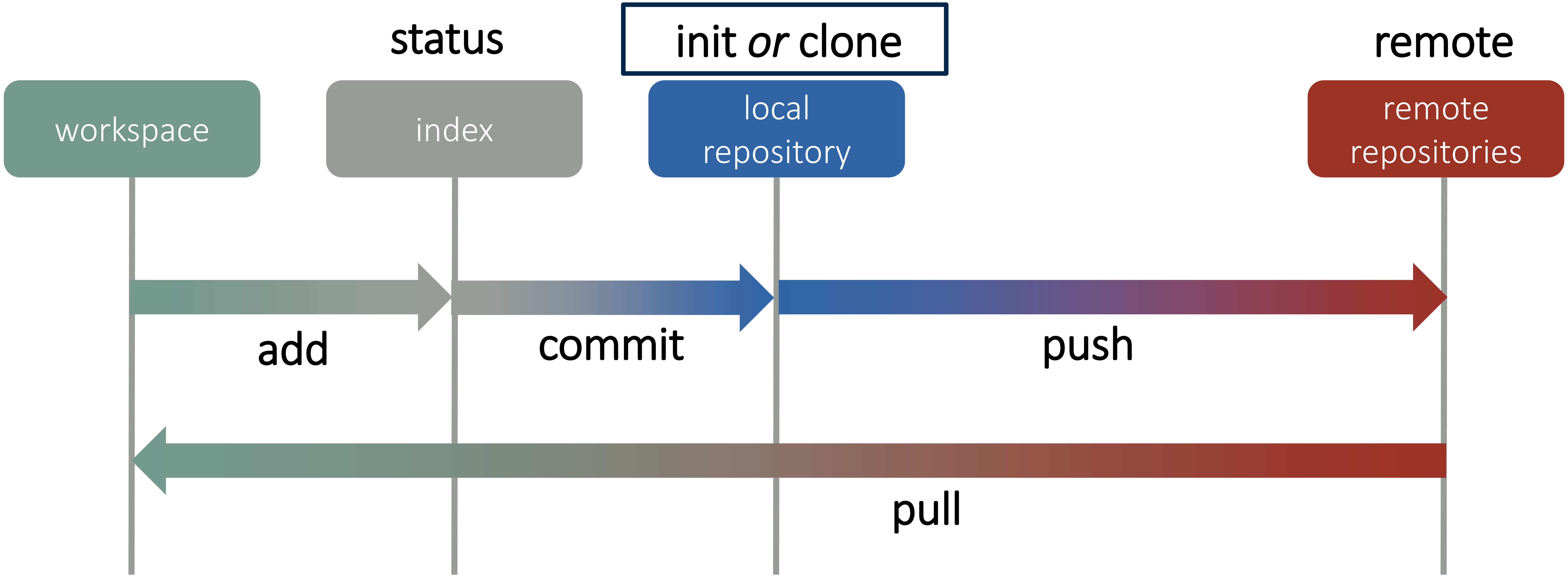
Maintainer

Step 1 Breakout Rooms

Get Your Terminal Ready



Basic Git Commands



status

init *or* clone

remote

workspace

index

local repository

remote repositories

add

commit

push

pull

Basic Git Commands

```
git init
```

start a new, blank repository in your working directory

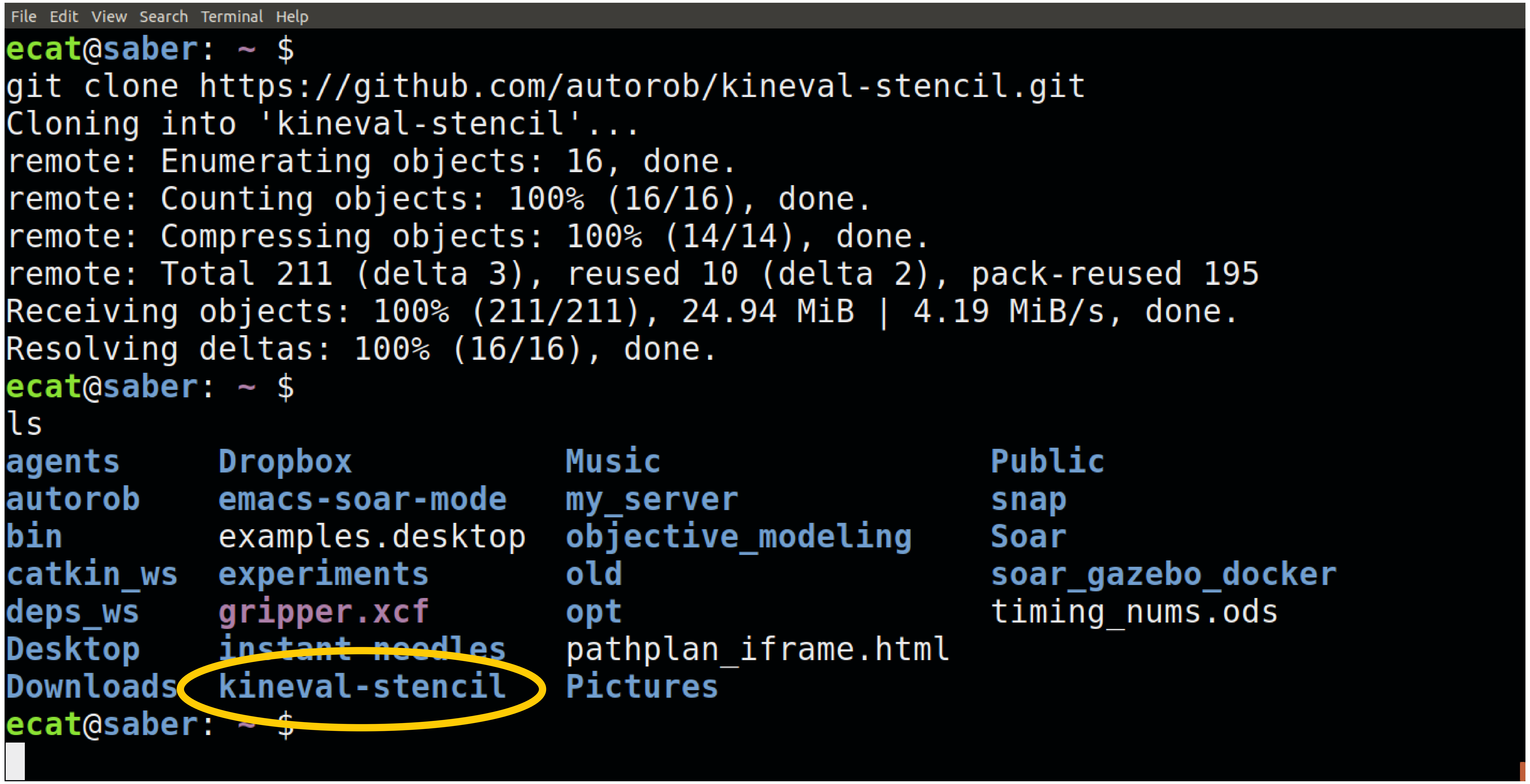
```
git clone <repo address>
```

copy an existing repository from the cloud into your working directory

Step 2: Clone

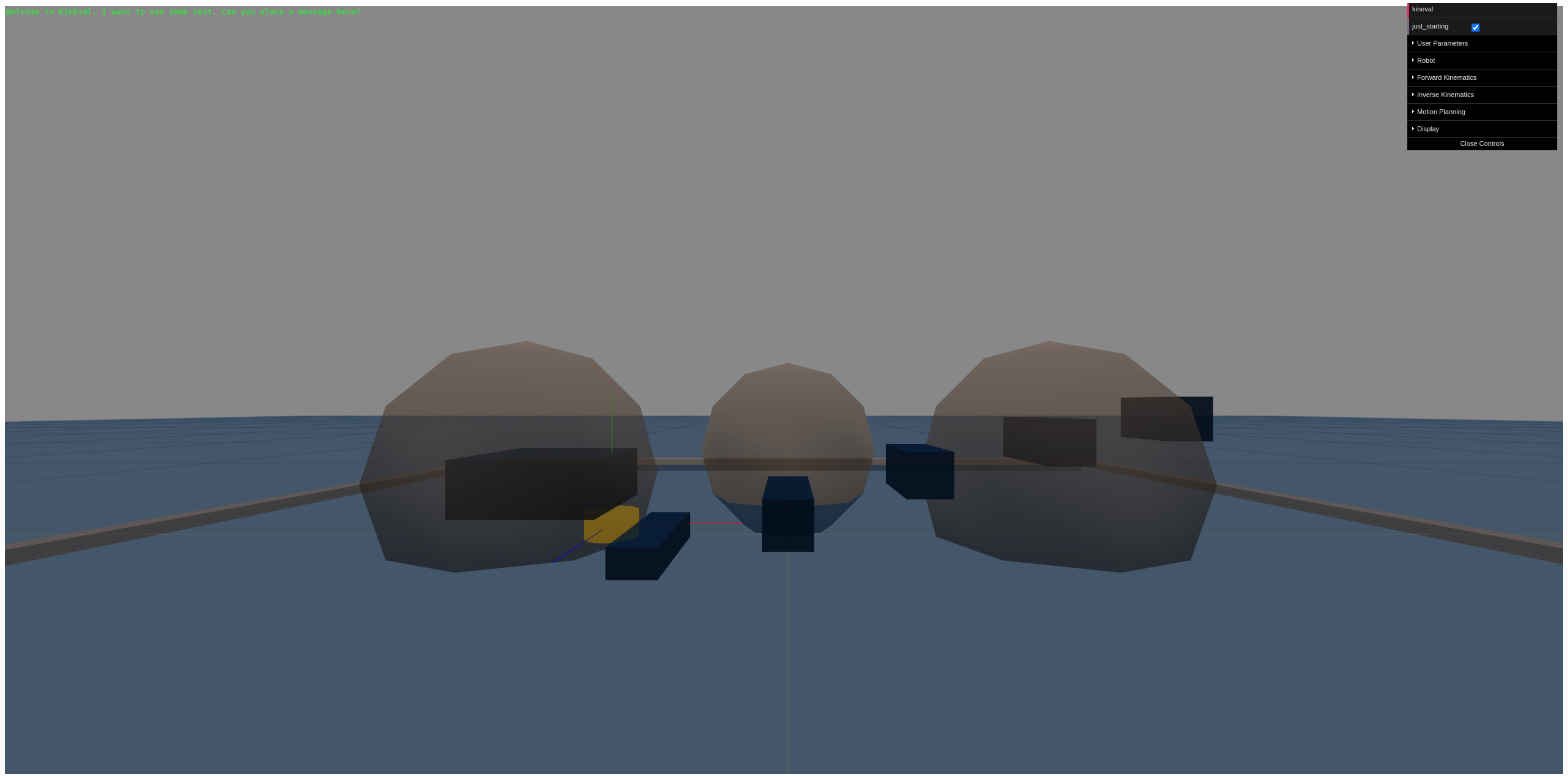
1. Clone the autorob kineval-stencil repo
   https://github.com/autorob/kineval-stencil
2. Open “home.html” and “project_pathplan/search_canvas.html” in a web browser to make sure stencil code runs properly

Step 2: Clone



```
ecat@saber: ~ $
git clone https://github.com/autorob/kineval-stencil.git
Cloning into 'kineval-stencil'...
remote: Enumerating objects: 16, done.
remote: Counting objects: 100% (16/16), done.
remote: Compressing objects: 100% (14/14), done.
remote: Total 211 (delta 3), reused 10 (delta 2), pack-reused 195
Receiving objects: 100% (211/211), 24.94 MiB | 4.19 MiB/s, done.
Resolving deltas: 100% (16/16), done.
ecat@saber: ~ $
ls
agents     Dropbox          Music                 Public
autorob    emacs-soar-mode  my_server             snap
bin        examples.desktop objective_modeling    Soar
catkin_ws  experiments      old                   soar_gazebo_docker
deps_ws    gripper.xcf      opt                   timing_nums.ods
Desktop    instant_needles  pathplan_iframe.html
Downloads  kineval-stencil  Pictures
ecat@saber: ~ $
```

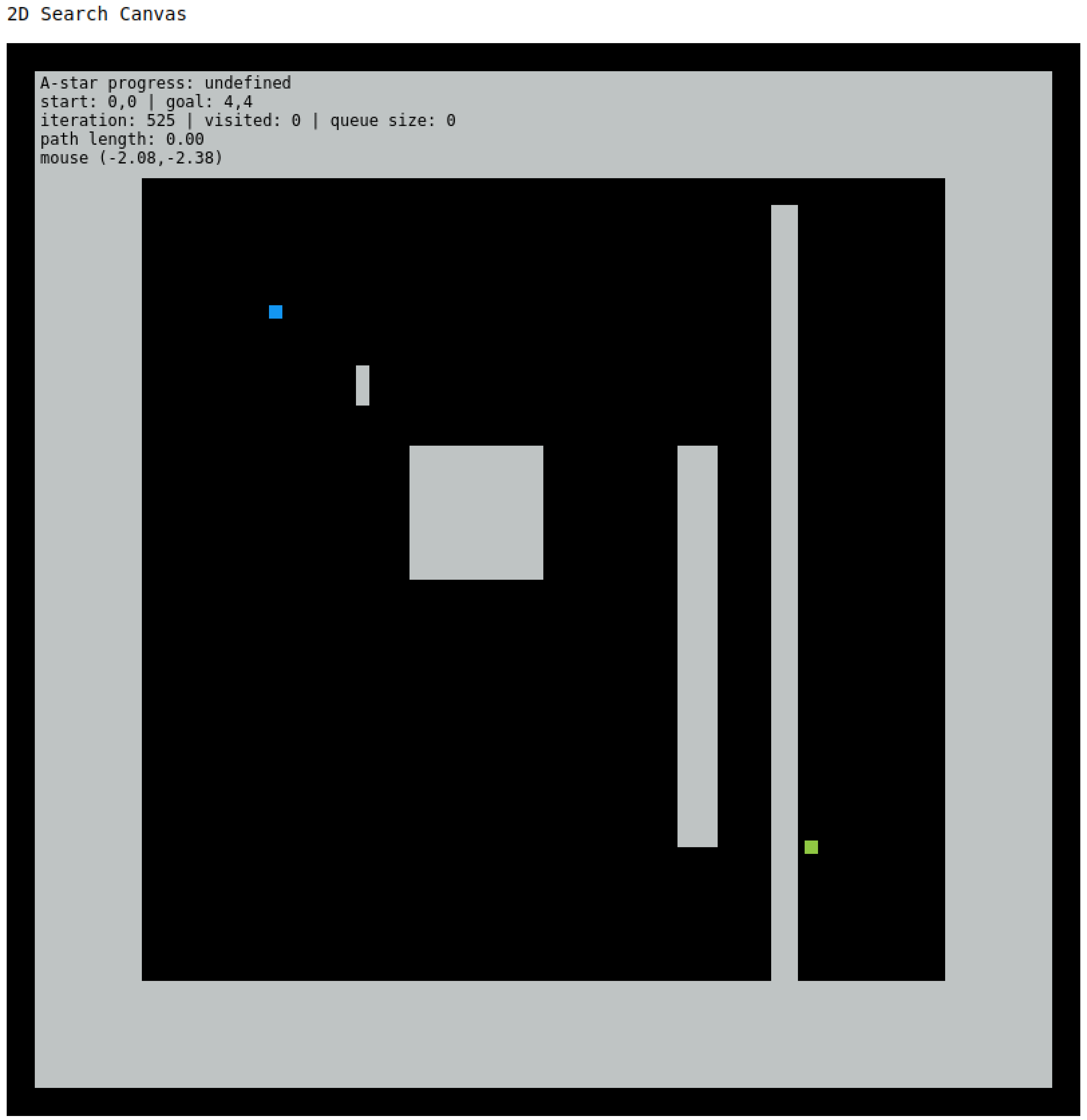
Step 2: Clone



Welcome to KinEval. I want to see some text. Can you place a message here?

kineval

just_starting ☑

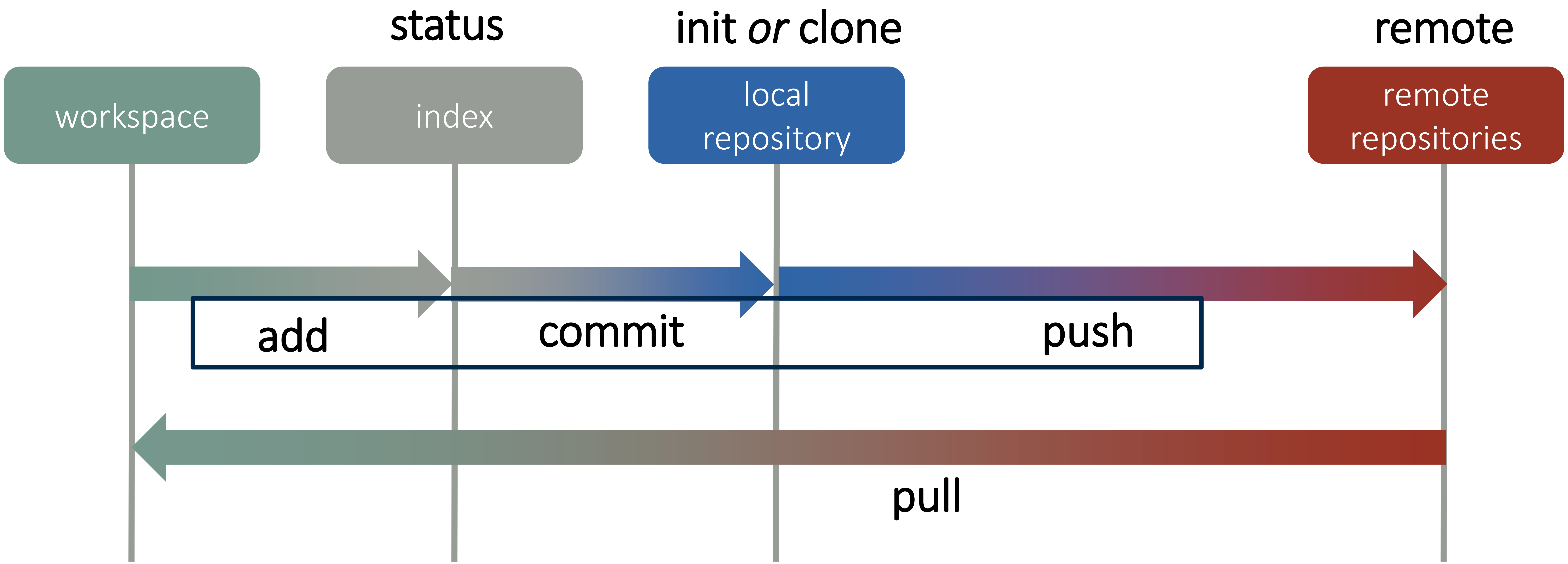


2D Search Canvas

```
A-star progress: undefined
start: 0,0 | goal: 4,4
iteration: 525 | visited: 0 | queue size: 0
path length: 0.00
mouse (-2.08,-2.38)
```

Step 2 Breakout Rooms

Basic Git Commands



Basic Git Commands

```
git add <filename>
```

add a file to the index, a.k.a. “stage” it

```
git commit –m “message”
```

save all of the edits to the files currently in the index as a “commit”

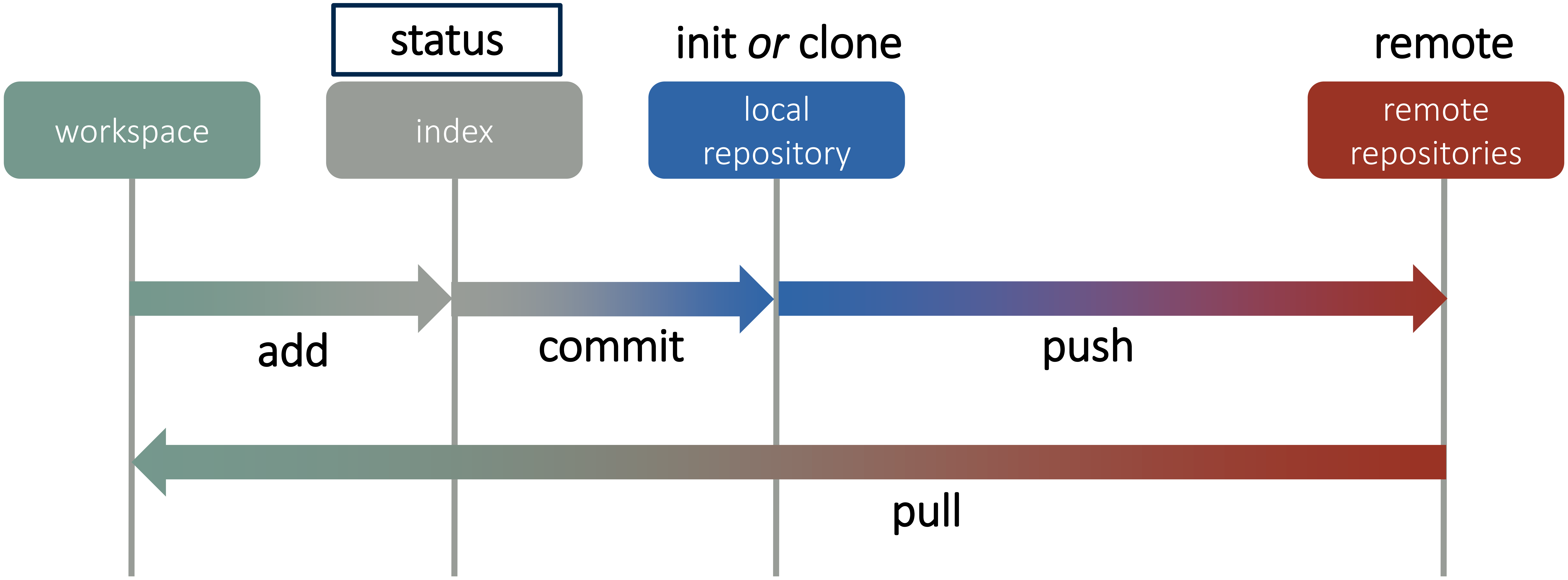
```
git push
```

push local changes to the default remote repository

```
git push <remote> <branch>
```

push local changes to a specific remote repository

Basic Git Commands



status

init *or* clone

remote

workspace

index

local repository

remote repositories

add

commit

push

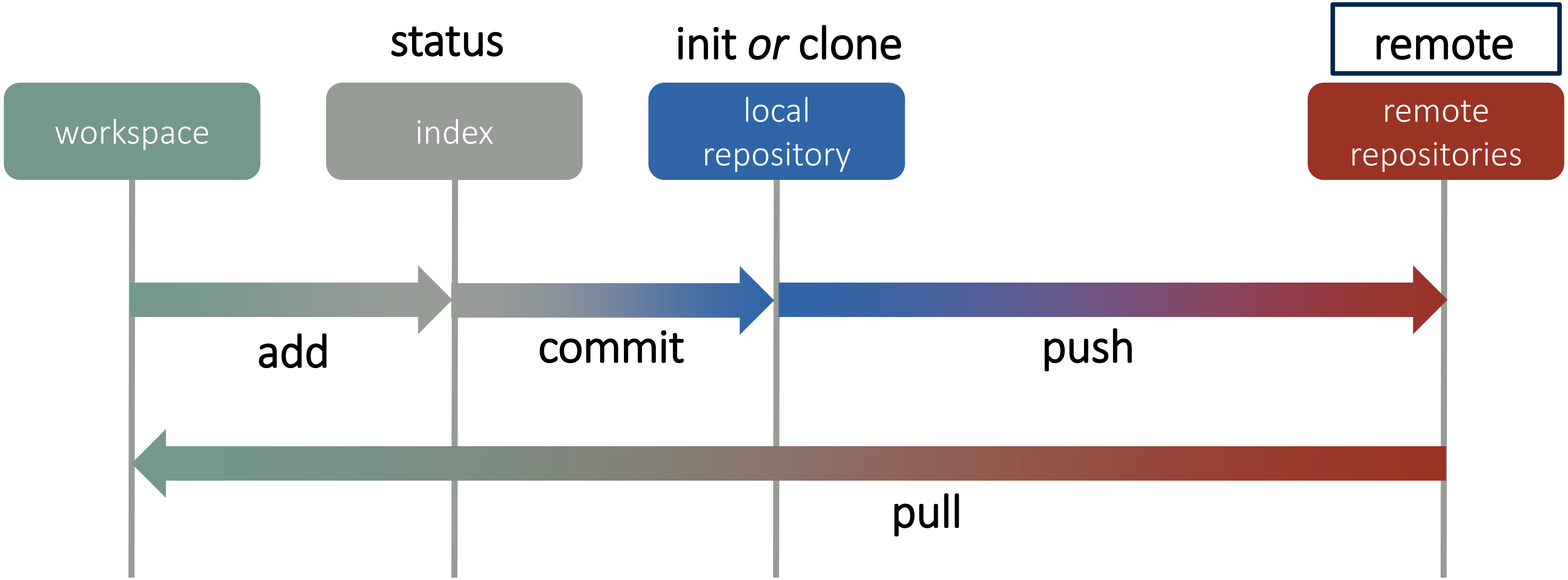
pull

Basic Git Commands

```
git status
```

list which files in the workspace have been modified and which have been added to the index

Basic Git Commands



Basic Git Commands

```
git remote -v
```

lists the addresses of your remote repositories

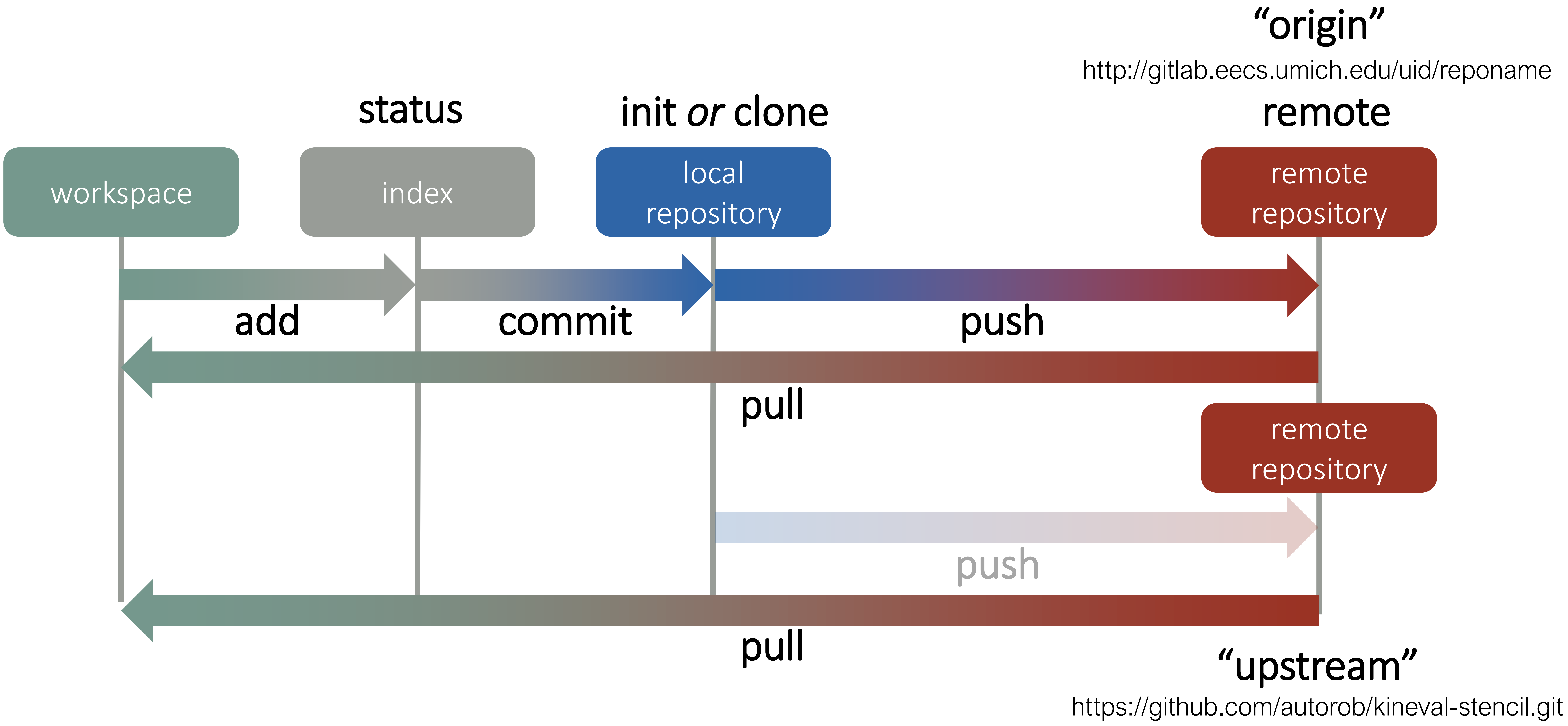
```
git remote add <name> <address>
```

adds a new remote

```
git remote rename <old-name> <new-name>
```

renames an existing remote

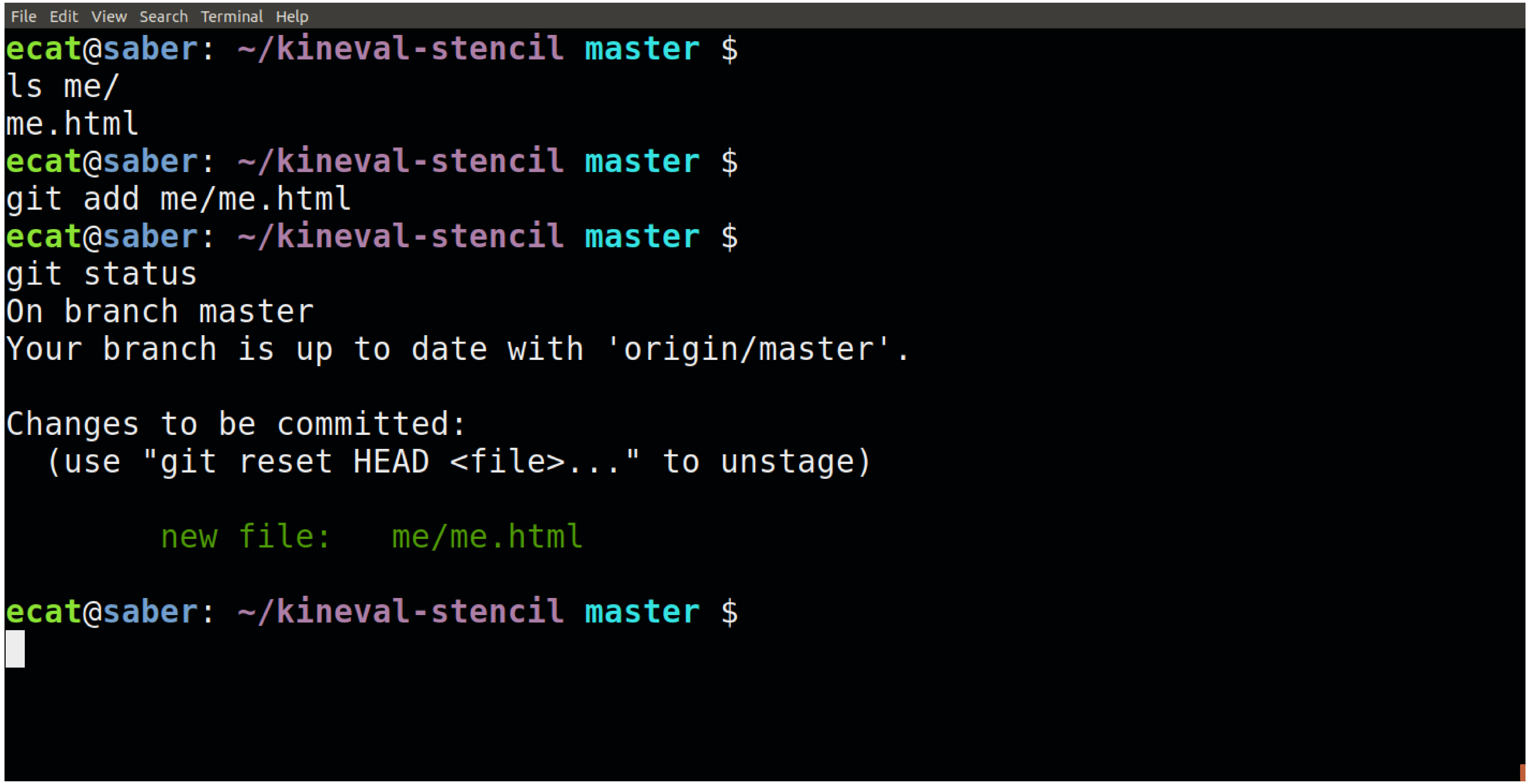
Basic Git Commands



Step 3: Commit + Push

1. Create kineval-stencil/me directory
2. Create me/me.html with just your name in the file
3. Add me/me.html to the index
4. Check the git status to see me.html in the index

Step 3: Commit + Push

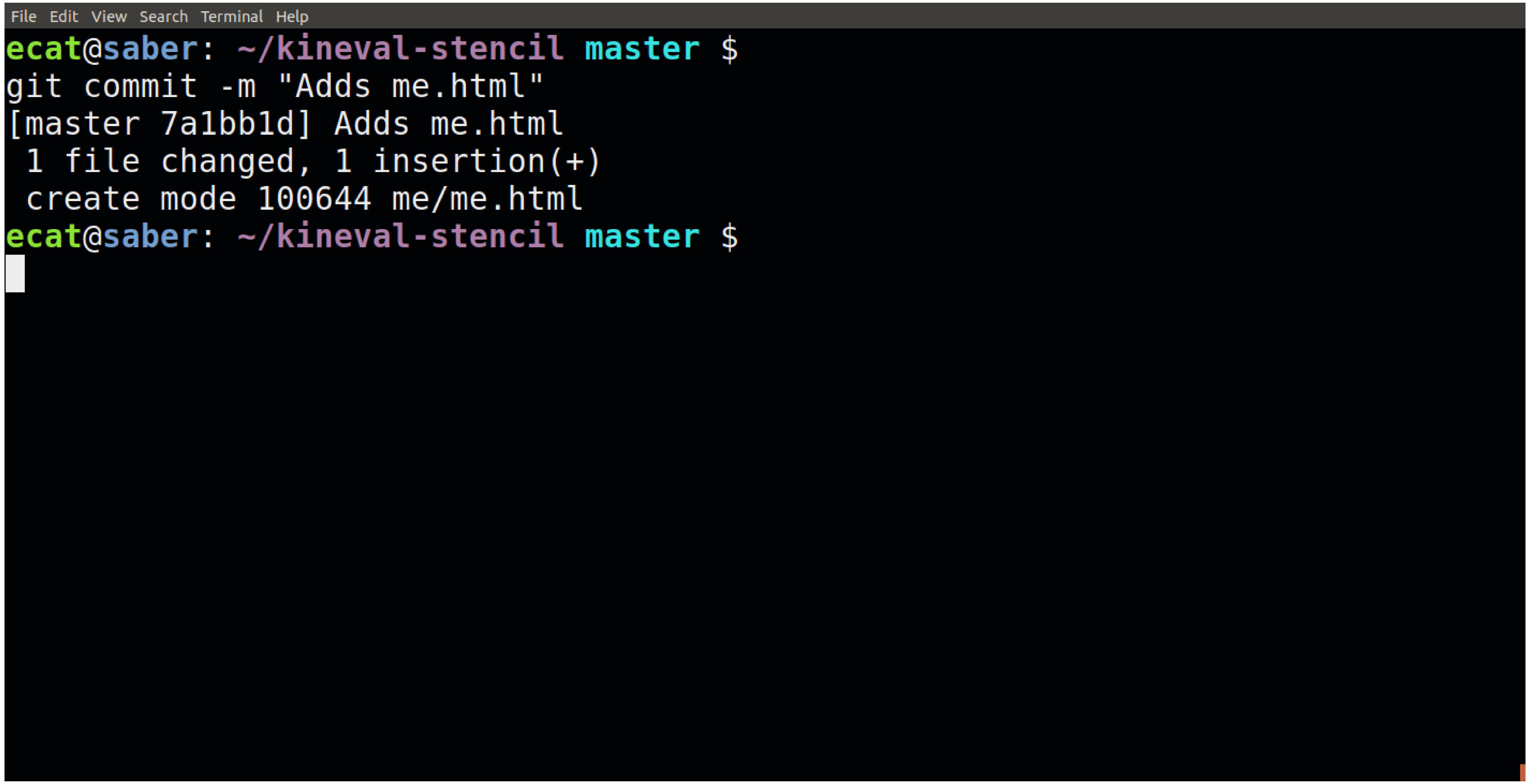


```
ecat@saber: ~/kineval-stencil master $
ls me/
me.html
ecat@saber: ~/kineval-stencil master $
git add me/me.html
ecat@saber: ~/kineval-stencil master $
git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        new file:   me/me.html

ecat@saber: ~/kineval-stencil master $
```

Step 3: Commit + Push

1. Create kineval-stencil/me directory
2. Create me/me.html with just your name in the file
3. Add me/me.html to the index
4. Check the git status to see me.html in the index
5. Make your first commit

Step 3: Commit + Push



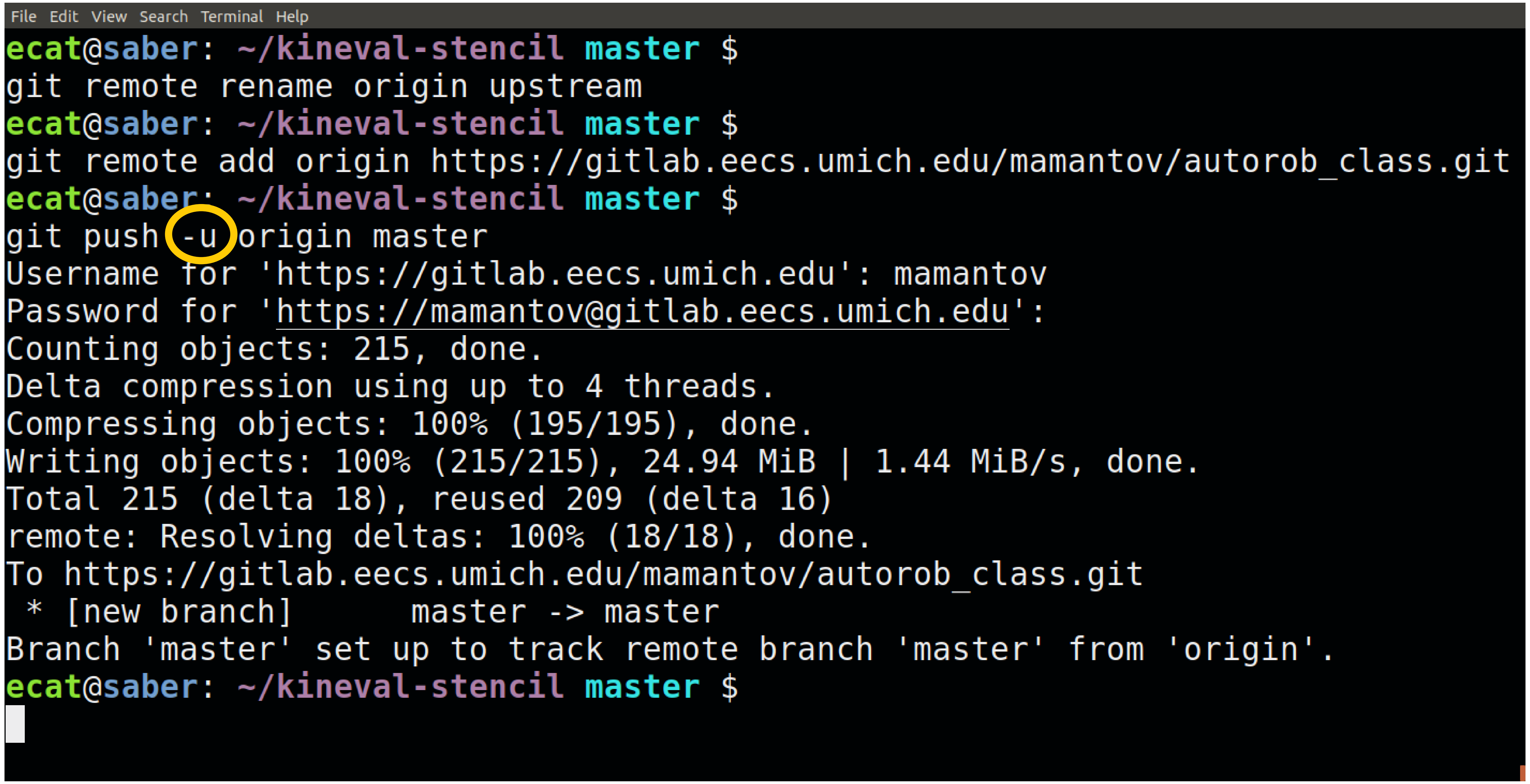
```
ecat@saber: ~/kineval-stencil master $
git commit -m "Adds me.html"
[master 7a1bb1d] Adds me.html
 1 file changed, 1 insertion(+)
 create mode 100644 me/me.html
ecat@saber: ~/kineval-stencil master $
```

Step 3: Commit + Push

1. Create kineval-stencil/me directory
2. Create me/me.html with just your name in the file
3. Add me/me.html to the index
4. Check the git status to see me.html in the index
5. Make your first commit
6. Make your personal remote repository the “origin” and the autorob remote repository the “upstream”
7. Push your commit

Step 3: Commit + Push

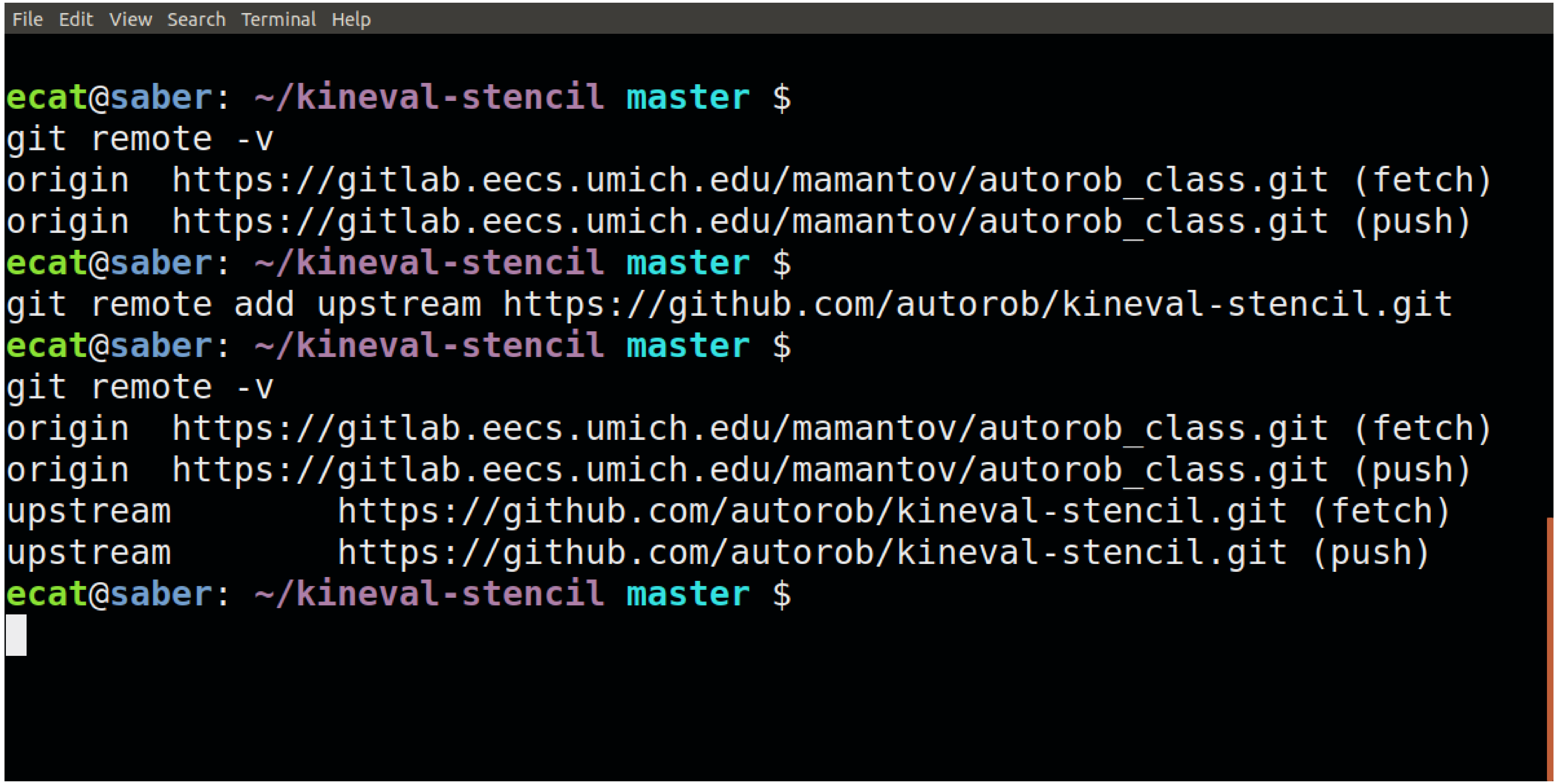
* -u only needed the first time you push!



```
ecat@saber: ~/kineval-stencil master $
git remote rename origin upstream
ecat@saber: ~/kineval-stencil master $
git remote add origin https://gitlab.eecs.umich.edu/mamantov/autorob_class.git
ecat@saber: ~/kineval-stencil master $
git push -u origin master
Username for 'https://gitlab.eecs.umich.edu': mamantov
Password for 'https://mamantov@gitlab.eecs.umich.edu':
Counting objects: 215, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (195/195), done.
Writing objects: 100% (215/215), 24.94 MiB | 1.44 MiB/s, done.
Total 215 (delta 18), reused 209 (delta 16)
remote: Resolving deltas: 100% (18/18), done.
To https://gitlab.eecs.umich.edu/mamantov/autorob_class.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
ecat@saber: ~/kineval-stencil master $
```

Step 3: Commit + Push

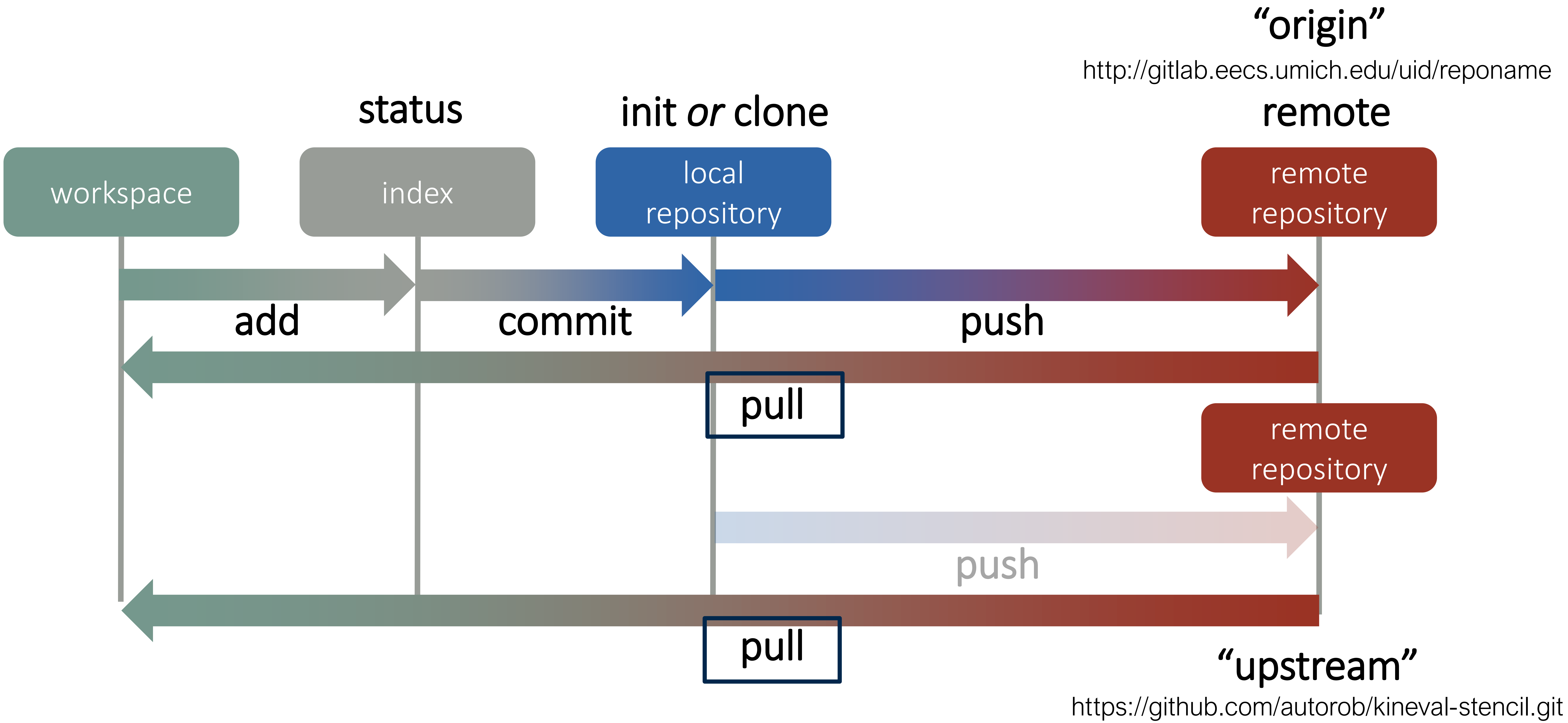
* Only if you are not following along with the slides and cloned your own remote!



```
ecat@saber: ~/kineval-stencil master $
git remote -v
origin  https://gitlab.eecs.umich.edu/mamantov/autorob_class.git (fetch)
origin  https://gitlab.eecs.umich.edu/mamantov/autorob_class.git (push)
ecat@saber: ~/kineval-stencil master $
git remote add upstream https://github.com/autorob/kineval-stencil.git
ecat@saber: ~/kineval-stencil master $
git remote -v
origin  https://gitlab.eecs.umich.edu/mamantov/autorob_class.git (fetch)
origin  https://gitlab.eecs.umich.edu/mamantov/autorob_class.git (push)
upstream        https://github.com/autorob/kineval-stencil.git (fetch)
upstream        https://github.com/autorob/kineval-stencil.git (push)
ecat@saber: ~/kineval-stencil master $
```

Step 3 Breakout Rooms

Basic Git Commands



Basic Git Commands

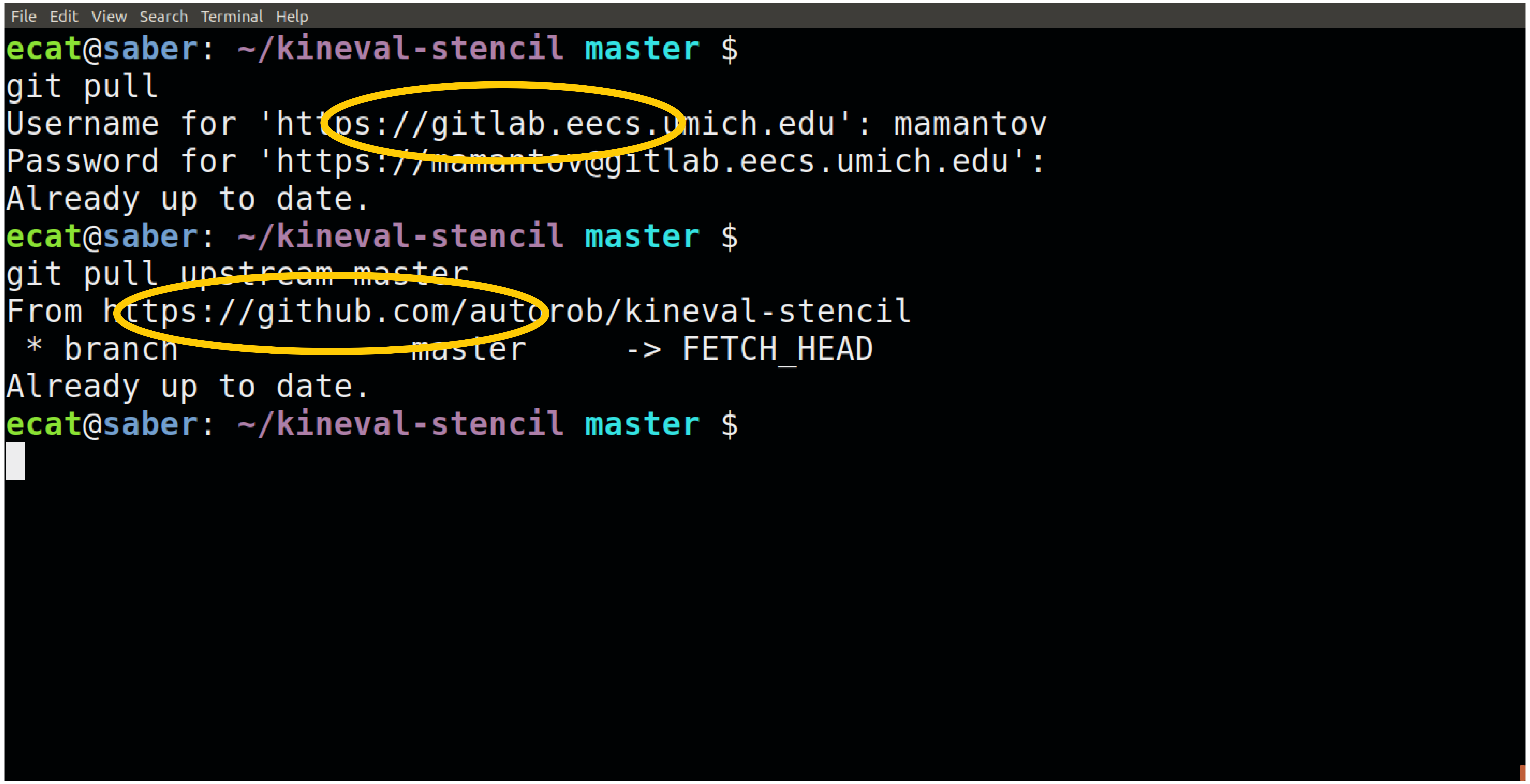
```
git pull
```

pull changes from the default remote into your local repository and workspace

```
git pull <remote> <branch>
```

pull changes from a specific remote into your local repository and workspace

Two Types of Pull



```
ecat@saber: ~/kineval-stencil master $
git pull
Username for 'https://gitlab.eecs.umich.edu': mamantov
Password for 'https://mamantov@gitlab.eecs.umich.edu':
Already up to date.
ecat@saber: ~/kineval-stencil master $
git pull upstream master
From https://github.com/autorob/kineval-stencil
 * branch            master     -> FETCH_HEAD
Already up to date.
ecat@saber: ~/kineval-stencil master $
```

Step 4: HTML Practice

1. Copy a picture that best represents you to kineval-stencil/me/me.png
2. Add the following to me/me.html:

   ```
   <img src=”me.png”>
   I am an awesome student. I’m most excited about:
   <script>
   console.log(Array(16).join(”wat”-1)+” Batman!”)
   </script>
   ```
3. Open me/me.html to see your photo

Step 5: Validate Changes

1. Commit and push your updates to me/me.html
2. Check the webpage of your remote repository to see your changes
3. Make sure the course staff has the correct address for your remote repository! Talk to us now or DM us on MS Team to correct the address if it has changed.

Steps 4+5 Breakout Rooms

Git Resources: Tutorials

1. Course website
   https://autorob.org/#git_tutorial
2. Bitbucket tutorials
   https://www.atlassian.com/git/tutorials
3. Pro Git book
   https://git-scm.com/book/en/v2
4. Learn Git Branching
   https://learngitbranching.js.org/
5. Git Magic tutorial
   http://www-cs-students.stanford.edu/~blynn/gitmagic/
6. The simple guide
   http://rogerdudler.github.io/git-guide/

Git Resources: SSH Keys

Github:

https://docs.github.com/en/github/authenticating-to-github/adding-a-new-ssh-key-to-your-github-account

Bitbucket:

https://support.atlassian.com/bitbucket-cloud/docs/set-up-an-ssh-key/

GitLab:

https://docs.gitlab.com/ee/ssh/

* Setting up SSH keys allows you to push/pull without entering your password

Git Resources: Your Computer

`git --help` and `git <command> -h`