UNIVERSITY
OF CRETE

CS-255 – Git Tutorial

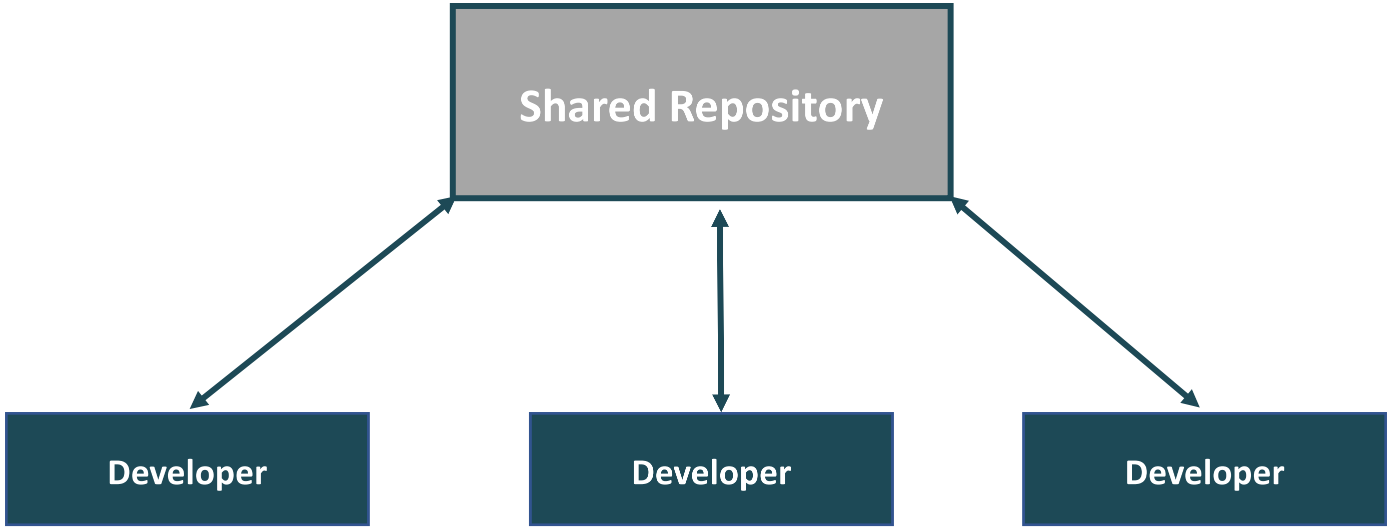
Giorgos Xanthakis
Iacovos Kolokasis
University of Crete
gxanth@csd.uoc.gr
kolokasis@csd.uoc.gr

Outline

- What is git
- How to join CS255 group
- What are ssh keys and how to use them
- How to fork a project
- Make a project private and add TAs as members
- How to use private.py
- How to use git

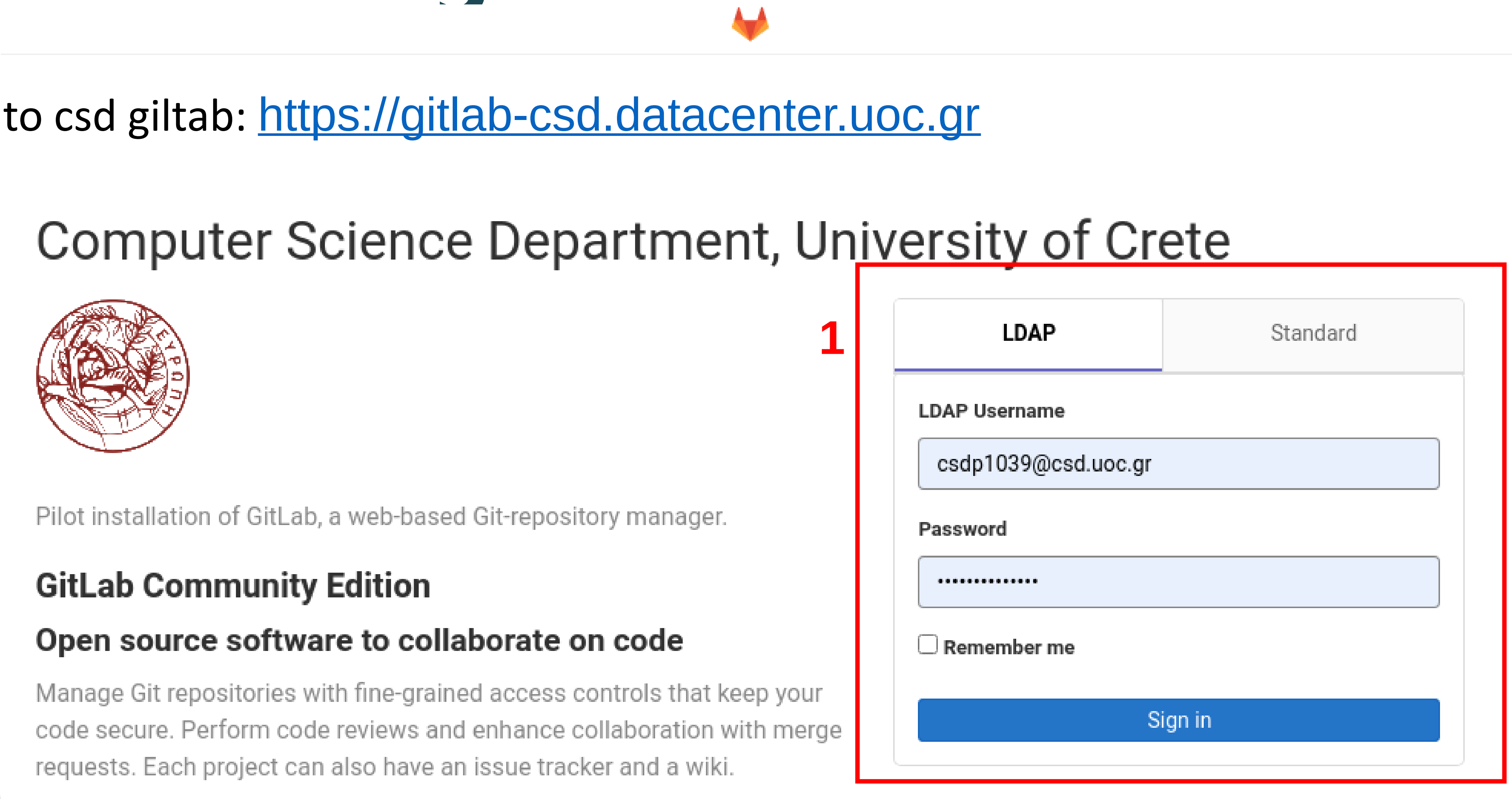
What is Git?

- Version control system
- Track file history
- Shared among multiple users



How to Login to CSD Git?

- Go to csd giltab: https://gitlab-csd.datacenter.uoc.gr



Computer Science Department, University of Crete

Pilot installation of GitLab, a web-based Git-repository manager.

GitLab Community Edition

Open source software to collaborate on code

Manage Git repositories with fine-grained access controls that keep your code secure. Perform code reviews and enhance collaboration with merge requests. Each project can also have an issue tracker and a wiki.

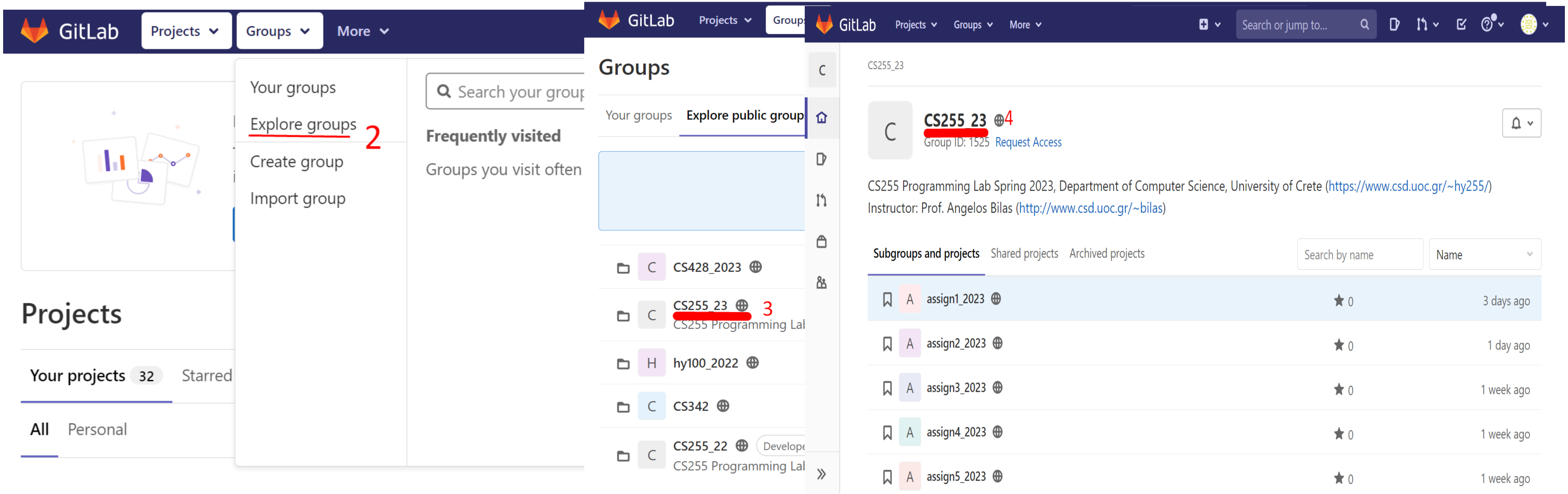
1

LDAP Username: csdp1039@csd.uoc.gr

Password: ••••••••••••••

□ Remember me

How to Join CS255 Group?

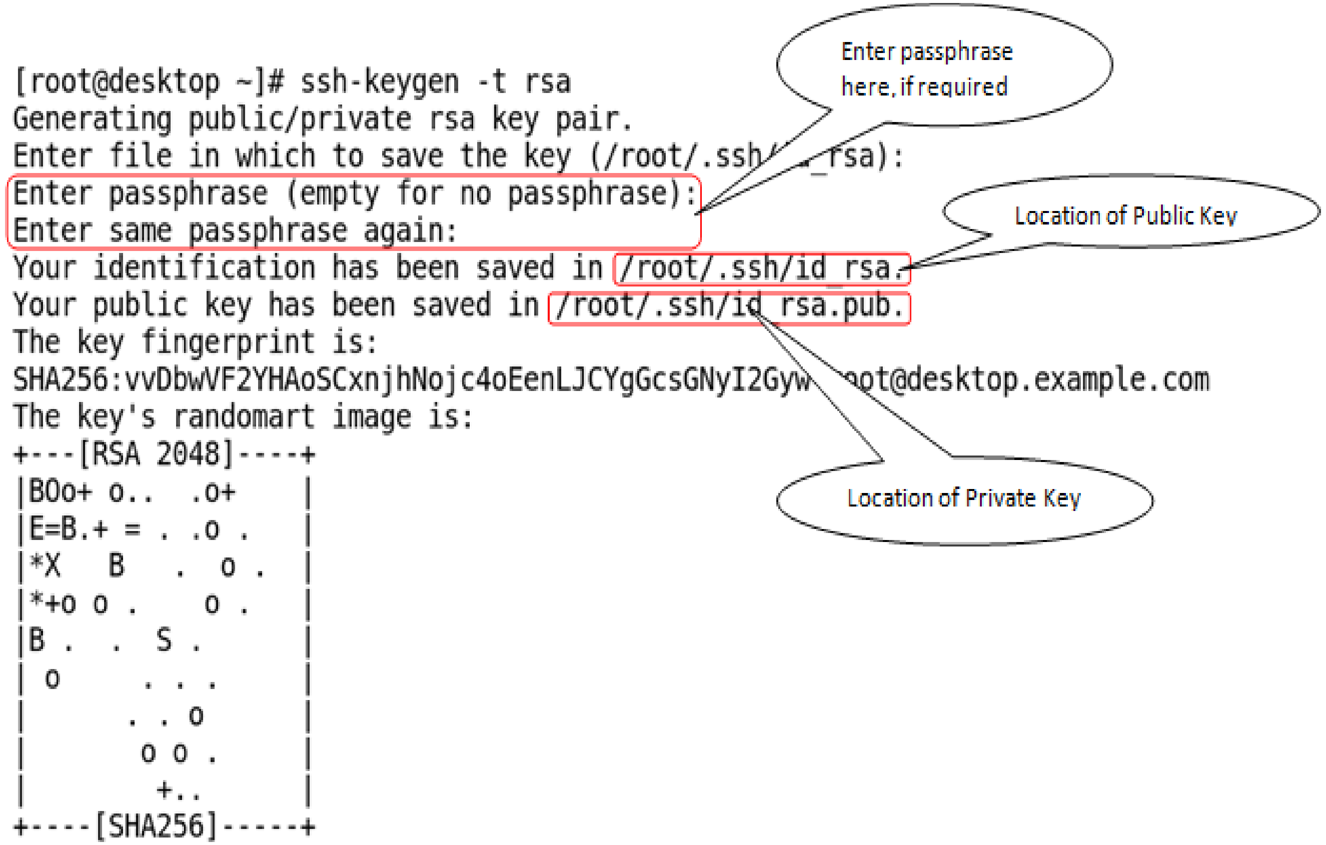


CS255 Programming Lab Spring 2023, Department of Computer Science, University of Crete (https://www.csd.uoc.gr/~hy255/)
Instructor: Prof. Angelos Bilas (http://www.csd.uoc.gr/~bilas)

assign1_2023	0	3 days ago
assign2_2023	0	1 day ago
assign3_2023	0	1 week ago
assign4_2023	0	1 week ago
assign5_2023	0	1 week ago

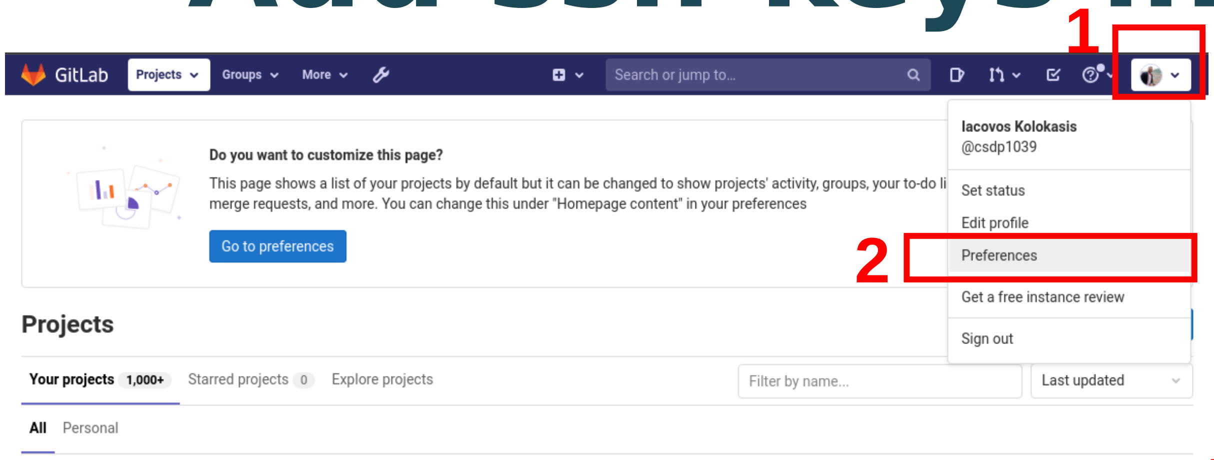
Generating ssh-keys

- Run: ssh-keygen -t rsa -b 2048 -C "kolokasis@ics.forth.gr"
- Press enter in the next three options
- Run: cat ~/.ssh/id_rsa.pub
- Copy the printed text



```
[root@desktop ~]# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:vvDbwVF2YHAoSCxnjhNojc4oEenLJCYgGcsGNyI2Gyw root@desktop.example.com
The key's randomart image is:
+---[RSA 2048]----+
|BOo+ o..  .o+    |
|E=B.+ = . .o .   |
|*X   B   .  o .  |
|*+o o .    o .   |
|B .  .  S .      |
| o     . . .     |
|      . . o      |
|       o o .     |
|        +..      |
+----[SHA256]-----+
```

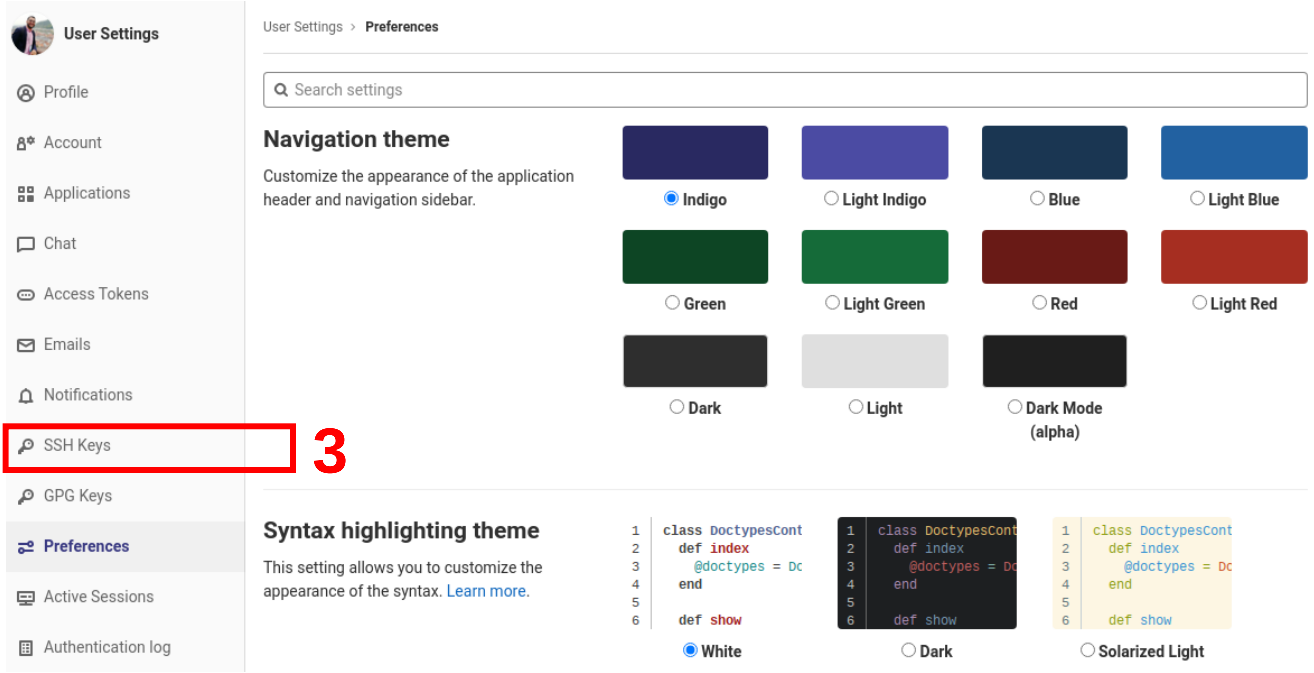
Add ssh-keys in your Account



Do you want to customize this page?

This page shows a list of your projects by default but it can be changed to show projects' activity, groups, your to-do li merge requests, and more. You can change this under "Homepage content" in your preferences

Projects



Navigation theme

Customize the appearance of the application header and navigation sidebar.

● Indigo ○ Light Indigo ○ Blue ○ Light Blue
○ Green ○ Light Green ○ Red ○ Light Red
○ Dark ○ Light ○ Dark Mode (alpha)

Syntax highlighting theme

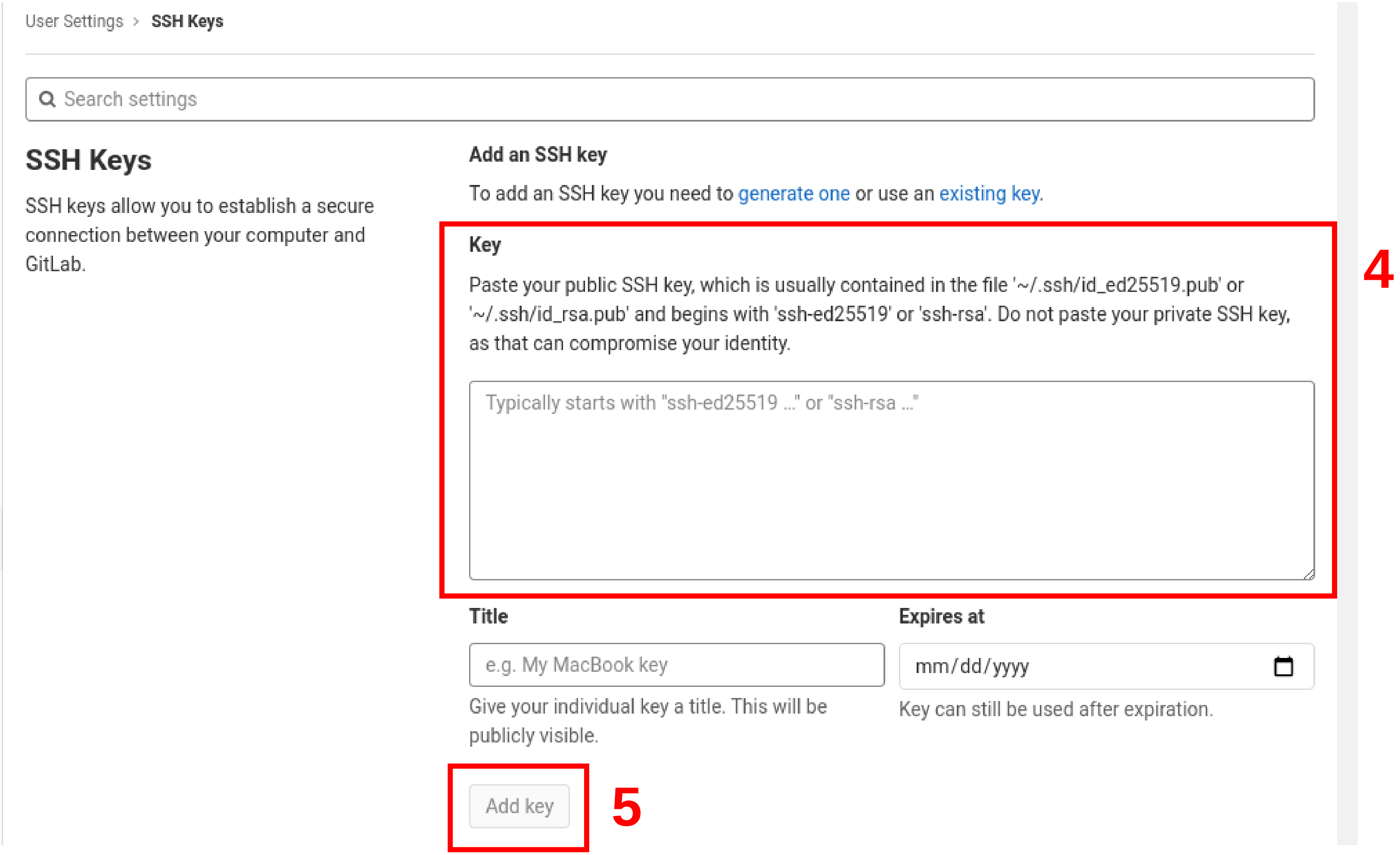
This setting allows you to customize the appearance of the syntax. Learn more.

```
class DoctypesCont
  def index
    @doctypes = Dc
  end

  def show
```

● White ○ Dark ○ Solarized Light

Add ssh-keys in your Account



User Settings › **SSH Keys**

Search settings

SSH Keys

SSH keys allow you to establish a secure connection between your computer and GitLab.

Add an SSH key

To add an SSH key you need to generate one or use an existing key.

Key

Paste your public SSH key, which is usually contained in the file '~/.ssh/id_ed25519.pub' or '~/.ssh/id_rsa.pub' and begins with 'ssh-ed25519' or 'ssh-rsa'. Do not paste your private SSH key, as that can compromise your identity.

Typically starts with "ssh-ed25519 …" or "ssh-rsa …"

Title

e.g. My MacBook key

Give your individual key a title. This will be publicly visible.

Expires at

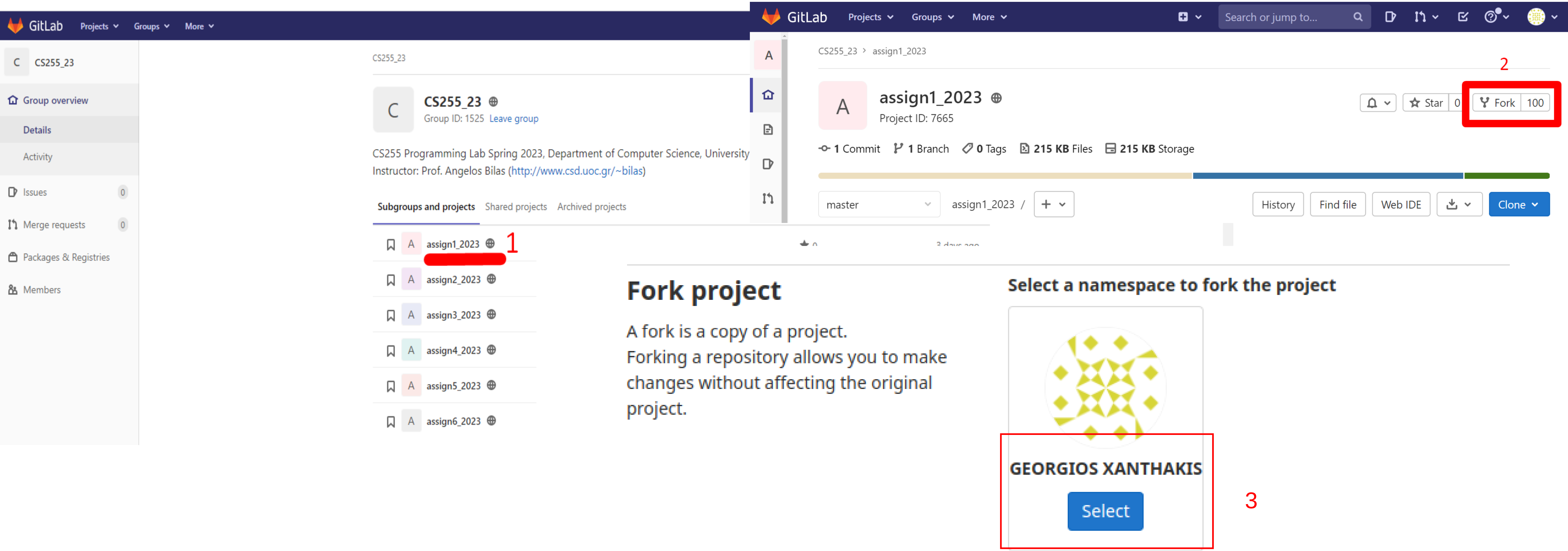
mm/dd/yyyy

Key can still be used after expiration.

4

5

How to Fork a Repository



CS255_23

CS255_23

Group ID: 1525 Leave group

CS255 Programming Lab Spring 2023, Department of Computer Science, University
Instructor: Prof. Angelos Bilas (http://www.csd.uoc.gr/~bilas)

1

CS255_23 > assign1_2023

assign1_2023

Project ID: 7665

1 Commit 1 Branch 0 Tags 215 KB Files 215 KB Storage

2

Fork project

A fork is a copy of a project.
Forking a repository allows you to make changes without affecting the original project.

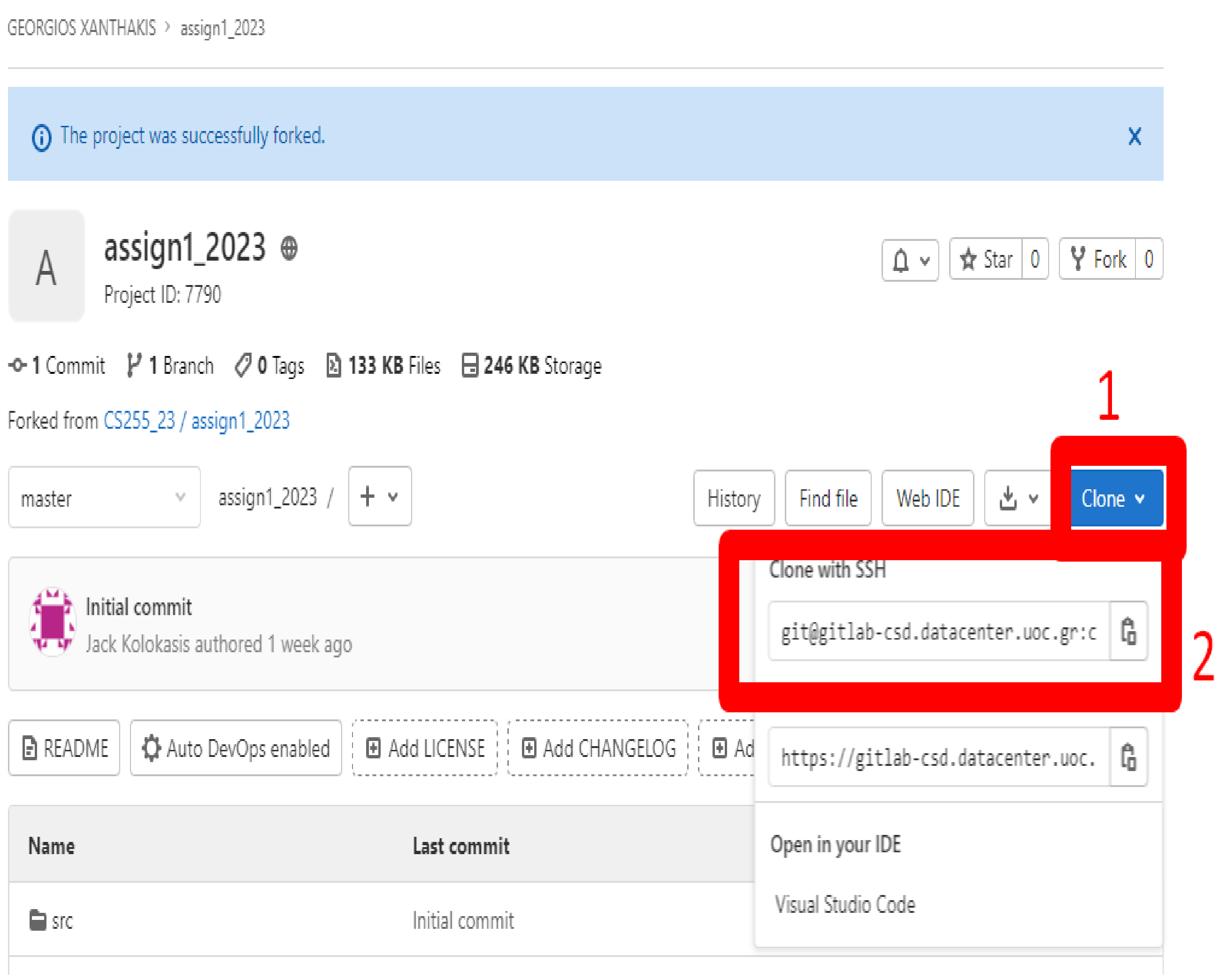
Select a namespace to fork the project

GEORGIOS XANTHAKIS

3

Info About Fork

- You need to fork each assignment **only once!**
- If you want to download your project in different devices (e.g., laptop, desktop PC) you need to clone it



GEORGIOS XANTHAKIS › assign1_2023

The project was successfully forked.

assign1_2023

Project ID: 7790

1 Commit 1 Branch 0 Tags 133 KB Files 246 KB Storage

Forked from CS255_23 / assign1_2023

Initial commit

Jack Kolokasis authored 1 week ago

Clone with SSH

`git@gitlab-csd.datacenter.uoc.gr:c`

`https://gitlab-csd.datacenter.uoc.`

Name	Last commit
src	Initial commit

Make your Project Private

- To make your project private
- Go to Project Settings -> General
- Visibility
- Change Project Visibility to Private

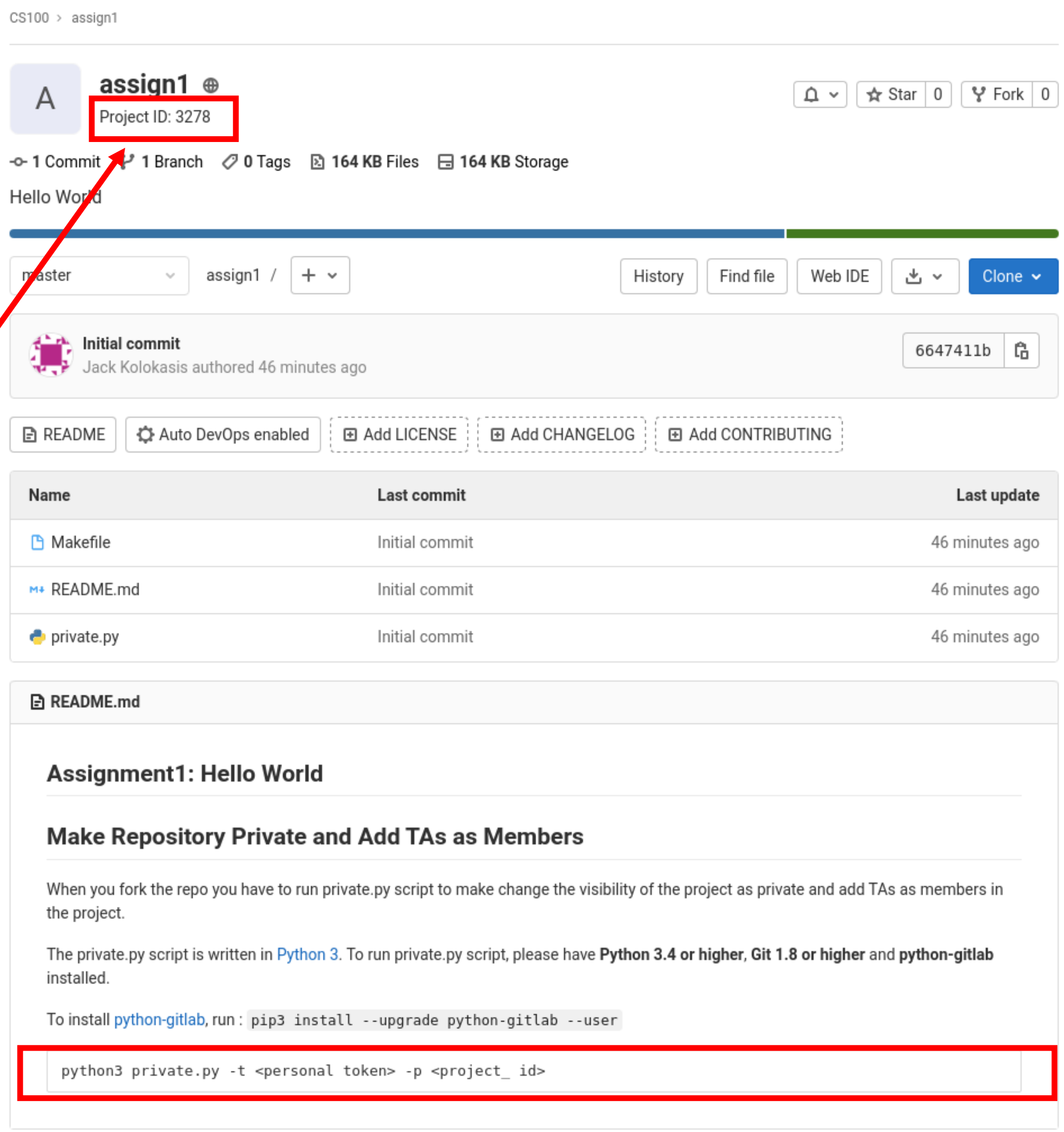
- To add the TAs in your project
- Go to Project Settings -> Members
- Add each TA as a Developer

- But we can avoid to do these steps on every assignment

Using private.py

- Run pip3 install --upgrade python-gitlab --user
- Copy your access token and the assignment project id

python3 private.py -t <accessToken> -p <projectID>



Project ID: 3278

Hello World

Initial commit
Jack Kolokasis authored 46 minutes ago

6647411b

Name	Last commit	Last update
Makefile	Initial commit	46 minutes ago
README.md	Initial commit	46 minutes ago
private.py	Initial commit	46 minutes ago

README.md

Assignment1: Hello World

Make Repository Private and Add TAs as Members

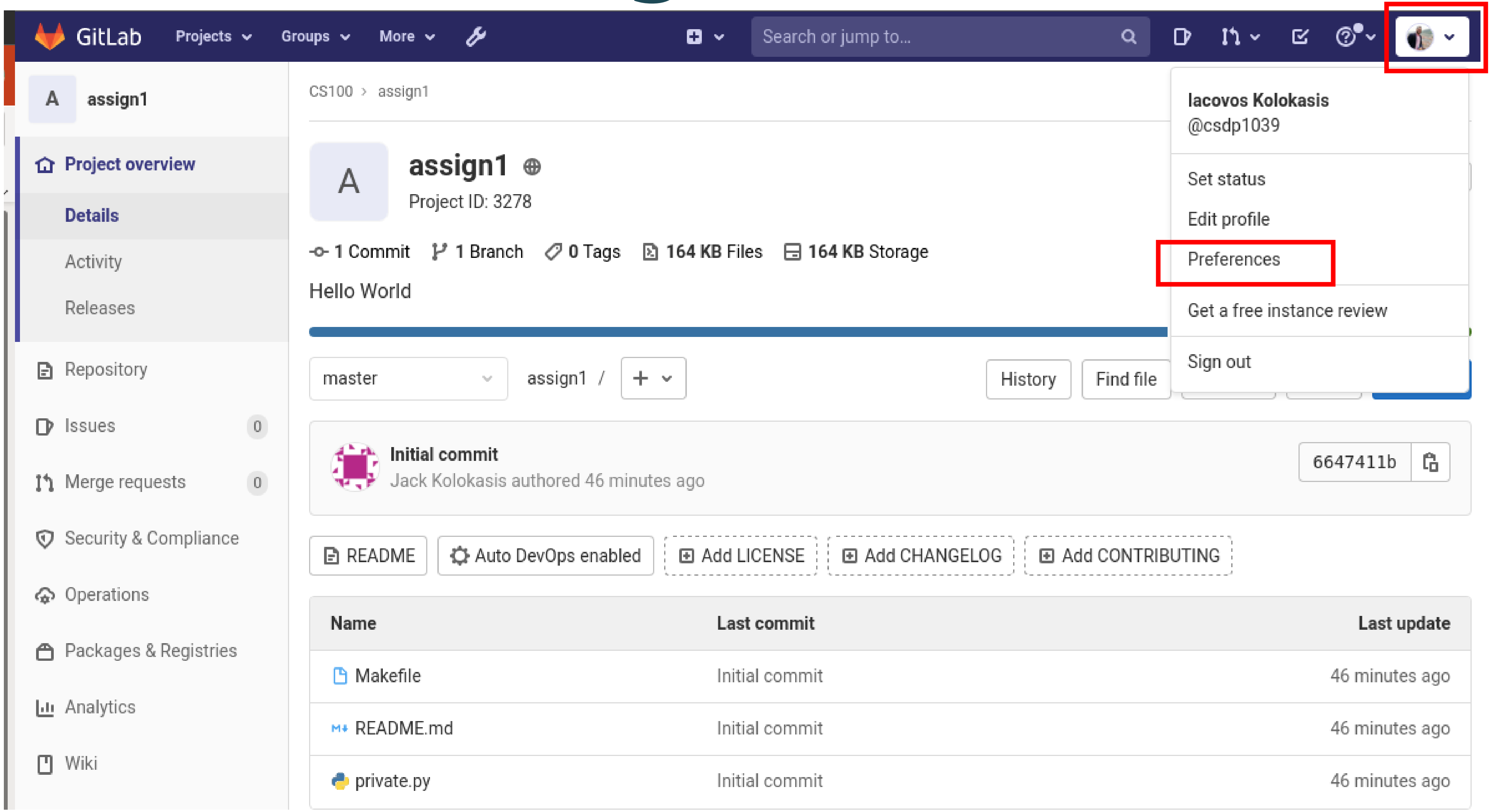
When you fork the repo you have to run private.py script to make change the visibility of the project as private and add TAs as members in the project.

The private.py script is written in Python 3. To run private.py script, please have **Python 3.4 or higher**, **Git 1.8 or higher** and **python-gitlab** installed.

To install python-gitlab, run : `pip3 install --upgrade python-gitlab --user`

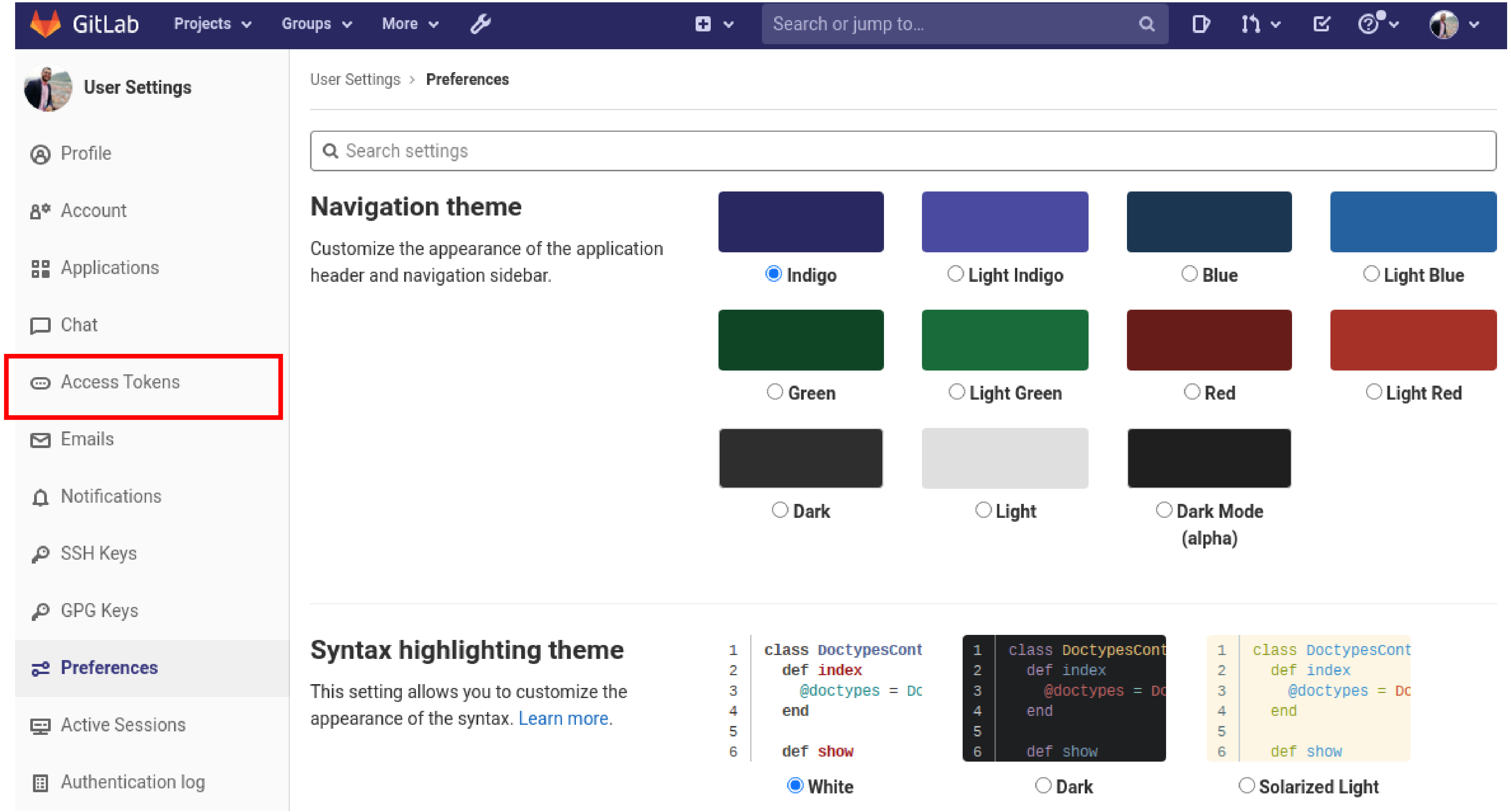
```
python3 private.py -t <personal token> -p <project_ id>
```

Generating an Access Token



Name	Last commit	Last update
Makefile	Initial commit	46 minutes ago
README.md	Initial commit	46 minutes ago
private.py	Initial commit	46 minutes ago

Generating an Access Token



GitLab   Projects ⌄   Groups ⌄   More ⌄

Search or jump to...

User Settings

- Profile
- Account
- Applications
- Chat
- Access Tokens
- Emails
- Notifications
- SSH Keys
- GPG Keys
- **Preferences**
- Active Sessions
- Authentication log

User Settings › **Preferences**

Search settings

Navigation theme

Customize the appearance of the application header and navigation sidebar.

● Indigo   ○ Light Indigo   ○ Blue   ○ Light Blue

○ Green   ○ Light Green   ○ Red   ○ Light Red

○ Dark   ○ Light   ○ Dark Mode (alpha)

Syntax highlighting theme

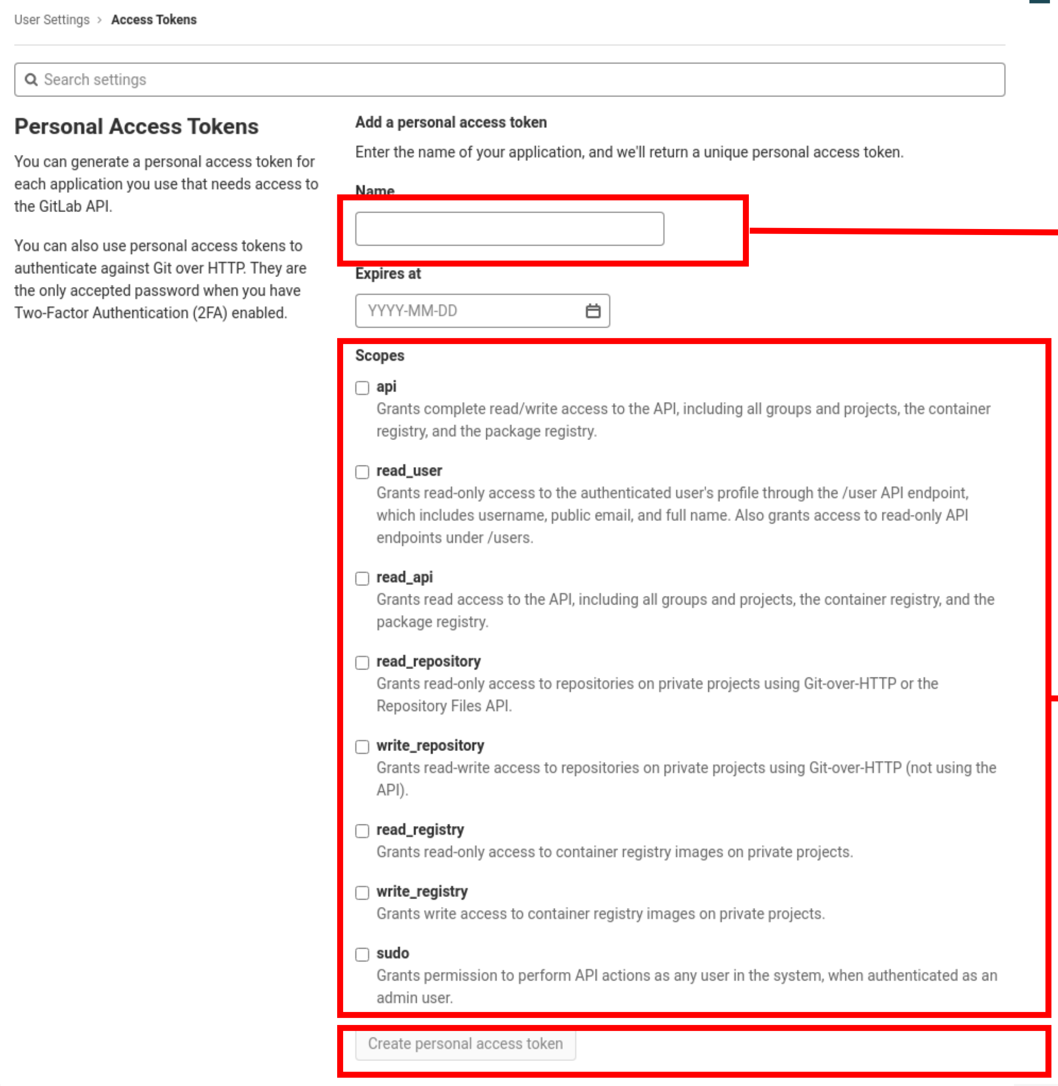
This setting allows you to customize the appearance of the syntax. Learn more.

```
class DoctypesCont
  def index
    @doctypes = Dc
  end

  def show
```

● White   ○ Dark   ○ Solarized Light

Generating an Access Token



User Settings › **Access Tokens**

Search settings

Personal Access Tokens

You can generate a personal access token for each application you use that needs access to the GitLab API.

You can also use personal access tokens to authenticate against Git over HTTP. They are the only accepted password when you have Two-Factor Authentication (2FA) enabled.

Add a personal access token

Enter the name of your application, and we'll return a unique personal access token.

Name

Expires at

YYYY-MM-DD

Scopes

☐ **api**
Grants complete read/write access to the API, including all groups and projects, the container registry, and the package registry.

☐ **read_user**
Grants read-only access to the authenticated user's profile through the /user API endpoint, which includes username, public email, and full name. Also grants access to read-only API endpoints under /users.

☐ **read_api**
Grants read access to the API, including all groups and projects, the container registry, and the package registry.

☐ **read_repository**
Grants read-only access to repositories on private projects using Git-over-HTTP or the Repository Files API.

☐ **write_repository**
Grants read-write access to repositories on private projects using Git-over-HTTP (not using the API).

☐ **read_registry**
Grants read-only access to container registry images on private projects.

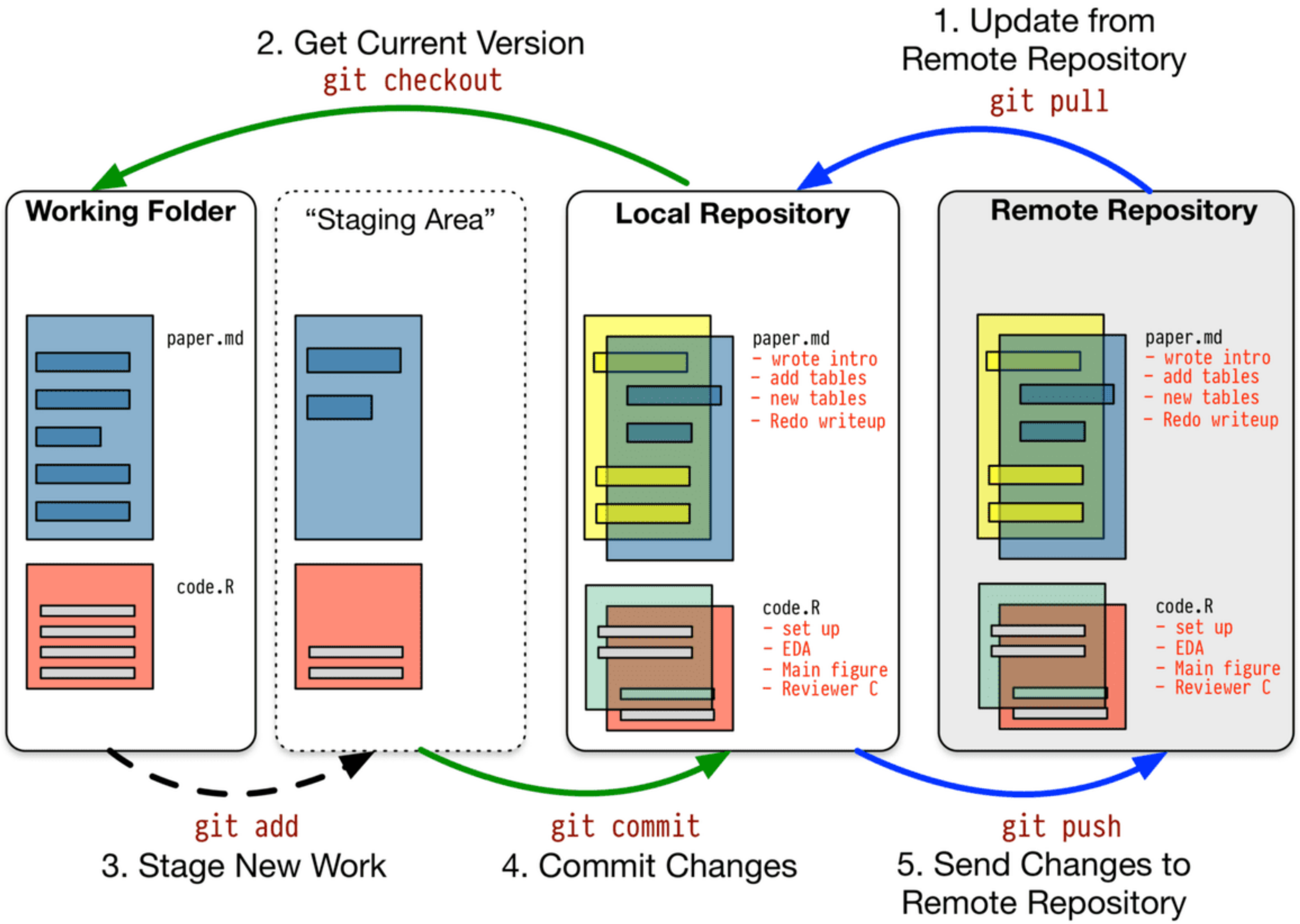
☐ **write_registry**
Grants write access to container registry images on private projects.

☐ **sudo**
Grants permission to perform API actions as any user in the system, when authenticated as an admin user.

Add your name e.g., Iacovos Kolokasis

Select all the options

How does Git Work?



How to Use Git?

- To download a git repo run: git clone repo-url
- To setup your credentials in the local git repo run:
  git config user.name "Giorgos Xanthakis"
  git config user.email "gxanth@csd.uoc.gr"
- To stage a file run: git add file
- To commit staged files run: git commit –m "Your Message"
- To save your code in the server run: git push

Guidelines

- Commit often – at least one commit for each step
- Pay attention to your commit messages
- Describe precisely the purpose of the commit
- Commit only relevant files and modifications!!
- Push frequently (In case something goes wrong with your PC/laptop)

Thank you for your attention

Giorgos Xanthakis

Iacovos Kolokasis

University of Crete & ICS – FORTH

gxanth@csd.uoc.gr

kolokasis@csd.uoc.gr