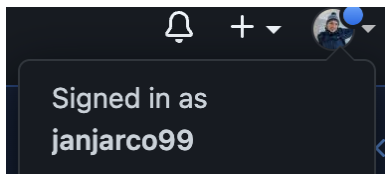


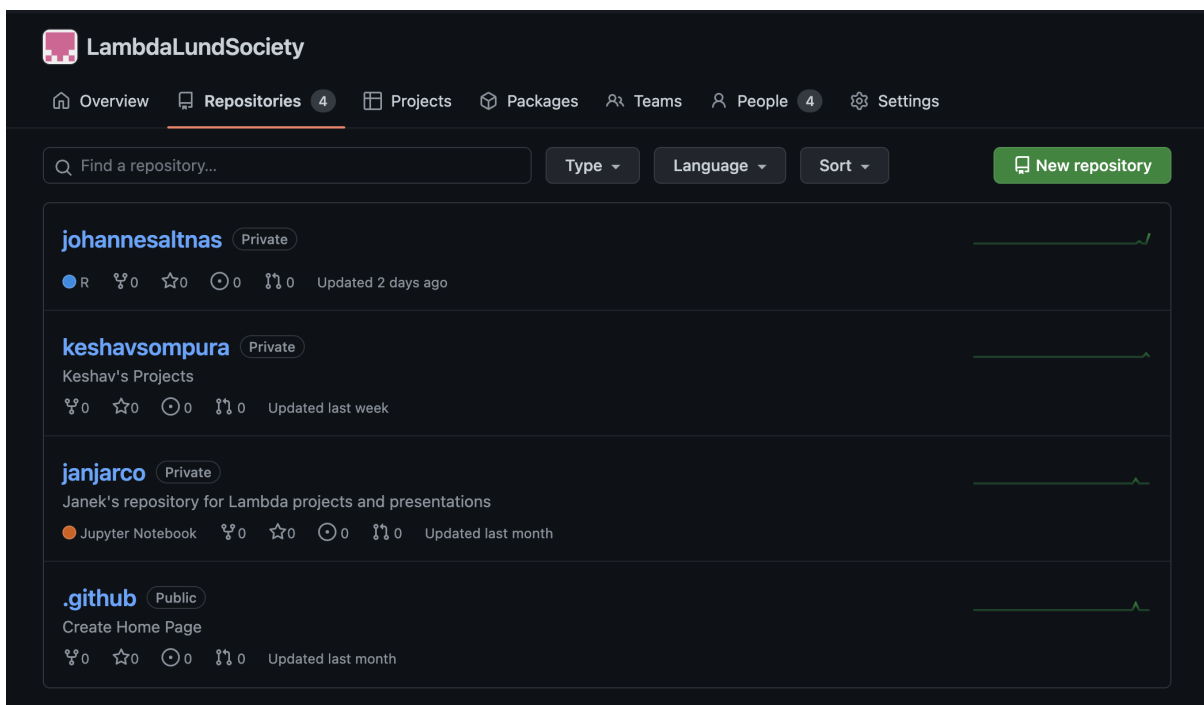
GitHub instructions

In this guide I will guide you through the steps that has to be taken to upload your code and get access to the code uploaded by other Lambda members ~ Janek

Text to Janek, Johannes or Keshav with your GitHub nickname to be added to the **LambdaLundSociety** organization on Git. You will find your nickname in the right upper corner:



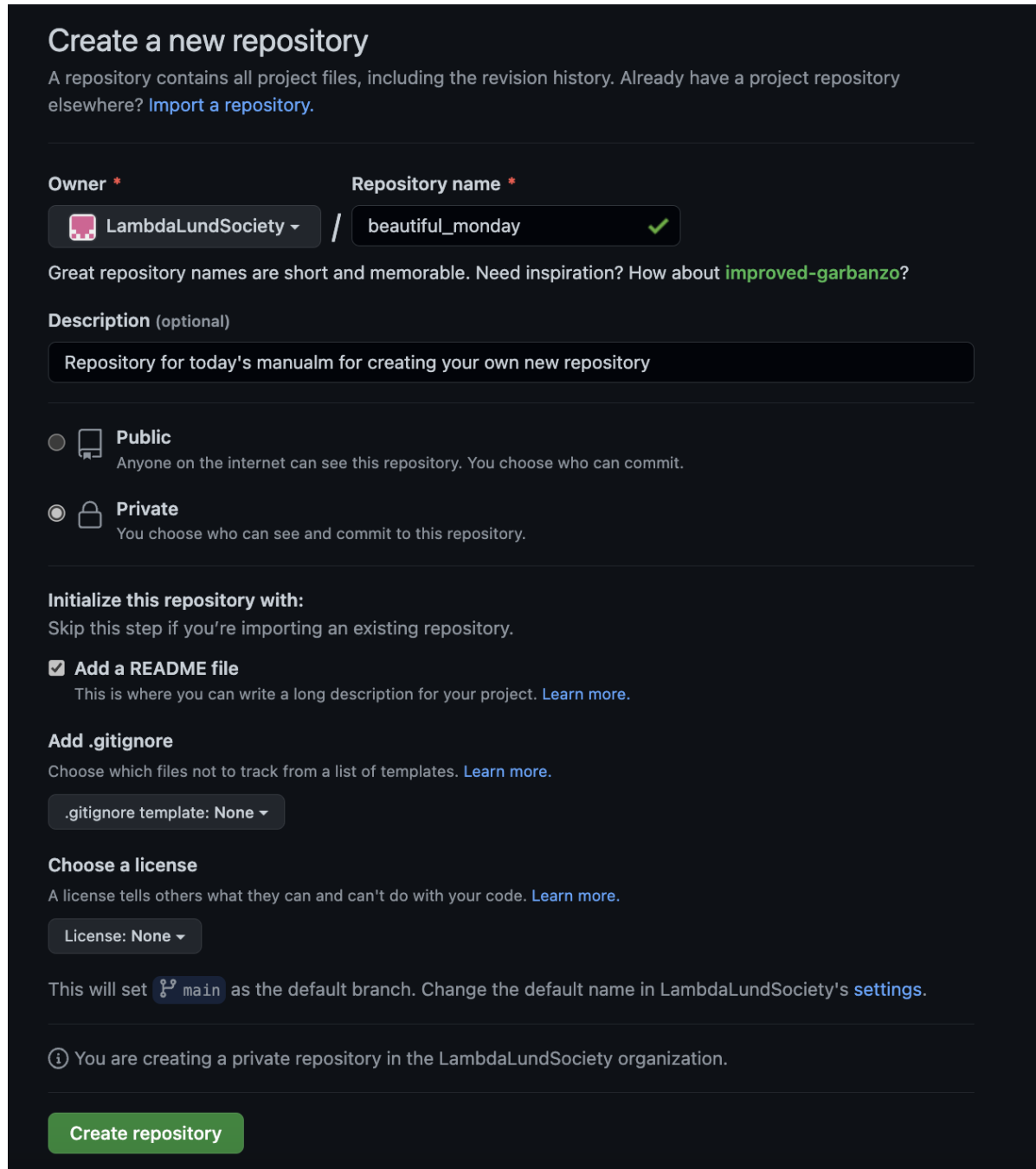
When you'll get access to the organization you will see this view with access to private repositories inside the organization. Otherwise you can see only public repositories, but we don't have right now



Creating new private repository inside the organization

Now we can create our own repository:


1. Click **New repository** button and fill out the template in this way:



Create a new repository

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


Owner * **Repository name ***


 LambdaLundSociety / beautiful_monday ✓

Great repository names are short and memorable. Need inspiration? How about **improved-garbanzo?**

Description (optional)

Repository for today's manualm for creating your own new repository

☐  **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](#).

.gitignore template: None ▾

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

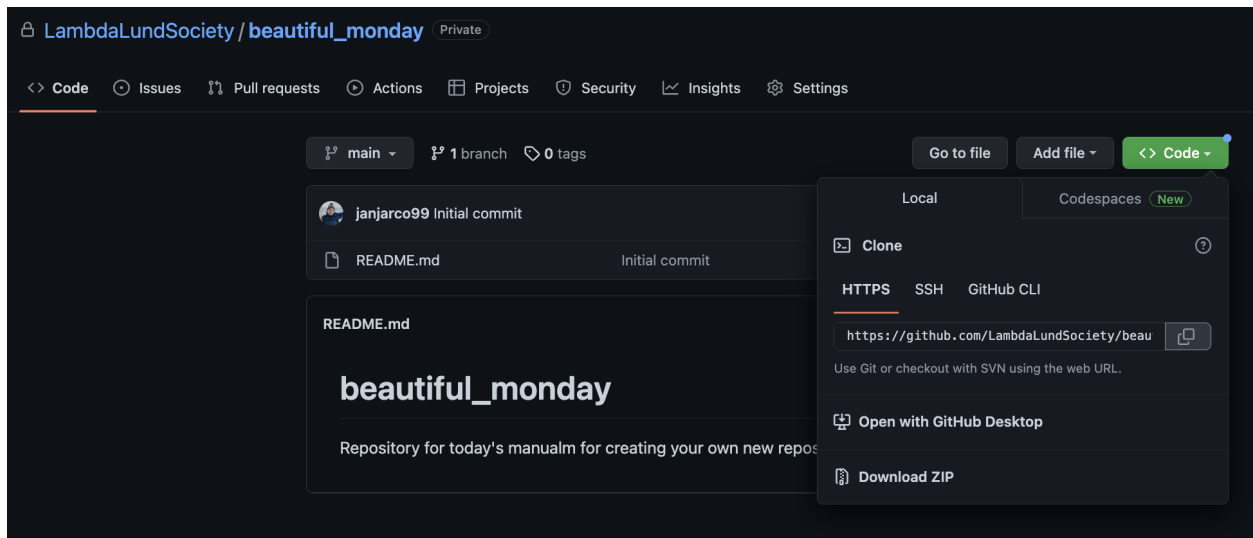
License: None ▾

This will set  **main** as the default branch. Change the default name in LambdaLundSociety's [settings](#).

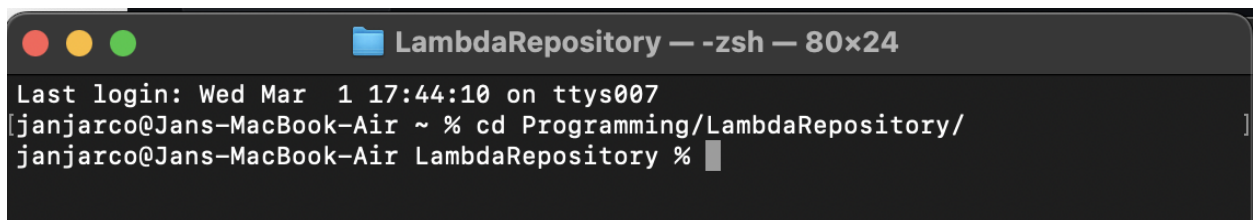
 You are creating a private repository in the LambdaLundSociety organization.

Create repository

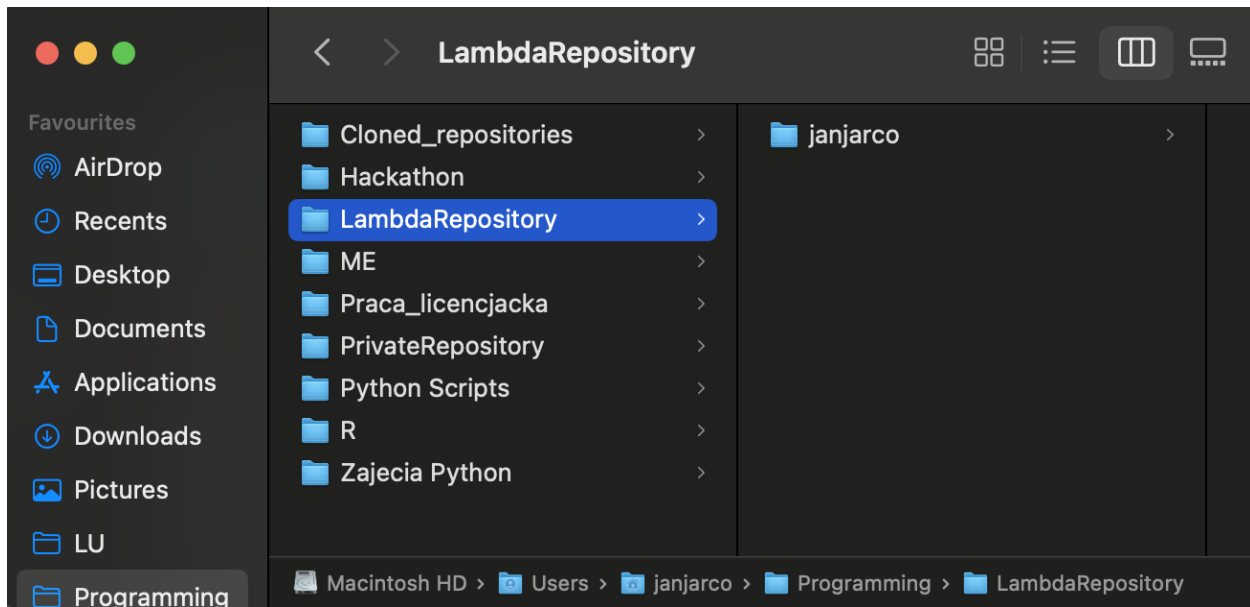
Now you're in the initial view of your own repository. Everyone can get access to it in the same away as below. Click **<> Code** and copy button next to HTTPS link:



Now open **Terminal** and with `cd` command change directory to the folder where you want to keep your files. You can start typing name of following folders and using **Tab** it will be autocompleted.



This is how my folder tree looks like to give you better understanding



Now enter `git clone` command with adding HTTPS link of the repository you want to clone after.

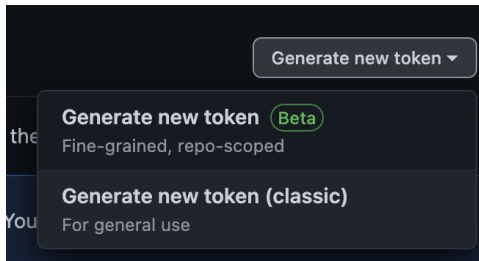
```
LambdaRepository — git-remote-https ◀ git clone https://github.com/Lam...
Last login: Wed Mar 1 17:44:10 on ttys007
[janjarco@Jans-MacBook-Air ~ % cd Programming/LambdaRepository/]
[janjarco@Jans-MacBook-Air LambdaRepository % git clone https://github.com/Lambda
LundSociety/beautiful_monday.git
\Cloning into 'beautiful_monday'...
Username for 'https://github.com': janjarco99
```

Now, you will be asked for your login and then password. But since 2021 GitHub doesn't allow for authentication password, so you have to use **Personal tokens** that you can generate in the settings as explained below.

Generating token for authentication:

Enter this link to the right page in settings: <https://github.com/settings/tokens>

Click **Generate new token (classic)**:



Now, fill the tag name to identify the token, set expiration time period and fill repo checkbox as probably you will only use this functionality:

A screenshot of the 'New personal access token (classic)' form in GitHub. The form has a title 'New personal access token (classic)' and a description: 'Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used to authenticate to the API over Basic Authentication or to Git over HTTPS, or can be used to authenticate to the API over Basic Authentication'. Below this is a 'Note' section with a text input for the token name, which is filled with 'TokenMarch2023'. Underneath is a question 'What's this token for?'. Then there is an 'Expiration' section with a dropdown menu set to '30 days' and a note 'The token will expire on Wed, Apr 5 2023'. Below that is a 'Select scopes' section with a link 'Read more about OAuth scopes.' and a list of scopes with checkboxes: 'repo' (checked), 'repo:status', 'repo_deployment', 'public_repo', 'repo:invite', and 'security_events'. Each scope has a description of its permissions.

After confirming the settings you will see your long personal token and keep it in some safe space, it's the last time you see this token so save it in a safe space.

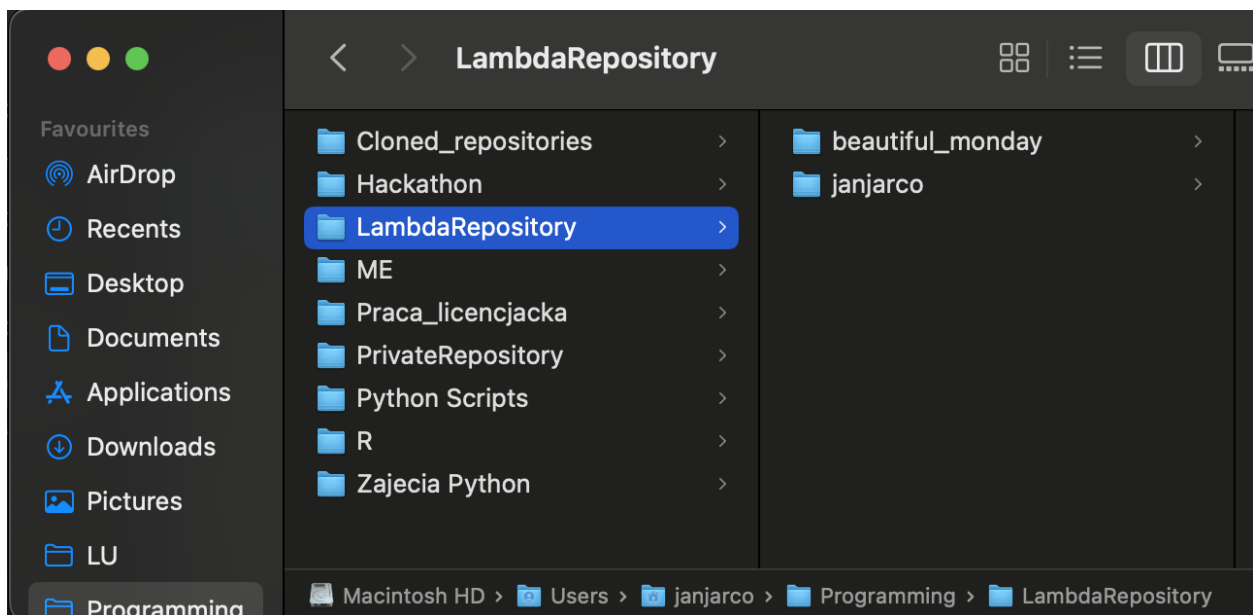
Alright! Now enter your username and then the token and press Enter!

Now, this is how your terminal will look like:

```
LambdaRepository — zsh — 80x24
fatal: Authentication failed for 'https://github.com/LambdaLundSociety/beautiful_monday.git/'
[Janjarco@Jans-MacBook-Air LambdaRepository % git clone https://github.com/LambdaLundSociety/beautiful_monday.git
Cloning into 'beautiful_monday'...
Username for 'https://github.com': janjarco99
[Password for 'https://janjarco99@github.com':
remote: Support for password authentication was removed on August 13, 2021.
remote: Please see https://docs.github.com/en/get-started/getting-started-with-git/about-remote-repositories#cloning-with-https-urls for information on currently recommended modes of authentication.
fatal: Authentication failed for 'https://github.com/LambdaLundSociety/beautiful_monday.git/'
[Janjarco@Jans-MacBook-Air LambdaRepository % git clone https://github.com/LambdaLundSociety/beautiful_monday.git
Cloning into 'beautiful_monday'...
Username for 'https://github.com': janjarco99
[Password for 'https://janjarco99@github.com':
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
Janjarco@Jans-MacBook-Air LambdaRepository %
```

And now you will see the repository itself on your computer, my is called

`beautiful_monday`:

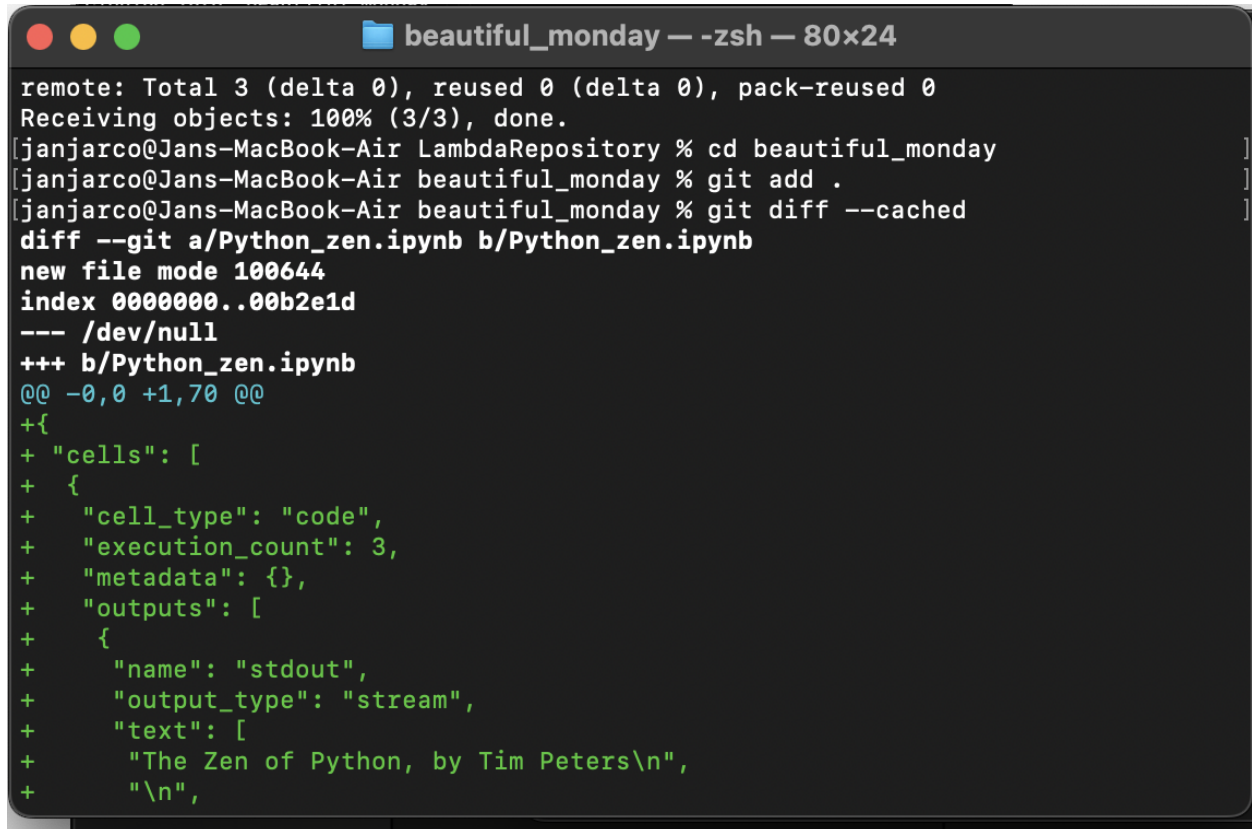


But it's a little empty here 😞 Okay, so let's upload some files there!

I copied two files into the repository. While doing a project of course this folder will be your main repository and you can easily update or upload new files with following commands:

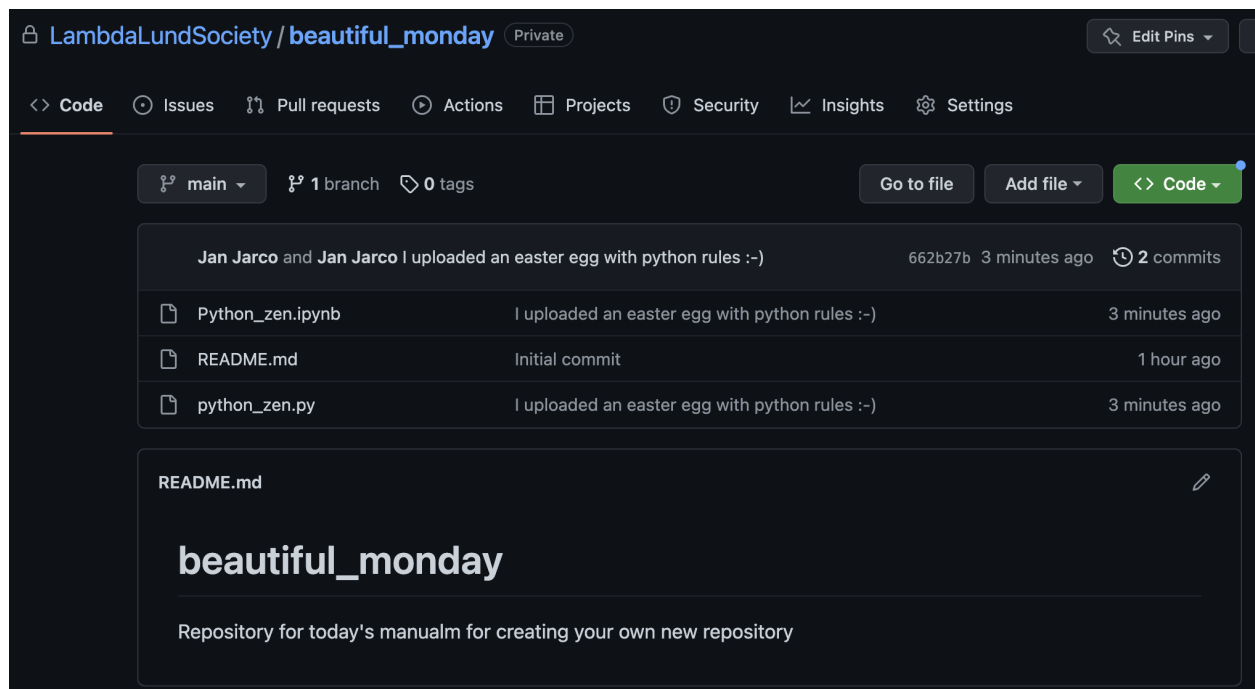
1. `git add .` - this will add all files with their updates

2. `git diff --cached` - you can see what changes will you execute in git repository. Green coloured are new lines, red coloured are deleted ones, etc. everything clearly explained. You can explore all the changes with clicking **Enter** or just click **Q** if you're done with checking these changes.



```
beautiful_monday — -zsh — 80x24
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
[janjarco@Jans-MacBook-Air LambdaRepository % cd beautiful_monday
[janjarco@Jans-MacBook-Air beautiful_monday % git add .
[janjarco@Jans-MacBook-Air beautiful_monday % git diff --cached
diff --git a/Python_zen.ipynb b/Python_zen.ipynb
new file mode 100644
index 0000000..00b2e1d
--- /dev/null
+++ b/Python_zen.ipynb
@@ -0,0 +1,70 @@
+{
+  "cells": [
+    {
+      "cell_type": "code",
+      "execution_count": 3,
+      "metadata": {},
+      "outputs": [
+        {
+          "name": "stdout",
+          "output_type": "stream",
+          "text": [
+            "The Zen of Python, by Tim Peters\n",
+            "\n",
```

3. `git commit -m "I uploaded an easter egg with python rules :-)"` -this comment will be shown with changes you made to the repository.
4. `git push origin main` - with this command all the changes will be pushed to the main branch



Now, these files are available in the repository 😊