GIT & GITHUB WORKSHOP

A SMALL INTODUCTION

WHAT IS GIT ?

- Git is a distributed version-control system tracking source code changes during software development.
- It is designed for coordinating work among programmers, but it can be used to track changes in any set of files.
- Its goals include speed, data integrity, and support for distributed, non-linear workflows.

WHY SHOULD I CARE ABOUT IT?

You can keep track of changes in your file/project.

- That means no more last.c, lastlast.c, truelast.c etc

You can have different workflows on the same project, without worrying about collateral damage (up to a certain point of course)

- More on that later…

HOW IT WORKS THEN?

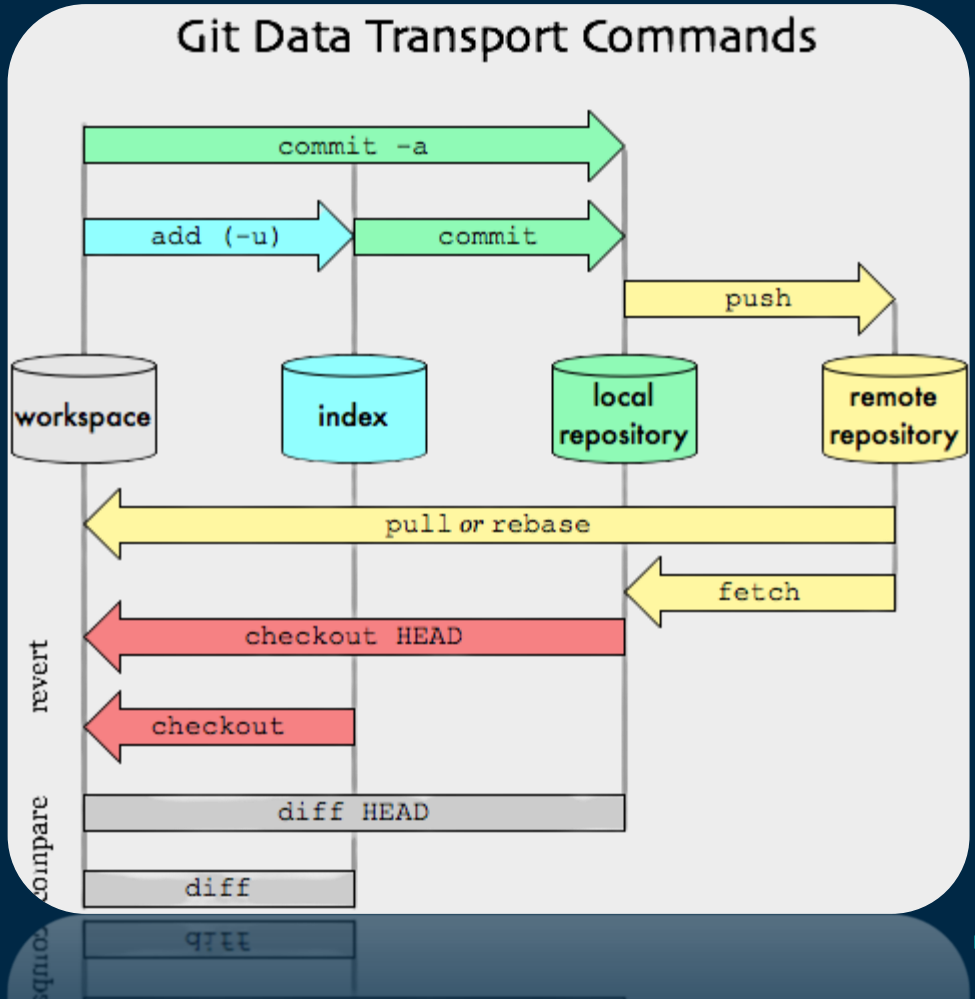
1.First, you start by initializing the repository.

- Repository is the word used to describe your project & all the complementary files git adds.

2. Start your project.

3. After a while, stage your project.

How git staging/saving works:



HOW DOES IT WORK? (again )

1. First, you start by initializing the repository. (Initial Step)

2. Start your project.

3. Commit the staged file(s) to the local repository. **(Whenever)**

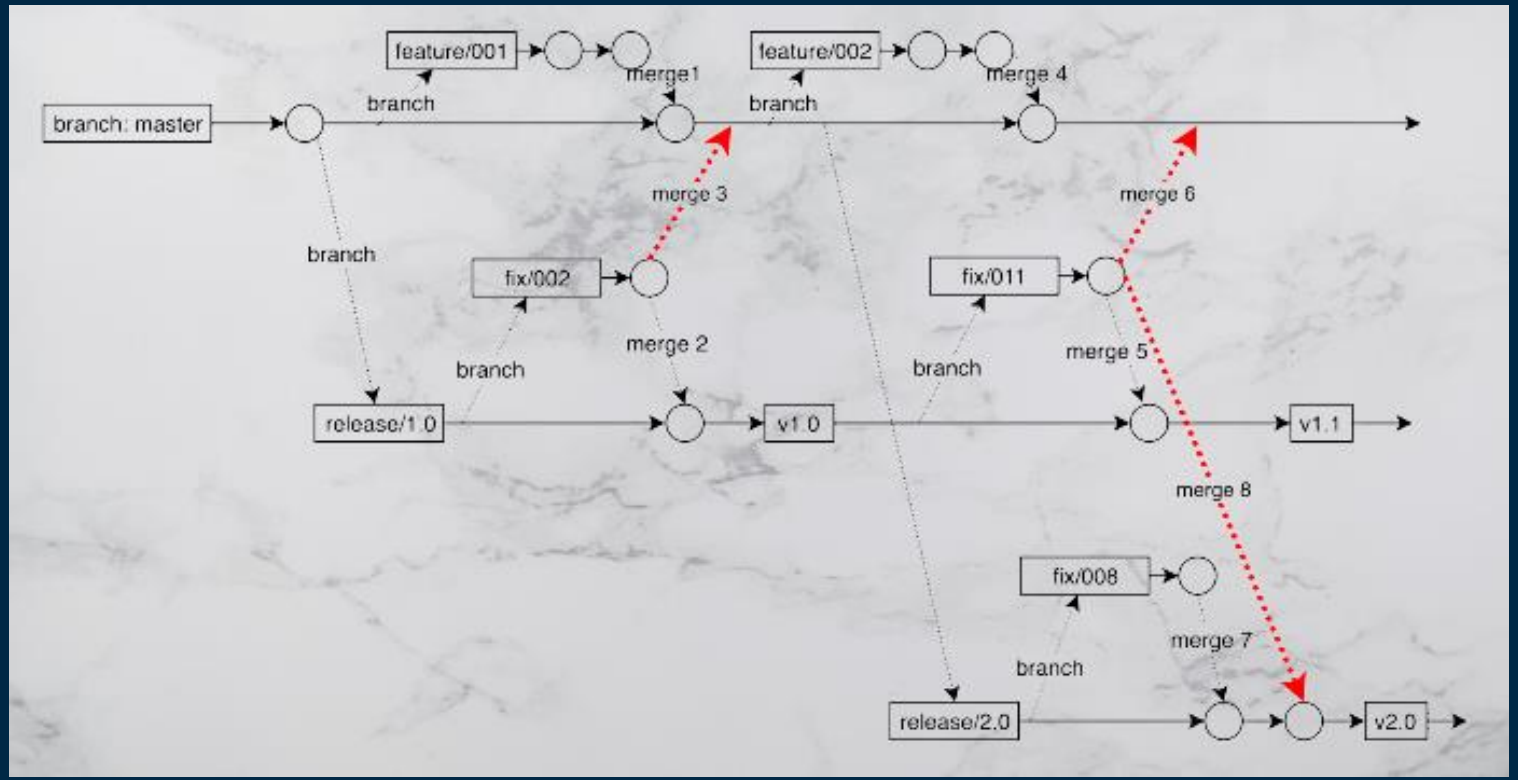
4. Push your commit to the remote repository. **(Whenever)**

NICE BUT IS THAT ALL?

Not at all! Git has more features that make it more robust (but we will not see them in the workshop):

- Tags
  - ➢ Mark specific commits with a label to identify/find/promote them.
- Submodules
  - ➢ Use different git repositories/projects as part of your project.
- Branches

BRANCH


feature/001
merge1
feature/002
merge 4
branch
branch
branch: master
merge 3
merge 6
branch
fix/002
fix/011
merge 2
merge 5
branch
branch
release/1.0
v1.0
v1.1
merge 8
fix/008
merge 7
branch
release/2.0
v2.0

Top **Source Code** Repository **Hosts**

GitHub

WHAT IS GITHUB?

- GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere.

- It supports remote repositories, which are either public or private, and for those, it supports wikis, custom webpages, issue tracking, etc.

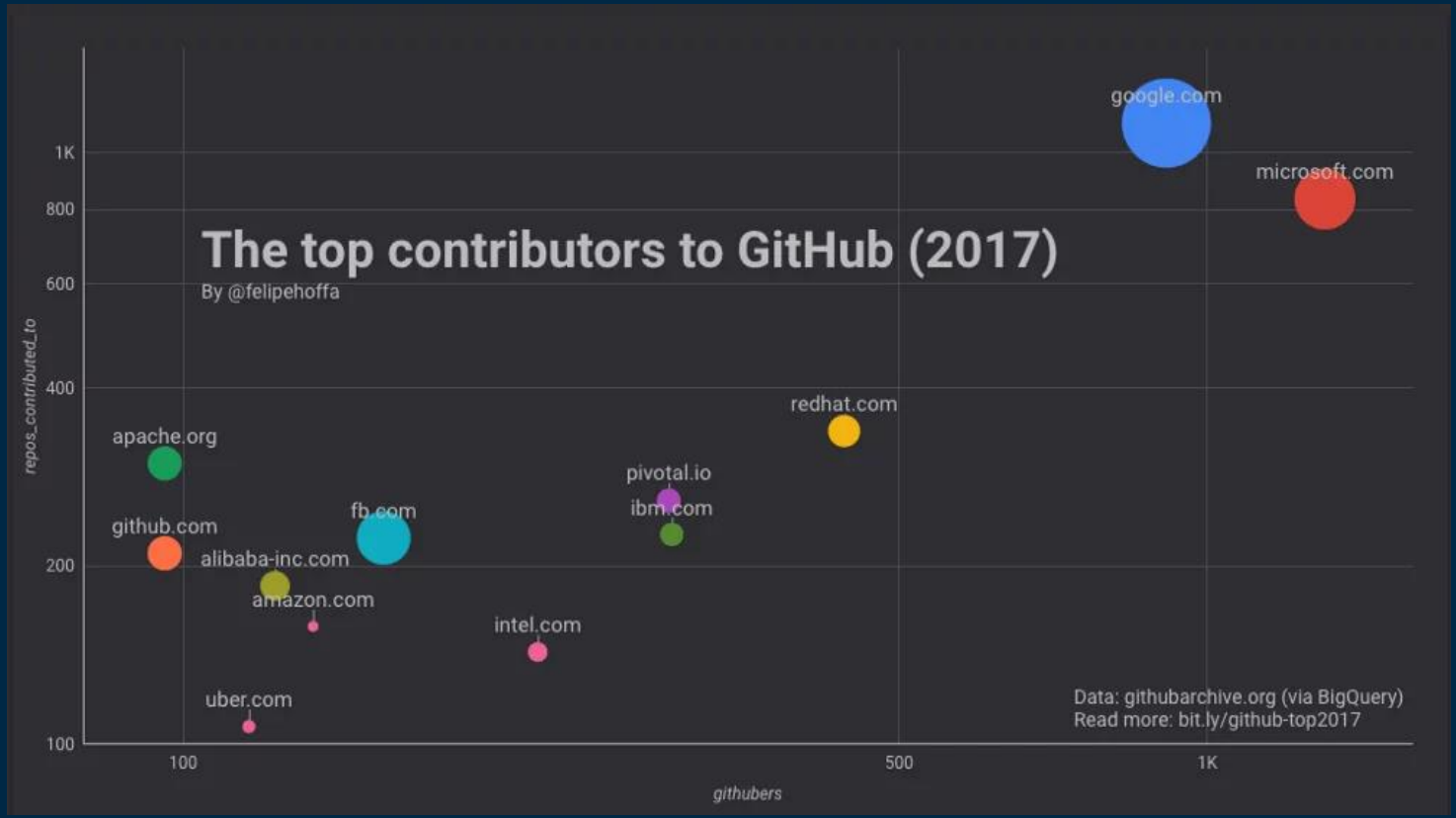
- GitHub is the largest code hosting platform.

WHY I CHOUSE GITHUB

- IS THE BEST GIT PROVIDER
- UNLIMITED PUBLIC AND PRIVATE REPOS
- UNLIMITED COLLABORATORS
- OPEN SOURCE
- GITHUB ACTIONS
- SECURITY OF THE CODE
- INSIGHTS
- MARKETPLACE
- CODESPACE
- STUDENT DEVELOPER PACK
- COPILOT

100 MILIONS

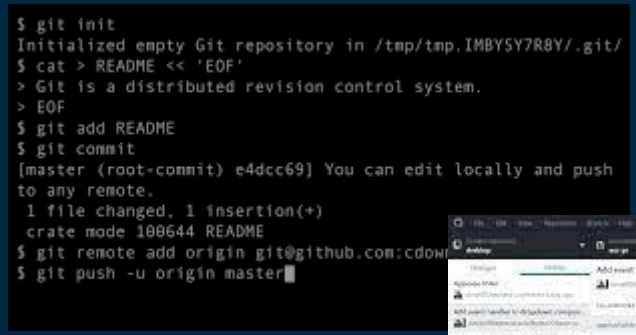
Developers on GitHub

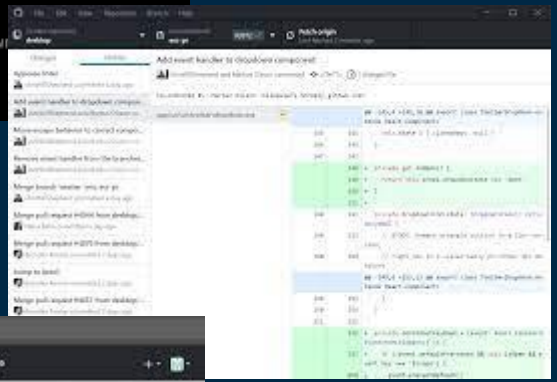

The top contributors to GitHub (2017)
By @felipehoffa
google.com
microsoft.com
redhat.com
apache.org
pivotal.io
ibm.com
fb.com
github.com
alibaba-inc.com
amazon.com
intel.com
uber.com
Data: githubarchive.org (via BigQuery)
Read more: bit.ly/github-top2017
repos_contributed_to
githubers
1K
800
600
400
200
100
100
500
1K

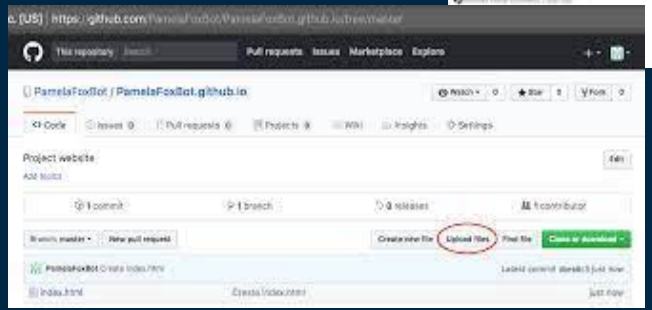
The Workshop

METHODS FOR GITHUB

- Git
- Web application
- GitHub Desktop
- Extensions from IDEs
- GitHub command line







Install Git

Linux

Different by Linux distribution
https://git-scm.com/download/linux

Mac OS

https://git-scm.com/download/mac

Windows

https://git-scm.com/download/win

For all!

Step 0: Configuration

Linux

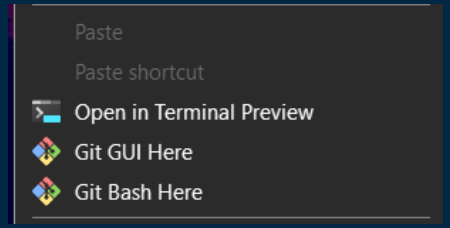
Open a terminal

Mac Os

Open a terminal

Windows

Right-click in an empty space.



Click on the option:

Git Bash Here

Step 0: Configuration

Write the following command on the terminal:

- git config --global user.name "Your name"

  The name must be written the same way as it is on GitHub.

- git config --global user.email "Your email"

  The email must be written the same way as it is on GitHub.

Step 1: Initialization

- Create a folder with the name:

  MyFirstGitRepo

Step 1: Initialization

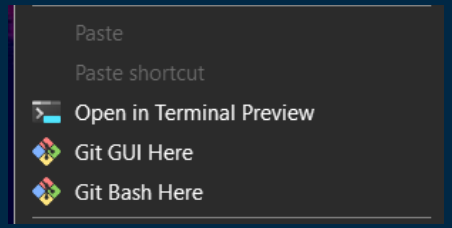
Linux / Mac / Win. bash

Open a terminal, go to the new folder and type the following command:

git init

Windows GUI

Right-click in an empty space.



Click on the option:

Git GUI Here

Step 1: Initialization

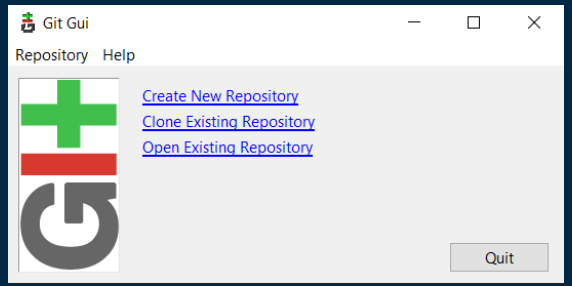
Linux / Mac / Win. bash

Open a terminal, go to the new folder and type the following command:

git init

Windows GUI

Click on the option:
Create New Repository



Step 1: Initialization

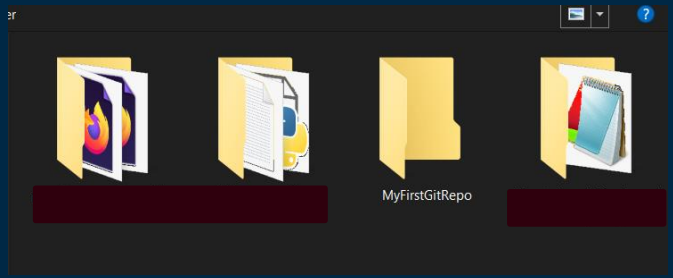
Linux / Mac / Win. bash

Open a terminal, go to the new folder and type the following command:

git init

Windows GUI

Use the navigator to find the folder



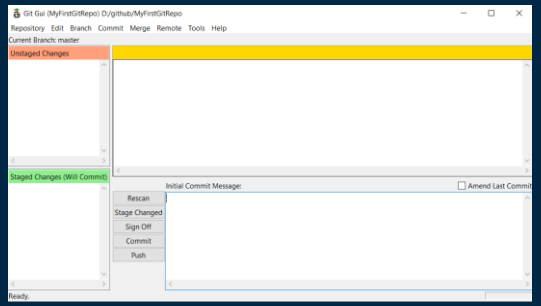
Step 1: Initialization

Linux / Mac / Win. bash

Open a terminal, go to the new folder and type the following command:

git init

Windows GUI



Step 2: Your Project

Create a text file with the name:
git_test.txt

Inside write the following sentence:
Hi, my name is <your_name>!

Step 3: Staging

Linux / Mac / Win. bash

Type the following command on the terminal:

git status

Windows GUI

On the Git GUI, click the button: **Rescan**

Step 3: Staging

Linux / Mac / Win. bash

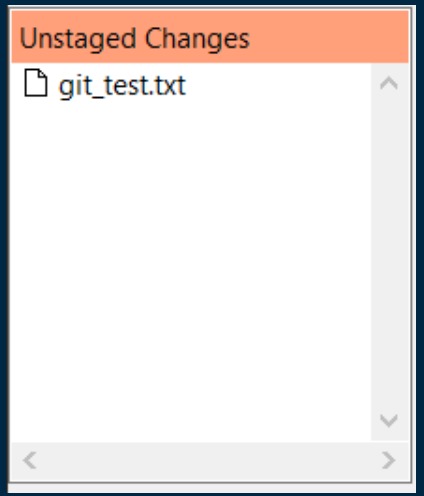
```
→  MyFirstGitRepo git:(master) X git status
On branch master

No commits yet

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

nothing added to commit but untracked files present (use "git add" to track)
→  MyFirstGitRepo git:(master) X
```

Windows GUI



Step 3: Staging

Linux / Mac / Win. bash

Type the following command on the terminal:

git add -A

Windows GUI

On the Git GUI, click the button:
Stage Changed

Step 3: Staging

Linux / Mac / Win. bash

Type the following command on the terminal:

git status

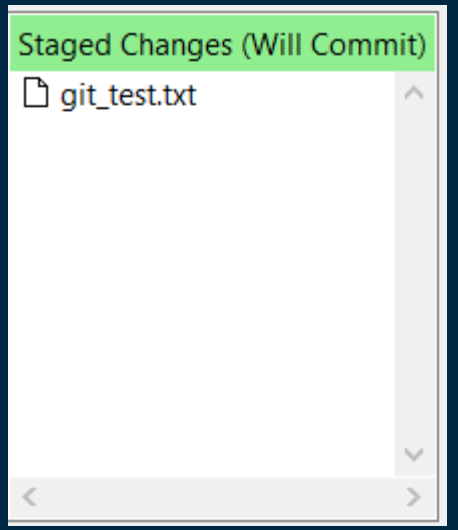
```
→  MyFirstGitRepo git:(master) ✗ git add -A
→  MyFirstGitRepo git:(master) ✗ git status
On branch master

No commits yet

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

→  MyFirstGitRepo git:(master) ✗
```

Windows GUI



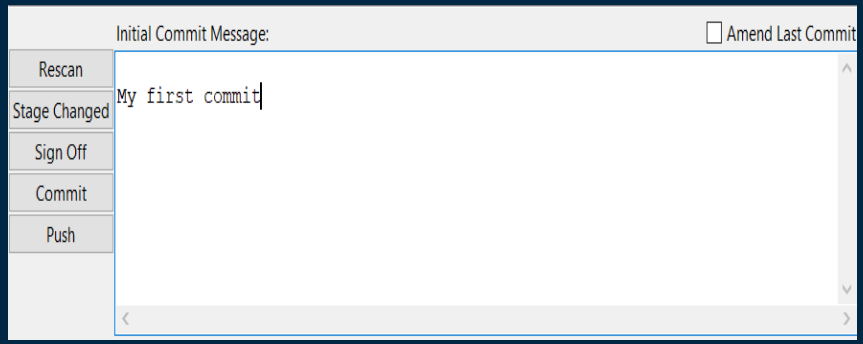
Step 4: Commit

Linux / Mac / Win. bash

Type the following command on the terminal:

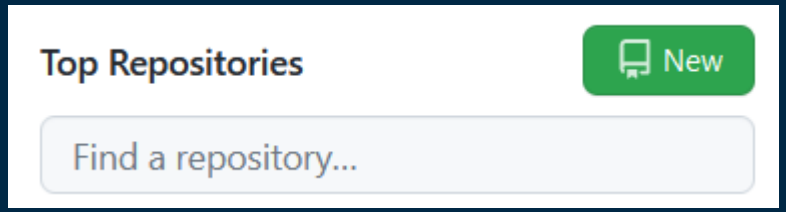
git commit –m “My first commit”

Windows GUI



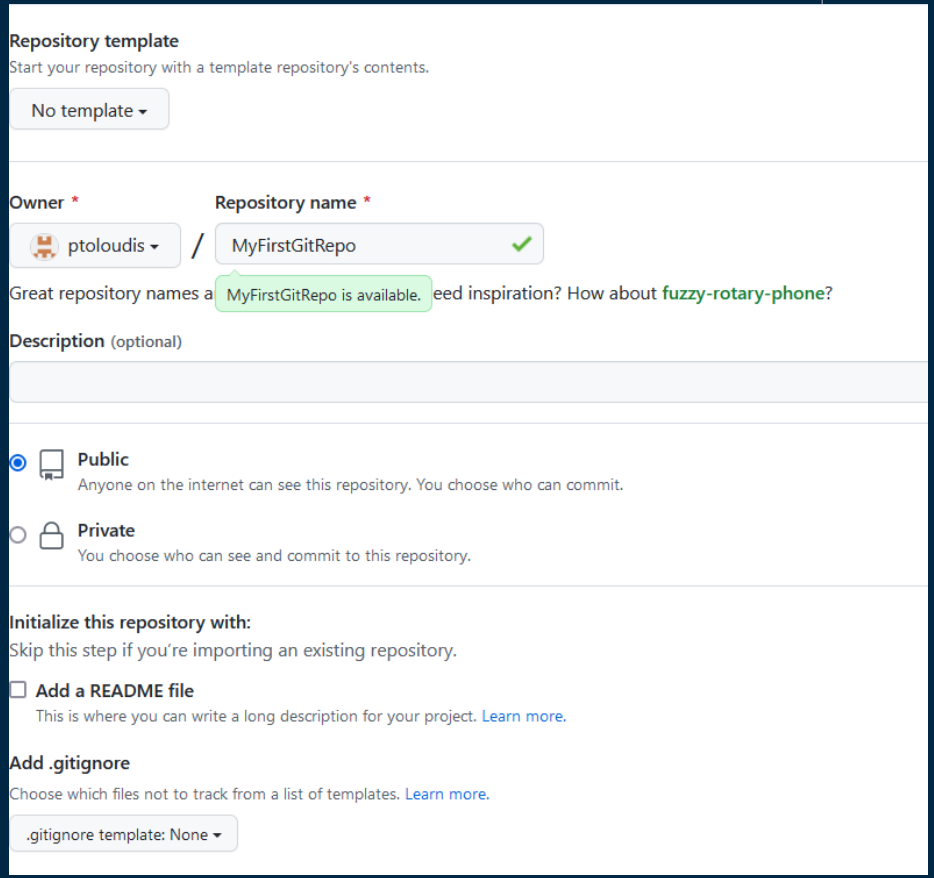
Step 5: Remote Repo / GitHub

Go to github.com, log in, and click on the new button.



Step 5: Remote Repo / GitHub

Name the repository:
MyFirstGitRepo
& Click **Create repository**



Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH https://github.com/ptoloudis/MyFirstGitRepo.git

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

```
echo "# MyFirstGitRepo" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/ptoloudis/MyFirstGitRepo.git
git push -u origin main
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/ptoloudis/MyFirstGitRepo.git
git branch -M main
git push -u origin main
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Import code

Windows Only User

Because this step is more complex for the purposes of this Workshop, the Windows user will have to follow the next steps on the terminal.

Step 5: Remote Repo / GitHub

Write the following command on the terminal:

```
git remote add origin
https://github.com/<your_user_name>/MyFirstGitRepo.git
```

Step 5: Remote Repo / GitHub

Write the following command on the terminal:

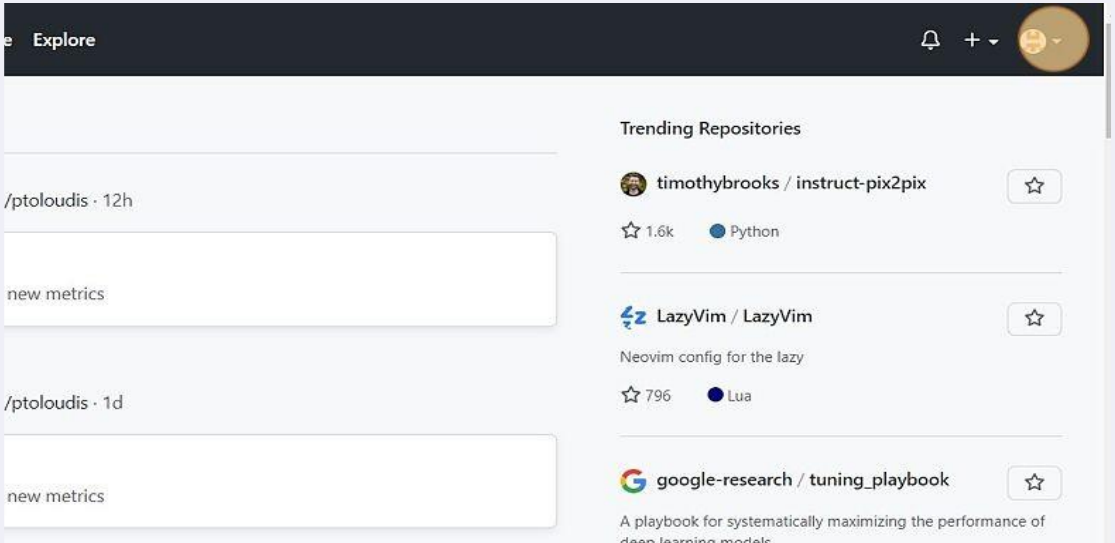
git push origin master

Your credentials for GitHub will be required.

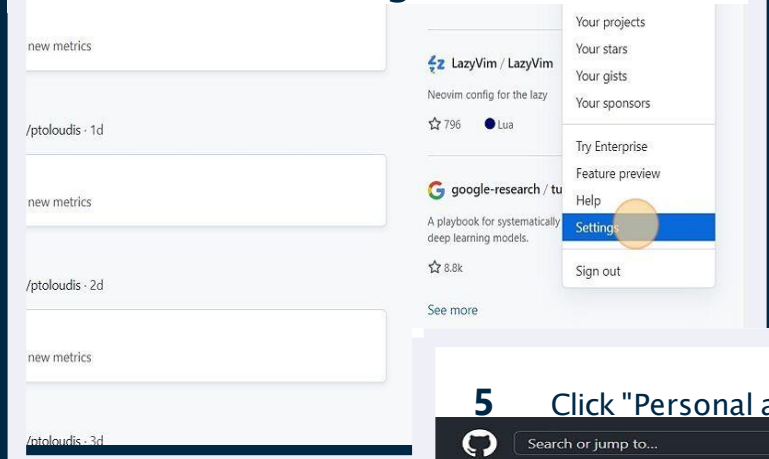
Step 5: Remote Repo / GitHub

If the password does not work:
use the Personal access tokens

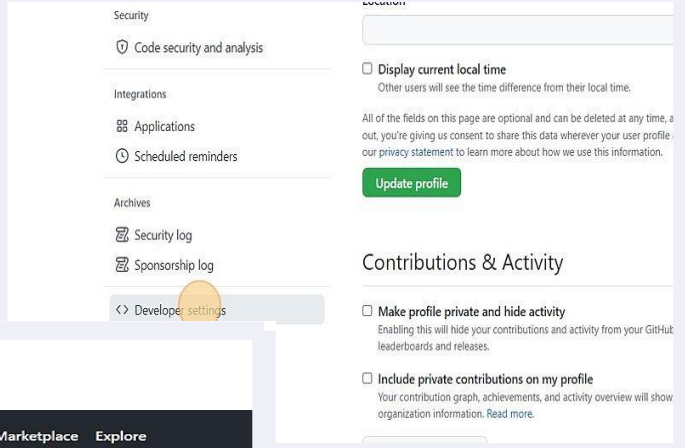
2 Click here.



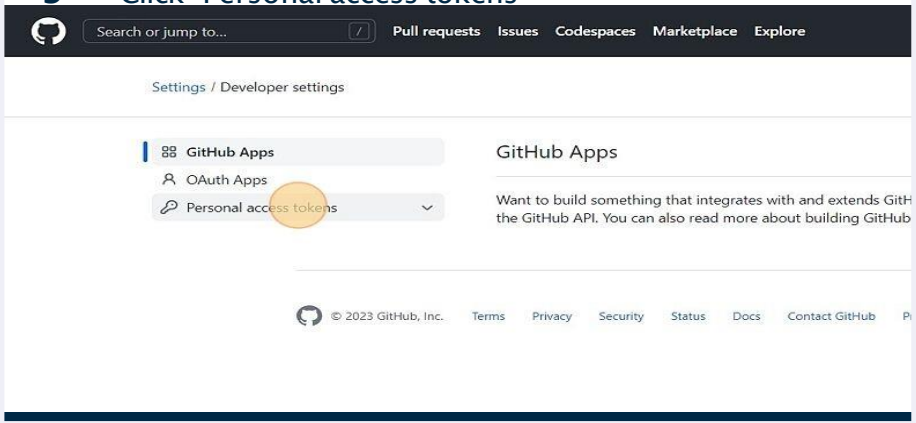
3 Click "Settings"



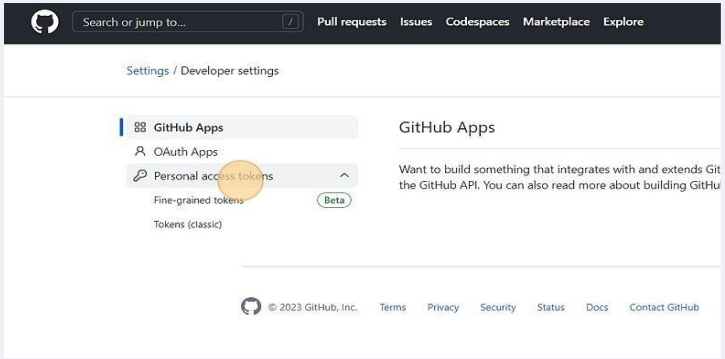
4 Click "Developer settings"



5 Click "Personal access tokens"

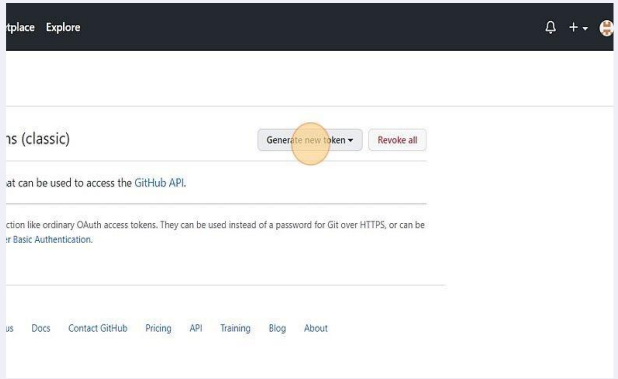


6 Click "Personal access tokens"



7 Click "Tokens (classic)"

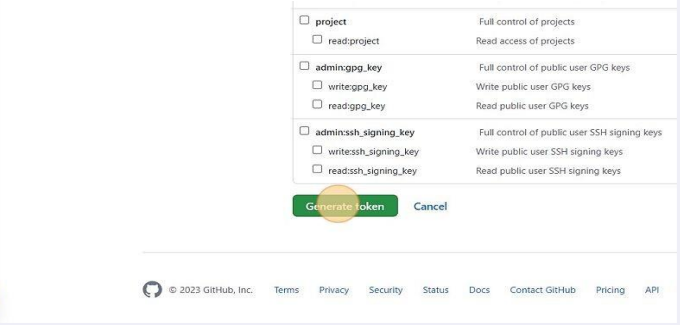
8 Click "Generate new token"



10 Click the "Note" field.

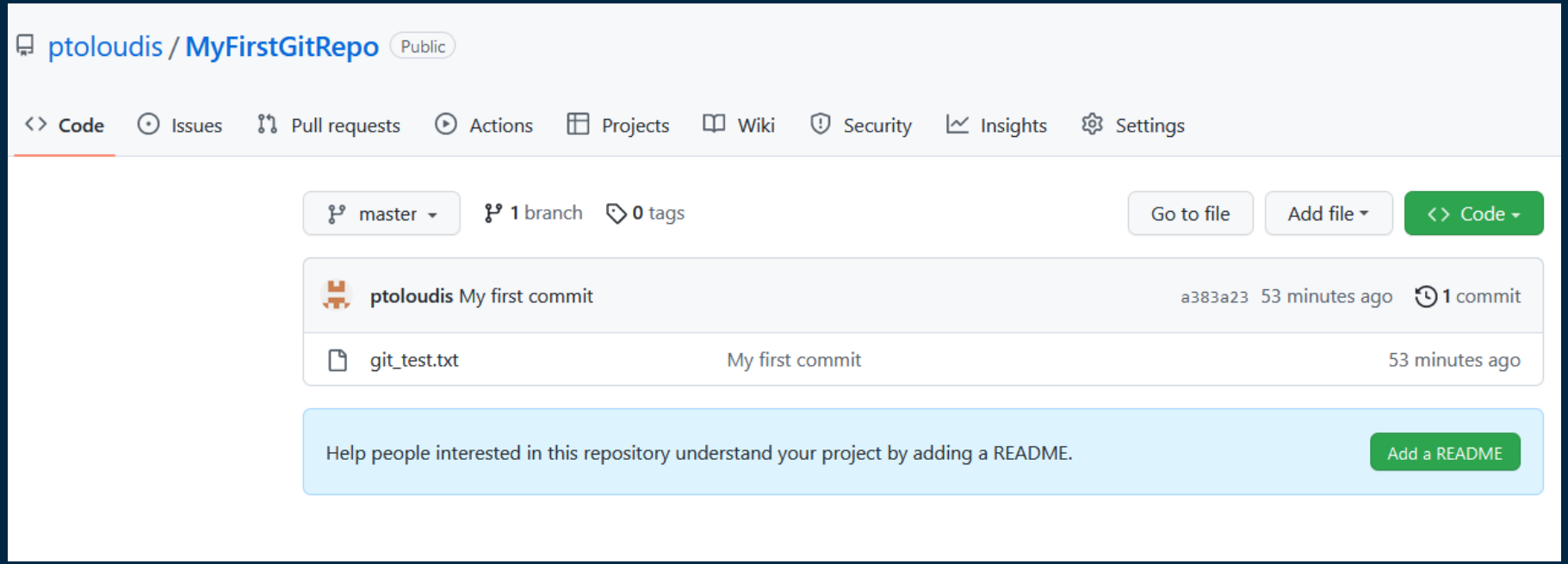
11 Type "workshop" and click the repo.

12 Click "Generate token"



13 Copy the code like this
ghp_VrgMblOAxnYKo68RLKoBa8dnuPXc2B2D43zE

```
→  MyFirstGitRepo git:(master) git push origin master
Username for 'https://github.com': ptoloudis
Password for 'https://ptoloudis@github.com':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 242 bytes | 1024 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/ptoloudis/MyFirstGitRepo.git
 * [new branch]      master -> master
→  MyFirstGitRepo git:(master) git status
On branch master
nothing to commit, working tree clean
→  MyFirstGitRepo git:(master)
```


ptoloudis / MyFirstGitRepo
Public
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
master
1 branch
0 tags
Go to file
Add file
Code
ptoloudis My first commit
a383a23
53 minutes ago
1 commit
git_test.txt
My first commit
53 minutes ago
Help people interested in this repository understand your project by adding a README.
Add a README

THANK YOU FOR YOUR TIME!