T05 - “Task Answers”

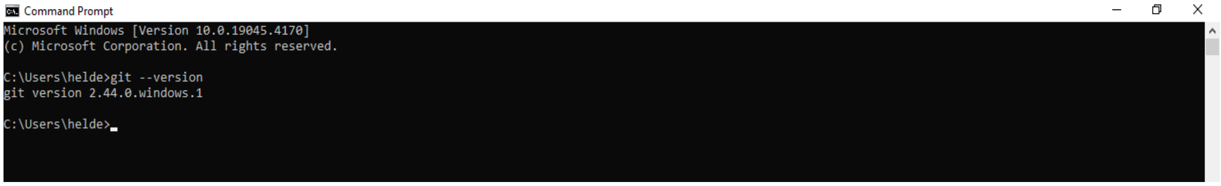
Technical Portfolio

HELDER PAIXAO - HP24010013265

helder.paixao@outlook.com

Part 1: Practical Task 1

- Install Git by following the instructions given in Part 1.
- Once you have successfully downloaded and installed Git, enter git --version into your terminal or command prompt to display your current version of Git.
- Take a screenshot of the displayed Git version and place it into your Google answers doc.

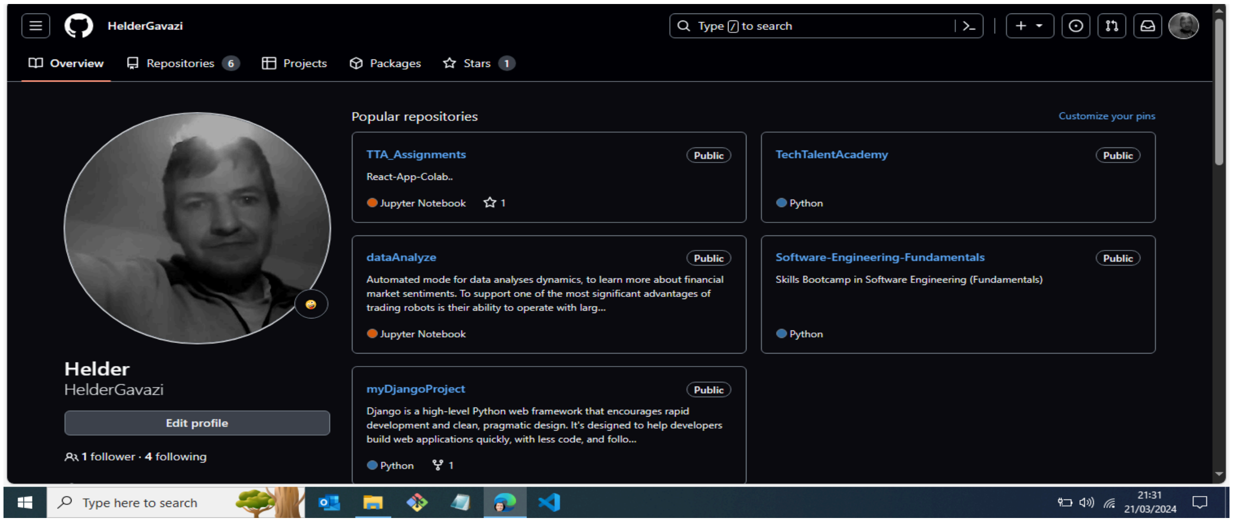


```
Microsoft Windows [Version 10.0.19045.4170]
(c) Microsoft Corporation. All rights reserved.

C:\Users\helde>git --version
git version 2.44.0.windows.1

C:\Users\helde>_
```

Git installed and updated to the latest version (git version 2.44.0.windows.1).

Part 1: Practical Task 2

- Create a free GitHub account by visiting https://github.com/join.
- After you have created your account, copy the URL of your GitHub profile and add it to your Google answers doc.



TTA_Assignments
React-App-Colab..
Jupyter Notebook ☆ 1

TechTalentAcademy
Python

dataAnalyze
Automated mode for data analyses dynamics, to learn more about financial market sentiments. To support one of the most significant advantages of trading robots is their ability to operate with larg...
Jupyter Notebook

Software-Engineering-Fundamentals
Skills Bootcamp in Software Engineering (Fundamentals)
Python

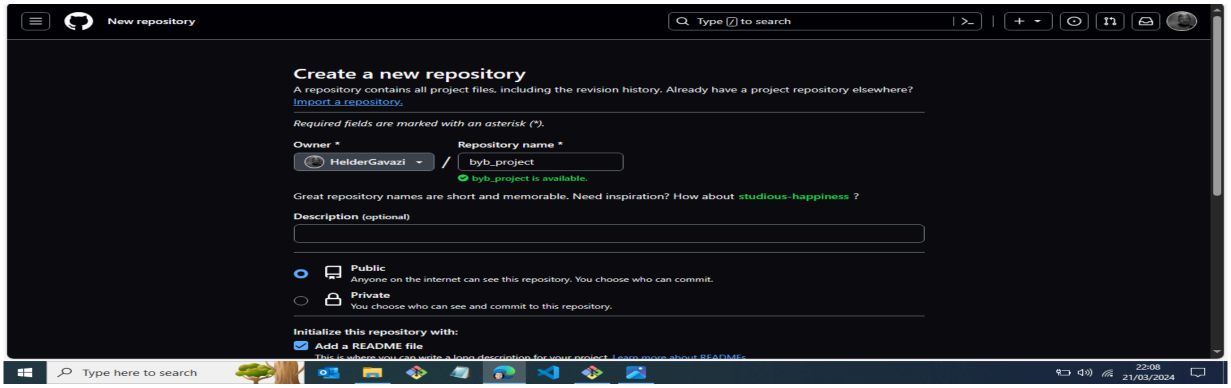
myDjangoProject
Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. It's designed to help developers build web applications quickly, with less code, and follo...
Python 1

Helder
HelderGavazi
1 follower · 4 following

My GitHub account URL: https://github.com/HelderGavazi

Part 2: Practical Task 1

- Login to GitHub using the account you created in Part 1 of this task.
- Create a new repository by selecting the ‘New’ button as shown in the image below.
- Name the repository ‘byb_project’ and make sure that it is public..



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Required fields are marked with an asterisk (*).

Owner *: HelderGavazi / Repository name *: byb_project

✓ byb_project is available.

Great repository names are short and memorable. Need inspiration? How about studious-happiness ?

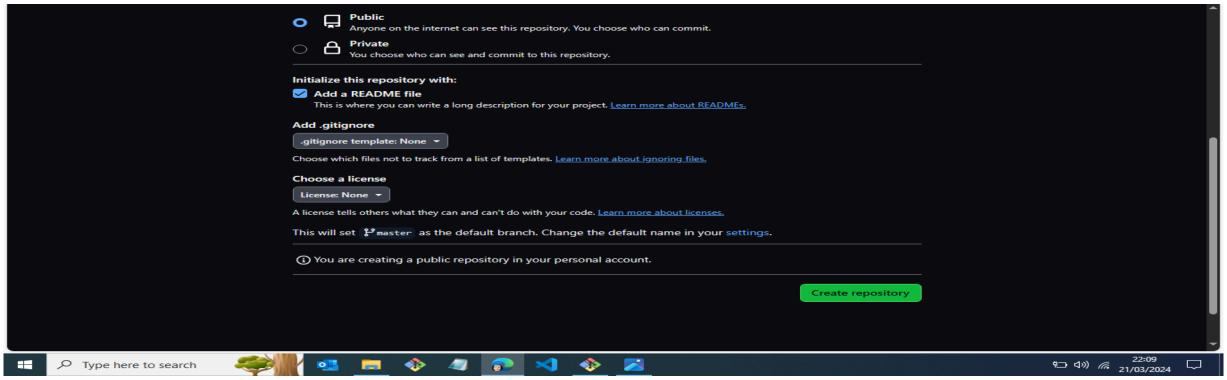
Description (optional)

● 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:

☑ Add a README file
This is where you can write a long description for your project. Learn more about READMEs.



● 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:

☑ Add a README file
This is where you can write a long description for your project. Learn more about READMEs.

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. Learn more about ignoring files.

Choose a license

License: None

A license tells others what they can and can't do with your code. Learn more about licenses.

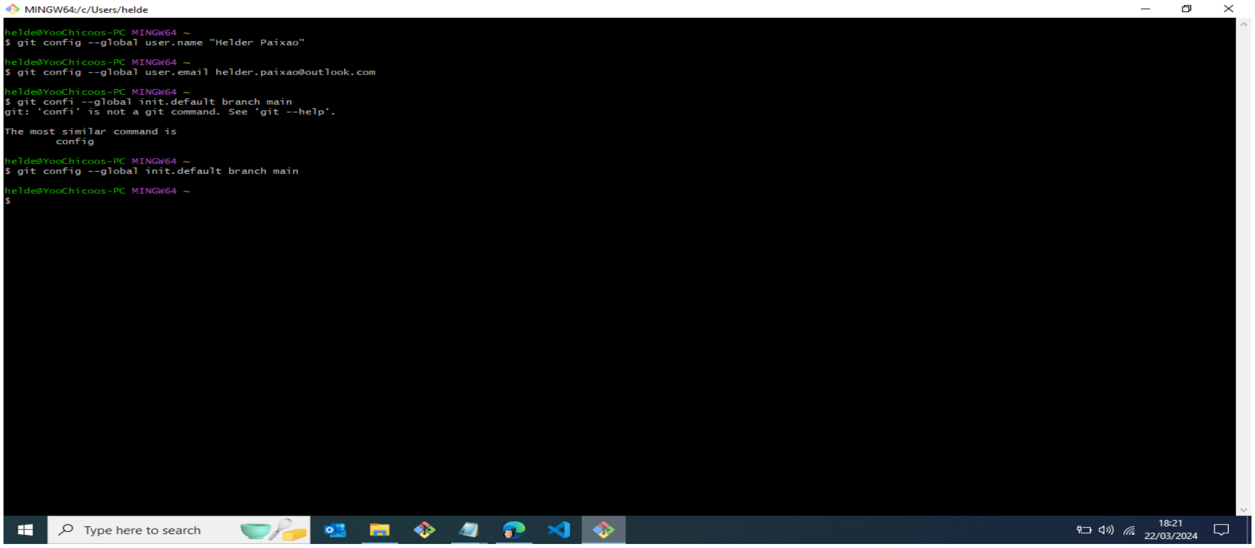
This will set `master` as the default branch. Change the default name in your settings.

ⓘ You are creating a public repository in your personal account.

Create repository

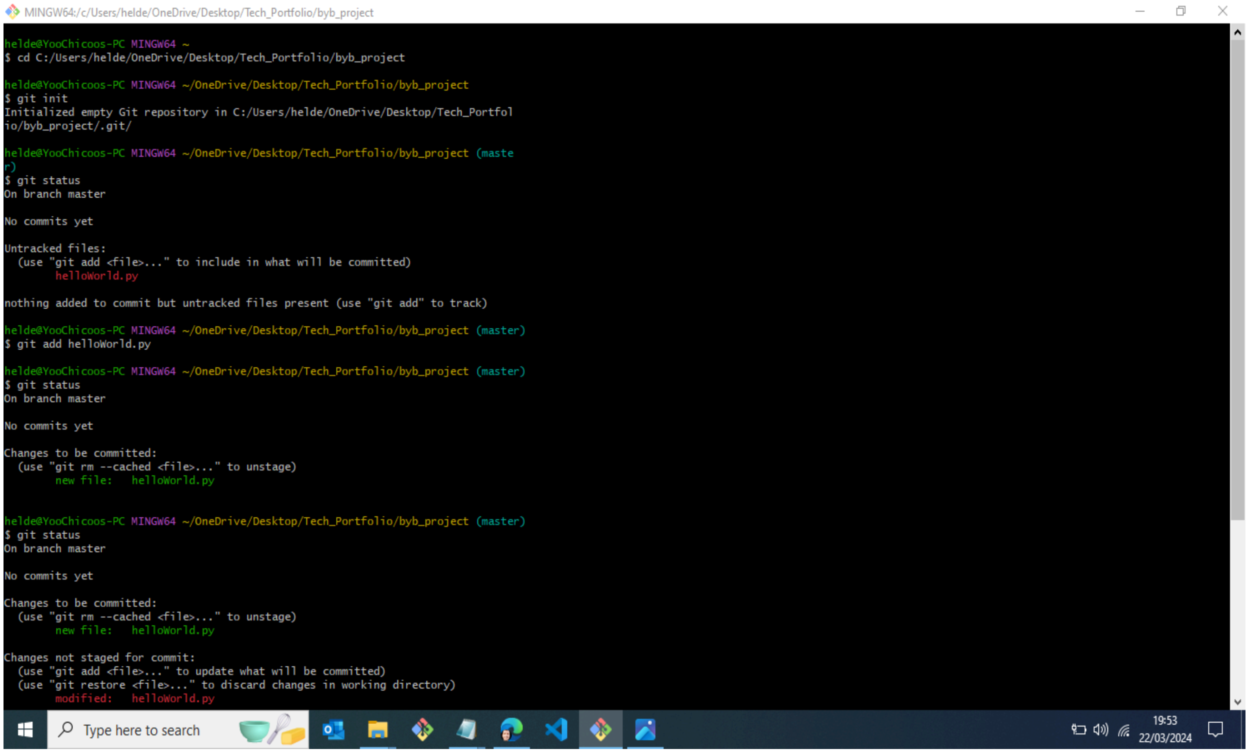
https://github.com/HelderGavazi/byb_project.git

- Next, create an empty folder called byb_project on your local machine.
- Open your terminal or command prompt and then change directory (cd) to your newly created folder.
- Enter the git init command to initialise your new repository.
- Enter the git status command and make a note of what you see. You should have a clean working directory.
- Create a new file in the byb_project folder called helloWorld.py and write a program that prints out the message “Hello World!”.
- Run the git status command again. You should now see that your helloWorld.py file is untracked.
- Enter the git add command followed by helloWorld.py to start tracking your new file.
- Once again, run the git status command. You should now see that your helloWorld.py file is tracked and staged to be committed.



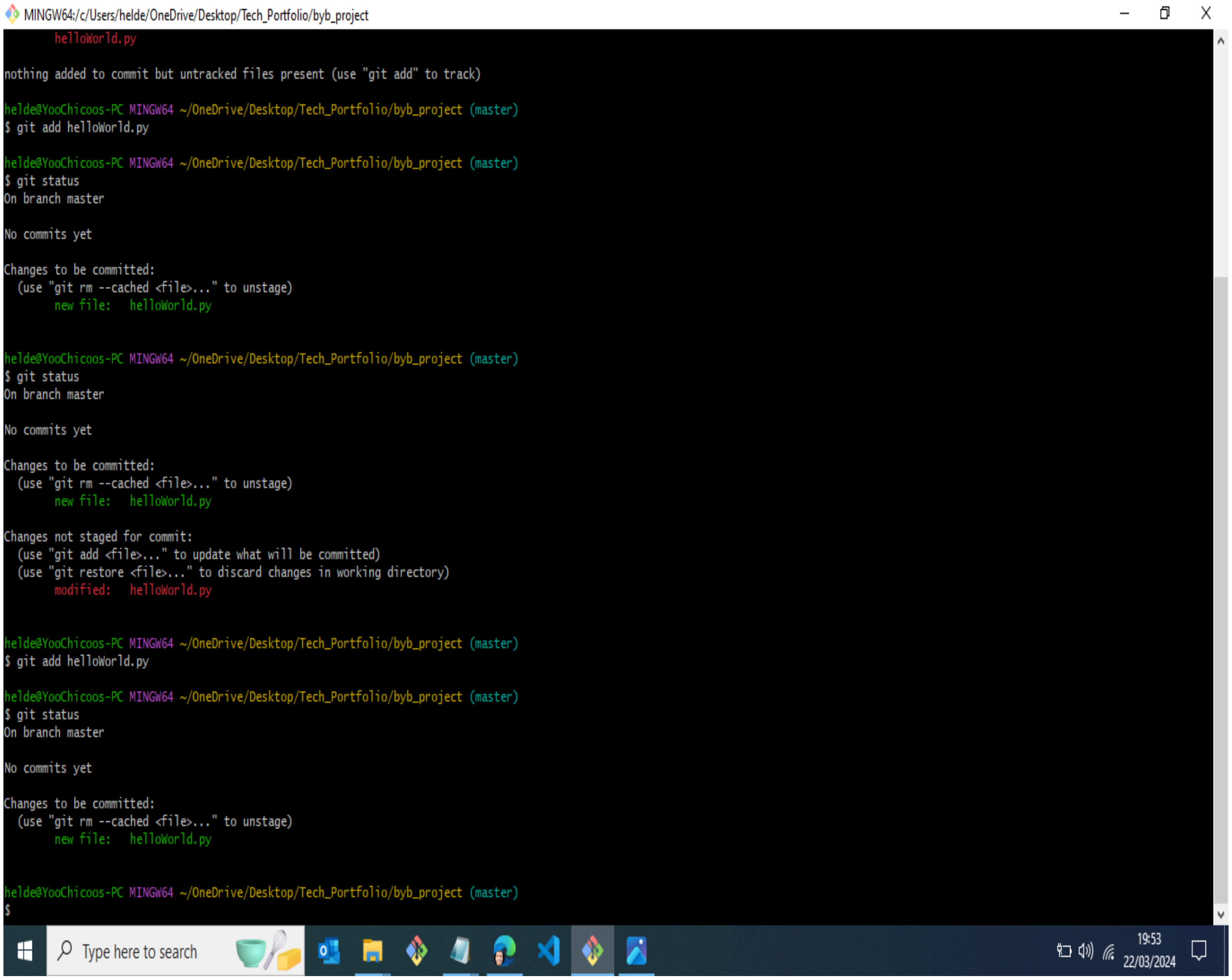
```
helde@YooChicoos-PC MINGW64 ~
$ git config --global user.name "Helder Paixao"

helde@YooChicoos-PC MINGW64 ~
$ git config --global user.email helder.paixao@outlook.com

helde@YooChicoos-PC MINGW64 ~
$ git confi --global init.default branch main
git: 'confi' is not a git command. See 'git --help'.

The most similar command is
        config

helde@YooChicoos-PC MINGW64 ~
$ git config --global init.default branch main

helde@YooChicoos-PC MINGW64 ~
$
```

- Now that it is tracked, let us change the file helloWorld.py. Change the message printed out by the program to “Git is Awesome!”
- Run git status again. You should see that helloWorld.py appears under a section called “Changes not staged for commit”. This means that the file is tracked but has been modified and not yet staged.
- To stage your file, simply run git add helloWorld.py again.
- If you run git status again you should see that it is once again staged for your next commit.


MINGW64:/c/Users/helde/OneDrive/Desktop/Tech_Portfolio/byb_project
Type here to search
19:53
22/03/2024

```
helde@YooChicoos-PC MINGW64 ~
$ cd C:/Users/helde/OneDrive/Desktop/Tech_Portfolio/byb_project

helde@YooChicoos-PC MINGW64 ~/OneDrive/Desktop/Tech_Portfolio/byb_project
$ git init
Initialized empty Git repository in C:/Users/helde/OneDrive/Desktop/Tech_Portfol
io/byb_project/.git/

helde@YooChicoos-PC MINGW64 ~/OneDrive/Desktop/Tech_Portfolio/byb_project (maste
r)
$ git status
On branch master

No commits yet

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

nothing added to commit but untracked files present (use "git add" to track)

helde@YooChicoos-PC MINGW64 ~/OneDrive/Desktop/Tech_Portfolio/byb_project (master)
$ git add helloWorld.py

helde@YooChicoos-PC MINGW64 ~/OneDrive/Desktop/Tech_Portfolio/byb_project (master)
$ git status
On branch master

No commits yet

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


helde@YooChicoos-PC MINGW64 ~/OneDrive/Desktop/Tech_Portfolio/byb_project (master)
$ git status
On branch master

No commits yet

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

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   helloWorld.py
```

- You can now commit your changes by running the git commit -m command. Remember to enter a suitable commit message after the -m switch.
- Running the git status command should now show a clean working directory once again.

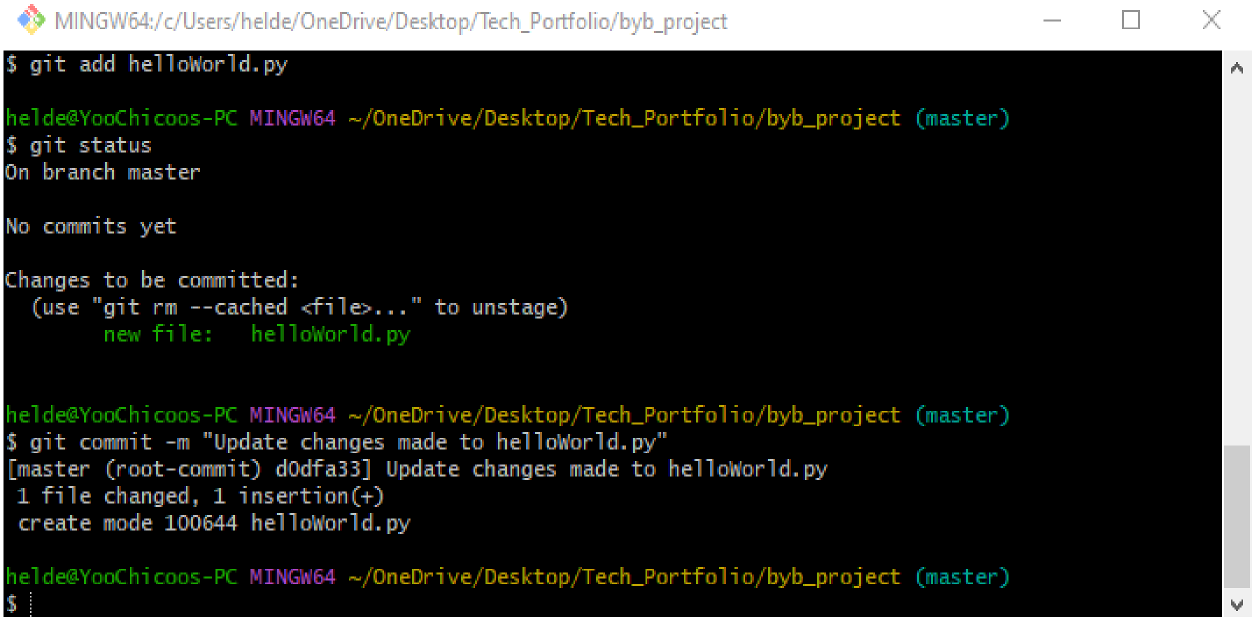


MINGW64:/c/Users/helde/OneDrive/Desktop/Tech_Portfolio/byb_project
Type here to search
19:53
22/03/2024

```
        helloWorld.py

nothing added to commit but untracked files present (use "git add" to track)

helde@YooChicoos-PC MINGW64 ~/OneDrive/Desktop/Tech_Portfolio/byb_project (master)
$ git add helloWorld.py

helde@YooChicoos-PC MINGW64 ~/OneDrive/Desktop/Tech_Portfolio/byb_project (master)
$ git status
On branch master

No commits yet

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


helde@YooChicoos-PC MINGW64 ~/OneDrive/Desktop/Tech_Portfolio/byb_project (master)
$ git status
On branch master

No commits yet

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

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   helloWorld.py


helde@YooChicoos-PC MINGW64 ~/OneDrive/Desktop/Tech_Portfolio/byb_project (master)
$ git add helloWorld.py

helde@YooChicoos-PC MINGW64 ~/OneDrive/Desktop/Tech_Portfolio/byb_project (master)
$ git status
On branch master

No commits yet

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


helde@YooChicoos-PC MINGW64 ~/OneDrive/Desktop/Tech_Portfolio/byb_project (master)
$
```

- Push the repository on your local machine to the remote repository on GitHub by following these steps:
  - Open your terminal or command prompt. Change directory (cd) into the byb_project folder you created.
  - Add your remote repository using the following command:
  - git remote add [remote-name] [url] e.g. git remote add origin https://github.com/HyperionDev/byb_project.git.
  - Push your local repository to your remote repository using the following command: git push -u [remote-name] master E.g. git push -u origin master

- Make the repository public and put a link to it into your Google answers doc

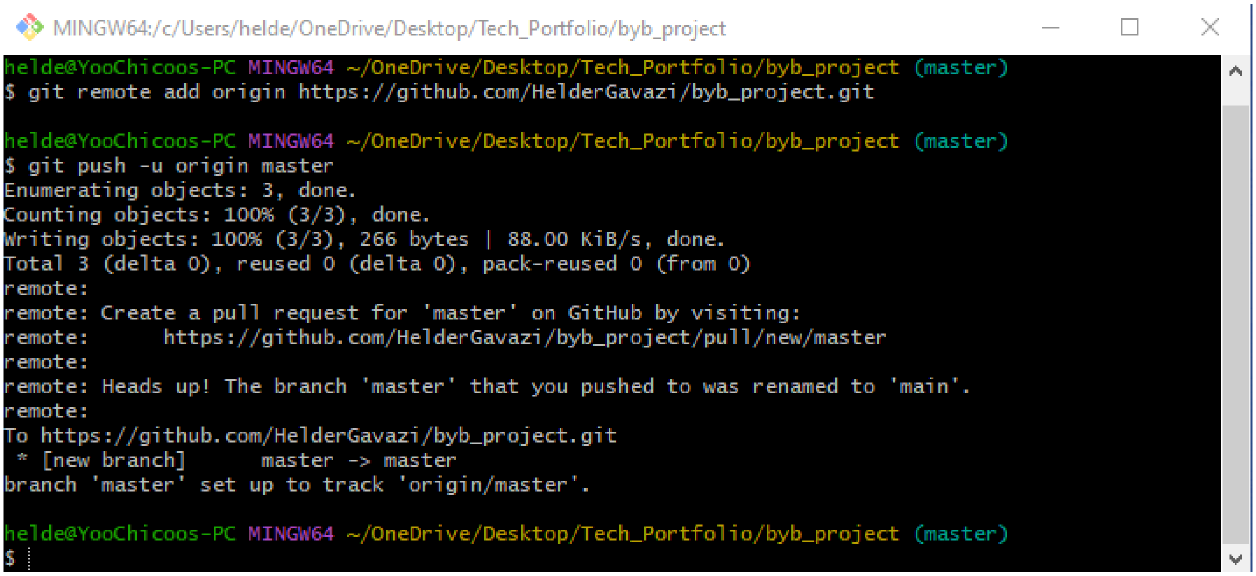


MINGW64:/c/Users/helde/OneDrive/Desktop/Tech_Portfolio/byb_project

```
$ git add helloWorld.py

helde@YooChicoos-PC MINGW64 ~/OneDrive/Desktop/Tech_Portfolio/byb_project (master)
$ git status
On branch master

No commits yet

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


helde@YooChicoos-PC MINGW64 ~/OneDrive/Desktop/Tech_Portfolio/byb_project (master)
$ git commit -m "Update changes made to helloWorld.py"
[master (root-commit) d0dfa33] Update changes made to helloWorld.py
 1 file changed, 1 insertion(+)
 create mode 100644 helloWorld.py

helde@YooChicoos-PC MINGW64 ~/OneDrive/Desktop/Tech_Portfolio/byb_project (master)
$
```

Now you can use the short name (e.g. origin) on the command line in lieu of the whole URL. The URL will be indicated under the heading shown below once you have created your repository on GitHub.

⇒ git remote add origin https://github.com/HelderGavazi/byb_project.git ⇐



MINGW64:/c/Users/helde/OneDrive/Desktop/Tech_Portfolio/byb_project

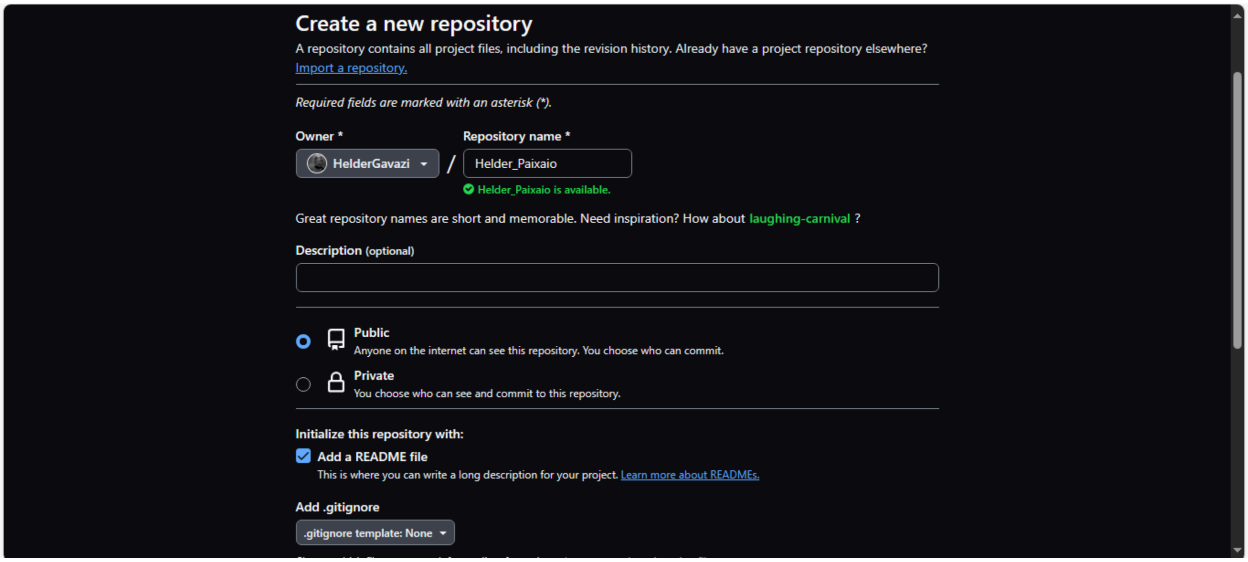
```
helde@YooChicoos-PC MINGW64 ~/OneDrive/Desktop/Tech_Portfolio/byb_project (master)
$ git remote add origin https://github.com/HelderGavazi/byb_project.git

helde@YooChicoos-PC MINGW64 ~/OneDrive/Desktop/Tech_Portfolio/byb_project (master)
$ git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 266 bytes | 88.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:      https://github.com/HelderGavazi/byb_project/pull/new/master
remote:
remote: Heads up! The branch 'master' that you pushed to was renamed to 'main'.
remote:
To https://github.com/HelderGavazi/byb_project.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.

helde@YooChicoos-PC MINGW64 ~/OneDrive/Desktop/Tech_Portfolio/byb_project (master)
$
```

- Push your local repository to your remote repository using the following command:
  - E.g. git push -u origin master

⇒ git push -u origin main ⇐

Part 2: Practical Task 2

- Create a second new repository on your GitHub account with the repo name spelt exactly the same as your username.



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Required fields are marked with an asterisk (*).

Owner * HelderGavazi / Repository name * Helder_Paixaio

✓ Helder_Paixaio is available.

Great repository names are short and memorable. Need inspiration? How about laughing-carnival ?

Description (optional)

● Public
Anyone on the internet can see this repository. You choose who can commit.

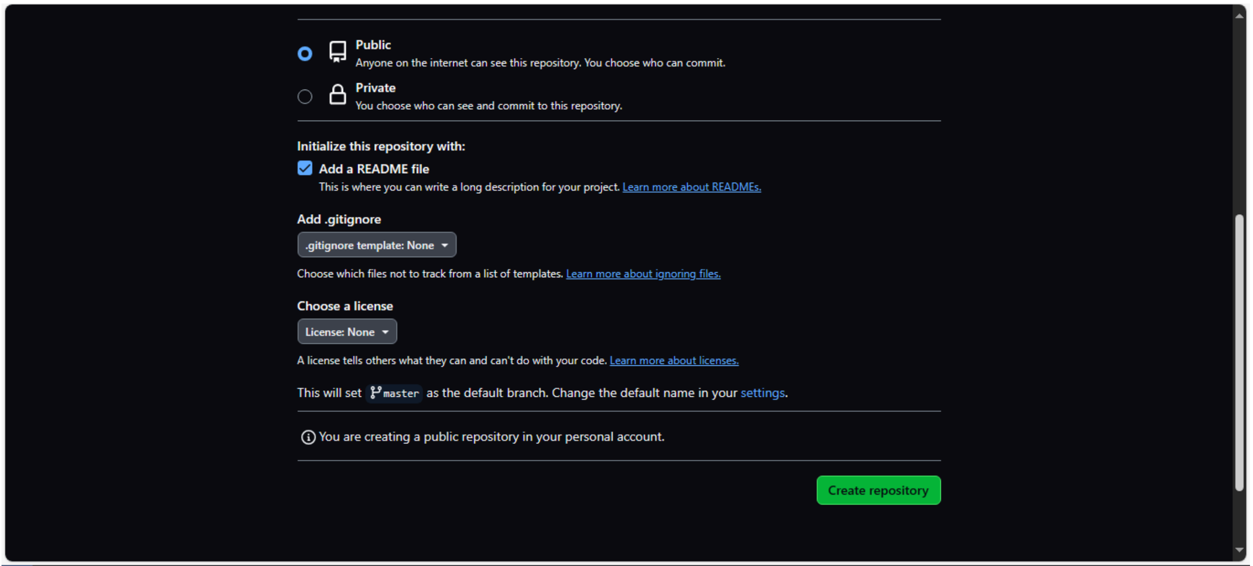
○ Private
You choose who can see and commit to this repository.

Initialize this repository with:

☑ Add a README file
This is where you can write a long description for your project. Learn more about READMEs.

Add .gitignore

.gitignore template: None



● 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:

☑ Add a README file
This is where you can write a long description for your project. Learn more about READMEs.

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. Learn more about ignoring files.

Choose a license

License: None

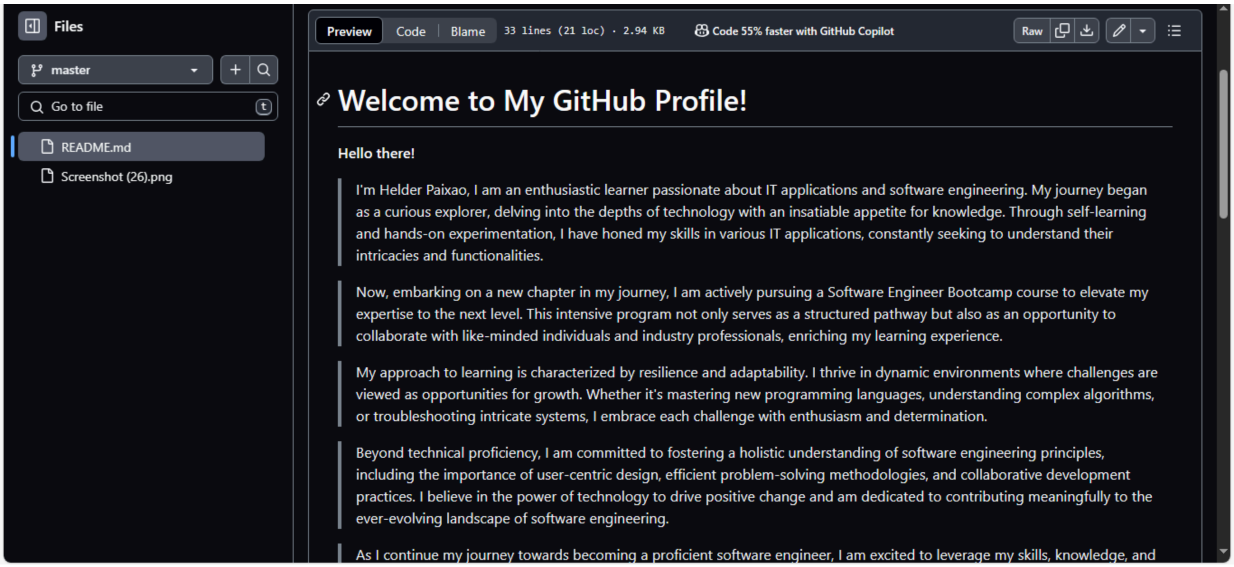
A license tells others what they can and can't do with your code. Learn more about licenses.

This will set `master` as the default branch. Change the default name in your settings.

ⓘ You are creating a public repository in your personal account.

⇒ https://github.com/HelderGavazi/Helder_Paixao.git ⇐

- Create a README.md file inside this newly created repo.



Welcome to My GitHub Profile!

Hello there!

> I'm Helder Paixao, I am an enthusiastic learner passionate about IT applications and software engineering. My journey began as a curious explorer, delving into the depths of technology with an insatiable appetite for knowledge. Through self-learning and hands-on experimentation, I have honed my skills in various IT applications, constantly seeking to understand their intricacies and functionalities.

> Now, embarking on a new chapter in my journey, I am actively pursuing a Software Engineer Bootcamp course to elevate my expertise to the next level. This intensive program not only serves as a structured pathway but also as an opportunity to collaborate with like-minded individuals and industry professionals, enriching my learning experience.

> My approach to learning is characterized by resilience and adaptability. I thrive in dynamic environments where challenges are viewed as opportunities for growth. Whether it's mastering new programming languages, understanding complex algorithms, or troubleshooting intricate systems, I embrace each challenge with enthusiasm and determination.

> Beyond technical proficiency, I am committed to fostering a holistic understanding of software engineering principles, including the importance of user-centric design, efficient problem-solving methodologies, and collaborative development practices. I believe in the power of technology to drive positive change and am dedicated to contributing meaningfully to the ever-evolving landscape of software engineering.

> As I continue my journey towards becoming a proficient software engineer, I am excited to leverage my skills, knowledge, and

==> **Helder_Paixao/README.md at master · HelderGavazi/Helder_Paixao (github.com)** ⇐

Final Submission Checklist

Part 1 – Ensure you hand in the following:

- Part 1: Practical Task 1 – A screenshot of the displayed git version after installation, pasted into your Google answers doc.
  - Page 1
- Part 1: Practical Task 2 – the URL of your GitHub profile, also pasted into your Google answers doc. Part 2 – Ensure you hand in the following:
  - **https://github.com/HelderGavazi/byb_project.git**
- Part 2: Practical Task 1 – a link to your byb-project repository, pasted into your Google answers doc, and your completed profile READ_ME file in the repository.
  - **Helder_Paixao/README.md at master · HelderGavazi/Helder_Paixao (github.com)**
- Part 2: Practical Task 2 – a link to your repository with the repo name spelt exactly the same as your username, pasted into your Google answers doc.
  - **https://github.com/HelderGavazi/Helder_Paixao.git**