

# Standard Operating Procedure (SOP)

## Uploading **new** Git Repository

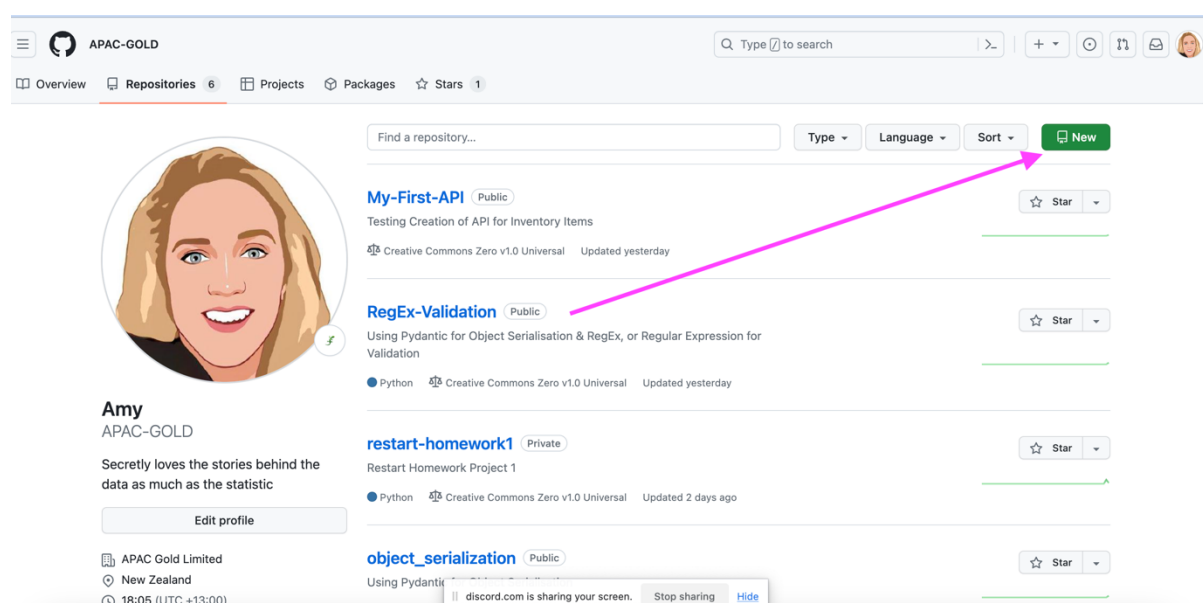
These steps are for people who want to create code locally, then move it to a folder on their local machine, to push into GitHub and share publicly with i.e. a Tutor or Peer.

---

Login GitHub

Select Repositories

Create New Repository



Create New Repository

Give it a Name & Description

Owner \* APAC-GOLD / Repository name \* My Second API

Great repository names are short and memorable. Need inspiration? How about [automatic-guacamole](#) ?

Description (optional)  
This API was used to test "Put" "Get" & "Delete" for Items in a Directory

☒ 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: Creative Commons Zero v1.0 Universal  
A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

This will set `main` as the default branch. Change the default name in your [settings](#).

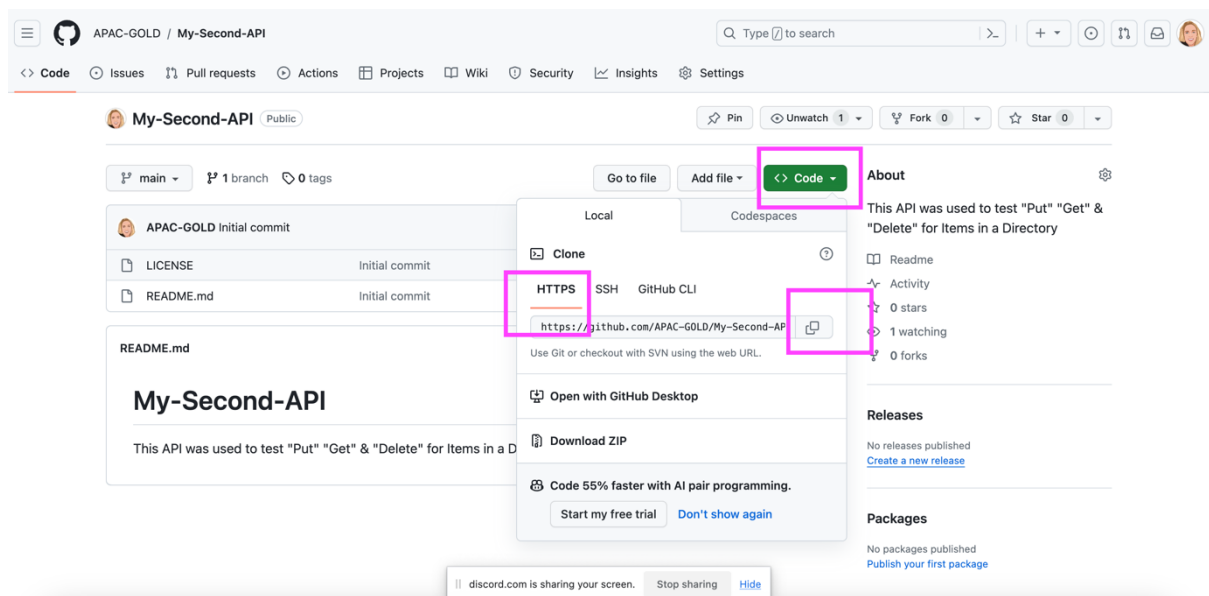
You are creating a public repository. [discord.com](#) is sharing your screen. [Stop sharing](#) [Hide](#)

Do NOT create a "ReadMe" File or  
Select a Licence  
(Or you will have to delete these files later from terminal, see last section of this SOP).

Click "Create" button in the bottom Right Hand Corner of your screen to create the new  
GitHub Repository

Click on "Code"

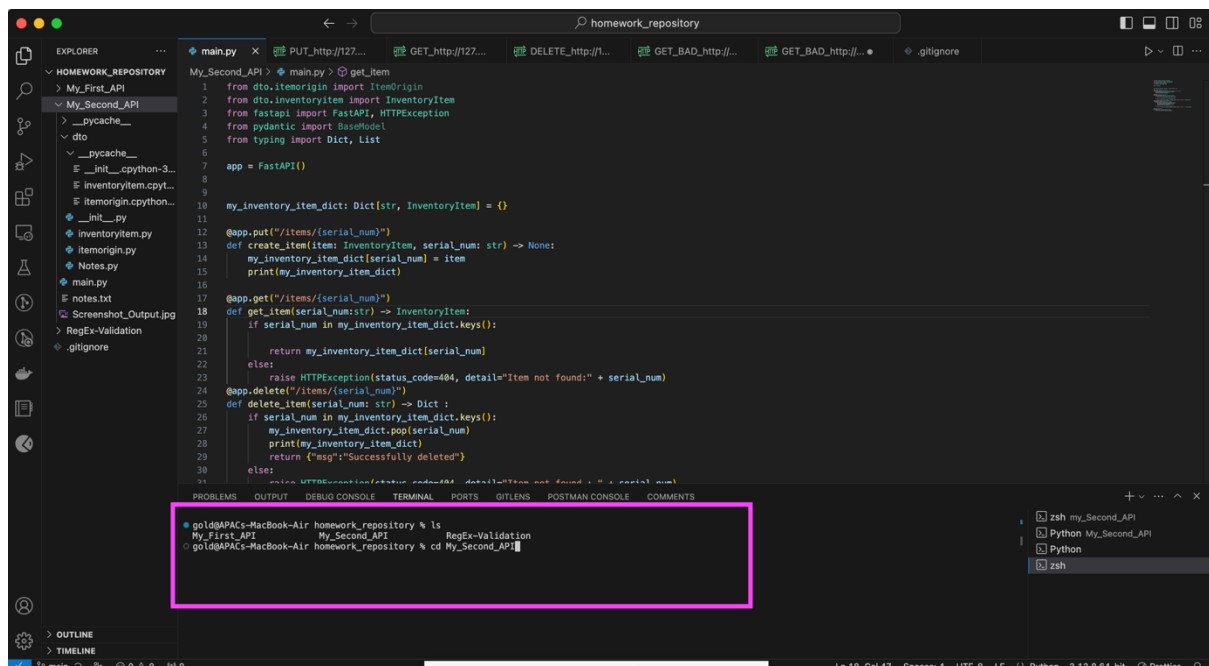
Choose HTTPS



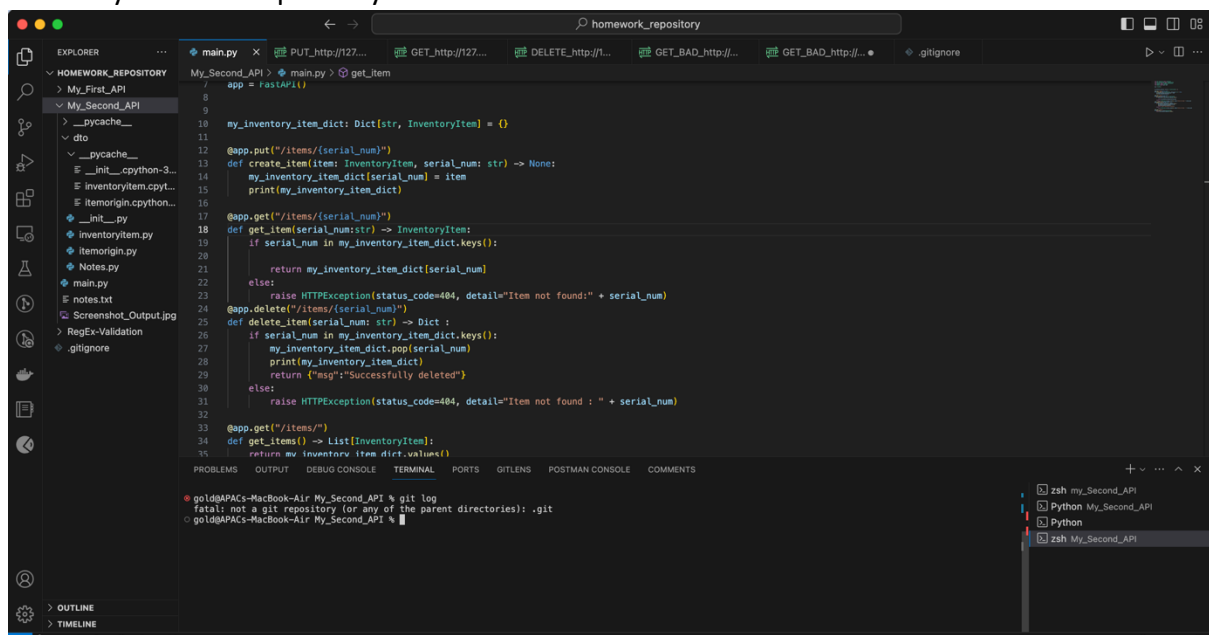
Copy the URL and save it for later  
<https://github.com/APAC-GOLD/My-Second-API.git>

Open Visual Studio

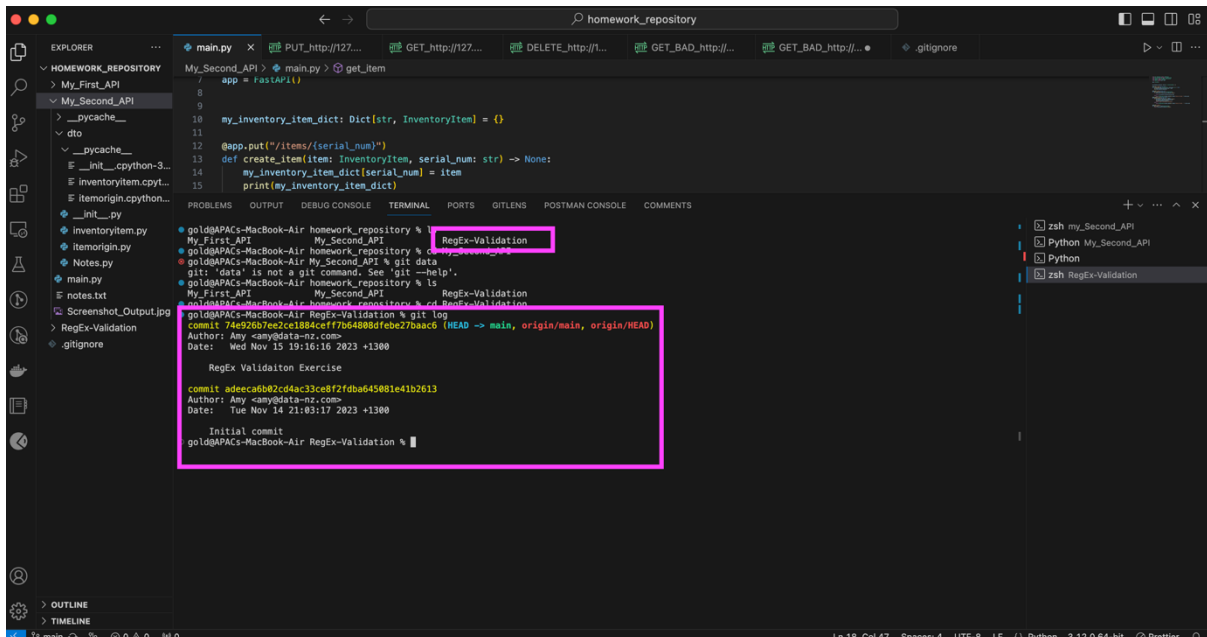
Navigate to the correct folder using Terminal to get to the correct (local) directory



Once you have selected your local directory, then use the command “git log” to check if the directory is a local repository.



Screenshot – showing this folder is only a local directory and contains the code



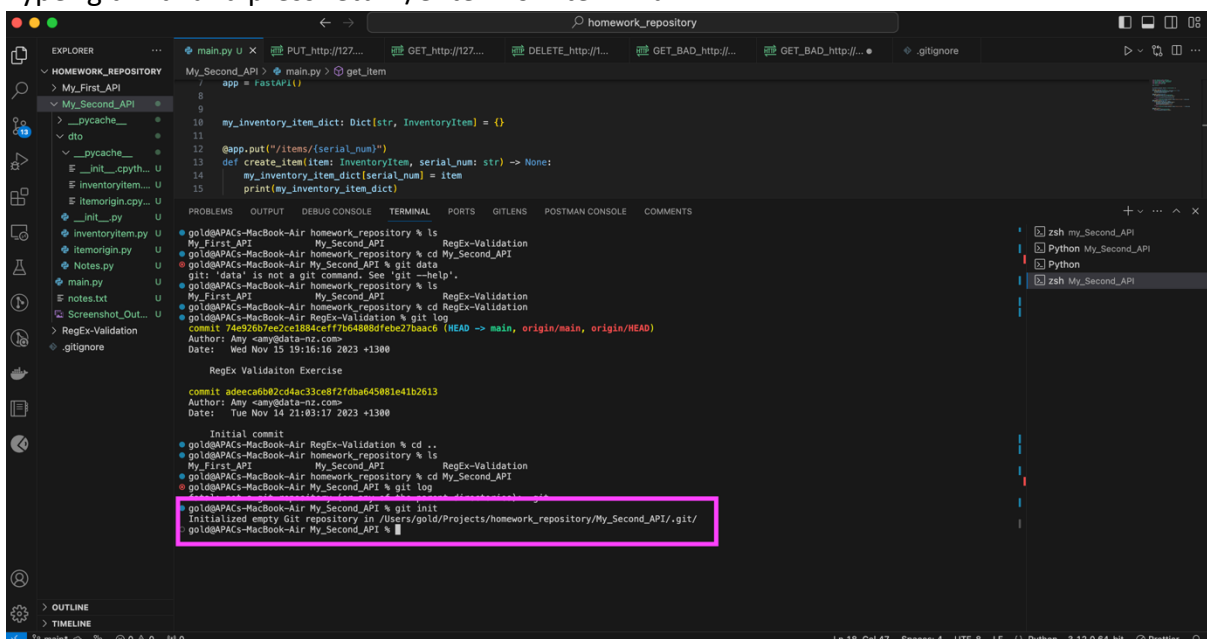
Screenshot – showing this folder IS a local repository linked to GitHub

There are Two Options:

1. git clone GitHub\_repository\_url to download the GitHub repos to local, then copy the local source code folder to the folder we just downloaded -> then git add . -> git commit -m "message" -> git push as usual
2. Use files on the local repository with git init -> then setup the connection to the GitHub repository on the local folder by using git remote set-url

We are going to demonstrate Option 2:

Type “git init” and press return/enter from terminal



Screenshot – showing git init is initialised

to add a new remote type “git remote add origin <https://github.com/APAC-GOLD/My-Second-API.git>”

Then type “git add .” into consol

```

git log
commit 74c926b7ec2c1884ceff7b54888dfebe27baac6 (HEAD -> main, origin/main, origin/HEAD)
Author: Amy <amy@data-nz.com>
Date:   Wed Nov 15 19:16:16 2023 +1300

    RegEx Validation Exercise

commit e8dec6087cd4ac33ce8f2f0ba645801e41b2613
Author: Amy <amy@data-nz.com>
Date:   Tue Nov 14 21:03:17 2023 +1300

    Initial commit

git status
No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .DS_Store
        Screenshot_Output.jpg
        __pycache__/
        dto/
        main.py
        notes.txt

nothing added to commit but untracked files present (use "git add" to track)
git remote add origin https://github.com/APAC-GOLD/My-Second-API.git
git status
On branch main

```

git commit -m "My Second API Commit 16 November 2023"

```

git log
fatal: not a git repository (or any of the parent directories): .git

git init
Initialized empty Git repository in /Users/gold/Projects/homework_repository/My_Second_API/.git/
git remote add origin https://github.com/APAC-GOLD/My-Second-API.git
git status
On branch main

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .DS_Store
        Screenshot_Output.jpg
        __pycache__/
        dto/
        main.py
        notes.txt

nothing added to commit but untracked files present (use "git add" to track)
git remote add origin https://github.com/APAC-GOLD/My-Second-API.git
git add .
git commit -m "My Second API Commit 16 November 2023"
[main (root-commit) c1833cf] My Second API Commit 16 November 2023
13 files changed, 181 insertions(+)
create mode 100644 .DS_Store
create mode 100644 Screenshot_Output.jpg
create mode 100644 __pycache__/main.cpython-312.pyc
create mode 100644 dto/.DS_Store
create mode 100644 dto/Notes.py
create mode 100644 dto/__init__.py
create mode 100644 dto/__pycache__/__init__.cpython-312.pyc
create mode 100644 dto/__pycache__/inventoryitem.cpython-312.pyc
create mode 100644 dto/inventoryitem.py
create mode 100644 dto/inventoryitem.cpython-312.pyc
create mode 100644 dto/inventoryitem.py
create mode 100644 main.py
create mode 100644 notes.txt
git status
On branch main

```

git commit -m "My Second API Commit 16 November 2023"

```

My_Second_API > main.py > get_item
7 app = FastAPI()
8
9
10 my_inventory_item_dict: Dict[str, InventoryItem] = {}
11
12 @app.put("/items/{serial_num}")
13 def create_item(item: InventoryItem, serial_num: str) -> None:
14     my_inventory_item_dict[serial_num] = item
15     print(my_inventory_item_dict)

fatal: not a git repository (or any of the parent directories): .git
gold@APACs-MacBook-Air My_Second_API % git init
Initialized empty git repository in /Users/gold/Projects/homework_repository/My_Second_API/.git/
gold@APACs-MacBook-Air My_Second_API % git remote -v
gold@APACs-MacBook-Air My_Second_API % git status
On branch main

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    .DS_Store
    Screenshot_Output.jpg
    __pycache__/
    dto/
    main.py
    notes.txt

nothing added to commit but untracked files present (use "git add" to track)
gold@APACs-MacBook-Air My_Second_API % git remote add origin https://github.com/APAC-GOLD/My-Second-API.git
gold@APACs-MacBook-Air My_Second_API % git add .
gold@APACs-MacBook-Air My_Second_API % git commit -m "My Second API Commit 16 November 2023"
[main (root-commit) c1833cf] My Second API Commit 16 November 2023
13 files changed, 181 insertions(+)
create mode 100644 .DS_Store
create mode 100644 Screenshot_Output.jpg
create mode 100644 __pycache__/main.cpython-312.pyc
create mode 100644 dto/.DS_Store
create mode 100644 dto/_init_.py
create mode 100644 dto/__pycache__/__init__.cpython-312.pyc
create mode 100644 dto/__pycache__/inventoryitem.cpython-312.pyc
create mode 100644 dto/__pycache__/itemorigin.cpython-312.pyc
create mode 100644 dto/inventoryitem.py
create mode 100644 dto/itemorigin.py
create mode 100644 main.py
create mode 100644 notes.txt
gold@APACs-MacBook-Air My_Second_API %
  
```

git status (to check)

```

My_Second_API > main.py > get_item
7 app = FastAPI()
8
9
10 my_inventory_item_dict: Dict[str, InventoryItem] = {}
11
12 @app.put("/items/{serial_num}")
13 def create_item(item: InventoryItem, serial_num: str) -> None:
14     my_inventory_item_dict[serial_num] = item
15     print(my_inventory_item_dict)

gold@APACs-MacBook-Air My_Second_API % git remote -v
gold@APACs-MacBook-Air My_Second_API % git status
On branch main

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    .DS_Store
    Screenshot_Output.jpg
    __pycache__/
    dto/
    main.py
    notes.txt

nothing added to commit but untracked files present (use "git add" to track)
gold@APACs-MacBook-Air My_Second_API % git remote add origin https://github.com/APAC-GOLD/My-Second-API.git
gold@APACs-MacBook-Air My_Second_API % git add .
gold@APACs-MacBook-Air My_Second_API % git commit -m "My Second API Commit 16 November 2023"
[main (root-commit) c1833cf] My Second API Commit 16 November 2023
13 files changed, 181 insertions(+)
create mode 100644 .DS_Store
create mode 100644 Screenshot_Output.jpg
create mode 100644 __pycache__/main.cpython-312.pyc
create mode 100644 dto/.DS_Store
create mode 100644 dto/_init_.py
create mode 100644 dto/__pycache__/__init__.cpython-312.pyc
create mode 100644 dto/__pycache__/inventoryitem.cpython-312.pyc
create mode 100644 dto/__pycache__/itemorigin.cpython-312.pyc
create mode 100644 dto/inventoryitem.py
create mode 100644 dto/itemorigin.py
create mode 100644 main.py
create mode 100644 notes.txt
gold@APACs-MacBook-Air My_Second_API % git status
On branch main
nothing to commit, working tree clean
gold@APACs-MacBook-Air My_Second_API %
  
```

Enter "git push origin main" to push the files to GitHub

If you get an error that there are files on your GitHub repository that are not on your local machine then use:

`git push -f origin main`

this will overwrite any files in the GitHub repository with your files from the local directory

Now your files are in the selected GitHub repository



The screenshot shows the GitHub interface for the repository 'My-Second-API' under the user 'APAC-GOLD'. The repository is public and has 1 commit, 1 branch (main), and 0 tags. The commit history shows a single commit 'c1833cf' from 8 minutes ago. The file list includes: \_\_pycache\_\_, dto, .DS\_Store, Screenshot\_Output.jpg, main.py, and notes.txt, all committed 8 minutes ago. The right sidebar contains sections for 'About' (describing the API's purpose), 'Activity' (0 stars, 1 watching, 0 forks), 'Releases' (no releases published), 'Packages' (no packages published), and 'Languages'. A notification at the bottom states 'discord.com is sharing your screen.' with 'Stop sharing' and 'Hide' buttons.

End. Kicking Goals! Learn More: <https://github.com/APAC-GOLD/How-To.git>