TD5

Exercice 1

1. Create an empty working directory called “td4”

> sudo apt-get update
> mkdir td4

2. Initialize a Git repository in it

> cd td4
> git init

3. Install the Linux python3-pip package using your Linux package manager.

> sudo apt-get install python3-pip

4. Install the VirtualEnv Python package using pip3

> pip3 install virtualenv

5. Create a Python virtual environment called “.env”

> sudo apt install python3.10-venv
> python3 -m venv .env

6. Activate your virtual environment.

> source .env/bin/activate

> On peut observer que juste avant le nom d’utilisateur on a maintenant (.env)

7. List the Python packages installed in your virtual environment.

> pip list

8. Does Git want you to commit something ? Do you think it is a good thing ? hint : You can find templates at https://github.com/github/gitignore

> Non ça ne commit rien
> Ce n’est pas une bonne idée, Si les packages sont installés et mis à jour, il n’y a aucune raison de livrer ces fichiers à Git

9. Create a .gitignore file to tell Git which files should be untracked.

> nano .gitignore
> *.env

10. Does Git want you to commit something ? Do you think it is a good thing this time ?

> Git veut nous faire commit le .gitignore

11. Do your first commit and check that Git is happy now.

> git add .
> git commit -m “Le fichier .gitignore”
> git status
> deactivate

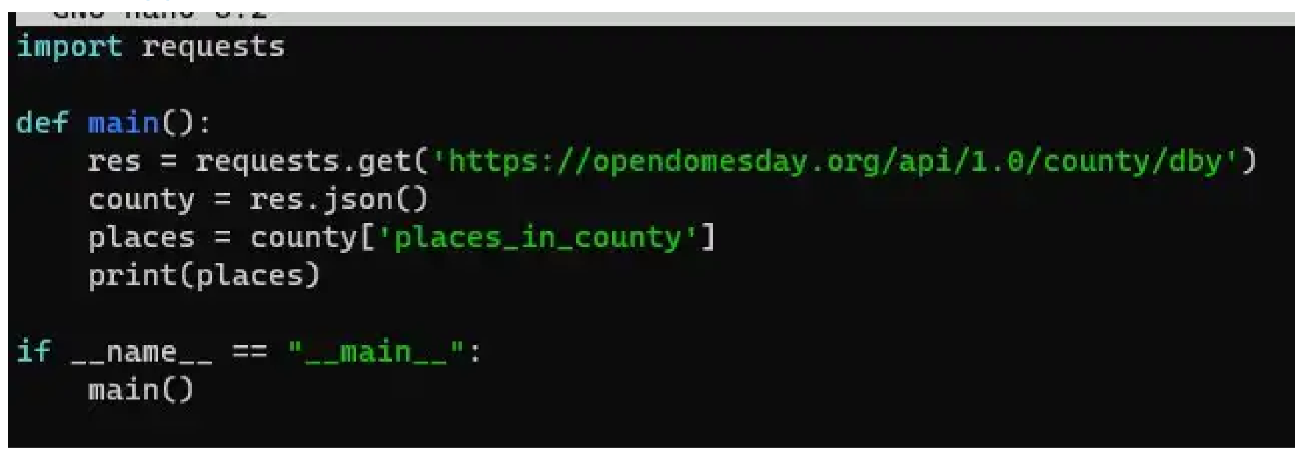
Exercice 2

1. Install the Python package Requests using pip

> pip install requests

2. Create a Python script that returns the list of all place ids in Derbyshire

> nano ex2.py



```
import requests

def main():
    res = requests.get('https://opendomesday.org/api/1.0/county/dby')
    county = res.json()
    places = county['places_in_county']
    print(places)

if __name__ == "__main__":
    main()
```

> python3 ex2.py

3. Commit your changes in Git

> git add ex2.py
> git commit -m “Python script that returns the list of all place ids in Derbyshire”

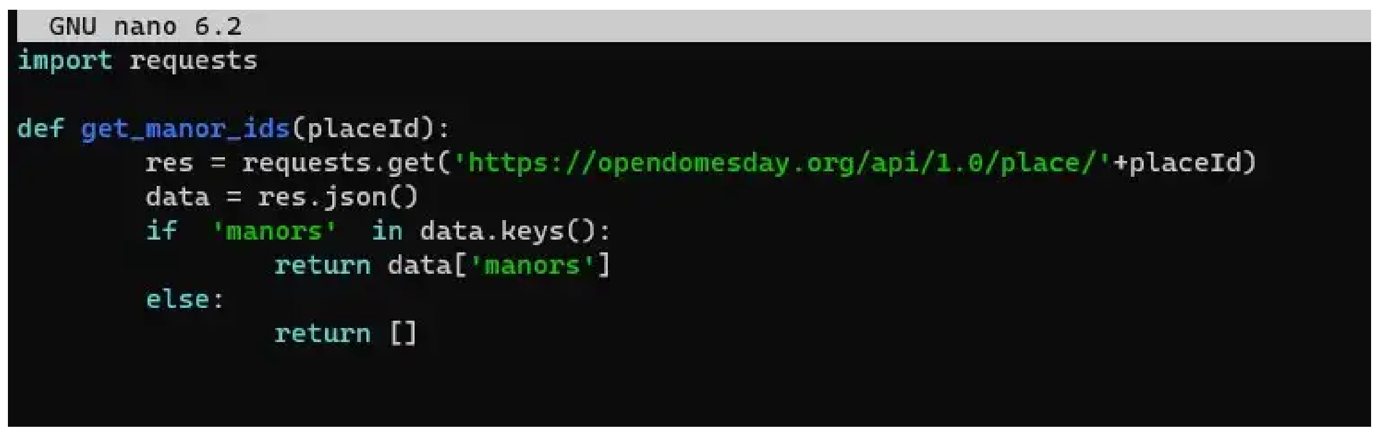
Exercice 3

1. Create a Python module with a get_manor_ids function that takes a place id as parameter and returns the list of manors.

> nano module.py

2. Check that calling your module does not produce any output

> 2. .env/bin/python3 module.py



```
  GNU nano 6.2
import requests

def get_manor_ids(placeId):
        res = requests.get('https://opendomesday.org/api/1.0/place/'+placeId)
        data = res.json()
        if  'manors'  in data.keys():
                return data['manors']
        else:
                return []
```

3. To test your module, open a python interpreter and call your function with the first place id from Derbyshire

> .env/bin/python3
> import module
> module.get_manor_ids(1036)

4. Add a if name == ’main’ : block with your previous test, at the end of your module, to make it usable as a script.

5. Check that calling your module now does produce an output.

> python3 module.py

6. Commit your changes in Git

> git add module.py
> git commit -m “Fonction qui cherche les manoirs”
> git push