TD5

Thomas Coudreau

March 2023

Exercise 1: Working Directory

1. `mkdir td4`
   `cd td4`

2. `git init`

3. `sudo apt-get install python3-pip`

4. `pip3 install virtualenv`

5. `python3 -m venv .env`

   There is a new directory called ".env" in the working directory.

6. `source .env/bin/activate`

   Yes, "(`.env`)" added as a prefix in the prompt.

7. `pip list`

8. `git status`

   Yes, Git shows the ".env" directory as an untracked file. It is not a good thing to commit the virtual environment files to Git, as they can contain system-specific and user-specific configurations.

9. `echo ".env" > .gitignore`

10. `git status`

    Yes, we can commit the .gitignore file.

11. `git add .gitignore`
    `git commit -m "Initial commit with .gitignore"`

Exercise 2: Python Script

1. `pip install requests`

2. ```
   import requests

   def get_place_ids_in_derbyshire():
       url = "https://opendomesday.org/api/1.0/county/dby/"
       response = requests.get(url)
       data = response.json()
       place_ids = [place["id"] for place in data["places_in_county"]]
       return place_ids

   if __name__ == "__main__":
       place_ids = get_place_ids_in_derbyshire()
       print(place_ids)
   ```

   Save this code in a file called ‘derbyshire_places.py‘.

3. Commit your changes in Git.

   ```
   git add derbyshire_places.py
   git commit -m "Add Python script for Derbyshire places"
   ```

Exercise 3: Python Module

1. ```
   import requests

   def get_manor_ids(place_id):
       url = f"https://opendomesday.org/api/1.0/place/{place_id}"
       response = requests.get(url)
       data = response.json()
       manor_ids = [manor["id"] for manor in data["manors"]]
       return manor_ids
   ```

2. `python3 manors.py`

3. `python3`

   ```
   from manors import get_manor_ids
   get_manor_ids(1036)
   ```

4. ```
   import requests
   from derbyshire_places import get_place_ids_in_derbyshire

   def get_manor_ids(place_id):
       url = f"https://opendomesday.org/api/1.0/place/{place_id}"
   ```
   ```

```
    response = requests.get(url)
    data = response.json()
    manor_ids = [manor["id"] for manor in data["manors"]]
    return manor_ids

if __name__ == "__main__":
    place_ids = get_place_ids_in_derbyshire()
    print(place_ids)

    test_place_id = place_ids[0]
    manor_ids = get_manor_ids(test_place_id)
    print("Manors in the first place:", manor_ids)
```

5. `python manors.py`

6. ```
   git add manors.py
   git commit -m "Add Python module manors.py"
   ```

Exercise 4: Python Program

```
pip install pandas
```

```
import requests
import pandas as pd
from derbyshire_places import get_place_ids_in_derbyshire

def get_manor_ids(place_id):
    """Return a list of manor ids for a given place id."""
    url = f"https://opendomesday.org/api/1.0/place/{place_id}"
    response = requests.get(url)
    data = response.json()
    manor_ids = [manor["id"] for manor in data["manors"]]
    return manor_ids

def get_manor_details(manor_id):
    """Return manor details for a given manor id."""
    url = f"https://opendomesday.org/api/1.0/manor/{manor_id}"
    response = requests.get(url)
    data = response.json()
    return data

if __name__ == "__main__":
    place_ids = get_place_ids_in_derbyshire()
    print("Place IDs:", place_ids)

    all_manor_ids = []
```
```

```
for place_id in place_ids:
    manor_ids = get_manor_ids(place_id)
    all_manor_ids.extend(manor_ids)

print("Manor IDs:", all_manor_ids)

manor_details = [get_manor_details(manor_id) for manor_id in all_manor_ids]

manor_data = [
    {"manor_id": manor["id"], "geld_paid": manor.get("geld", 0), "ploughs": manor.get("t
    for manor in manor_details
]

df = pd.DataFrame(manor_data)

total_geld_paid = df["geld_paid"].sum()
total_ploughs = df["ploughs"].sum()

print("Total geld paid:", total_geld_paid)
print("Total ploughs:", total_ploughs)

git add manors.py
git commit -m "Enrich manors.py to compute total geld paid and ploughs owned in Derbyshi
```