Cheatsheet for Django Project

Use Case	Commands (Windows)	Commands (MacOS)
Check Python Version **(Ensure it's 3.x)**	`py --version / python --version`	`python3 --version`
Setting up a Virtual Environment	`py -m venv <environment_name> / python -m venv <environment_name>`	`python3 -m venv <environment_name>`
Activate Virtual Environment	`.\env\Scripts\activate`	`source env/bin/activate`
Exit Virtual Environment	`deactivate`	`deactivate`
Install Django	`py -m pip install django / pip install django`	`pip3 install django`
Create a new Django project	`py -m django startproject <project_name> . / django-admin startproject <project_name> .`	`django-admin startproject <project_name> .`
Create a new Django App	`py manage.py startapp <app_name> / python manage.py startapp <app_name>`	`python3 manage.py startapp <app_name>`
Generate Database Migration Files	`py manage.py makemigrations / python manage.py makemigrations`	`python3 manage.py makemigrations`
Apply Migrations	`py manage.py migrate / python manage.py migrate`	`python3 manage.py migrate`
List Migrations along with their Status	`py manage.py showmigrations / python manage.py showmigrations`	`python3 manage.py showmigrations`
Access Django Shell	`py manage.py shell / python manage.py shell`	`python3 manage.py shell`
Run the Development Server	`py manage.py runserver / python manage.py runserver`	`python3 manage.py runserver`
Run the Server on Custom Port	`py manage.py runserver 1234 / python manage.py runserver 1234`	`python3 manage.py runserver 1234`

Collaborating with Teammates

1. Always use the **<u>Same Django version</u>** for consistency.
    a. Create and Activate the **Virtual Environment** to isolate dependencies
    b. Check the Django version using
        i. `py -m django --version` on Windows
        ii. `python3 -m django --version` on MacOS
    c. Install the specific Django version using
        i. `py -m pip install django==a.b.c` on Windows
        ii. `python3 -m pip install django==a.b.c` on MacOS
2. **(Teammate 1)**
    a. Create and export dependencies using
        i. `py -m pip freeze > requirements.txt` on Windows
        ii. `pip3 freeze > requirements.txt` on MacOS
    b. Forward **requirements.txt** file to the other teammate
3. **(Teammate 2)**
    a. Install dependencies from **requirements.txt** using
        i. `py -m pip install -r requirements.txt` on Windows
        ii. `pip3 install -r requirements.txt` on MacOS

Recommendation:

Use Version Control Tools like **GitHub** to track progress and efficiently collaborate with teammates!