Django Cheat Sheet - Python ORM Course @ Software University

Create a virtual environment and activate it

`python3 -m venv myenv`	Creates a virtual environment named 'myenv', isolating your project dependencies
`source myenv/bin/activate # Linux/Mac`	Activates a virtual environment named 'myenv'
`myenv\Scripts\activate # Windows`	

Create/Install dependencies

`pip install -r requirements.txt`	Installs all required Python packages specified in the requirements.txt file
`pip freeze > requirements.txt`	Generates a requirements.txt file listing all installed Python packages and their versions

Create a Django app within a project

`python manage.py startapp app_name`	Sets up a new Django application within your project

Run migrations to create/update database schema

`python manage.py makemigrations`	Generates migration files based on changes to your models
`python manage.py migrate`	Applies the changes from migration files to the database

Unapply all migrations for a specific app

`python manage.py migrate app_name zero`	Reverts all migrations for the specified app

Show migration status and SQL for a specific app

`python manage.py showmigrations app_name`	Displays the migration status for a particular app
`python manage.py sqlmigrate app_name migration_number`	Shows the SQL for a specific migration

Reset the database

`python manage.py flush`	Deletes all data from the database and resets the tables

Create a superuser for the admin site

`python manage.py createsuperuser`	Creates a new superuser account to access the Django admin interface

Start the development server

`python manage.py runserver`	Launches the development server to preview your Django project

Generate Django shell for interactive database operations

`python manage.py shell`	Opens a Python shell with Django environment loaded for database operations

Access database shell

`python manage.py dbshell`	Opens a command-line interface to interact directly with your database