**Code Explanation**

**Django:** Django is a web application framework written in Python programming language. It is based on MVT (Model View Template) design pattern.

**Django Application:** A Django project contains the following packages and files. The outer directory is just a container for the application. We can rename it further.

* **manage.py:** It is a command-line utility which allows us to interact with the project in various ways and also used to manage an application that we will see later on in this tutorial.
* **django app:** A directory located inside, is the actual application package name. Its name is the Python package name which we'll need to use to import module inside the application.
* **\_\_init\_\_.py:** It is an empty file that tells to the Python that this directory should be considered as a Python package.
* **settings.py:** This file is used to configure application settings such as database connection, static files linking etc.
* **urls.py:** This file contains the listed URLs of the application. In this file, we can mention the URLs and corresponding actions to perform the task and display the view.
* **views.py:** A view function is a Python function that takes a Web request and returns a Web response. This response can be the HTML contents of a Web page, or a redirect.
* **wsgi.py:** It is an entry-point for WSGI-compatible web servers to serve Django project.

**Office\_File\_Tracking\_System:** This Directory contain the main Project logic and the description of the files are given below.

* **\_\_init\_\_.py:** It is an empty file that tells to the Python that this directory should be considered as a Python package.
* **settings.py:** This file is used to configure application settings such as database connection, static files linking, media directory, html templates etc.
* **urls.py:** This file contains the listed URLs of the content and account application.
* **wsgi.py:** It is an entry-point for WSGI-compatible web servers to serve Django project.
* **asgi.py:** It sets the setting environment for our project.

**Accounts App:** This application deals with all the accounts related functionalities for student, staff and admin.

* **\_\_init\_\_.py:** It is an empty file that tells to the Python that this directory should be considered as a Python package.
* **models.py:** It contains the essential fields and behaviors of the data that will be stored in the database.
* **urls.py:** This file contains the urls and path for the appropriate action to be taken.
* **admin.py:** Django has a builtin admin interface that reads metadata from your models, such as fields, and lets you perform CRUD operations for free.To be able to perform such operations, you need to register your models in the admin.py file
* **apps.py:** This file is created to help the user include any application configuration for the app.
* **views.py:** This file defines all the function which handle the logical part for the action performed by the user in this fille basically every function fetches data from the model and processes it then returns a html page or httpResponseas a result.

**Content App:** This application deals with all the content like application, home pages, about page etc.

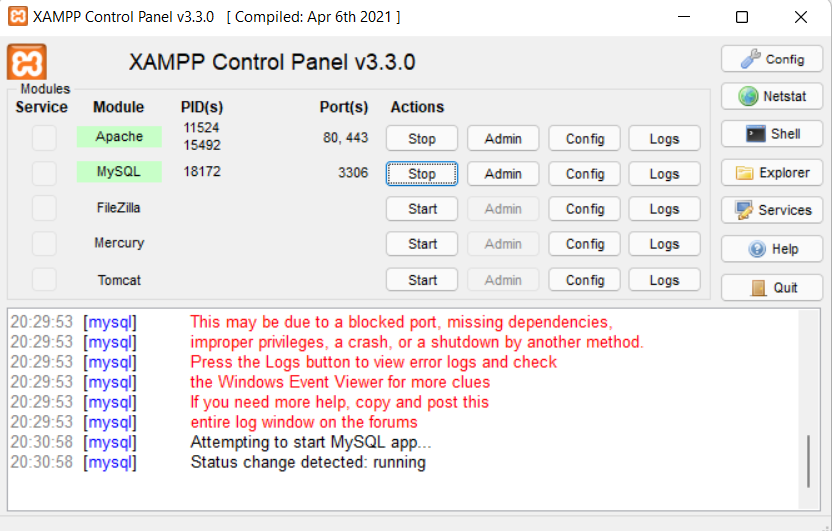
* **\_\_init\_\_.py:** It is an empty file that tells to the Python that this directory should be considered as a Python package.
* **models.py:** It contains the essential fields and behaviors of the data that will be stored in the database.
* **urls.py:** This file contains the urls and path for the appropriate action to be taken.
* **admin.py:** Django has a builtin admin interface that reads metadata from your models, such as fields, and lets you perform CRUD operations for free.To be able to perform such operations, you need to register your models in the admin.py file
* **apps.py:** This file is created to help the user include any application configuration for the app.
* **views.py:** This file defines all the function which handle the logical part for the action performed by the user in this fille basically every function fetches data from the model and processes it then returns a html page or httpResponseas a result.

**Requirements**

* 1. **Django**
  2. **MySql**
  3. **Xampp Control Panel**

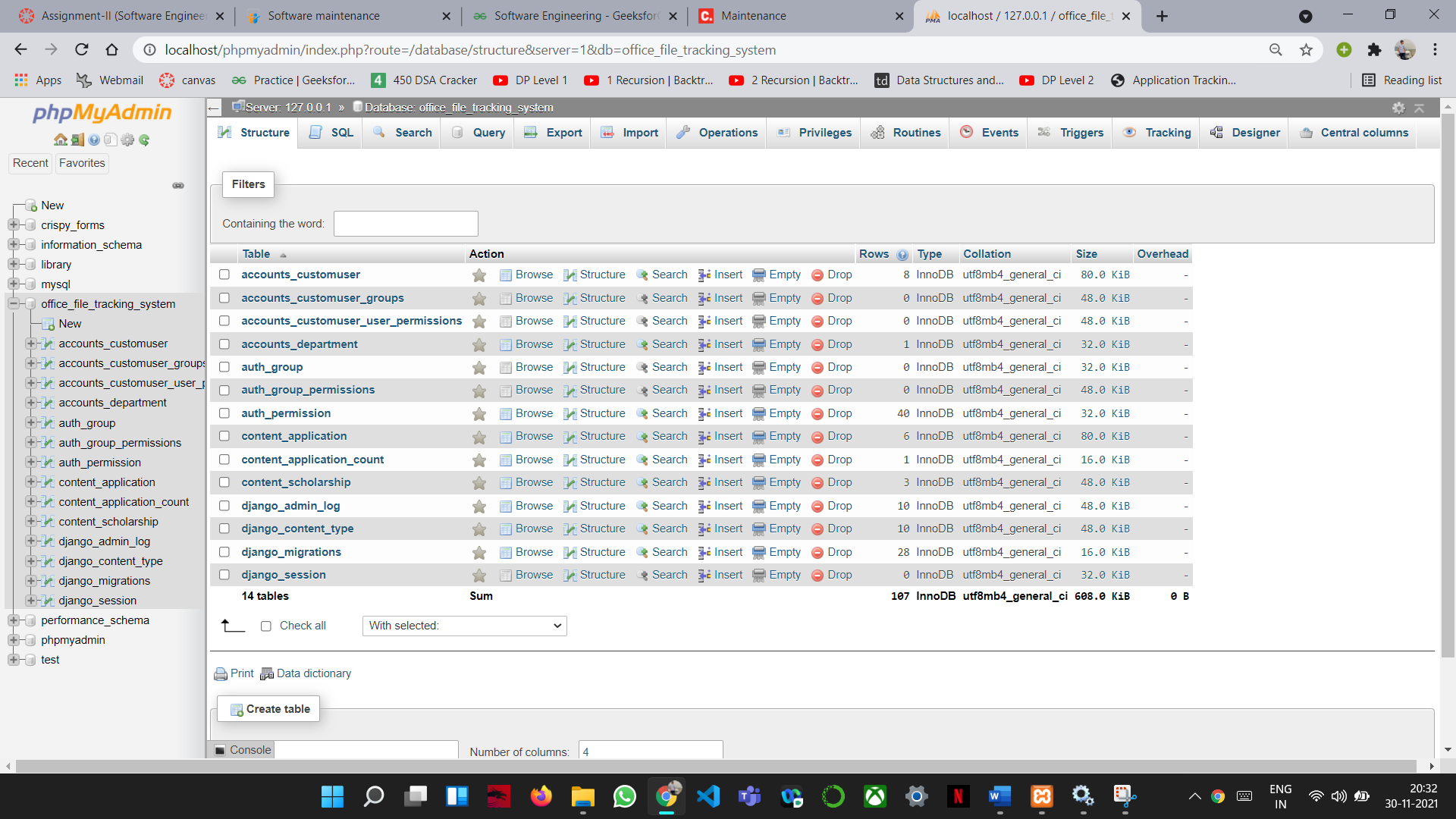
**How to Run The Code**

**Step 1:** Open Xampp Control Panel and run the Apache web server and MySql server.

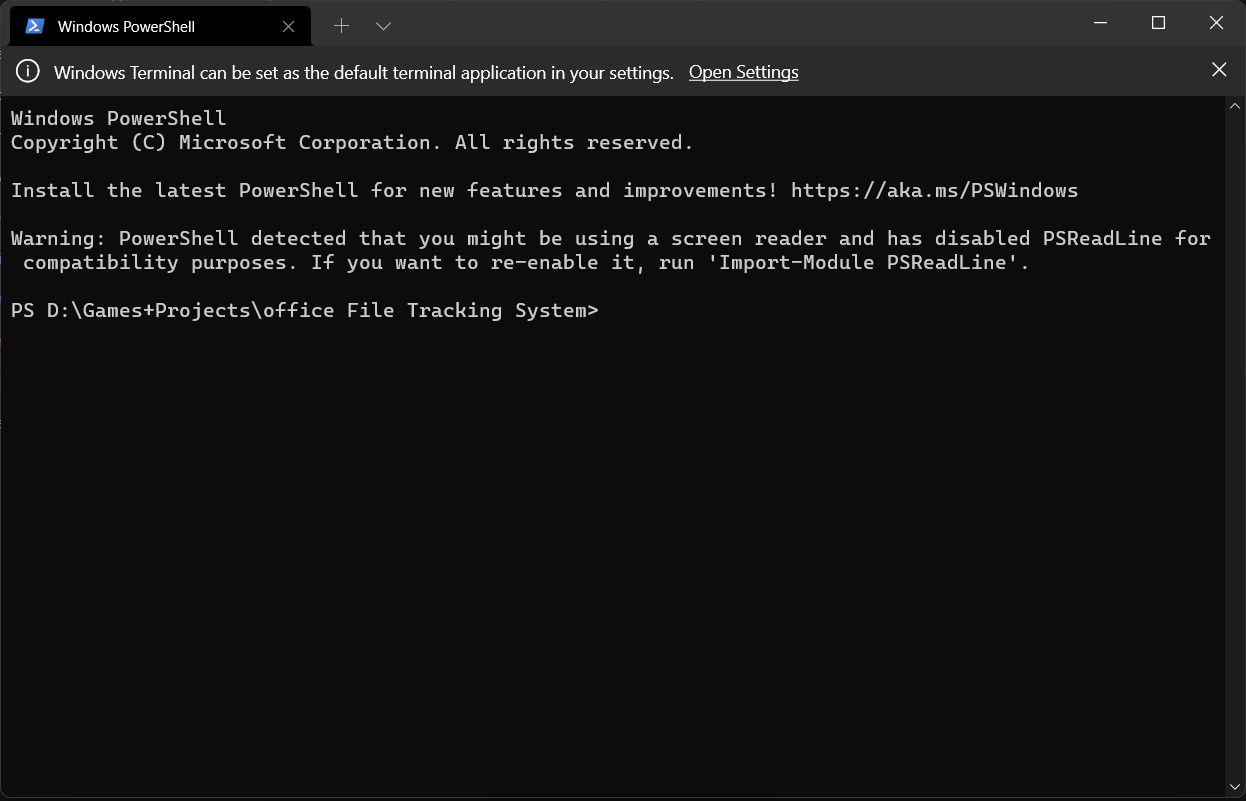


**Step 2:** Now open MySql admin from Xampp Control Panel and create the database

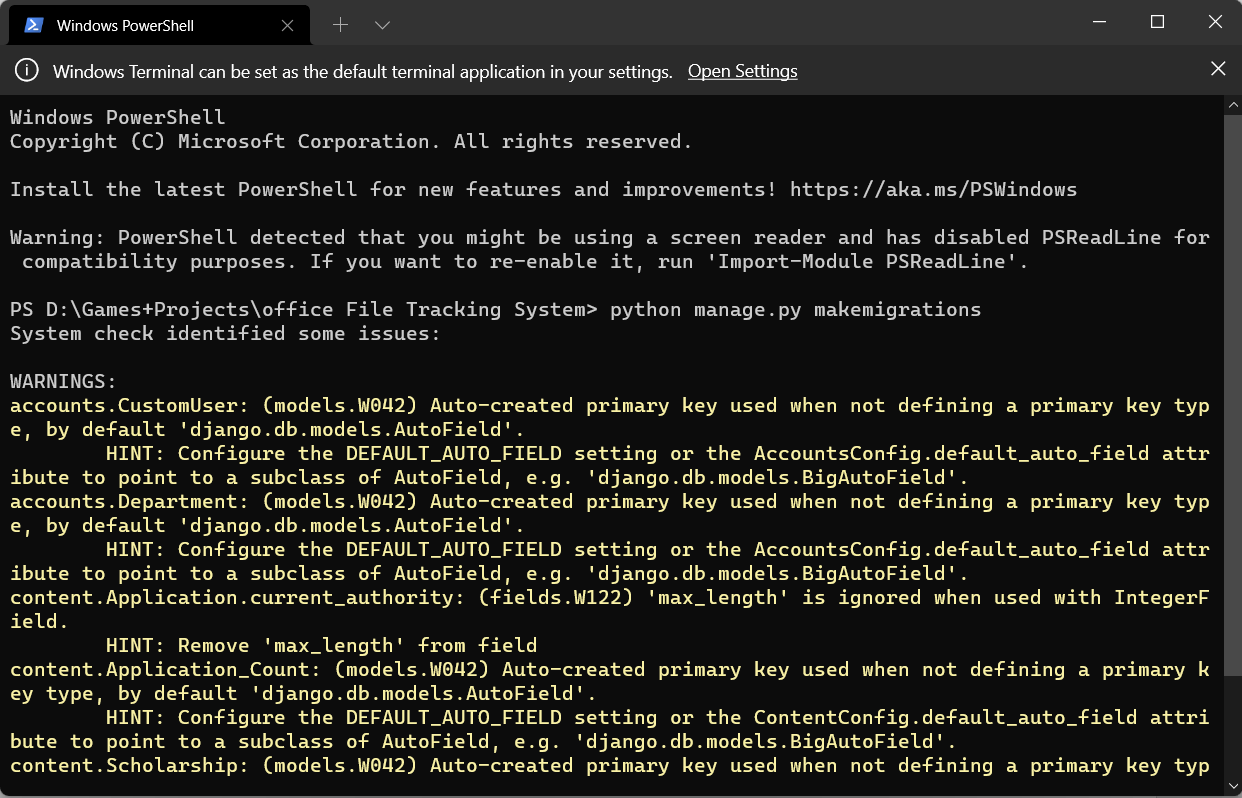
office\_file\_tracking\_system.



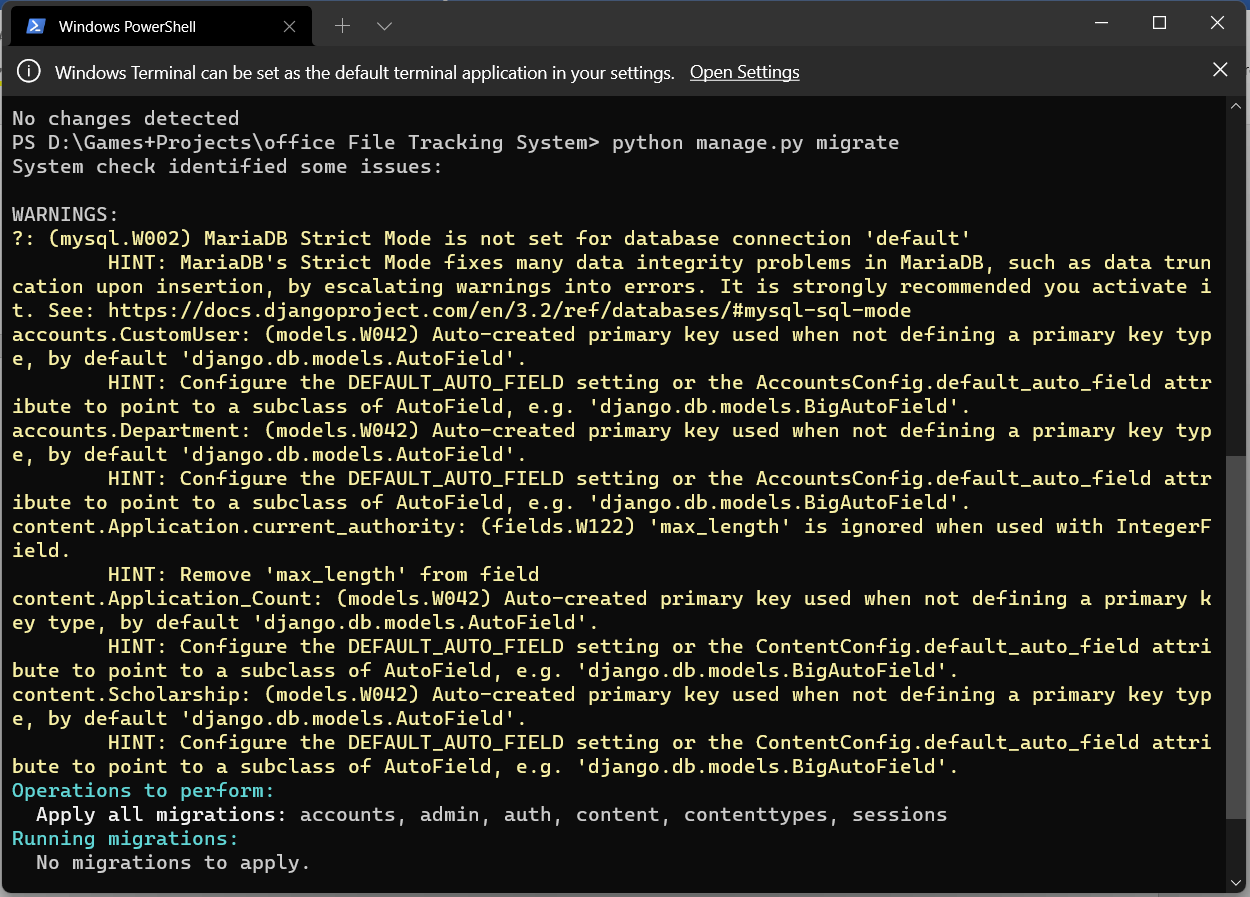
**Step 3:** Open command prompt at the location where your code is stored.



**Step 4:** Run command -python manage.py makemigrations.

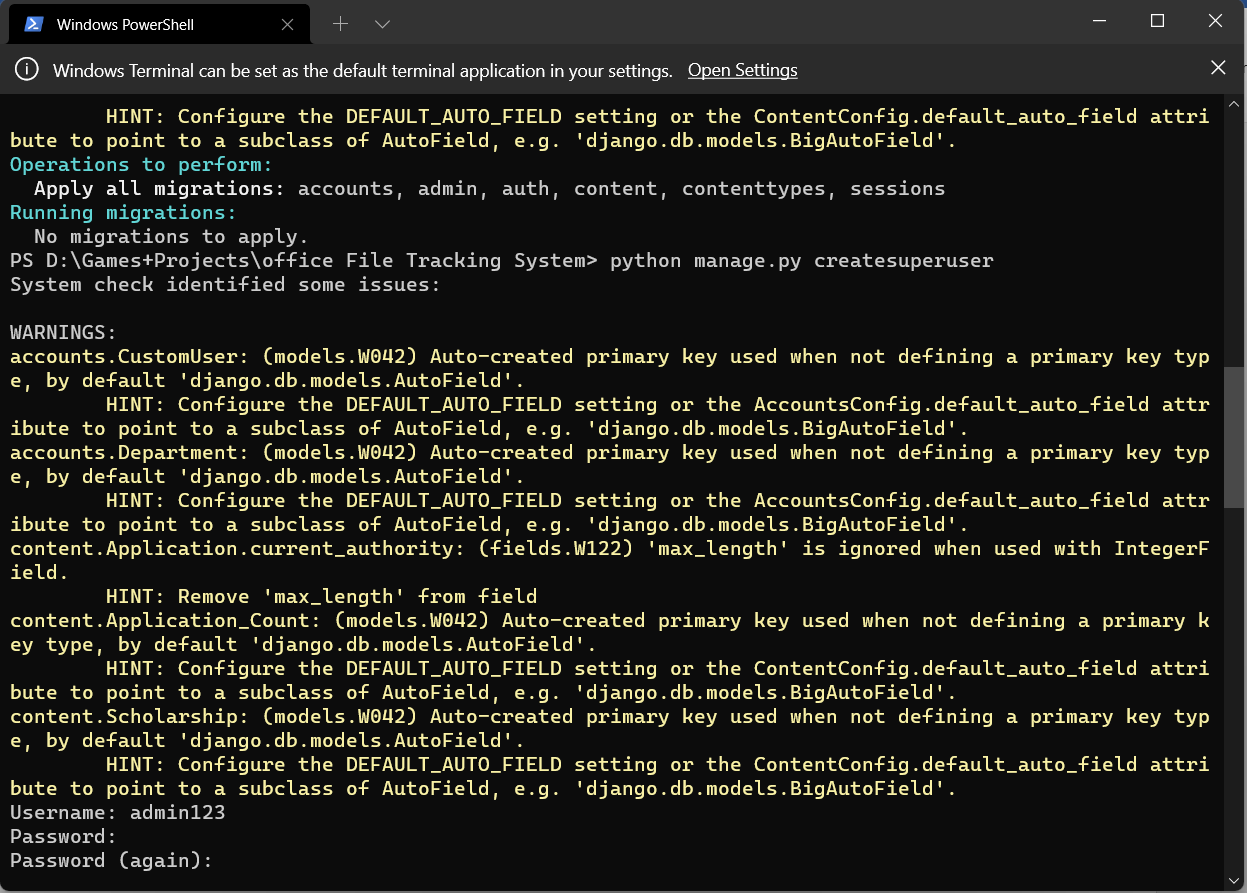


**Step 5:** Run command -python manage.py migrate.

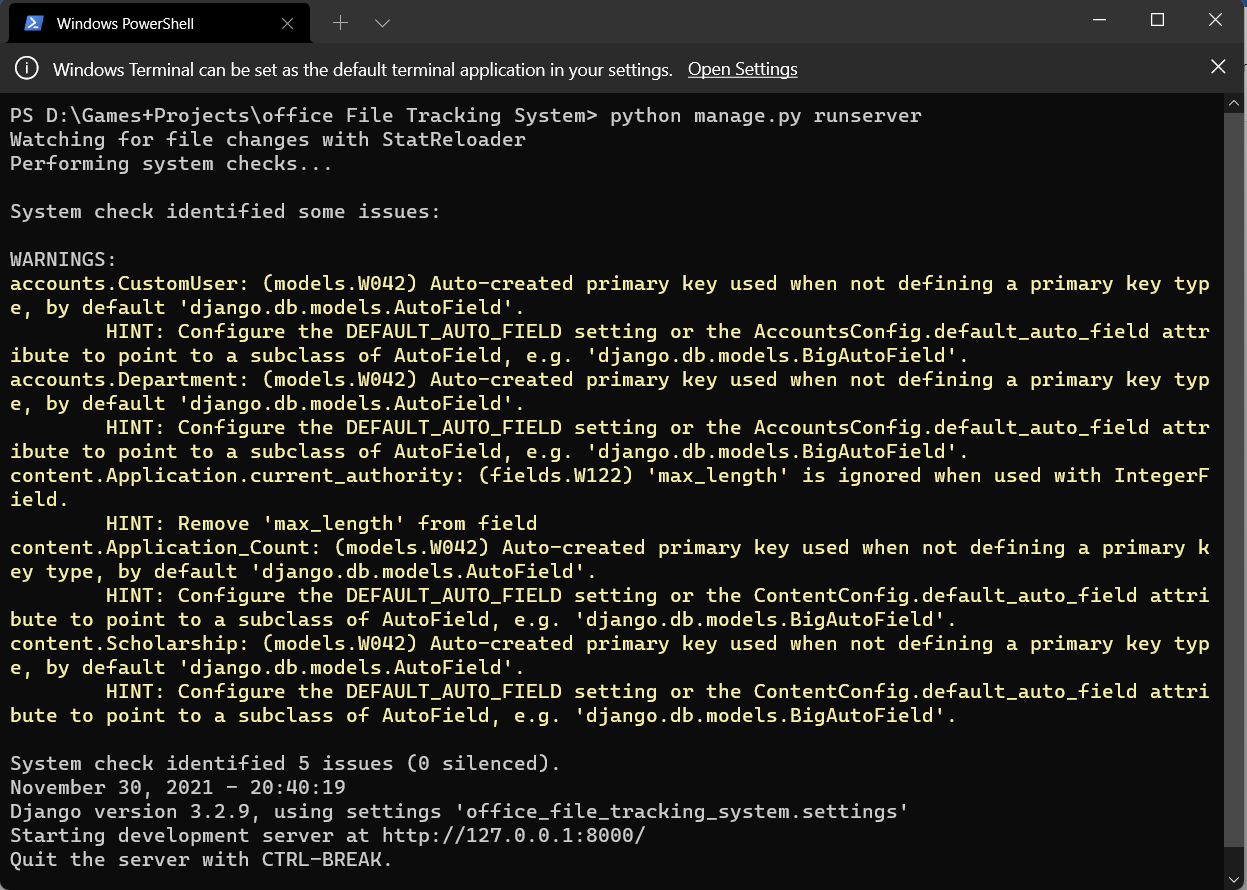


**Step 6:** Run command -python manage.py createsuperuser

Now Enter the username and password for django admin.



**Step 7:** Run command -python manage.py runserver



**Step 8:** Now copy paste the link http://127.0.0.1:8000/ in your web browser and you are ready to

use this Web Application.

