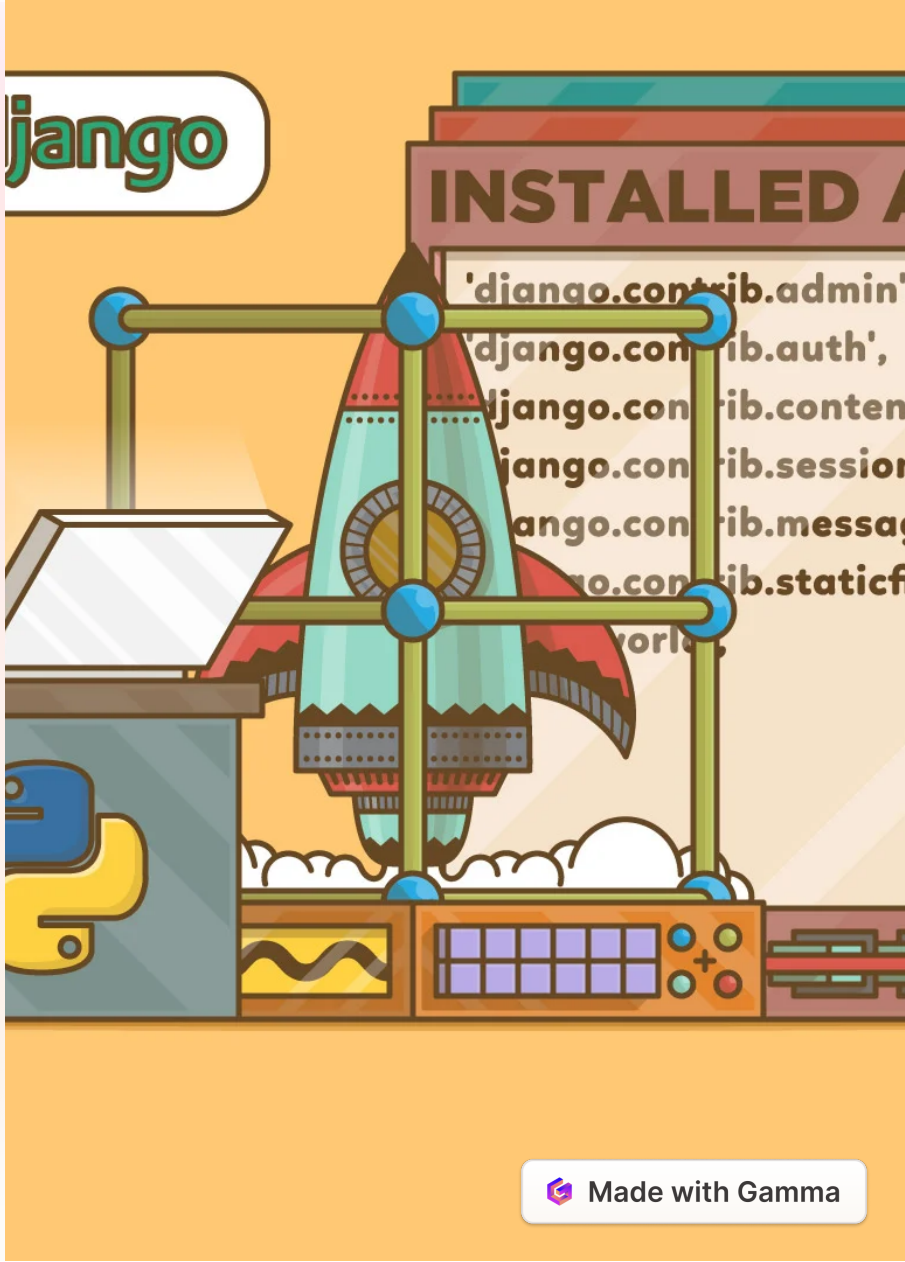
Models and Databases in Django

prepared by :

ENG : / Ghada Atef



A Django model is a table in your database.

Up until now in this tutorial, output has been static data from Python or HTML templates.

Now we will see how Django allows us to work with data, without having to change or upload files in the prosess.

In Django, data is created in objects, called Models, and is actually tables in a database.

Django Models

Creating models and fields

Models are created as classes that inherit from the Django `models.Model` class. Fields define the data that a table can store, such as text, numbers, or dates.

1

Definition and Purpose

Models are the blueprint for creating database tables in Django. Each model represents a table in the database.

2

3

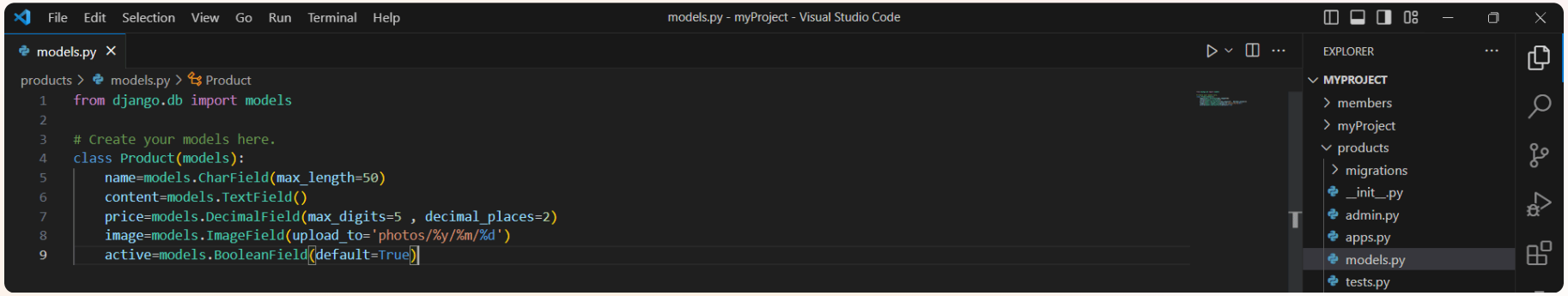
Database operations in Django

Django provides many built-in methods to create, read, update, and delete data from the database.

Create Table (Model)

To create a model, navigate to the `models.py` file in the `/products/` folder. (new app)

Open it, and add a Product table by creating a Product `class`, and describe the table fields in it:



SQLite Database

When we created the Django project, we got an empty SQLite database.

It was created in the `myproject` root folder, and has the filename `db.sqlite3`.

By default, all Models created in the Django project will be created as tables in this database.

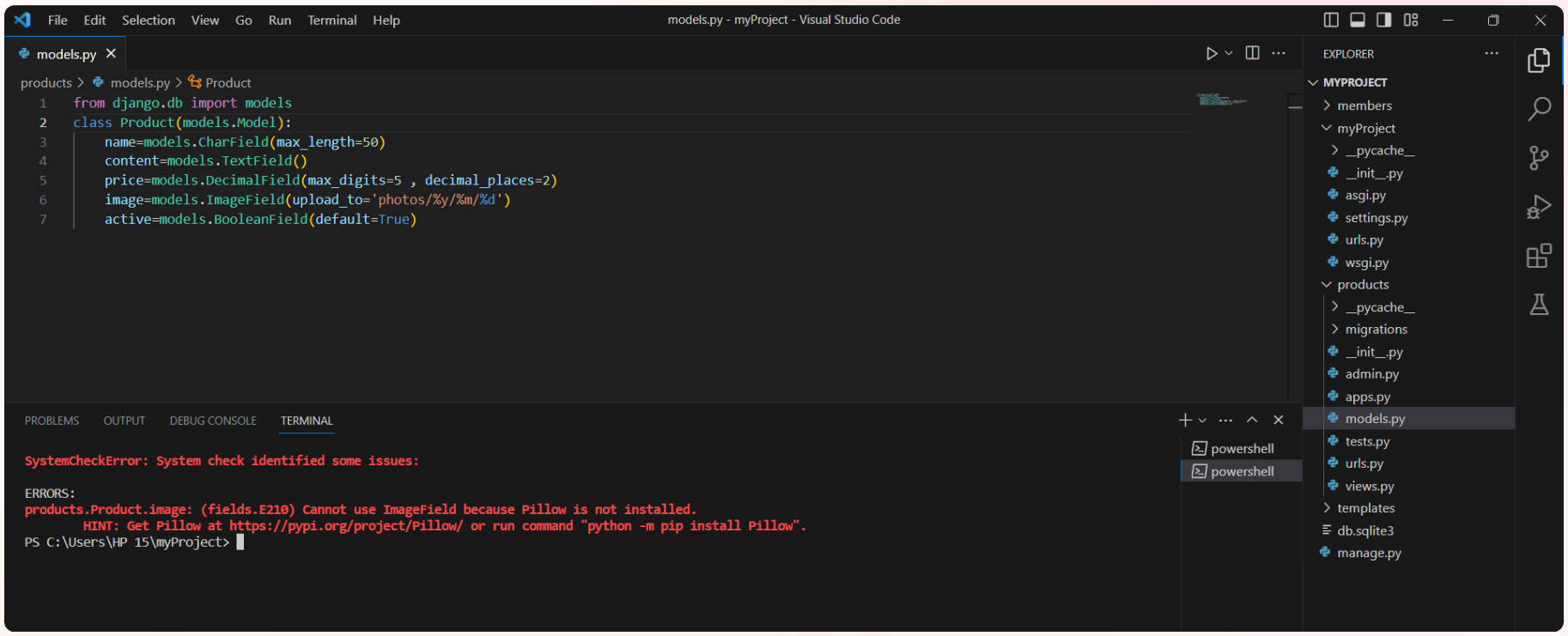
Migrate

Now when we have described a Model in the `models.py` file, we must run a command to actually create the table in the database.

Navigate to the `/myproject/` folder and run this command:

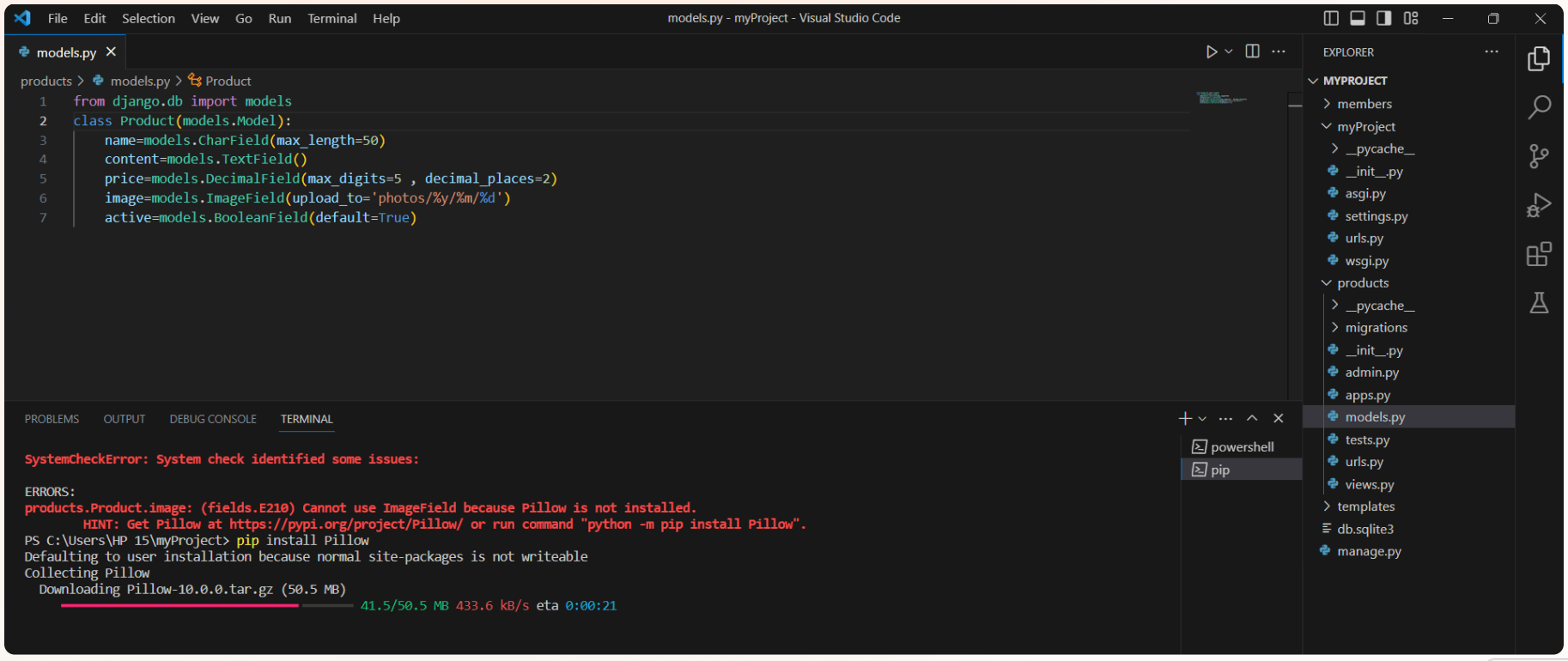
py manage.py makemigrations

may be appear error like this



to solve this we should write this command :

pip install Pillow



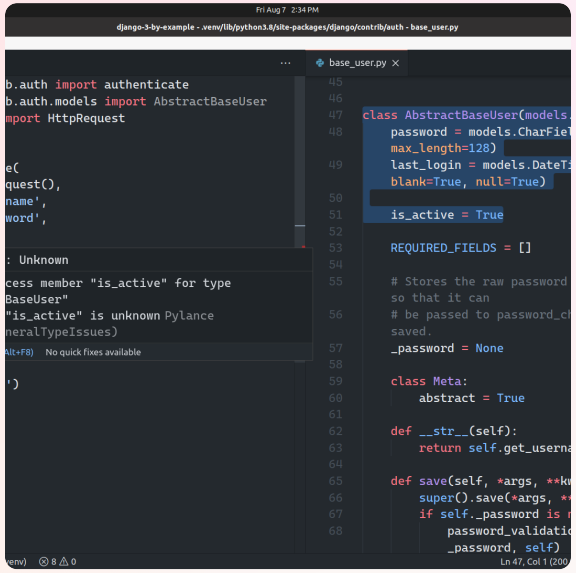
Django creates a file describing the changes and stores the file in the `/migrations/` folder:

Run the migrate command:

py manage.py migrate

Now you have a Product table in you database!

Django ORM



What is ORM?

ORM stands for Object-Relational Mapping. In Django, the ORM is used to translate between Python objects and database tables.

Advantages of using ORM

ORM allows developers to write database queries using Python instead of SQL, which makes querying the database much easier and less error-prone.

Sample code snippet

To create a new record in the database using Django ORM, we can use the following code:

Querying the database using Django

1

Retrieving all objects

To retrieve all records from a table in the database, we can use the `objects.all()` method.

2

Filtering records

We can filter records in the database using the `objects.filter()` method and specifying the conditions for the filter.

3

Ordering records

We can order the records in the database using the `objects.order_by()` method and specifying the field by which we want to order.

Conclusion

Models and databases are a crucial aspect of web development with Django. By following best practices and utilizing Django's built-in tools, we can create efficient and maintainable web applications that scale with ease.

Thank You

Any Questions ?