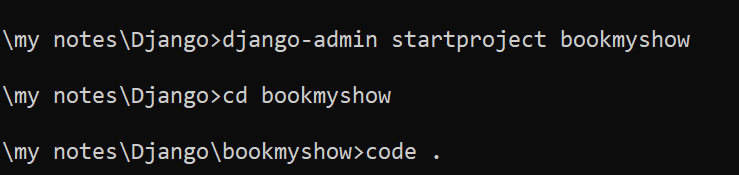
Model :

1. Prerequisite : engine, database name, username, password
   1. Ex : mysql, student, root, root
2. Model represents structure of table in django
3. To represent model, we have create a class in models.py
   1. File models.py is present in every application
4. Every model class that we create is must be subclass of Model class
   1. Model class is present inside models package (from django.db import models)
   2. models package has multiple useful componets like
      1. models.Model -> Model class
      2. models.IntergerField() -> IntegerFiled() class to provide data type to column of table
5. After creating model class, register it into admin.py
   1. admin.py present in every application

Creating a project

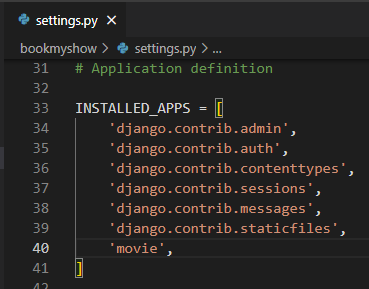
1. create project and open in vs code



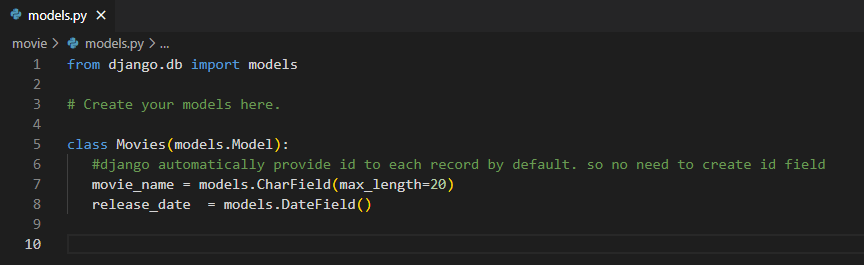
1. create application : movie



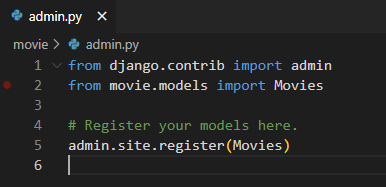
1. register application in settings.py



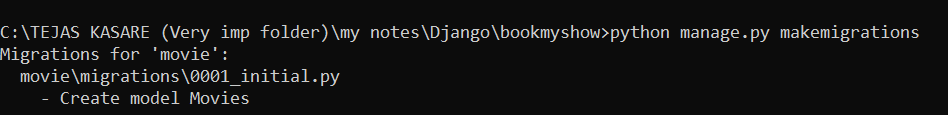
1. open model.py from movie and create Movies model in it



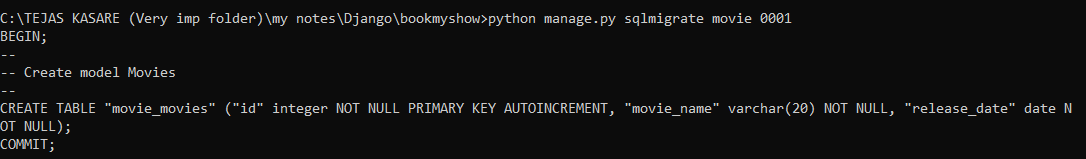
1. Register Movie model in admin.py



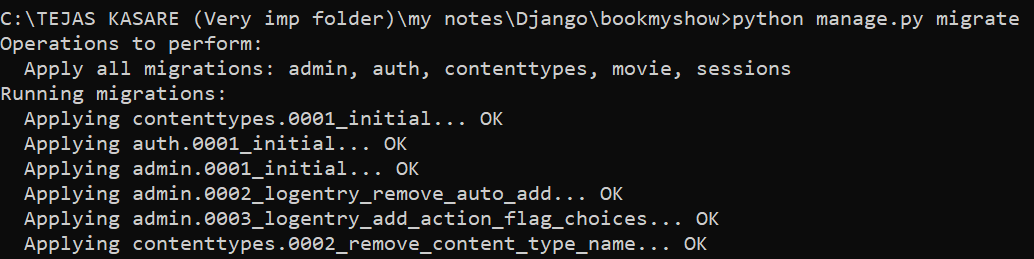
1. Convert model class code into respective sql code



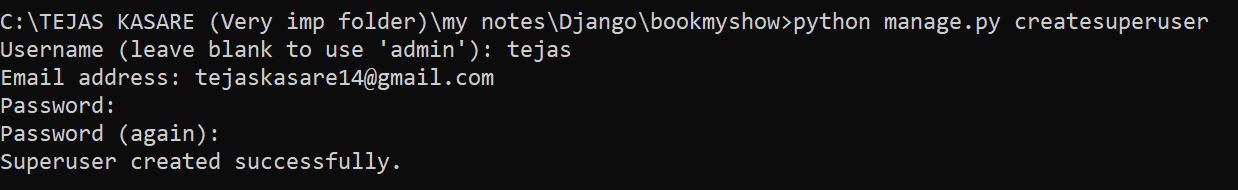
1. You can check respective sql code also by :



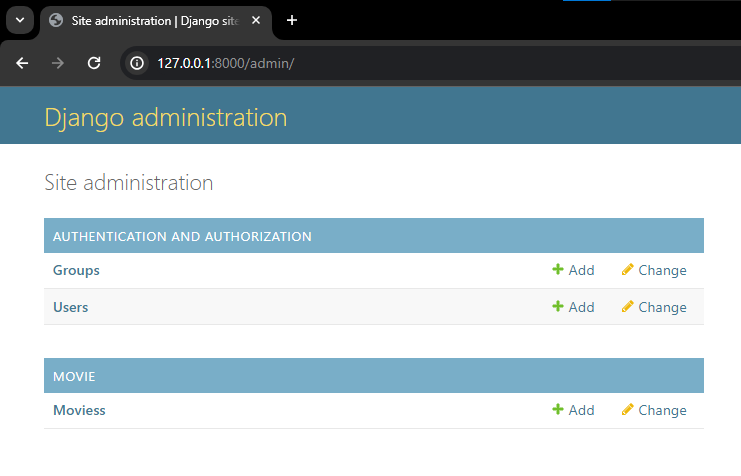
1. Executing sql code



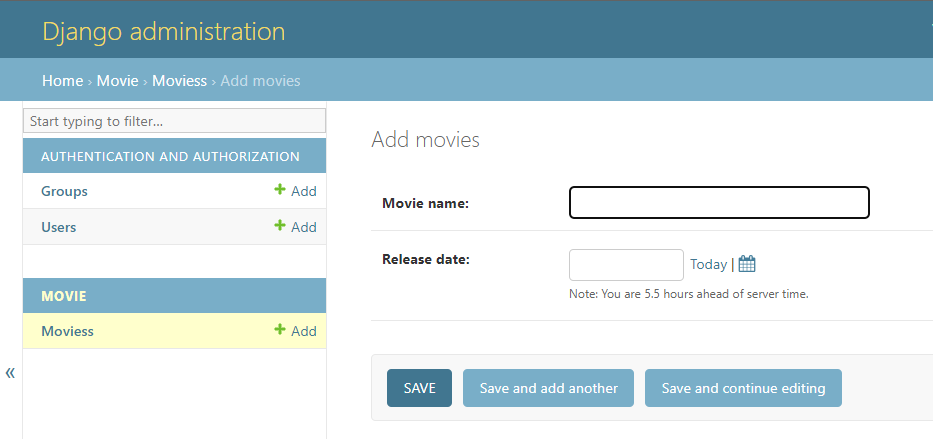
1. Creating super user for admin panel (tejas, django@123)



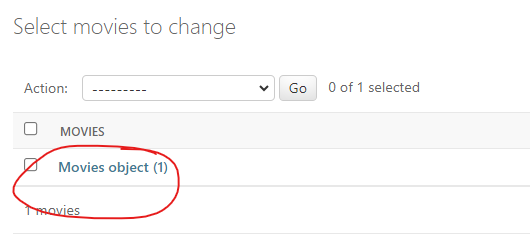
1. Start server
2. Go to admin url : <http://127.0.0.1:8000/admin> and add above credentials
   1. You will get following page with movie app



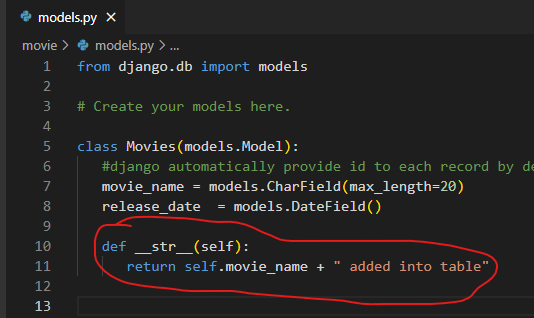
* 1. Click on add and add a movie



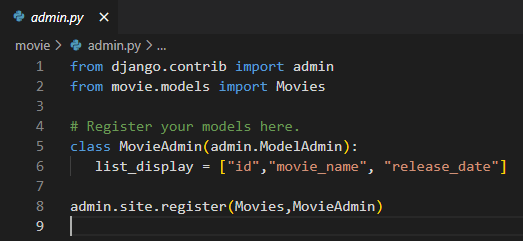
* 1. After adding movie, you will get movie object, not columns

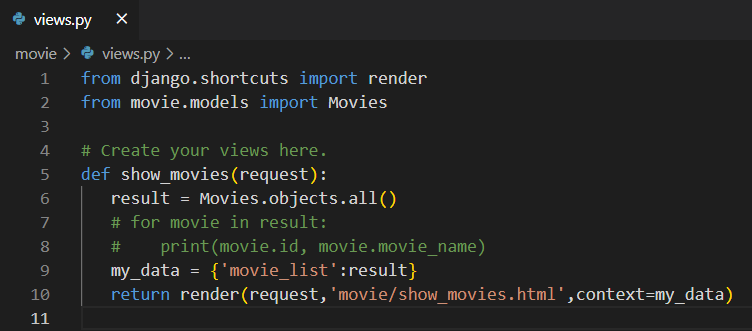


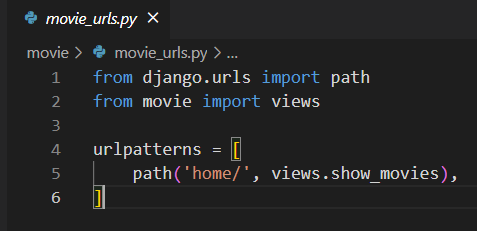
1. Sending response like “xyz movie added to table”. Add following in models.py



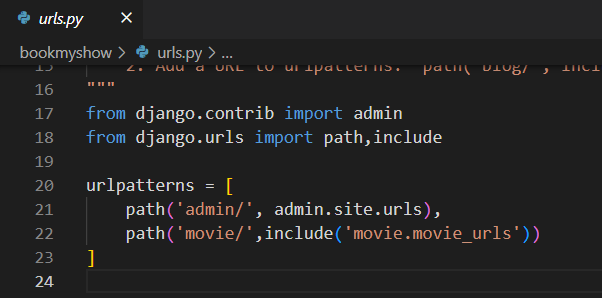
1. Display column name instead on model object (step 11-c) in admin panel. Add following in admin.py



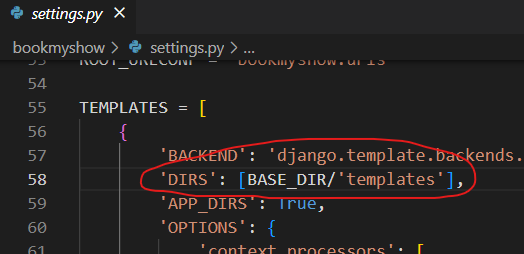
1. Displaying data into html
   1. Create show\_movie() view in movie -> views.py
      1. 
   2. Create application level url for above view
      1. Create movie\_urls.py in movie folder



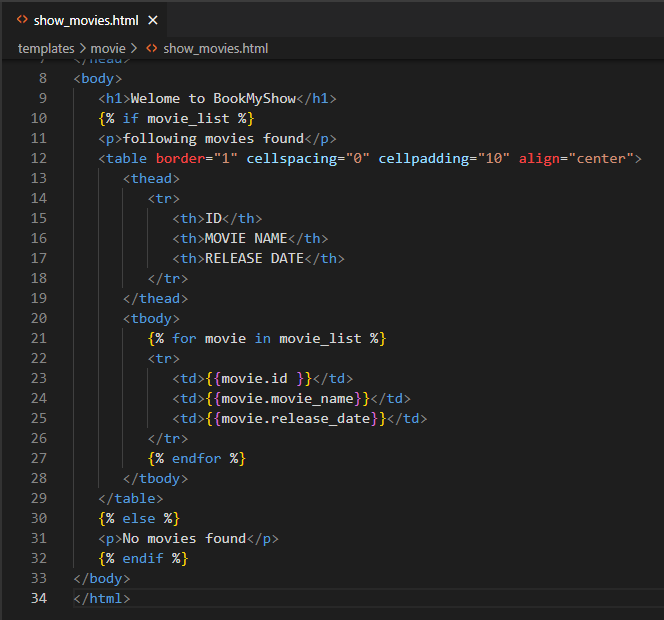
* 1. Project level url for above movie application (urls.py)



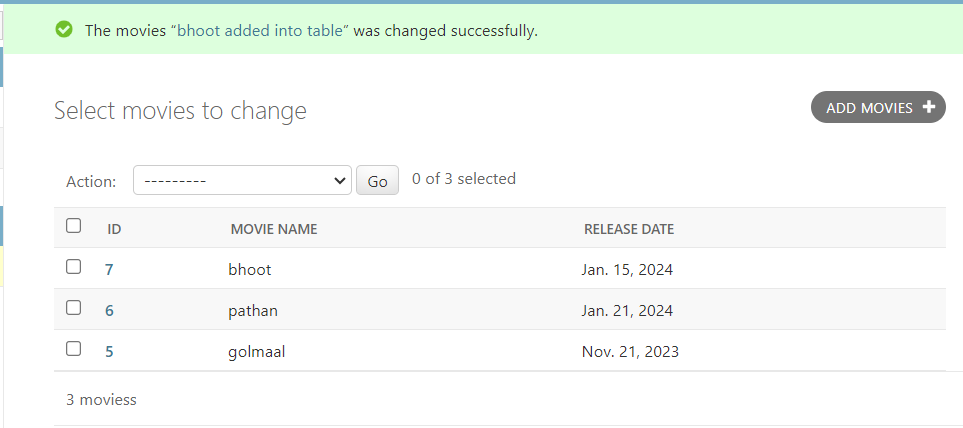
* 1. Create templates folder in bookmyshow folder (project level)
  2. Register templates folder in settings.py



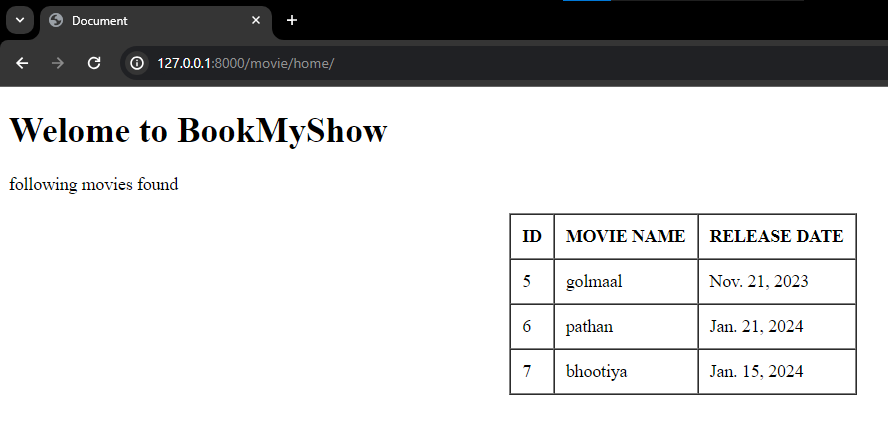
* 1. Create movie folder in template folder
     1. Create show\_movie.html in templates -> movie -> show\_movie.html



* 1. Runserver and check for output
     1. Admin panel : we are having 3 movies



* + 1. Get all movies in our url : http://127.0.0.1:8000/movie/home/



* + 1. DONE !