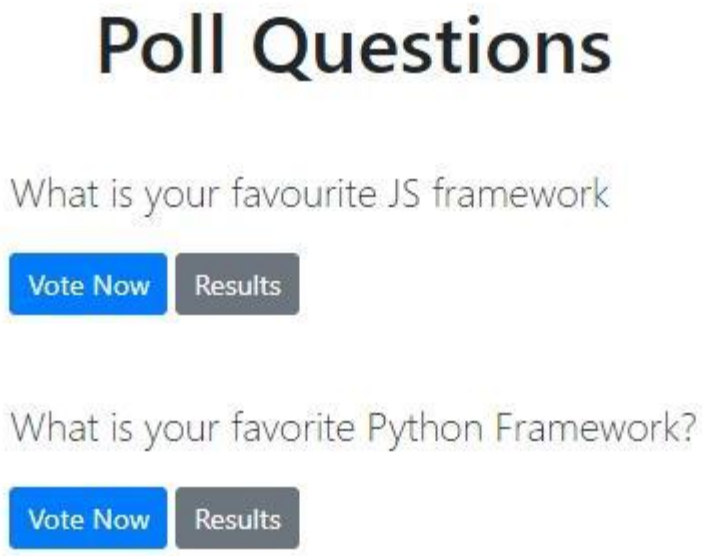
Voting system project using Django

Voting System Project Using Django Framework

Project Title: Pollster (Voting System) web application using Django framework

Type of Application (Category): Web application.

Introduction: We will create a pollster (voting system) web application using Django. This application will conduct a series of questions along with many choices. A user will be allowed to give voting for that question by selecting a choice. Based on the answer the total votes will be calculated and it will be displayed to the user. Users can also check the result of the total votes for specific questions on the website directly. We will also build the admin part of this project. Admin user will be allowed to add questions and manage questions in the application.



Pre-requisite: Knowledge of Python and basics of Django Framework. Python should be installed in the system. Visual studio code or any code editor to work on the application.

Technologies used in the project: Django framework and SQLite database which comes by default with Django.

Implementation of the Project

Creating Project

Step-1: Create an empty folder pollster_project in your directory.

Step-2: Now switch to your folder and create a virtual environment in this folder using the following command.

Pip install pipenv

Pipenv shell

Step-3: A Pipfile will be created in your folder from the above step. Now install Django in your folder using the following command.

Pipenv install django

Step-4: Now we need to establish the Django project. Run the following command in your folder and initiate a Django project.

Django-admin startproject pollster

A New Folder with name pollster will be created. Switch to the pollster folder using the following command.

Cd pollster

The folder structure will look something like this.

Python manage.py runserver

Step-5: Create an app ‘polls‘ using the following command

Python manage.py startapp polls

Below is the folder structure after creating ”polls’ app in the project.

Create Models

Step-1: In your models.py file write the code given below to create two tables in your database. One is ‘Question‘ and the other one is ‘Choice‘. ‘Question’ will have two fields of ‘question_text’ and a ‘pub_date’. Choice has three fields: ‘question’, ‘choice_text’, and ‘votes’. Each Choice is associated with a Question.

```
From django.db import models

# Create your models here.


Class Question(models.Model):

    Question_text = models.CharField(max_length = 200)

    Pub_date = models.DateTimeField('date published')


    Def __str__(self):

        Return self.question_text
```

```
Class Choice(models.Model):

    Question = models.ForeignKey(Question, on_delete = models.CASCADE)

    Choice_text = models.CharField(max_length = 200)

    Votes = models.IntegerField(default = 0)

    Def __str__(self):

        Return self.choice_text
```

Step-2:Go to the settings.py file and in the list, INSTALLED_APPS write down the code below to include the app in our project. This will refer to the polls -> apps.py -> PollsConfig class.

```
INSTALLED_APPS = [

    ‘polls.apps.PollsConfig’,
```

‘django.contrib.admin’,

‘django.contrib.auth’,

‘django.contrib.contenttypes’,

‘django.contrib.sessions’,

‘django.contrib.messages’,

‘django.contrib.staticfiles’,

]

Step-3: We have made changes in our database and created some tables but in order to reflect these changes we need to create migration here and then Django application will stores changes to our models. Run the following command given below to create migrations.

Python manage.py makemigrations polls

Inside polls->migrations a file 0001_initial.py will be created where you can find the database tables which we have created in our models.py file. Now to insert all the tables in our database run the command given below…

Python manage.py migrate

Create an Admin User

Step-1: Run the command given below to create a user who can login to the admin site.

Python manage.py createsuperuser

It will prompt username which we need to enter.

Username: geeks123

Now it will prompt an email address which again we need to enter here.

Email address: xyz@example.com

The final step is to enter the password. We need to enter the password twice, the second time as a confirmation of the first.

Password: ******

Password (again): ******

Superuser created successfully.

Python3

From django.db import models

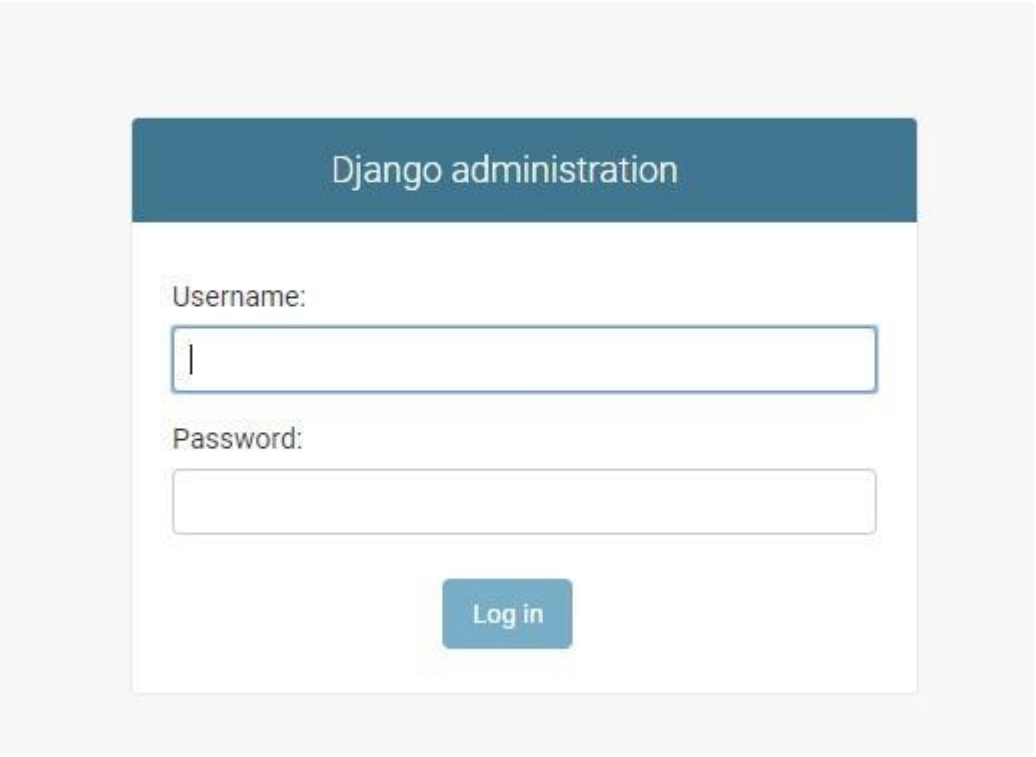
Create your models here.

Class Question(models.Model):

    Question_text = models.CharField(max_length = 200)

    Pub_date = models.DateTimeField(‘date published’)

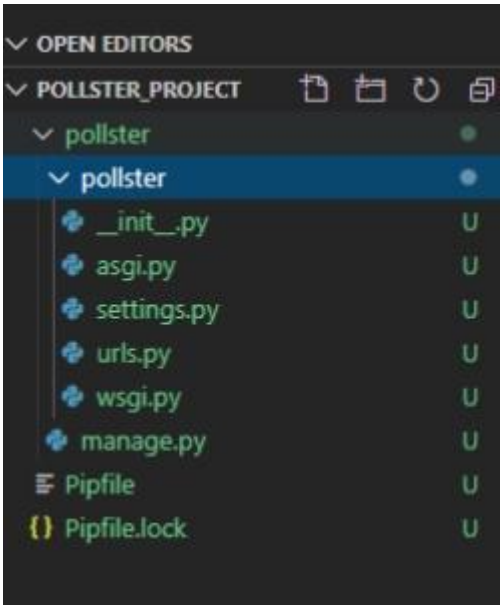
    Def __str__(self):



```
    Return self.question_text


Class Choice(models.Model):
    Question = models.ForeignKey(Question, on_delete = models.CASCADE)
    Choice_text = models.CharField(max_length = 200)
    Votes = models.IntegerField(default = 0)


    Def __str__(self):
        Return self.choice_text
```



Python3

```
INSTALLED_APPS = [
    'polls.apps.PollsConfig',
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
]
```

Step-3: We have made changes in our database and created some tables but in order to reflect these changes we need to create migration here and then Django application will stores changes to our models. Run the following command given below to create migrations.

```
Python manage.py makemigrations polls
```

Inside polls->migrations a file 0001_initial.py will be created where you can find the database tables which we have created in our models.py file. Now to insert all the tables in our database run the command given below…

```
Python manage.py migrate
```

Create an Admin User

Step-1: Run the command given below to create a user who can login to the admin site.

Python manage.py createsuperuser

It will prompt username which we need to enter.

Username: geeks123

Now it will prompt an email address which again we need to enter here.

Email address: xyz@example.com

The final step is to enter the password. We need to enter the password twice, the second time as a confirmation of the first.

Password: ******

Password (again): ******

Superuser created successfully.