WP Lab 9

Q1.Create a Register page and Success page with the following requirements:

i.Register page should contain four input TextBoxes for UserName, Password,

Email id and Contact Number and also a button to submit. Make the username as

compulsory field and other fields as optional.

ii.On button click, Success page is displayed with message "Welcome

{UserName}" and also his Email and Contact Number has to be displayed.

iii. Use secure technique to send details to the Success page (Hint: use csrftoken)

Design a website with two pages.

**user\_registration/forms.py**

from django import forms

class RegistrationForm(forms.Form):

username = forms.CharField(max\_length=100, required=True, label='Username')

password = forms.CharField(widget=forms.PasswordInput, required=False, label='Password')

email = forms.EmailField(required=False, label='Email')

contact\_number = forms.CharField(max\_length=15, required=False, label='Contact Number')

**user\_registration/urls.py**

from django.urls import path

from . import views

urlpatterns = [

path('register/', views.register, name='register'),

path('success/<str:username>/<str:email>/<str:contact\_number>/', views.success, name='success'),

]

**user\_registration/views.py**

from django.shortcuts import render, redirect

from .forms import RegistrationForm

def register(request):

if request.method == "POST":

form = RegistrationForm(request.POST)

if form.is\_valid():

username = form.cleaned\_data.get('username')

email = form.cleaned\_data.get('email')

contact\_number = form.cleaned\_data.get('contact\_number')

# Redirect with parameters in the URL

return redirect('success', username=username, email=email, contact\_number=contact\_number)

else:

form = RegistrationForm()

return render(request, 'user\_registration/register.html', {'form': form})

def success(request, username, email, contact\_number):

return render(request, 'user\_registration/success.html', {

'username': username,

'email': email,

'contact\_number': contact\_number

})

**user\_registration/templates/user\_registration/register.html**

<html>

<head><title>Register</title></head>

<body>

<h1>Register</h1>

<form method="post">

{% csrf\_token %}

{{ form.as\_p }}

<button type="submit">Submit</button>

</form>

</body>

</html>

**user\_registration/templates/user\_registration/success.html**

<html>

<head><title>Success</title></head>

<body>

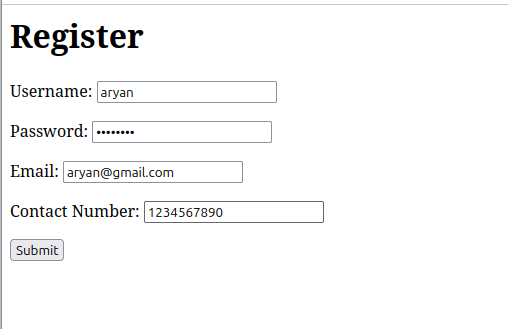
<h1>Welcome {{ username }}</h1>

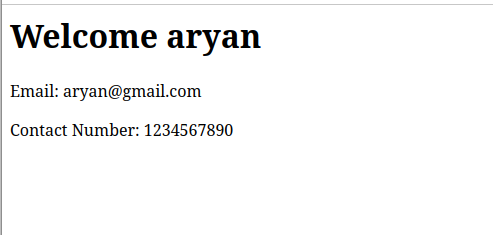
<p>Email: {{ email }}</p>

<p>Contact Number: {{ contact\_number }}</p>

</body>

</html>

**Output:**



Q2.“How is the book ASP.NET with c# by Vipul Prakashan?” Give the user three choice :

i) Good

ii) Satisfactory

iii) Bad.

Provide a VOTE button. After user votes, present the result in

percentage using labels next to the choices.

**book\_vote/models.py**

from django.db import models

class Vote(models.Model):

choice = models.CharField(max\_length=20)

count = models.IntegerField(default=0)

def \_\_str\_\_(self):

return self.choice

**book\_vote/forms.py**

from django import forms

class VoteForm(forms.Form):

CHOICES = [

('Good', 'Good'),

('Satisfactory', 'Satisfactory'),

('Bad', 'Bad'),

]

vote = forms.ChoiceField(choices=CHOICES, widget=forms.RadioSelect, required=True)

**book\_vote/views.py**

from django.shortcuts import render

from .forms import VoteForm

from .models import Vote

def vote(request):

if request.method == "POST":

form = VoteForm(request.POST)

if form.is\_valid():

# Get the vote choice from the form

vote\_choice = form.cleaned\_data['vote']

# Update the vote count for the selected choice

vote, created = Vote.objects.get\_or\_create(choice=vote\_choice)

vote.count += 1

vote.save()

return render(request, 'book\_vote/results.html', {'vote': vote\_choice})

else:

form = VoteForm()

# Get the total votes and percentage for each choice

total\_votes = sum(vote.count for vote in Vote.objects.all())

votes = Vote.objects.all()

percentages = {vote.choice: (vote.count / total\_votes \* 100) if total\_votes else 0 for vote in votes}

return render(request, 'book\_vote/vote.html', {'form': form, 'percentages': percentages})

**book\_vote/urls.py**

from django.urls import path

from . import views

urlpatterns = [

path('vote/', views.vote, name='vote'),

]

**book\_vote/templates/book\_vote/vote.html**

<html>

<head><title>Vote for Book</title></head>

<body>

<h1>How is the book "ASP.NET with C# by Vipul Prakashan?"</h1>

<form method="post">

{% csrf\_token %}

{{ form.as\_p }}

<button type="submit">VOTE</button>

</form>

<h2>Current Vote Percentages:</h2>

<ul>

{% for choice, percentage in percentages.items %}

<li>{{ choice }}: {{ percentage|floatformat:2 }}%</li>

{% endfor %}

</ul>

</body>

</html>

**book\_vote/templates/book\_vote/results.html**

<html>

<head><title>Vote Results</title></head>

<body>

<h1>Thank you for voting!</h1>

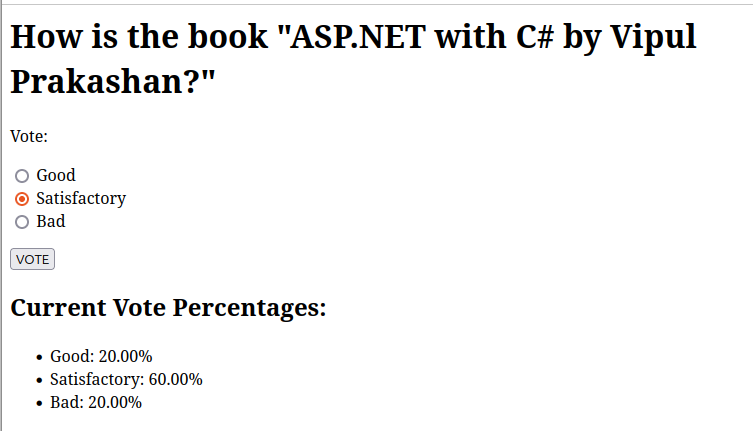
<h2>Your vote: {{ vote }}</h2>

<a href="{% url 'vote' %}">Back to voting</a>

</body>

</html>

**Output:**

****

Q3.Create a website with two pages. Page 1 has two TextBoxes (name and total marks) and one ‘Calculate’ Button as shown in the figure. On clicking the ‘Calculate’ Button, CGPA (total marks/50) along with the name should be displayed in the Page 2 . Use Django sessions to store the information.

**cgpa\_calculator/forms.py**

from django import forms

class CGPAForm(forms.Form):

name = forms.CharField(max\_length=100, label='Name', required=True)

total\_marks = forms.IntegerField(label='Total Marks', required=True)

**cgpa\_calculator/views.py**

from django.shortcuts import render, redirect

from .forms import CGPAForm

def calculate(request):

if request.method == 'POST':

form = CGPAForm(request.POST)

if form.is\_valid():

# Get data from the form

name = form.cleaned\_data['name']

total\_marks = form.cleaned\_data['total\_marks']

# Store the data in session

request.session['name'] = name

request.session['total\_marks'] = total\_marks

return redirect('result')

else:

form = CGPAForm()

return render(request, 'cgpa\_calculator/form.html', {'form': form})

def result(request):

# Get the data from session

name = request.session.get('name')

total\_marks = request.session.get('total\_marks')

# Calculate CGPA (total\_marks / 50)

if total\_marks is not None:

cgpa = total\_marks / 50

else:

cgpa = 0

return render(request, 'cgpa\_calculator/result.html', {'name': name, 'cgpa': cgpa})

**cgpa\_calculator/urls.py**

from django.urls import path

from . import views

urlpatterns = [

path('calculate/', views.calculate, name='calculate'),

path('result/', views.result, name='result'),

]

**cgpa\_calculator/templates/cgpa\_calculator/form.html**

<html>

<head><title>CGPA Calculator</title></head>

<body>

<h1>Enter Your Name and Total Marks</h1>

<form method="post">

{% csrf\_token %}

{{ form.as\_p }}

<button type="submit">Calculate</button>

</form>

</body>

</html>

**cgpa\_calculator/templates/cgpa\_calculator/result.html**

<html>

<head><title>CGPA Result</title></head>

<body>

<h1>Hello {{ name }}</h1>

<p>Your CGPA is: {{ cgpa }}</p>

<a href="{% url 'calculate' %}">Go back and calculate again</a>

</body>

</html>

**Output:**

