Name :- Janvi Chauhan

Rollno:- 03

Msc(Cs)-6

Django Assignment

Assignment 1

---

1. Create a Django project with specification

Urls and views

---```

`urls.py` in the project directory:

```python
# specification_project/urls.py
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('main.urls')),  # Include main app's URLs
]
```

`urls.py` in the `main` app directory:

Create a new Django app called `main`:

```bash
cd specification_project
python manage.py startapp main
```

```python
# main/urls.py
from django.urls import path
from . import views

urlpatterns = [
    path('', views.home, name='home'),
    path('about/', views.about, name='about'),
    path('contact/', views.contact, name='contact'),
]
```

`views.py` in the `main` app directory:

```python
# main/views.py
from django.shortcuts import render
from django.http import HttpResponse

def home(request):
    return HttpResponse("Welcome to the homepage!")
```

```
def about(request):
    return HttpResponse("This is the about page.")

def contact(request):
    return HttpResponse("Contact us at: contact@example.com")
```

Update `urlpatterns` in `urls.py` in the project directory:

```python
# specification_project/urls.py
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('main.urls')),  # Include main app's URLs
]
```

---

2. Create a form.html for enter 2 numbers and perform calculator by

using post method.

```html
<!-- form.html -->
<!DOCTYPE html>
```

```
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Calculator</title>
</head>
<body>
    <h1>Calculator</h1>
    <form method="post">
        {% csrf_token %}
        <label for="num1">Enter first number:</label><br>
        <input type="number" id="num1" name="num1" required><br>
        <label for="num2">Enter second number:</label><br>
        <input type="number" id="num2" name="num2" required><br>
        <label for="operation">Select operation:</label><br>
        <select id="operation" name="operation">
            <option value="add">Addition (+)</option>
            <option value="subtract">Subtraction (-)</option>
            <option value="multiply">Multiplication (*)</option>
            <option value="divide">Division (/)</option>
        </select><br><br>
        <button type="submit">Calculate</button>
    </form>
</body>
</html>
```

4. By using admin-panel insert username and password verification

program.

python manage.py startapp verification

# verification/models.py
from django.db import models

class UserCredentials(models.Model):
    username = models.CharField(max_length=100)
    password = models.CharField(max_length=100)

    def __str__(self):
        return self.username
```

---

```python
# verification/admin.py
from django.contrib import admin
from .models import UserCredentials

admin.site.register(UserCredentials)
```

```bash
python manage.py makemigrations
python manage.py migrate

```

python manage.py createsuperuser
```

```python
# In views.py
from django.shortcuts import render, redirect
from .models import UserCredentials

def verify_user(request):
    if request.method == 'POST':
        username = request.POST.get('username')
        password = request.POST.get('password')

        # Retrieve user credentials from the database
        try:
            user = UserCredentials.objects.get(username=username, password=password)
            # User is verified
            return render(request, 'verification/success.html')
        except UserCredentials.DoesNotExist:
            # User credentials are incorrect
            return render(request, 'verification/failure.html')

    return render(request, 'verification/verification_form.html')
```

- `verification_form.html`:

```html
<!-- templates/verification/verification_form.html -->
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>User Verification</title>
</head>
<body>
    <h1>User Verification</h1>
    <form method="post">
        {% csrf_token %}
        <label for="username">Username:</label><br>
        <input type="text" id="username" name="username" required><br>
        <label for="password">Password:</label><br>
        <input type="password" id="password" name="password" required><br>
        <button type="submit">Verify</button>
    </form>
</body>
</html>
```

```

- `success.html`:

```html
<!-- templates/verification/success.html -->
<!DOCTYPE html>
<html lang="en">
<head>
```
```

```
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Verification Success</title>
</head>
<body>
    <h1>Verification Successful</h1>
    <!-- Add success message or redirect to another page -->
</body>
</html>
```

- `failure.html`:

```html
<!-- templates/verification/failure.html -->
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Verification Failure</title>
</head>
<body>
    <h1>Verification Failed</h1>
    <!-- Add failure message or redirect to another page -->
</body>
</html>
```

---

5. Create index.html page and as per need make changes in url.py

and views.py and settings.py also create CSS file.

```html
<!-- templates/index.html -->
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Welcome to My Website</title>
    <link rel="stylesheet" type="text/css" href="{% static 'css/styles.css' %}">
</head>
<body>
    <div class="container">
        <h1>Welcome to My Website</h1>
        <p>This is the index page.</p>
    </div>
</body>
</html>
```

```css
/* static/css/styles.css */
.container {
    max-width: 800px;
    margin: 0 auto;
    padding: 20px;
    text-align: center;
}
```

Define a view to render the `index.html` template.

`views.py` in the project directory:

```python
# views.py
from django.shortcuts import render

def index(request):
    return render(request, 'index.html')
```

`urls.py` in the project directory:

```python
# urls.py
from django.contrib import admin
```

```
from django.urls import path
from django.conf import settings
from django.conf.urls.static import static
from . import views

urlpatterns = [
    path('', views.index, name='index'),
    path('admin/', admin.site.urls),
] + static(settings.STATIC_URL, document_root=settings.STATIC_ROOT)
```

`settings.py` in the `my_project` directory:

```python
# my_project/settings.py



# Static files (CSS, JavaScript, Images)
STATIC_URL = '/static/'
```

---

6. Create form.py file with

Firstname

Lastname

Password

Gender

Submit

On click submit display this data in another page

django-admin startproject form_project

```

---

```

`forms.py` in the `form_app` app directory:

```python

# form_app/forms.py

from django import forms

class MyForm(forms.Form):

    first_name = forms.CharField(max_length=100)
```

```
    last_name = forms.CharField(max_length=100)
    password = forms.CharField(widget=forms.PasswordInput)
    GENDER_CHOICES = (
        ('M', 'Male'),
        ('F', 'Female'),
        ('O', 'Other'),
    )
    gender = forms.ChoiceField(choices=GENDER_CHOICES)
```

```python
# form_app/views.py
from django.shortcuts import render
from .forms import MyForm

def form_view(request):
    if request.method == 'POST':
        form = MyForm(request.POST)
        if form.is_valid():
            data = form.cleaned_data
            return render(request, 'form_app/display.html', {'data': data})
    else:
        form = MyForm()
    return render(request, 'form_app/form.html', {'form': form})
```

```html
<!-- templates/form_app/form.html -->
<!DOCTYPE html>
```

```
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Form</title>
</head>
<body>
    <h1>Enter Your Details</h1>
    <form method="post">
        {% csrf_token %}
        {{ form.as_p }}
        <button type="submit">Submit</button>
    </form>
</body>
</html>
```

```html
<!-- templates/form_app/display.html -->
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Display Data</title>
</head>
<body>
    <h1>Submitted Data</h1>
    <p>First Name: {{ data.first_name }}</p>
```

```
    <p>Last Name: {{ data.last_name }}</p>
    <p>Password: ********</p>
    <p>Gender: {{ data.gender }}</p>
</body>
</html>
```

```python
# form_app/urls.py
from django.urls import path
from . import views

urlpatterns = [
    path('', views.form_view, name='form_view'),
]
```

```python
# form_project/urls.py
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('form_app.urls')),
]
```

---

7. Enter any number in textbox and press button and check that

number is prime or not.

---

django-admin startproject prime_check_project

```

```python

# prime_checker/views.py

from django.shortcuts import render

def prime_check(request):

    if request.method == 'POST':

        number = int(request.POST.get('number'))

        is_prime = True if number > 1 and all(number % i != 0 for i in range(2, int(number**0.5) + 1)) else False

        return render(request, 'prime_checker/result.html', {'number': number, 'is_prime': is_prime})

    return render(request, 'prime_checker/form.html')

```

```

```html
<!-- templates/prime_checker/form.html -->
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Prime Checker</title>
</head>
<body>
    <h1>Prime Number Checker</h1>
    <form method="post">
        {% csrf_token %}
        <label for="number">Enter a number:</label>
        <input type="number" id="number" name="number" required>
        <button type="submit">Check Prime</button>
    </form>
</body>
</html>
```

`result.html` in the `templates/prime_checker` directory:

```html
<!-- templates/prime_checker/result.html -->
<!DOCTYPE html>
<html lang="en">
<head>
```

```
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Prime Checker Result</title>
</head>
<body>
    <h1>Prime Number Checker Result</h1>
    <p>The number {{ number }} is {% if is_prime %}prime{% else %}not prime{% endif %}.</p>
    <a href="{% url 'prime_check' %}">Back to form</a>
</body>
</html>
```

```python
# prime_checker/urls.py
from django.urls import path
from . import views

urlpatterns = [
    path('', views.prime_check, name='prime_check'),
]
```

Include the app's URLs in the project's `urls.py`:

`urls.py` in the project directory:

```python
# prime_check_project/urls.py
```

```
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('prime_checker.urls')),
]
```
``

8. Create a font dialog box by using text box and list box for font size

and font style as per the selection perform operation in new page.

```bash
cd font_dialog_project
python manage.py startapp font_dialog
```

```python
# font_dialog/views.py
from django.shortcuts import render

def font_dialog(request):
    if request.method == 'POST':
```

```
        font_size = request.POST.get('font_size')
        font_style = request.POST.get('font_style')
        return render(request, 'font_dialog/result.html', {'font_size': font_size, 'font_style': font_style})
    return render(request, 'font_dialog/form.html')
```

```html
<!-- templates/font_dialog/form.html -->
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Font Dialog</title>
</head>
<body>
    <h1>Font Dialog</h1>
    <form method="post">
        {% csrf_token %}
        <label for="font_size">Font Size:</label>
        <input type="text" id="font_size" name="font_size" required>
        <label for="font_style">Font Style:</label>
        <select id="font_style" name="font_style">
            <option value="normal">Normal</option>
            <option value="italic">Italic</option>
            <option value="bold">Bold</option>
        </select>
        <button type="submit">Apply</button>
    </form>
```

</body>

</html>

```

---

```html
<!-- templates/font_dialog/result.html -->
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Font Dialog Result</title>
    <style>
        .text {
            font-size: {{ font_size }}px;
            font-style: {{ font_style }};
        }
    </style>
</head>
<body>
    <h1>Font Dialog Result</h1>
    <p class="text">This is a sample text with the selected font size and style.</p>
</body>
</html>
```

```

```python
# font_dialog/urls.py
from django.urls import path
from . import views

urlpatterns = [
    path('', views.font_dialog, name='font_dialog'),
]
```

Include the app's URLs in the project's `urls.py`:

`urls.py` in the project directory:

```python
# font_dialog_project/urls.py
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('font_dialog.urls')),
]
```

```

# 9. Create a model and fetch the content from model in our page.
```

```python
# models.py
from django.db import models

class Content(models.Model):
    title = models.CharField(max_length=100)
    body = models.TextField()

    def __str__(self):
        return self.title
```

---

python manage.py makemigrations

python manage.py migrate

```

---

```python
# views.py
from django.shortcuts import render
from .models import Content

def content_page(request):
    content = Content.objects.all()
    return render(request, 'your_app/content_page.html', {'content': content})
```

---<!--
content_page.html -->
```

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Content Page</title>
</head>
<body>
    <h1>Content Page</h1>
    <ul>
        {% for item in content %}
            <li>{{ item.title }}</li>
            <p>{{ item.body }}</p>
        {% endfor %}
    </ul>
</body>
</html>
```

```python
# urls.py
from django.urls import path
from . import views

urlpatterns = [
    path('content/', views.content_page, name='content_page'),
]
```

10. Insert data from the form in model .

```python
# models.py
from django.db import models

class FormData(models.Model):
    field1 = models.CharField(max_length=100)
    field2 = models.CharField(max_length=100)
    # Add more fields as needed

    def __str__(self):
        return f"{self.field1} - {self.field2}"  # Customize the display as needed
```

```python
# forms.py
from django import forms
from .models import FormData

class FormDataForm(forms.ModelForm):
    class Meta:
        model = FormData
        fields = ['field1', 'field2']  # Add more fields as needed
```

```python
```

```
# views.py
from django.shortcuts import render, redirect
from .forms import FormDataForm

def form_view(request):
    if request.method == 'POST':
        form = FormDataForm(request.POST)
        if form.is_valid():
            form.save()
            return redirect('success_page')  # Redirect to success page after saving data
    else:
        form = FormDataForm()
    return render(request, 'your_app/form.html', {'form': form})
```

```html
<!-- form.html -->
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Form</title>
</head>
<body>
    <h1>Form</h1>
    <form method="post">
        {% csrf_token %}
        {{ form.as_p }}
        <button type="submit">Submit</button>
```

```
    </form>
</body>
</html>
```

```html
<!-- success.html -->
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Success</title>
</head>
<body>
    <h1>Form Submitted Successfully!</h1>
    <!-- Add any success message or redirection link here -->
</body>
</html>
```

```python
# urls.py
from django.urls import path
from . import views

urlpatterns = [
    path('form/', views.form_view, name='form_view'),
    path('success/', views.success_view, name='success_page'),  # Define success page URL
]
```