**Set up a Django Project**

1. **Install Django**: Make sure you have Django installed. If not, run:

pip install django

1. **Create a Django Project**:

django-admin startproject registration\_project

1. **Navigate to the Project Directory**:

cd registration\_project

1. **Create a Django App**:

manage.py startapp users

1. **Register the App**: Add users to your INSTALLED\_APPS in settings.py:

INSTALLED\_APPS = [

...

'users',

]

**Step 2: Define the Model**

1. In users/models.py, define a model to store user registration details:

from django.db import models

class User(models.Model):

name = models.CharField(max\_length=100)

age = models.IntegerField()

address = models.TextField()

mail\_id = models.EmailField()

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

return self.name

1. **Run Migrations**:

manage.py makemigrations

manage.py migrate

**Step 3: Create a Form**

1. In users/forms.py, create a Django form to handle user input:

from django import forms

from .models import User

class UserRegistrationForm(forms.ModelForm):

class Meta:

model = User

fields = ['name', 'age', 'address', 'mail\_id']

**Step 4: Create the View**

1. In users/views.py, create a view to handle the registration form submission:

from django.shortcuts import render, redirect

from .forms import UserRegistrationForm

def register(request):

if request.method == 'POST':

form = UserRegistrationForm(request.POST)

if form.is\_valid():

form.save()

return redirect('success')

else:

form = UserRegistrationForm()

return render(request, 'users/register.html', {'form': form})

def success(request):

return render(request, 'users/success.html')

**Step 5: Create URLs**

1. In users/urls.py, create URL patterns for the registration form:

from django.urls import path

from . import views

urlpatterns = [

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

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

]

1. In registration\_project/urls.py, include the users app's URLs:

from django.contrib import admin

from django.urls import path, include

urlpatterns = [

path('admin/', admin.site.urls),

path('users/', include('users.urls')),

]

**Step 6: Create Templates**

1. **Create a folder named templates/users/** in the users app.
2. **Create the register.html template**:

html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>User Registration</title>

<style>

body {

font-family: Arial, sans-serif;

}

.form-container {

max-width: 500px;

margin: auto;

padding: 20px;

background-color: #f9f9f9;

border: 1px solid #ccc;

border-radius: 5px;

}

.form-container input, .form-container textarea {

width: 100%;

padding: 10px;

margin: 10px 0;

border: 1px solid #ccc;

border-radius: 5px;

}

.form-container input[type="submit"] {

background-color: #4CAF50;

color: white;

border: none;

}

</style>

</head>

<body>

<div class="form-container">

<h2>Register</h2>

<form method="POST">

{% csrf\_token %}

{{ form.as\_p }}

<input type="submit" value="Submit">

</form>

</div>

</body>

</html>

1. **Create the success.html template**:

html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Registration Successful</title>

</head>

<body>

<h1>Registration Successful!</h1>

<p>Thank you for registering.</p>

</body>

</html>

**Step 7: Add JavaScript (Optional)**

You can add a simple JavaScript validation script to the register.html template if needed:

html

<script>

document.querySelector('form').addEventListener('submit', function(event) {

const name = document.querySelector('input[name="name"]').value;

const age = document.querySelector('input[name="age"]').value;

if (name === '' || age === '') {

event.preventDefault();

alert('Please fill in all fields.');

}

});

</script>

**Step 8: Run the Project**

1. **Start the Django development server**:

manage.py runserver

1. **Access the registration form**: Open a browser and go to http://127.0.0.1:8000/users/register/.

**Final Folder Structure**

registration\_project/

manage.py

registration\_project/

\_\_init\_\_.py

settings.py

urls.py

wsgi.py

users/

migrations/

templates/

users/

register.html

success.html

\_\_init\_\_.py

admin.py

apps.py

forms.py

models.py

urls.py

views.py

With this setup, users can register with their name, age, address, and mail ID, and the data will be stored in the database.