Activity 6:- Create a Django application that reads data from JSON and displays it on the page

Solution:-

Step 1:- Create a media directory in your app directory and then create a JSON file (data.json) and save it.

Like this:-

```
[
    {
        "name": "Ankita Shukla",
        "age": 24,
        "is_student": false,
        "hobbies": ["reading", "teaching", "creating"],
        "address": {
            "street": "123 Main St",
            "city": "noida",
            "country": "India"
        }
    },
    {
        "name": "Rakhi",
        "age": 25,
        "is_student": true,
        "hobbies": ["teaching", "food_lovers", "traveling"],
        "address": {
            "street": "456 Elm Rd",
            "city": "noida",
            "country": "india"
        }
    }
]
```

Step 2:- now go to your project setting and do some changes in file like:-

- “import os”,
- “MEDIA_URL = '/media/'
- MEDIA_ROOT = os.path.join(BASE_DIR, 'jsonreader/media')”

Here In 3rd point, jsonreader is my app name You write the app name that you created.

Step 3:- Now write some code in your app/views.py file

```
import json
import os
from django.conf import settings
from django.shortcuts import render


def json_data(request):
    json_file_path = os.path.join(settings.MEDIA_ROOT, 'data.json')

    try:
        with open(json_file_path, 'r') as json_file:
            data = json.load(json_file)
    except FileNotFoundError:
        data = []
        print("JSON file not found")

    print(data)

    context = {'data_list': data}
    return render(request, 'json_data.html', context)
```

and save it.

Step 4:- Now to read the data from the file we create an HTML file to read all the data

Just create a templates directory in your app create an HTML file and do some code:-

```
<!DOCTYPE html>
<html>
<head>
    <title>JSON Data</title>
    <style>
        table {
            border-collapse: collapse;
            width: 100%;
        }
        th, td {
```

```
            border: 2px solid black;
            padding: 8px;
            text-align: left;
        }
    </style>
</head>
<body>
    <h1>JSON Data</h1>
    <table>
        <tr>
            <th>Name</th>
            <th>Age</th>
            <th>Is Student</th>
            <th>Hobbies</th>
            <th>Street</th>
            <th>City</th>
            <th>Country</th>
        </tr>
        {% for item in data_list %}
            <tr>
                <td>{{ item.name }}</td>
                <td>{{ item.age }}</td>
                <td>{{ item.is_student }}</td>
                <td>{{ item.hobbies|join:", " }}</td>
                <td>{{ item.address.street }}</td>
                <td>{{ item.address.city }}</td>
                <td>{{ item.address.country }}</td>
            </tr>
        {% endfor %}
    </table>
</body>
</html>
```

Step 5:- Now define the URL for accessing the file with this code:-

1) First import our views methods in the project URL

```
from jsonreader.views import *
```

```
path('json', json_data, name='json_data'),
```

Step 6:- Now run the server and you are able to see the JSON file data:-

JSON Data

Name	Age	Is Student	Hobbies	Street	City	Country
Ankita Shukla	24	False	reading, teaching, creating	123 Main St	noida	India
Rakhi	25	True	teaching, food_lovers, traveling	456 Elm Rd	noida	india

Activity no 8:-

Create a Django application that validates user credentials on the login page

Step 1:- Create a project (project 3) and app (authentication)

Step 2:- Now make some changes in the setting.py file of your app

AUTHENTICATION_BACKENDS = [

'django.contrib.auth.backends.ModelBackend',

]

Step 3:- Now do some code in views.py file

import auth in views.py file as shown in 2nd line of code

```
from django.shortcuts import render, redirect
from django.contrib.auth import authenticate, login
from django.contrib.auth.models import User
from django.contrib.auth import logout


def login_view(request):
    if request.method == 'POST':
        username = request.POST['username']
        password = request.POST['password']
        user = authenticate(request, username=username, password=password)
        if user is not None:
            login(request, user)
            request.session['user_id'] = user.id  # Store user ID in session
            return redirect('home')  # Redirect to a successful login page
        else:
            error_message = "Invalid username or password"
            return render(request, 'login.html', {'error_message': error_message})
    return render(request, 'login.html')


def home(request):
    user_id = request.session.get('user_id')
    if user_id:
        user = User.objects.get(id=user_id)
        return render(request, 'home.html', {'user': user})
    else:
        return redirect('login')


def logout_view(request):
        logout(request)
        return redirect('login')
```

Step 4:- Now we have to create 2 HTML file in the templates directory

1) login.html:-

```
<!DOCTYPE html>
<html>
<head>
    <title>Login</title>
</head>
<body>
    <h1>Login</h1>
    {% if error_message %}
    <p>{{ error_message }}</p>
    {% endif %}
    <form method="post">
        {% csrf_token %}
        <label for="username">Username:</label>
        <input type="text" id="username" name="username"><br><br>
        <label for="password">Password:</label>
        <input type="password" id="password" name="password"><br><br>
        <input type="submit" value="Login">
    </form>
</body>
</html>
```

2) home.html:-

```
<!DOCTYPE html>
<html>
<head>
    <title>Home</title>
</head>
<body>
    <h1>Welcome to the Home Page</h1>
    {% if user %}
    <p>Hello, {{ user.username }}!</p>
    <a href="{% url 'logout' %}">Logout</a>
    {% else %}
    <p>Please log in.</p>
    {% endif %}
</body>
</html>
```

Step 5:- Now for validation, we need to set a username and password using database

So we used the default database & run the make migrations and migrate your project

python manage.py makemigrations

python manage.py migrate

Step 6:- After running migrations, you might want to create a superuser account to access the Django admin interface and test your login system. Run the following command:

python manage.py createsuperuser and set username and password

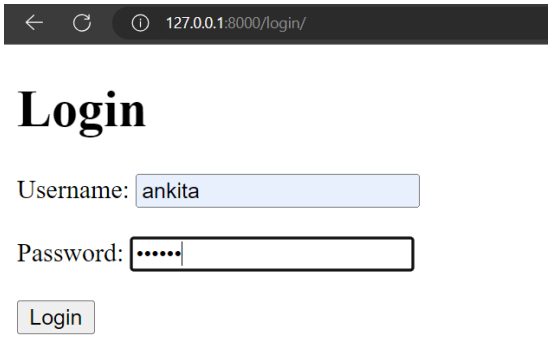
```
(venv) PS C:\Users\Ankita Shukla\PycharmProjects\webproject\project3> python manage.py createsuperuser
Username (leave blank to use 'ankitashukla'): ankita
Email address: sankita1218@gmail.com
Password:
Password (again):
This password is too short. It must contain at least 8 characters.
This password is too common.
This password is entirely numeric.
Bypass password validation and create user anyway? [y/N]: y
Superuser created successfully.
```

Step 7:- after that we need to define the all path in app/url.py file for all define method of our views.py file :-

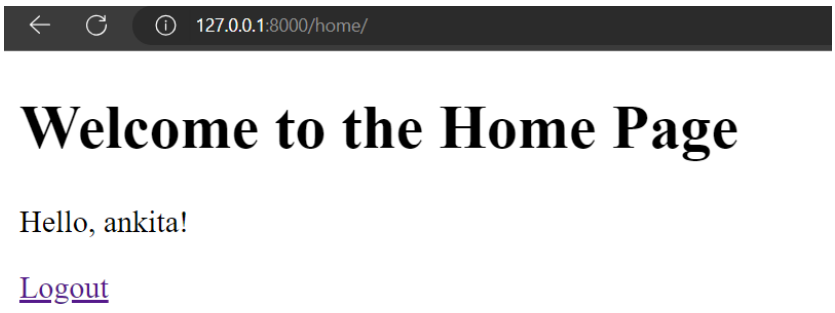
```
from django.contrib import admin
from django.urls import path
from authentication.views import *

urlpatterns = [
    path('login/', login_view, name= "login" ),
    path('home/', home, name="home"),
    path('logout/',logout_view, name='logout'),
    path('admin/', admin.site.urls),
]
```

Step 8:- At last run the server py manage.py runserver

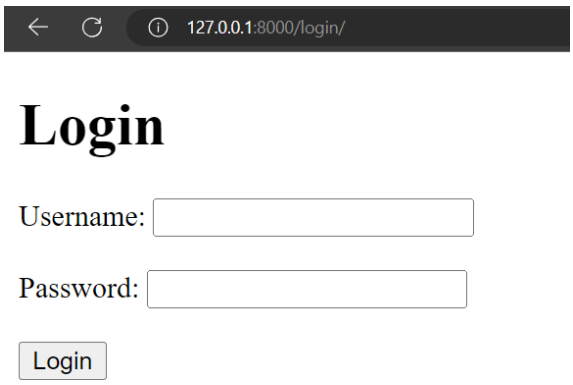
Step 9:- Type the first route /login and able to see the form



Step 10:- fill the data and click on login and you redirect home page



Step 11:- if you clicked on logout button if you again reached on your login page



Completed 😊 😊 😊 😊 ....