

## HAZARA UNIVERSITY, MANSEHRA

**DEPARTMENT OF CS & IT**

**Lab Work**

**Database Systems**

|  |  |
| --- | --- |
| Course Name | Database Systems |
| Student’s Name | **Hassan Ali** |
| Roll Number | **302-2017** |
| Course Code | **CS202** |
| Credit Hours | **4 (3+1)** |
| Class | **BSSE-V Section A** |

INSTALLATION

Required Softwares:

* Pycharm Community Edition
* Python 3.10 or later
* Postgres SQL
* Web Browser (Chrome preferred)

Framework:

* Django

Libraries :

* Psycopg2
* Pillow
* Virtualenv

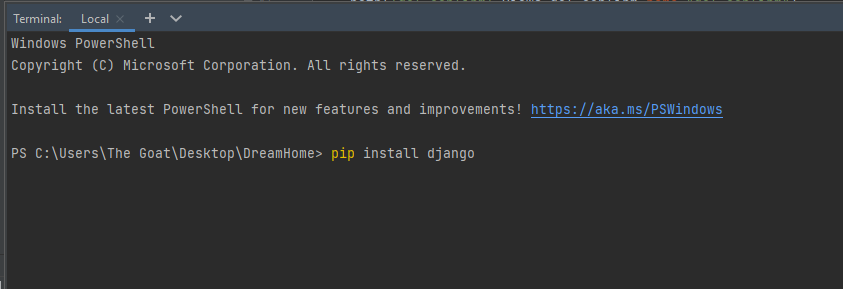
Notes:

We can install Django and other mentioned libraries from terminal of pycharm using following commands:

pip install Django

pip install psycopg2

pip install pillow



**Creating a Django Project**

After installation , we can create a Django project using command prompt CMD.

Go to cmd , navigate to your desired directory.

Type the following command

django-admin startproject name\_of\_your\_project

~~~~

As we know there are several apps inside a project , so let’s create an app in our project.

Navigate to your project folder using cd command in CMD.

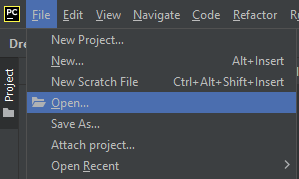
And type the following command

django-admin startapp mytestapp

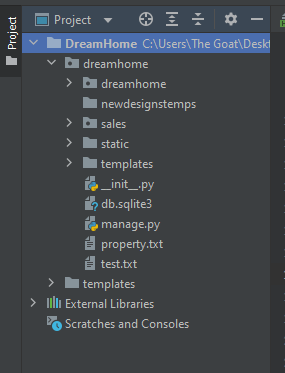


**Django Project Configuration**

Open your project folder in Pycharm through open.



After opening project you will see the project files on left side of pycharm.



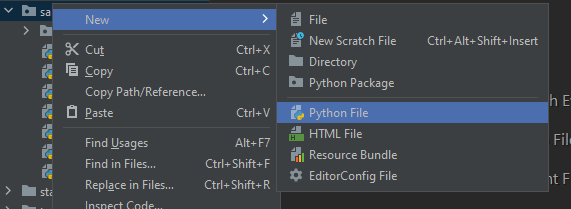
**DREAMHOME PROPERTY PROJECT**

**NOTE:**

**Whole project folder is uploaded to Google Drive , you can download it from the following link:**[**https://drive.google.com/drive/folders/1vYjBpH8ZMlXrwk-nz9CRqSXOFT\_JCh0m?usp=sharing**](https://drive.google.com/drive/folders/1vYjBpH8ZMlXrwk-nz9CRqSXOFT_JCh0m?usp=sharing)

In this project we will create a project named as dreamhome , created an app named as sales.

Now inside sales directory we will create a python file names as urls.py for url routing.



Now open the urls.py file

Code the following:

from django.urls import path  
from . import views  
  
urlpatterns = [  
  
 path('',views.login,name="login"),  
 path('index',views.index,name="index"),  
 path('add\_prop',views.add\_prop,name="add\_prop"),  
 path('home\_search',views.home\_search,name="home\_search"),  
 path('home\_delete',views.home\_delete,name="home\_delete"),  
 path('home\_add',views.home\_add,name="home\_add"),  
 path('add\_new',views.add\_new,name="add\_new"),  
 path('search\_property',views.search\_property\_for\_rent,name="search\_property"),  
 path('search\_result',views.search\_result,name="search\_result"),  
 path('del\_rec',views.del\_record,name="del\_rec"),  
 path('del\_confirm',views.del\_confirm,name="del\_confirm"),  
 path('login',views.login,name="login"),  
 path('register',views.register,name="register"),  
 path('logout',views.logout,name="logout"),  
 path('search\_owner\_prop',views.search\_owner\_prop,name="search\_owner\_prop"),  
 path('owner\_with\_prop',views.search\_owner\_prop,name="owner\_with\_prop"),  
]

Go to urls.py file of main project folder i,e dreamhome.

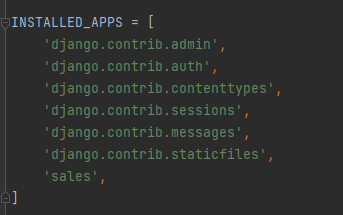
Replace code with :

from django.contrib import admin  
from django.urls import path,include  
urlpatterns = [  
 path('admin/', admin.site.urls),  
 path('',include('sales.urls')),  
]

Now go to settings.py file of dreamhome folder.

Navigate to installed apps

And add your app at last of list like I added sales in last:

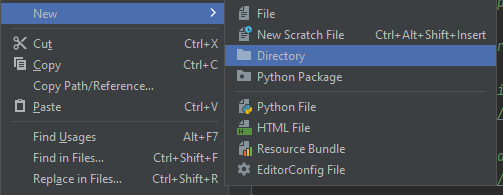


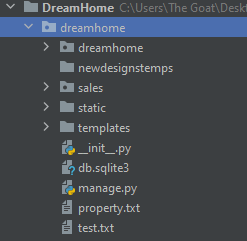
Now navigate templates in settings.py file and add link to your templates :

In DIRS list add BASE\_DIR and templates as I added below:

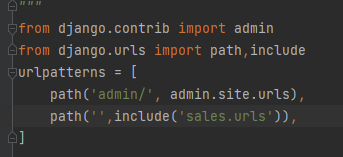


Now create a new directory in main folder named as templates:



As you can see templates folder here.

Now go to urls.py folder of main project



Import include and add a path link of your app’s urls.py file. I,e sales.urls

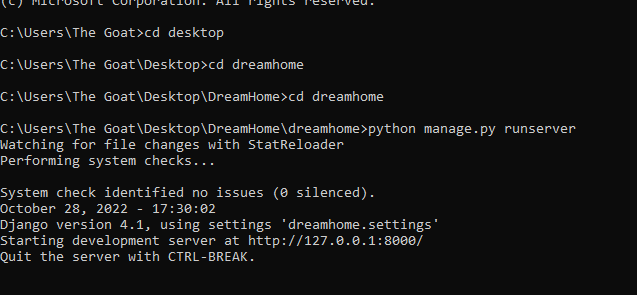
from django.contrib import admin  
from django.urls import path,include  
urlpatterns = [  
 path('admin/', admin.site.urls),  
 path('',include('sales.urls')),  
]

Now run the server through cmd :

Go to your main project directory using cd command where manage,py file is present.

Type python manage.py runserver

And server will run.cd d



Now copy the url of server which is 127.0.0.1 and paste it in your web browser.

Following are the working code of my project with file name:

Sales/Urls.py

from django.urls import path  
from . import views  
  
urlpatterns = [  
  
 path('',views.login,name="login"),  
 path('index',views.index,name="index"),  
 path('add\_prop',views.add\_prop,name="add\_prop"),  
 path('home\_search',views.home\_search,name="home\_search"),  
 path('home\_delete',views.home\_delete,name="home\_delete"),  
 path('home\_add',views.home\_add,name="home\_add"),  
 path('add\_new',views.add\_new,name="add\_new"),  
 path('search\_property',views.search\_property\_for\_rent,name="search\_property"),  
 path('search\_result',views.search\_result,name="search\_result"),  
 path('del\_rec',views.del\_record,name="del\_rec"),  
 path('del\_confirm',views.del\_confirm,name="del\_confirm"),  
 path('login',views.login,name="login"),  
 path('register',views.register,name="register"),  
 path('logout',views.logout,name="logout"),  
 path('search\_owner\_prop',views.search\_owner\_prop,name="search\_owner\_prop"),  
 path('owner\_with\_prop',views.search\_owner\_prop,name="owner\_with\_prop"),  
]

Sales/views.py

from django.contrib.auth.models import User  
from django.shortcuts import render  
# from django.templatetags import static  
from django.http import HttpResponse  
from django.shortcuts import redirect  
from django.contrib.auth import authenticate,login  
from django.contrib import messages, auth  
# from django.core.files import File  
import psycopg2  
# Create your views here.  
def index(request):  
 return render(request,'index.html')  
  
def add\_prop(request):  
 return render(request,'add\_prop.html')  
 return HttpResponse("Add Prop")  
def home\_delete(request):  
 delete = request.POST.get('delete')  
 if len(delete) != 0:  
 return redirect('/del\_rec')  
def home\_search(request):  
 search = request.POST.get('search')  
 if len(search) != 0:  
 return redirect('/search\_property')  
def home\_add(request):  
 add = request.POST.get('add')  
 if len(add) != 0:  
 return redirect('/add\_prop')  
  
def add\_new(request):  
 p\_no = request.POST.get('p\_no')  
 s\_treet = str(request.POST.get('street'))  
 c\_ity = request.POST.get('city')  
 p\_ostcode = request.POST.get('postcode')  
 t\_ype = request.POST.get('type')  
 r\_ooms = request.POST.get('rooms')  
 r\_ent = request.POST.get('rent')  
 o\_no = request.POST.get('o\_num')  
  
 conn = psycopg2.connect(database="dreamhomeproperty", user="postgres", password="hassan", host="127.0.0.1",port="5432")  
 cur = conn.cursor()  
 cur.execute("""INSERT INTO dreamhome\_schema.property\_for\_rent (property\_n,street,city,postcode,typee,rooms,rent,owner\_number) \  
 VALUES (%s,%s,%s,%s,%s,%s,%s,%s);""",(p\_no,s\_treet,c\_ity,p\_ostcode,t\_ype,r\_ooms,r\_ent,o\_no))  
 conn.commit()  
 conn.close()  
 return HttpResponse("Data Saved into table of database")  
  
def search\_property\_for\_rent(request):  
 return render(request,'search\_property.html')  
def search\_result(request):  
 pr\_num = int(request.POST.get('pr\_no'))  
 conn = psycopg2.connect(database="dreamhomeproperty", user="postgres", password="hassan", host="127.0.0.1",  
 port="5432")  
 cur = conn.cursor()  
 cur.execute(f''' SELECT \* FROM dreamhome\_schema.property\_for\_rent WHERE property\_n={pr\_num}''')  
 rows = cur.fetchall()  
 result = []  
 for t in rows: #list of tuples into single list  
 for x in t:  
 result.append(x)  
 cur.close()  
 if len(result) == 0:  
 return HttpResponse("<h1>NO RECORD FOUND </h1>")  
 else:  
 p\_n = result[0]  
 street = result[1]  
 city = result[2]  
 postcode = result[3]  
 typee = result[4]  
 rooms = result[5]  
 rent = result[6]  
 o\_n = result[7]  
 return render(request, 'searched.html', {  
 'p\_n': p\_n,  
 'street': street,  
 'city': city,  
 'postcode': postcode,  
 'type': typee,  
 'rooms': rooms,  
 'rent': rent,  
 'o\_n': o\_n  
 })  
  
def del\_record(request):  
  
 return render(request,'del.html')  
  
def del\_confirm(request):  
 pr\_num = int(request.POST.get('pr\_no'))  
 conn = psycopg2.connect(database="dreamhomeproperty", user="postgres", password="hassan", host="127.0.0.1",  
 port="5432")  
 cur = conn.cursor()  
 cur.execute(f''' DELETE FROM dreamhome\_schema.property\_for\_rent WHERE property\_n={pr\_num}''')  
 conn.commit()  
 cur.close()  
 return render(request,'del\_confirm.html')  
def login(request):  
 if request.method == 'POST':  
 u\_name = str(request.POST.get('username'))  
 pas = str(request.POST.get('password'))  
 user = authenticate(username=u\_name,password=pas)  
 u1 = "admin"  
 u2 = 'user'  
 if user is not None:  
 auth.login(request,user)  
 if u1 in u\_name:  
 return redirect('/index')  
 elif u2 in u\_name:  
 return redirect('/search\_owner\_prop')  
  
 else:  
 messages.info(request, 'invalid username or password')  
 return redirect('login')  
 else:  
 return render(request, 'login.html')  
  
def register(request):  
 if request.method == 'POST':  
 username = request.POST.get('username')  
 e\_mail = request.POST.get('email')  
 password = request.POST.get('password')  
 user = User.objects.create\_user(username=username, password=password, email=e\_mail)  
 user.save()  
 return HttpResponse("User created")  
 else:  
 return render(request, 'register.html')  
def logout(request):  
 auth.logout(request)  
 return redirect('/')  
  
def search\_owner\_prop(request):  
 if request.method == 'POST':  
 o\_no = request.POST.get('owner\_num')  
 conn = psycopg2.connect(database="dreamhomeproperty", user="postgres", password="hassan", host="127.0.0.1",  
 port="5432")  
 cur = conn.cursor()  
 cur.execute(f''' SELECT property\_for\_rent.property\_n,private\_owner.fname,property\_for\_rent.street,property\_for\_rent.rooms,property\_for\_rent.typee FROM dreamhome\_schema.property\_for\_rent   
INNER JOIN dreamhome\_schema.private\_owner  
ON property\_for\_rent.owner\_number={o\_no} AND private\_owner.owner\_no={o\_no}''')  
 rows = cur.fetchall()  
 conn.close()  
 if len(rows) != 0:  
 return render(request, 'owner\_with\_prop.html', {'data': rows})  
 else:  
 return HttpResponse("No Record Found")  
 else:  
 return render(request, 'search\_owner\_prop.html')

dreamhome.urls.py

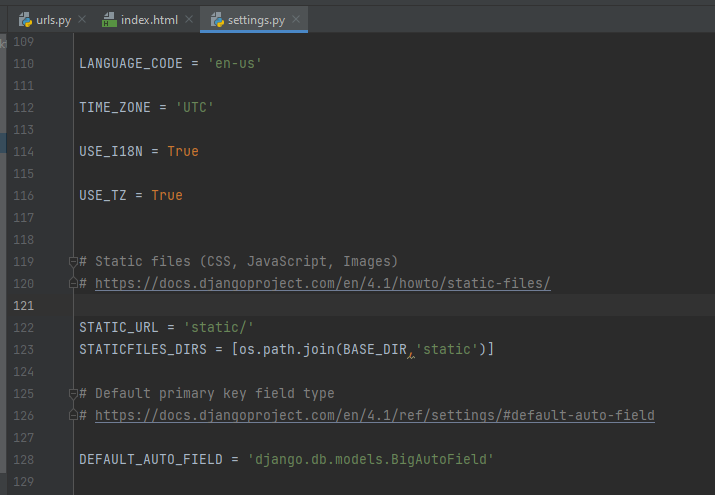
*"""dreamhome URL Configuration  
  
The `urlpatterns` list routes URLs to views. For more information please see:  
 https://docs.djangoproject.com/en/4.1/topics/http/urls/  
Examples:  
Function views  
 1. Add an import: from my\_app import views  
 2. Add a URL to urlpatterns: path('', views.home, name='home')  
Class-based views  
 1. Add an import: from other\_app.views import Home  
 2. Add a URL to urlpatterns: path('', Home.as\_view(), name='home')  
Including another URLconf  
 1. Import the include() function: from django.urls import include, path  
 2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))  
"""*from django.contrib import admin  
from django.urls import path,include  
urlpatterns = [  
 path('admin/', admin.site.urls),  
 path('',include('sales.urls')),  
]

Before going to templates part we have to create a directory in main folder named as static , in this static directory we will add all the files which will be used with our templates like css files javascript files etc.

So Create a new directory named as static :

Then go to settings.py fille and add path of this directory as shown below:

STATICFILES\_DIRS = [os.path.join(BASE\_DIR,'static')]



**TEMPLATES or HTML FILES**

**Login.html**

{% load static %}  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Login</title>  
 <link rel="stylesheet" href="{% static 'hassan.css' %}">  
</head>  
<body>  
<div class="bg-image"> </div>  
<h1 style="color:white;">DREAMHOME LOGIN PAGE</h1>  
<form method="post" action="login">  
 {% csrf\_token %}  
 <h2 style="margin-top:100px;color:white">Username : </h2>  
 <input type="text" name="username" placeholder="Enter username here..." size="30" style="height:30px">  
 <br>  
 <h2 style="color:white;" >Password : </h2>  
 <input type="password" name="password" placeholder="Enter password here..." size="30" style="height:30px">  
 <br>  
 <br>  
 <input type="submit" value="Login" style="height:30px; width:200px">  
  
  
</form>  
  
</body>  
</html>

**Index.html**

{%load static%}  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Sales</title>  
 <link rel="stylesheet" href="{% static 'hassan.css' %}">  
</head>  
<body>  
<h1>Dream Home Property Dealing </h1>  
<h1>ADMIN PAGE</h1>  
<h1>Home Page</h1>  
{% if user.is\_authenticated %}  
<form method="post" action="home\_add">  
 {% csrf\_token %}  
 <h1></h1>  
 <input type="submit" name="add" value="ADD NEW PROPERTY" style="height:30px; width:200px">  
</form>  
<form method="post" action="home\_search">  
 {% csrf\_token %}  
 <br>  
 <input type="submit" name="search" value="SEARCH FOR PROPERTY" style="height:30px; width:200px">  
</form>  
<form method="post" action="home\_delete">  
 {% csrf\_token %}  
 <br>  
 <input type="submit" name="delete" value="DELETE A PROPERTY" style="height:30px; width:200px">  
</form>  
<br><br><br>  
<a href="logout"><input type="button" value="logout" style="height:30px; width:200px"></a>  
{% else %}  
<h2>Please Login First ... </h2>  
<a href="/login"> <input type="button" value="login" style="height:30px; width:200px;"></a>  
{% endif %}  
</body>  
</html>

**add\_prop.html**

{%load static%}  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Sales</title>  
 <link rel="stylesheet" href="{% static 'hassan.css' %}">  
</head>  
<body>  
<h1>Dream Home Property Dealing </h1>  
<form method="post" action="add\_new">  
 {% csrf\_token %}  
 {% if user.is\_authenticated %}  
 <h2 style="coloe:black;">Property for Rent <br> Add New Property </h2>  
 <h3>Enter Property Number :</h3>  
 <input type="text" name="p\_no" size="30" style="height:30px">  
 <h3>Enter Street : </h3>  
 <input type="text" name="street" size="30" style="height:30px">  
 <h3>Enter City Name :</h3>  
 <input type="text" name="city" size="30" style="height:30px">  
 <h3>Enter postcode :</h3>  
 <input type="text" name="postcode" size="30" style="height:30px">  
 <h3>Enter type :</h3>  
 <input type="text" name="type" size="30" style="height:30px">  
 <h3>Enter no of rooms :</h3>  
 <input type="text" name="rooms" size="30" style="height:30px">  
 <h3>Enter rent :</h3>  
 <input type="text" name="rent" size="30" style="height:30px">  
 <h3>Enter owner number :</h3>  
 <input type="text" name="o\_num" size="30" style="height:30px">  
 <br>  
 <input type="submit" value="ADD PROPERTY" style="height:30px; width:200px">  
 <br><br>  
 <a href="logout"> <input type="button" value="Logout...." style="height:30px; width:200px"></a>  
 </form>  
 {% else %}  
 <h1>Please Login First .....</h1>  
 <a href="login"><input type="button" value="login" style="height:30px; width:200px"></a>  
{% endif %}  
</body>  
</html>

**Search\_property.html**

{% load static %}  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>SEARCH PROPERTY</title>  
 <link rel="stylesheet" href="{% static 'hassan.css' %}">  
</head>  
<body>  
<h1>DREAM HOME PROPERTY DEALERS</h1>  
<h2 style="color:white;">SEARCH A PROPERTY</h2>  
<form method="post" action="search\_result">  
 {% csrf\_token %}  
 {% if user.is\_authenticated %}  
 <h2 style="color:white;">Please provide property number : </h2>  
 <input type="text" name="pr\_no" size="30" style="height:30px">  
 <input type="submit" value="Find" style="height:30px; width:100px">  
 <br>  
 <br>  
 <a href="logout"><input type="button" value="Logout" style="height:30px; width:200px"></a>  
 {% else %}  
 <h1>Please Login First...</h1>  
 <a href="login"><input type="button" value="Login"></a>  
 {% endif %}  
 <h3>THIS PROJECT IS DESIGNED BY HASSAN ALI<br> 302-2017 <br> BSSE 5th A</h3>  
</form>  
</body>  
</html>

**del.html**

{% load static %}  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>SEARCH PROPERTY</title>  
 <link rel="stylesheet" href="{% static 'hassan.css' %}">  
</head>  
<body>  
<h1>DREAM HOME PROPERTY DEALERS</h1>  
<h2 style="color:white;">DELETE A PROPERTY</h2>  
<form method="post" action="del\_confirm">  
 {% csrf\_token %}  
 {%if user.is\_authenticated %}  
 <h2 style="color:white;">Please provide property number you want to delete : </h2>  
 <input type="text" name="pr\_no" size="30" style="height:30px">  
 <input type="submit" value="Delete" style="height:30px; width:100px">  
 <h3 style="color:white">THIS PROJECT IS DESIGNED BY HASSAN ALI<br> 302-2017 <br> BSSE 5th A</h3>  
 <a href="logout"> <input type="button" value="Logout...." style="height:30px; width:100px"></a>  
 {%else%}  
 <h1>Please Login First .....</h1>  
 <a href="login"><input type="button" value="login"></a>  
{% endif %}  
</form>  
</body>  
</html>

**del\_confirm.html**

{% load static %}  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Del Confirm</title>  
 <link rel="stylesheet" href="{% static 'style3.css '%}">  
</head>  
<body>  
<h1>Record has been Deleted Successfully!</h1>  
</body>  
</html>

**search\_owner\_prop.html**

{% load static %}  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Search Owner's Properties</title>  
 <link rel="stylesheet" href="{% static 'style3.css'%}">  
</head>  
<body>  
{% if user.is\_authenticated %}  
<h1>Welcome to User Portal.</h1>  
<h2> You can search properties of an owner here:</h2>  
<h2>Please Enter Owner Number</h2>  
<form method="post" action="search\_owner\_prop">  
 {% csrf\_token %}  
 <input type="text" name="owner\_num">  
 <br>  
 <br>  
 <input type="submit" value="Search">  
 <br>  
 <br>  
 <br>  
 <a href="logout">  
 <input type="button" value="Logout"></a>  
</form>  
{% else %}  
<h2>You are not allowed to access this page.<br> Please Login first ... </h2>  
<a href="/login"> <input type="submit" value="Login"></a>  
{% endif %}  
</body>  
</html>

**register.html**

{% load static %}  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Register</title>  
 <link rel="stylesheet" href="{% static 'style3.css' %}">  
</head>  
<body>  
<form method="post" action="register">  
 {% csrf\_token %}  
 <h1>USER REGISTRATION FORM</h1>  
 <h2>Select a username : </h2>  
 <input type="text" name="username">  
 <h2>Please enter your email address : </h2>  
 <input type="email" name="email">  
 <h2>Create a password : </h2>  
 <input type="password" name="password">  
 <br>  
 <input type="submit" value="Register">  
 <h1>OR Login</h1>  
 <br>  
 <a href="login">  
 <input type="button" value="Login"></a>  
  
</form>  
  
</body>  
</html>

**searched.html**

{% load static %}  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Result</title>  
 <link rel="stylesheet" href="{% static 'style.css' %}">  
</head>  
<body>  
<h1>RECORD FOUND</h1>  
  
<table>  
 <tr>  
 <th>Property Number</th>  
 <th>Street</th>  
 <th>City</th>  
 <th>Post Code</th>  
 <th>Type</th>  
 <th>Rooms</th>  
 <th>Rent</th>  
 <th>Owner Number</th>  
  
 </tr>  
 <tr>  
 <td>{{p\_n}}</td>  
 <td>{{street}}</td>  
 <td>{{city}}</td>  
 <td>{{postcode}}</td>  
 <td>{{type}}</td>  
 <td>{{rooms}}</td>  
 <td>{{rent}}</td>  
 <td>{{o\_n}}</td>  
 </tr>  
</table>  
  
  
  
</body>  
</html>

**owner\_with\_prop.html**

{% load static %}  
  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Owner With Properties</title>  
 <link rel="stylesheet" href="{% static 'style5.css '%}">  
</head>  
<body>  
<h2>Record Found {{data.0.1}} has the following properties.</h2>  
<h2>Property Number:&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;Street :  
 &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp; Rooms :&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;  
 &nbsp;&nbsp;&nbsp;&nbsp;Type :&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;  
</h2>  
  
{% for row in data %}  
<h3 style="text-indent:-10px;">{{row.0}}  
 &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;  
 &nbsp;&nbsp;&nbsp;{{row.2}}  
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;  
 &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;{{row.3}}&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;  
 &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;{{row.4}}</h3>  
{% endfor %}  
<a href="search\_owner\_prop"><input type="button" value="Search Again"></a>  
  
</body>  
</html>

**search\_owner\_prop.html**

{% load static %}  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Search Owner's Properties</title>  
 <link rel="stylesheet" href="{% static 'style3.css'%}">  
</head>  
<body>  
{% if user.is\_authenticated %}  
<h1>Welcome to User Portal.</h1>  
<h2> You can search properties of an owner here:</h2>  
<h2>Please Enter Owner Number</h2>  
<form method="post" action="search\_owner\_prop">  
 {% csrf\_token %}  
 <input type="text" name="owner\_num">  
 <br>  
 <br>  
 <input type="submit" value="Search">  
 <br>  
 <br>  
 <br>  
 <a href="logout">  
 <input type="button" value="Logout"></a>  
</form>  
{% else %}  
<h2>You are not allowed to access this page.<br> Please Login first ... </h2>  
<a href="/login"> <input type="submit" value="Login"></a>  
{% endif %}  
</body>  
</html>

**Static/Style0.css**

body{  
text-align:center;  
background-image:url('abc.jpg');  
 background-size: cover; /\* <------ \*/  
 background-repeat: no-repeat;  
 background-position: center center;  
}

**DATABASE CONNECTIVITY**

**I added these lines of codes in settings.py file to connect my database with Django project.**

DATABASES = {  
 'default': {  
 'ENGINE': 'django.db.backends.postgresql',  
 'NAME': 'DreamHome',  
 'USER': 'postgres',  
 'PASSWORD': 'hassan',  
 'HOST': 'localhost',  
 }  
}

**Similarly to execute a query or accessing database from python code we use psycopg2**

conn = psycopg2.connect(database="dreamhomeproperty", user="postgres", password="hassan", host="127.0.0.1",  
 port="5432")  
cur = conn.cursor()  
cur.execute(f''' DELETE FROM dreamhome\_schema.property\_for\_rent WHERE property\_n={pr\_num}''')  
conn.commit()  
cur.close()

**This code make a connection to our database and delete a property by its property number.**

**Property number is fetched from html form using POST method.**

**Change parameters according to your project like database name password and names of attributes etc.**

**CREATING DATABASE**

**I opened my pgadmin and set username and password for it.**

**Then I created a new database named as dreamhome.**

**In this database I created a new schema named as dreamhome\_schema**

**Inside this schema I created my tables.**

**One for property for rent and other for private owner**

**The property number in property table and owner number in owner tables are primary keys respectively.**

**But the owner number is acting as a foreign key in property table.**

**Client Server Architecture**

**There are two conditions for creating client server environment.**

**First is that we connect different computers with same internet connection through WIFI , now we can add IP of server pc in allowed hosts of my settings.py file as:**

ALLOWED\_HOSTS = ["127.0.0.1" , "192.168.1.1" , "localhost"]

**Here 192.168.1.1 is ip of my server.**

**Now go to cmd and runserver using :**

**Python manage.py runserver 0.0.0.0:8000**

**Now any other computer connected wirelessly through this server computer can access and manipulate database using the ip of server followed by port which is 8000 in this case and separated by a colon.**

**Like client will go to his/her browser and type (in my case , your ip will be different)**

**192.168.1.1:8000**

**How to check Ip of a computer:**

**Go to cmd and type**

**ipconfig**

**this will give you a list of addresses , you can take your ip from ipv4 address column.**

**Second condition is that we connect our server to clients by the use of a ethernet cable. This method’s benefit is that if you are running out of internet , you can create a network using wires.**

**So First you have to connect your clients and server using ethernet cable.**

**Then go to control panel of computer you are wishing to create server.**

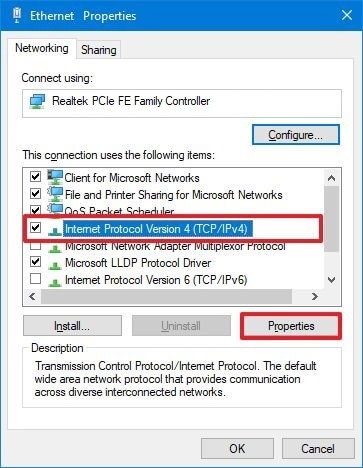
**Then go to network and sharing**

**At left column go to advance option and turn on network discovery and printer file sharing.**

**Go to All networks and turn off password protected sharing.**

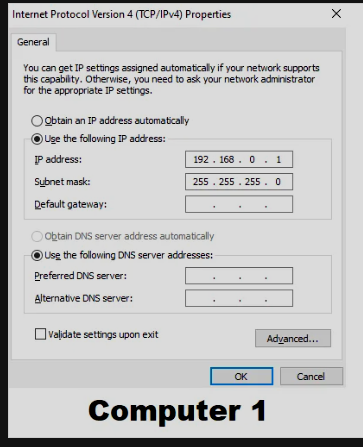
**Now come back to network and sharing and click on ethernet small symbol.**

**Go to**

 **Go to internet protocol Version4 TCP/IPv4**

**Now you have to mark use this ip address radio button.**

**Then**

****

**In default gateway ad 192.168.0.2**

**In Preffered DNS add 8.8.8.8**

**Click Ok**

**Repeat all the steps in your second computer which is going to be client.**

**But this time you have to shuffle the gateway and IP**

**In IP address add the gateway to previous computer which is 192.168.0.1**

**In Gateway add the Ip of previous computer which is 192.168.0.2**

**In Preffered DNS add 8.8.8.8**

**Click Ok**

**Now both computers are connected through ethernet cable.**

**Go to settings.py file in your server pc and add your ip in this case 192.168.0.1 in allowed hosts.**

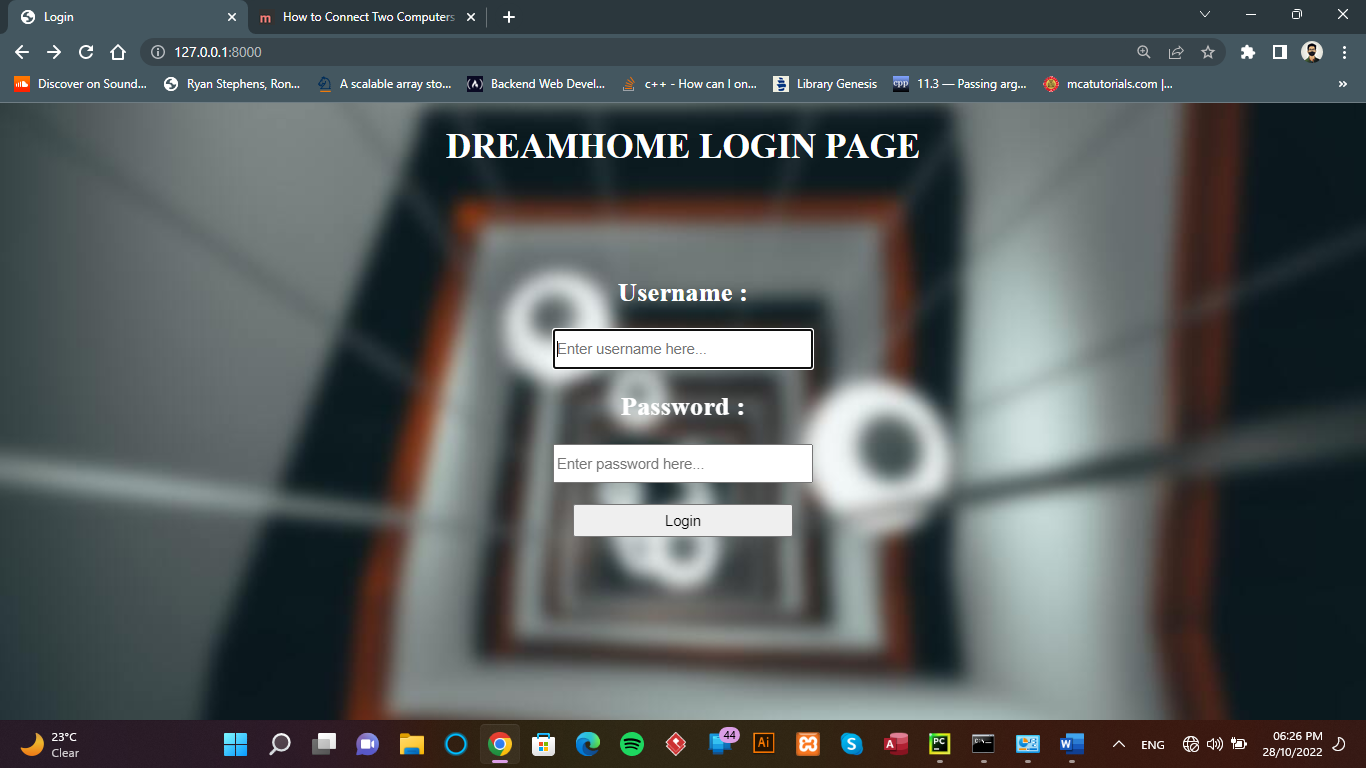
**And run the server using python manage.py runserver 0.0.0.0:8000**

**Now you can access database from other computer by typing 192.168.0.1:8000 in web browser.**

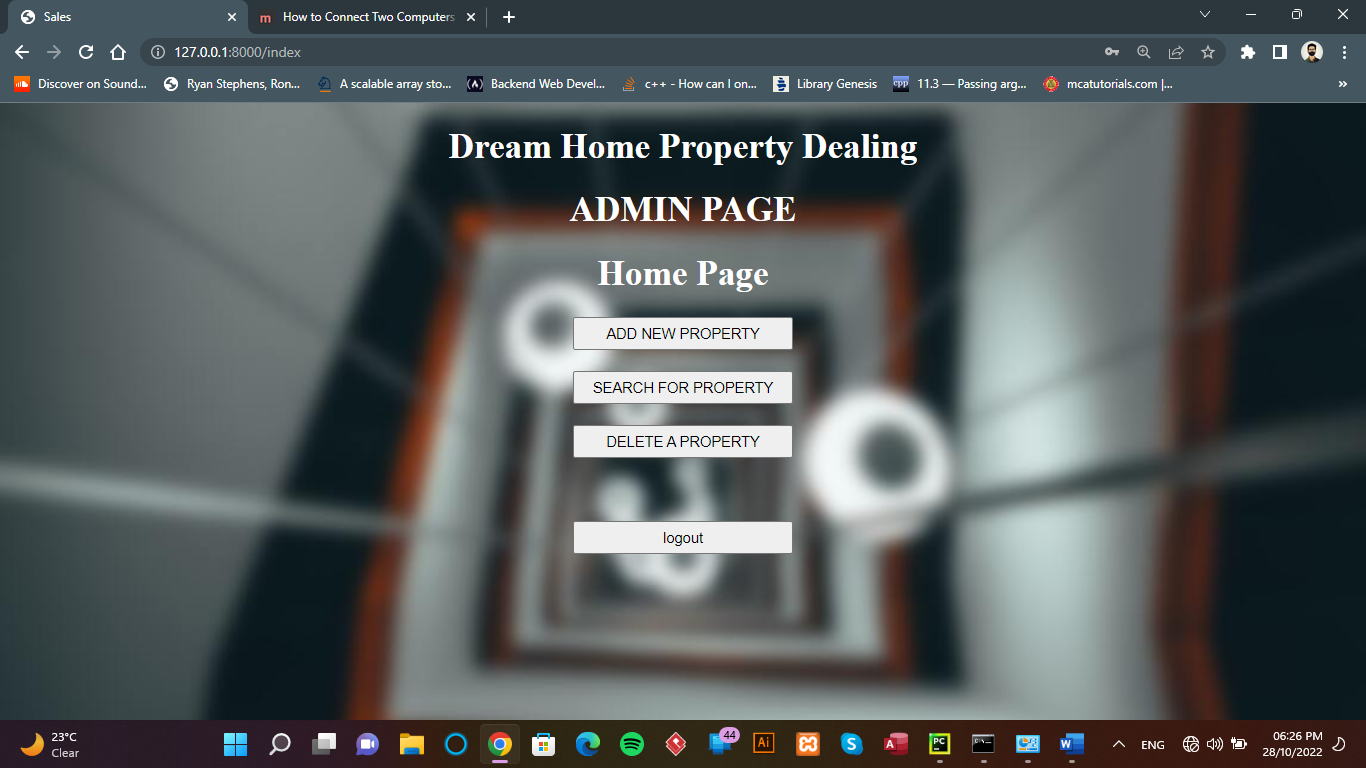
**FINAL RESULT OF PROJECT:  
Dreamhome project has a login page. There are two types of users , clients and admins.**

**Clients are allowed only to search the data while admin can search , add and delete the record from database.**

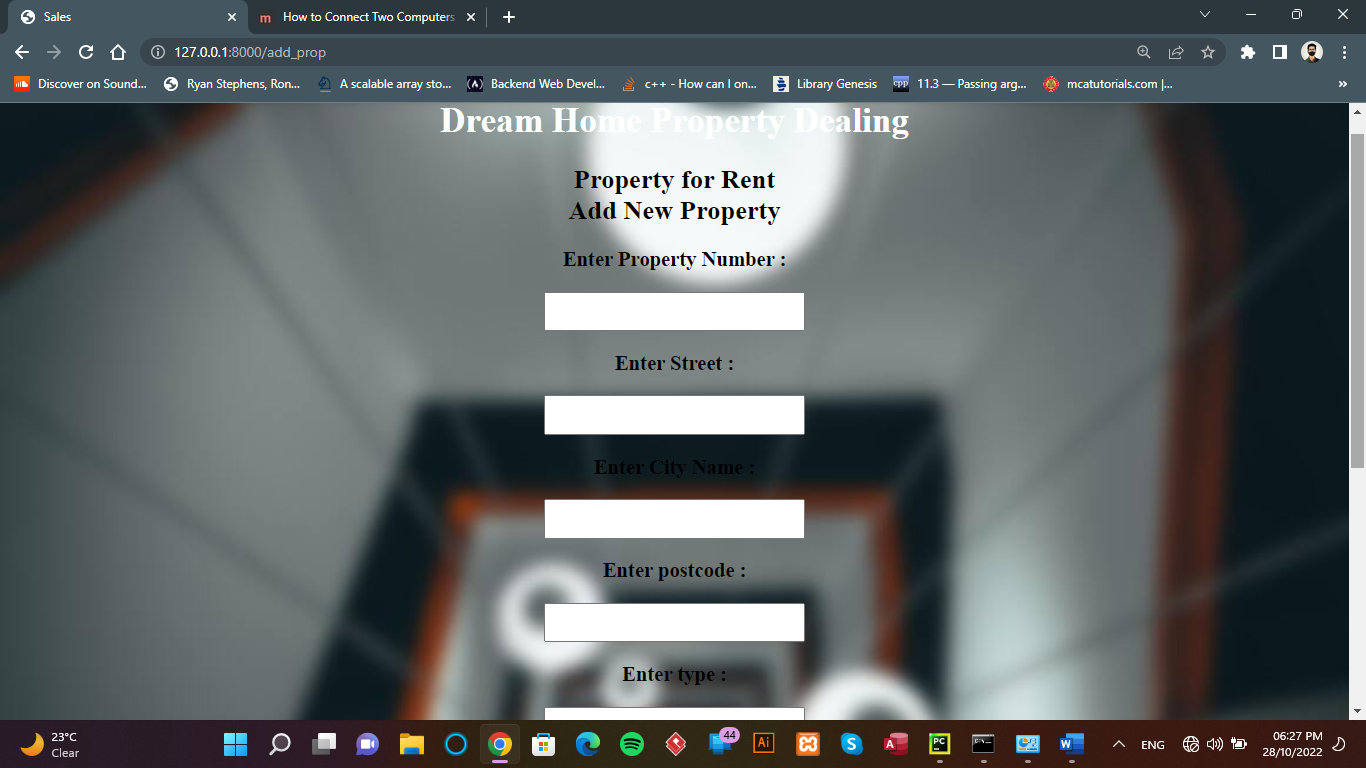
**Login**



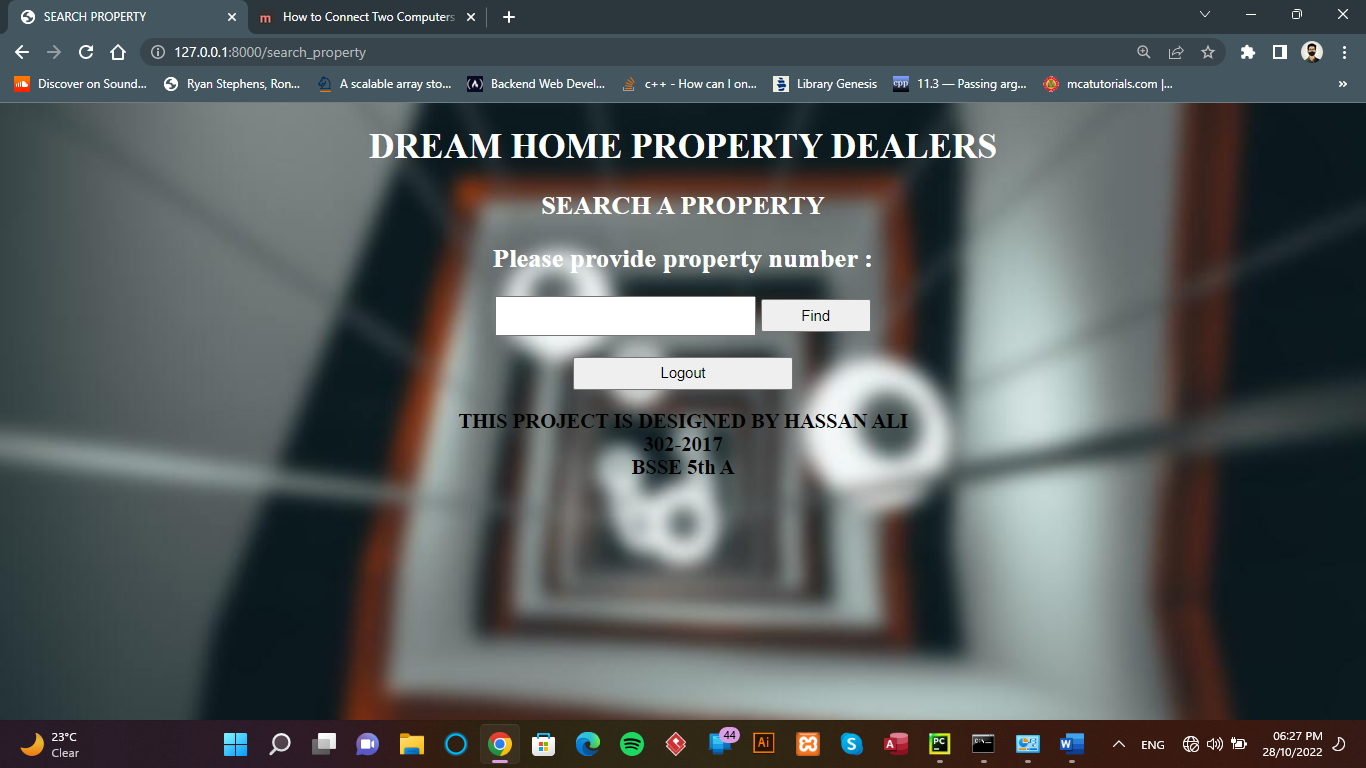
**Admin Portal**



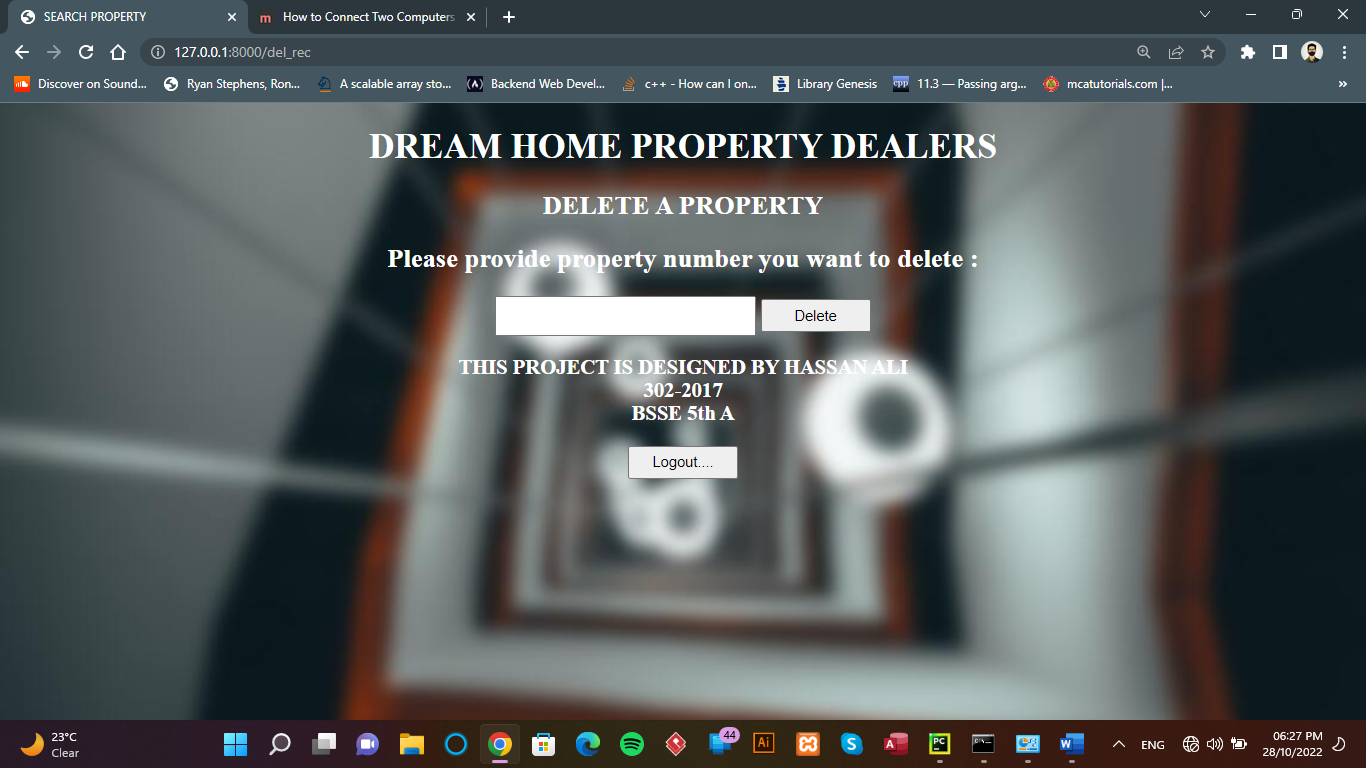
**ADD Property**



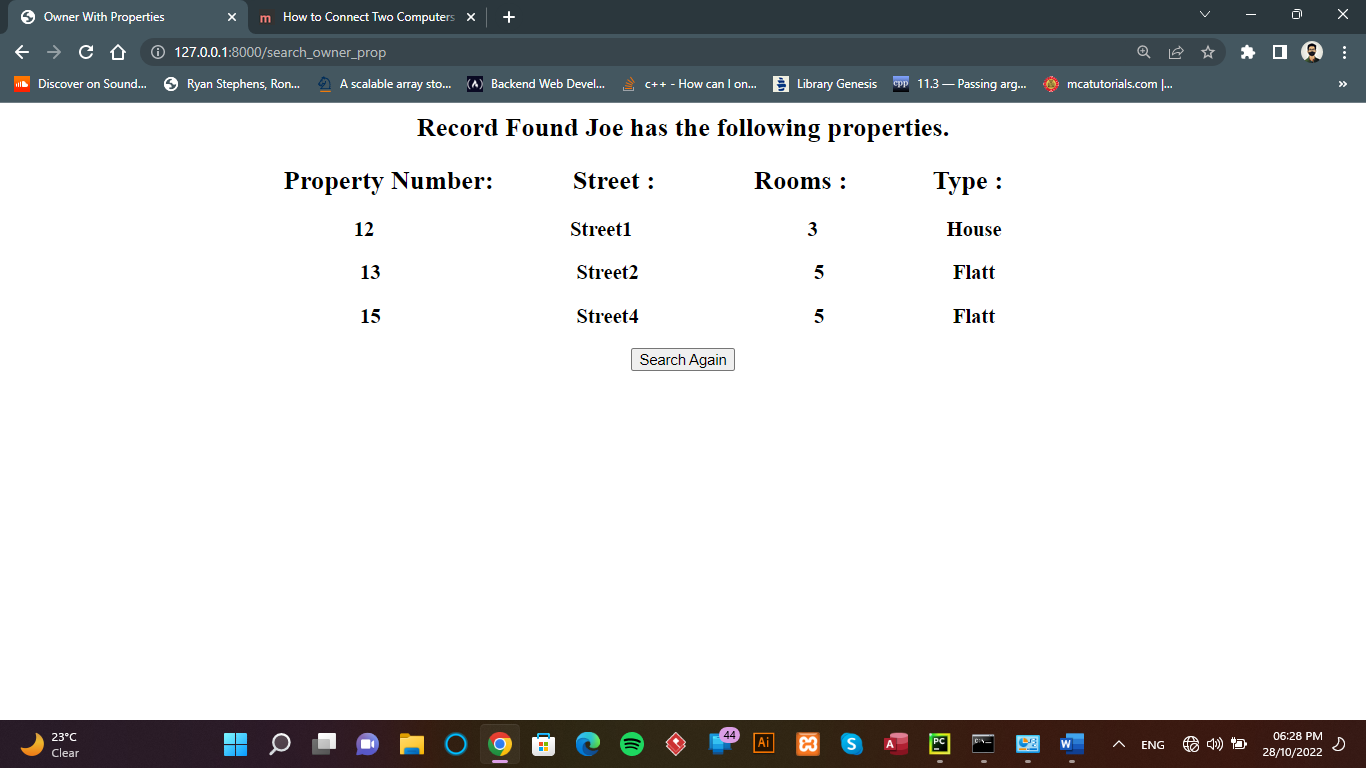
**Search a property**



**Delete Property**



**User Find Result**



**Registering a user**

