CS471 – Web Technologies (Laboratory)		**Lab 12** **Django Contrib Package**

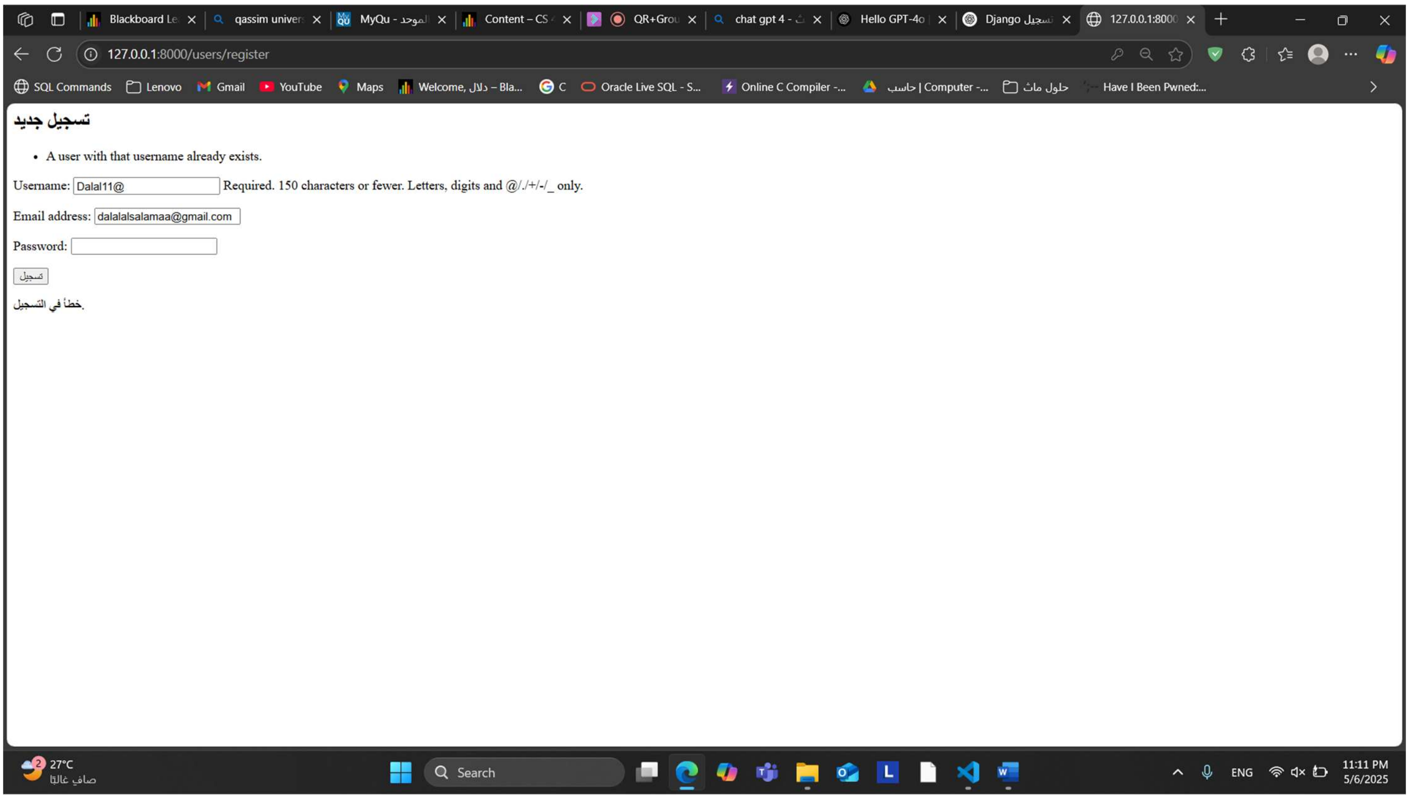
This lab session covers Django Contrib Package (admin, sessions and messages). By the end of this session, students should have a very good understanding of how to use these packages.

Pre-lab Preparation:

1. Pages of the previous lab.
2. Add the necessary contrib packages in settings.py

Lab Activities:

Task 1: Create a register page with a URL '/users/register' with any necessary HTML files and view functions.



تسجيل جديد

- A user with that username already exists.

Username: Dalal11@ Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

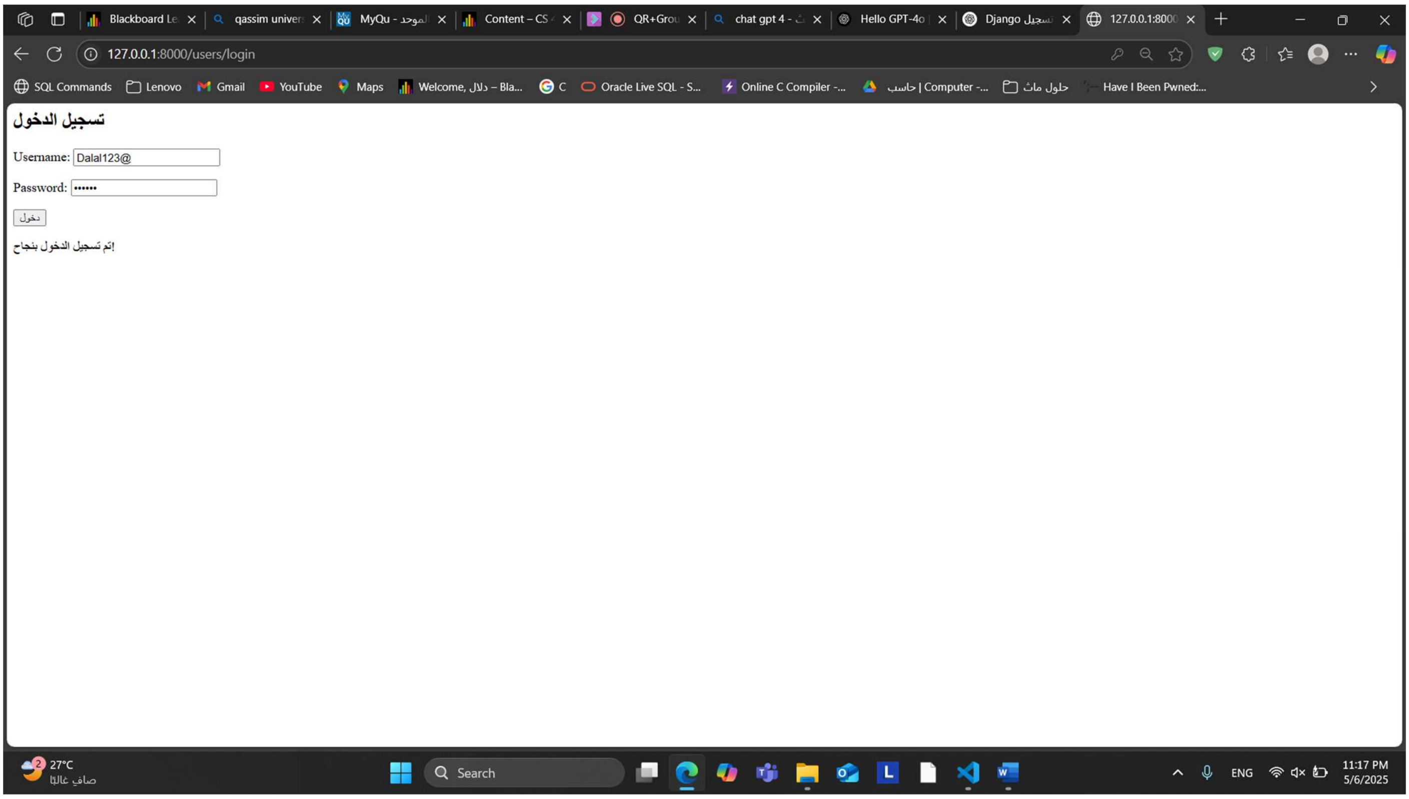
Email address: dalalalsalamaa@gmail.com

Password:

تسجيل

خطأ في التسجيل.

Task 2: Create a login page with a URL '/users/login' with any necessary HTML files and view functions.



127.0.0.1:8000/users/login

تسجيل الدخول

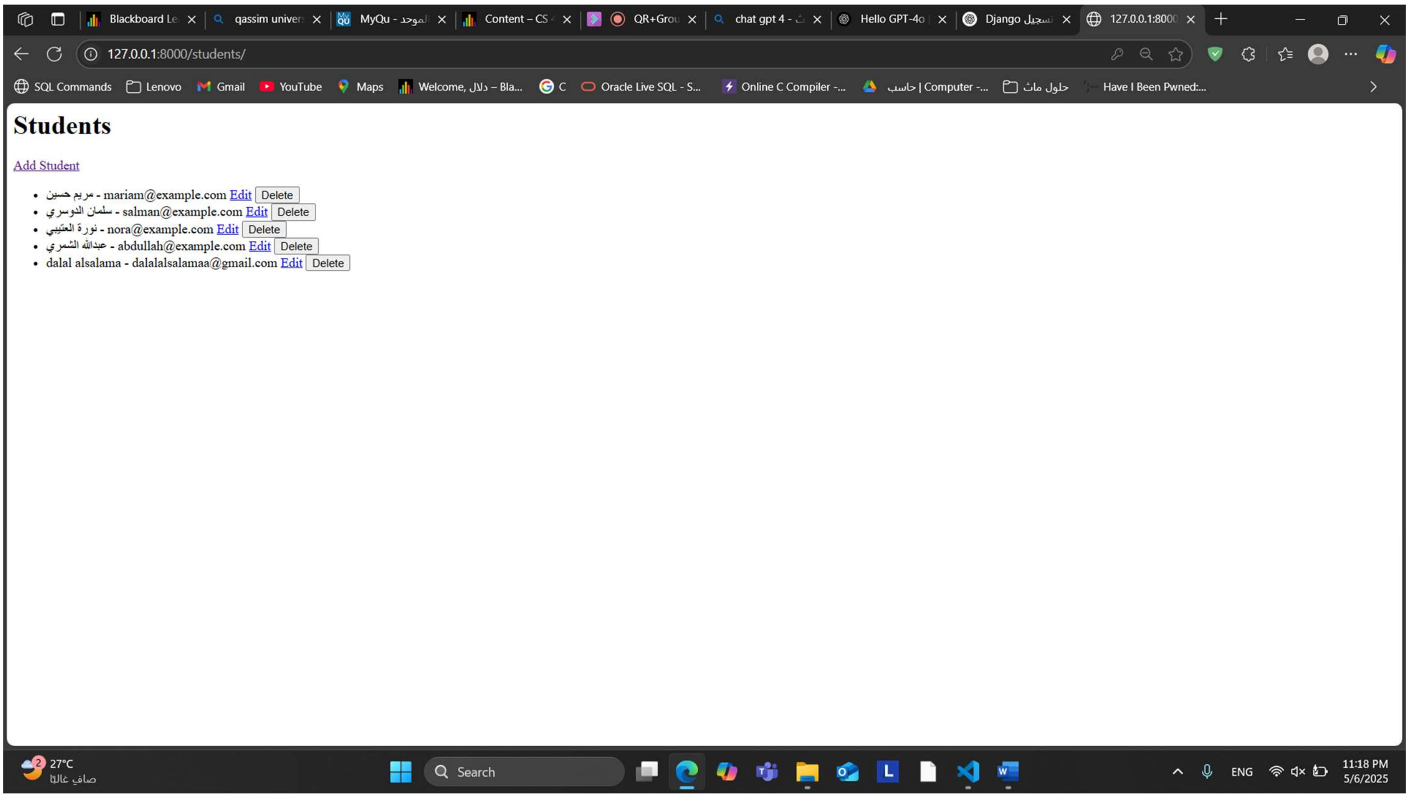
Username: Dalal123@

Password: ••••••

دخول

!إتم تسجيل الدخول بنجاح

Task 3: Apply login permission to the previous lab. If the user is not logged in, redirect the user to the login page.



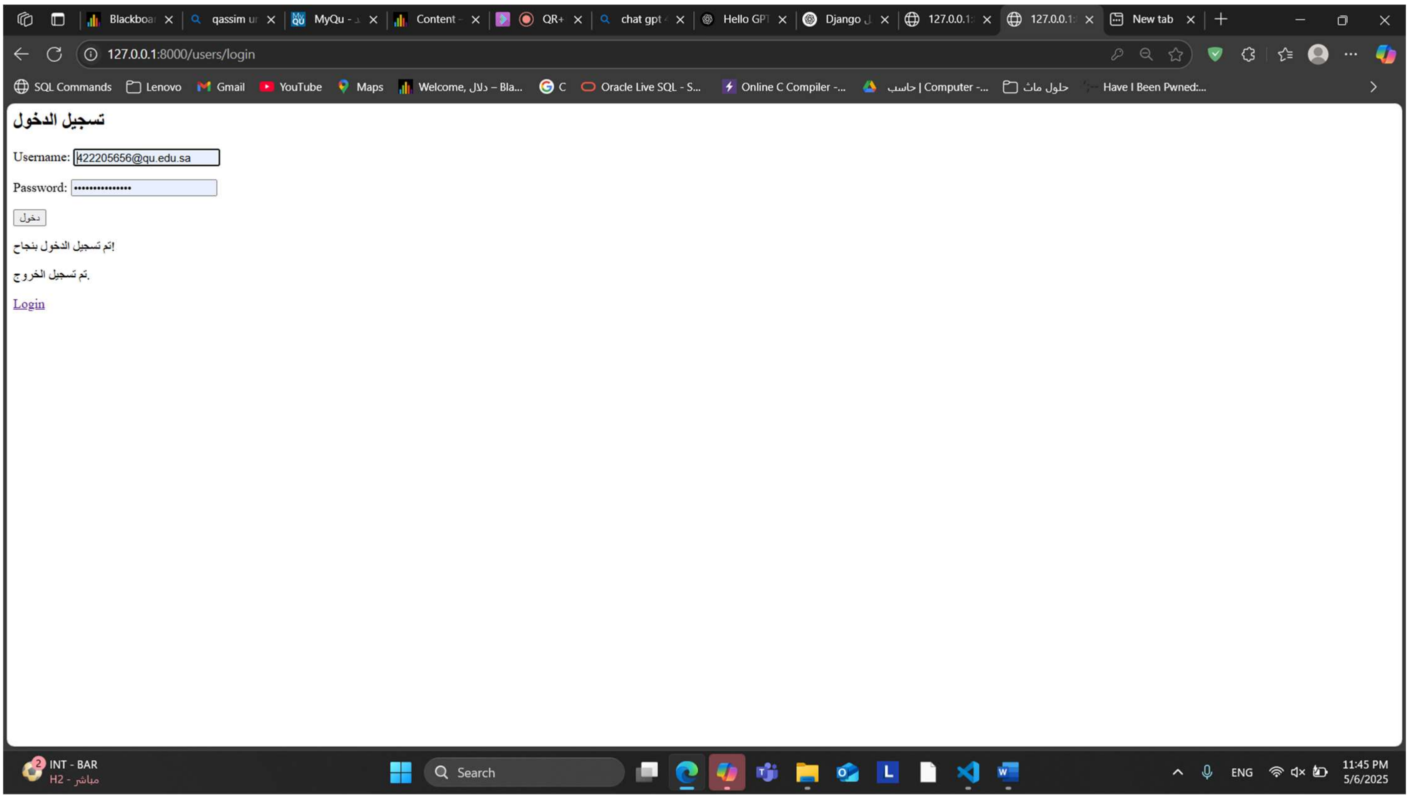
127.0.0.1:8000/students/

Students

Add Student

- مريم حسين - mariam@example.com Edit Delete
- سلمان الدوسري - salman@example.com Edit Delete
- نورة العتيبي - nora@example.com Edit Delete
- عبدالله الشمري - abdullah@example.com Edit Delete
- dalal alsalama - dalalalsalamaa@gmail.com Edit Delete

Task 4: Add a link '/users/logout' to sign out users to all pages where the user is logged in.



127.0.0.1:8000/users/login

تسجيل الدخول

Username: 422205656@qu.edu.sa

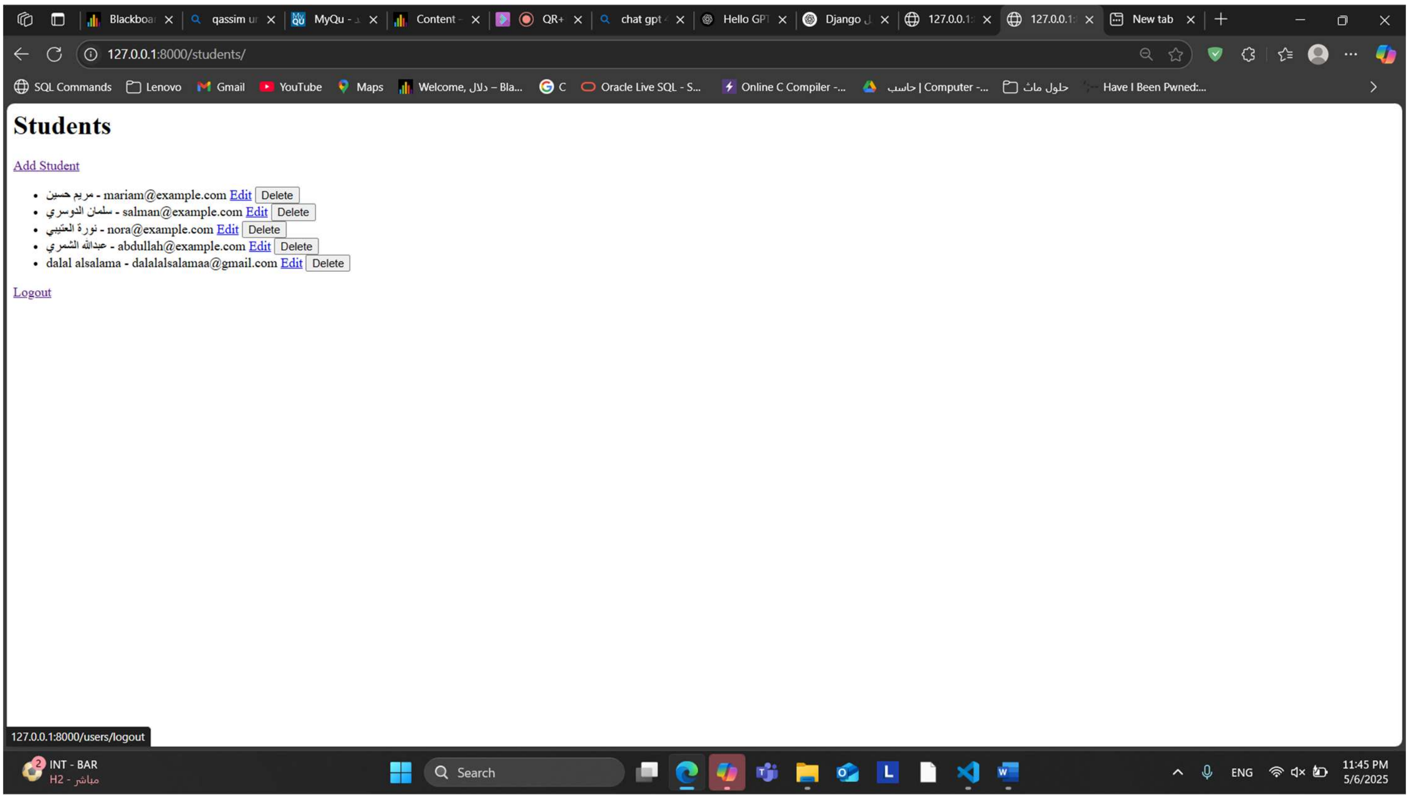
Password: •••••••••••••••

دخول

إتم تسجيل الدخول بنجاح

.تم تسجيل الخروج

Login



127.0.0.1:8000/students/

Students

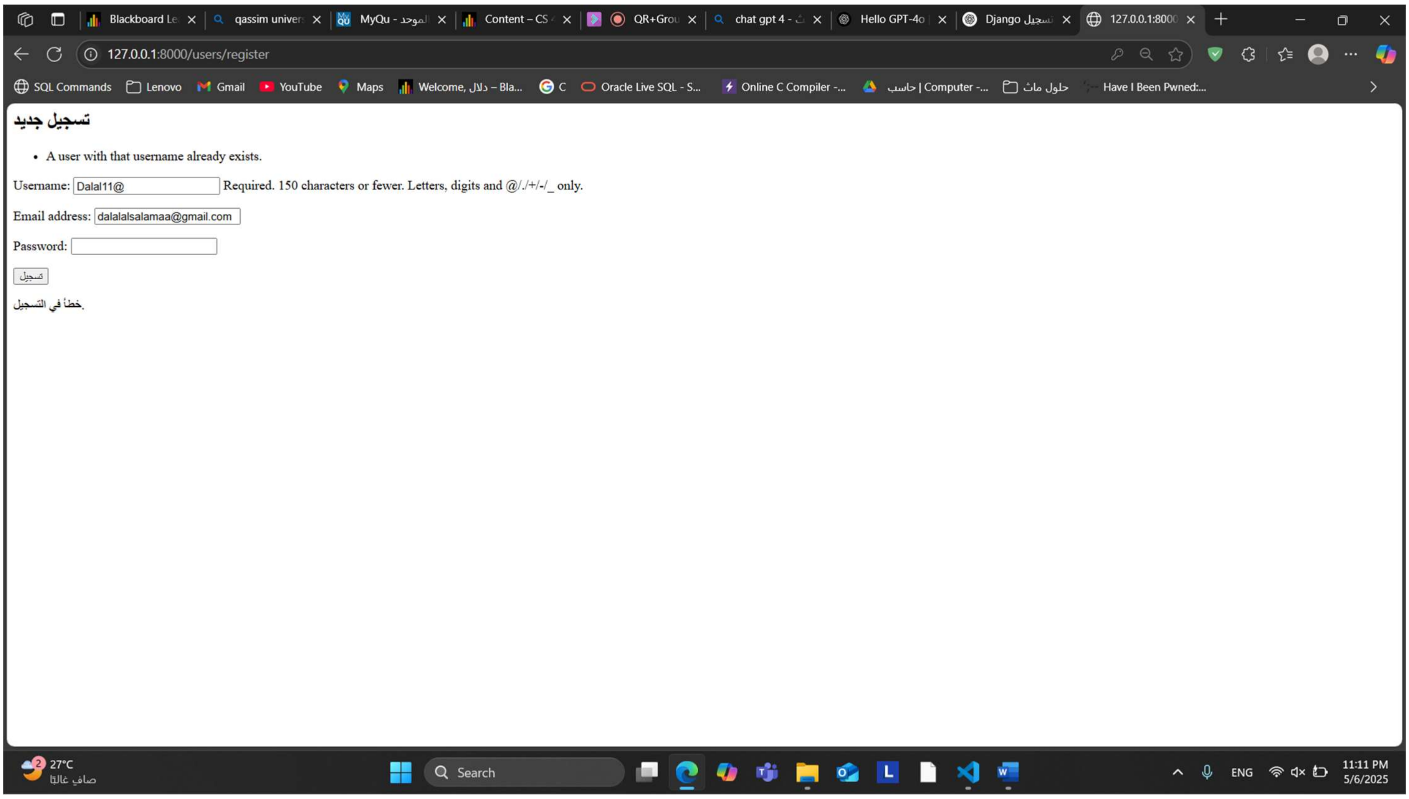
Add Student

- مريم حسين - mariam@example.com Edit Delete
- سلمان الدوسري - salman@example.com Edit Delete
- نورة العتيبي - nora@example.com Edit Delete
- عبدالله الشمري - abdullah@example.com Edit Delete
- dalal alsalama - dalalalsalamaa@gmail.com Edit Delete

Logout

127.0.0.1:8000/users/logout

Task 5: Provide messages to users, like, 'you have successfully registered', 'login successfully', or 'error message.'



تسجيل جديد

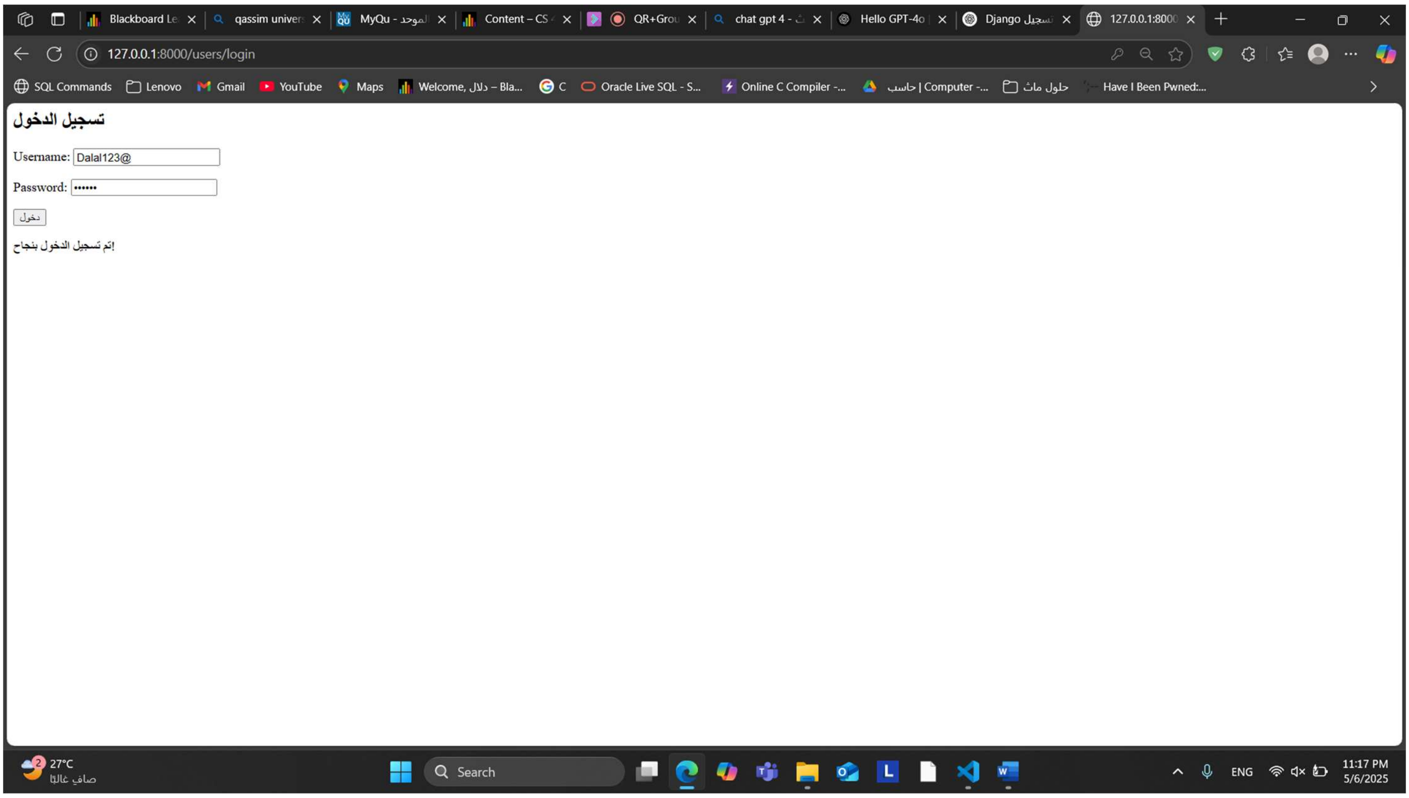
- A user with that username already exists.

Username: Dalal11@ Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Email address: dalalalsalamaa@gmail.com

Password:

خطأ في التسجيل.

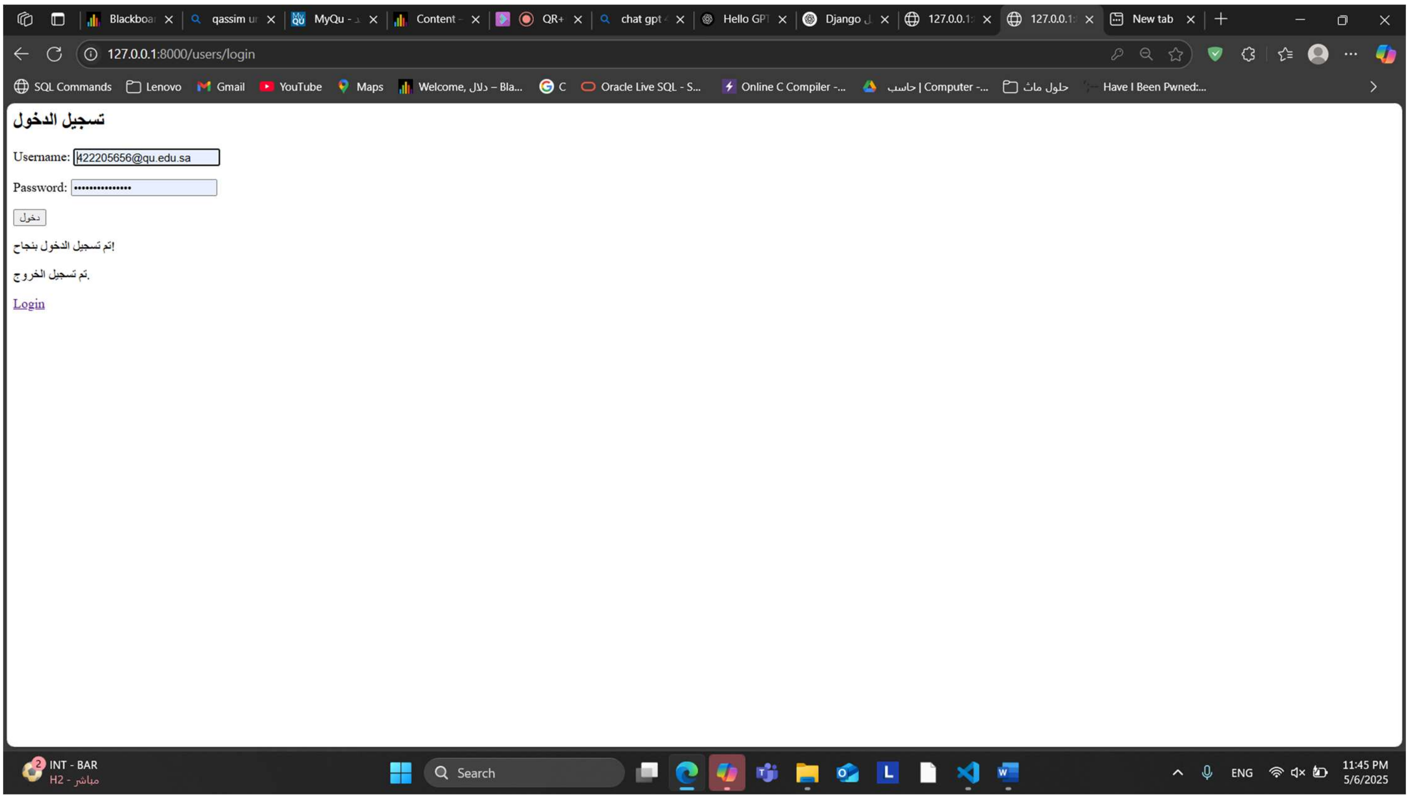


تسجيل الدخول

Username: Dalal123@

Password: ••••••

إتم تسجيل الدخول بنجاح!



تسجيل الدخول

Username: 422205656@qu.edu.sa

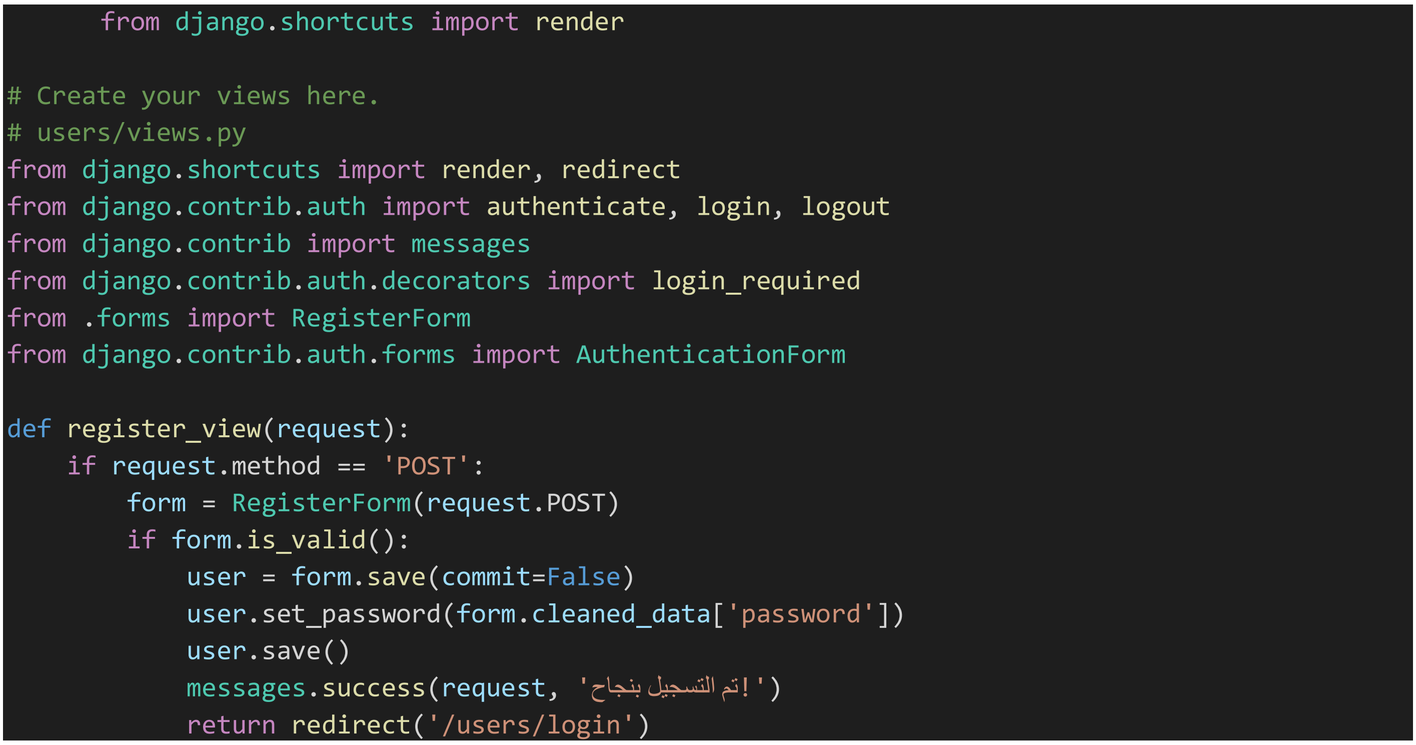
Password: •••••••••••••••

دخول

تم تسجيل الدخول بنجاح!

تم تسجيل الخروج.

Login



```
      from django.shortcuts import render

# Create your views here.
# users/views.py
from django.shortcuts import render, redirect
from django.contrib.auth import authenticate, login, logout
from django.contrib import messages
from django.contrib.auth.decorators import login_required
from .forms import RegisterForm
from django.contrib.auth.forms import AuthenticationForm

def register_view(request):
    if request.method == 'POST':
        form = RegisterForm(request.POST)
        if form.is_valid():
            user = form.save(commit=False)
            user.set_password(form.cleaned_data['password'])
            user.save()
            messages.success(request, 'تم التسجيل بنجاح!')
            return redirect('/users/login')
```

```python
        else:
            messages.error(request, 'خطأ في التسجيل.')
    else:
        form = RegisterForm()
    return render(request, 'users/register.html', {'form': form})

def login_view(request):
    if request.method == 'POST':
        form = AuthenticationForm(request, data=request.POST)
        if form.is_valid():
            user = authenticate(username=form.cleaned_data['username'],
password=form.cleaned_data['password'])
            if user:
                login(request, user)
                messages.success(request, 'تم تسجيل الدخول بنجاح!')
                return redirect('student_list')
            else:
                messages.error(request, 'اسم المستخدم أو كلمة المرور غير صحيحة.')
        else:
            messages.error(request, 'الرجاء التأكد من صحة البيانات.')
    else:
        form = AuthenticationForm()
    return render(request, 'users/login.html', {'form': form})

def logout_view(request):
    logout(request)
    messages.success(request, 'تم تسجيل الخروج.')
    return redirect('/users/login')
```

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

urlpatterns = [
    path('register', views.register_view, name='register'),
    path('login', views.login_view, name='login'),
    path('logout', views.logout_view, name='logout'),

]
```

```html
<h2>تسجيل الدخول</h2>
<form method="post">
  {% csrf_token %}
  {{ form.as_p }}
  <button type="submit">دخول</button>
</form>
{% for msg in messages %}
  <p>{{ msg }}</p>
{% endfor %}
{% if user.is_authenticated %}
    <a href="{% url 'logout' %}">Logout</a>  <!-- رابط تسجيل الخروج -->
{% else %}
    <a href="{% url 'login' %}">Login</a>  <!-- رابط تسجيل الدخول في حال عدم تسجيل الدخول -->
{% endif %}
```

```
<h2>تسجيل جديد</h2>
<form method="post">
  {% csrf_token %}
  {{ form.as_p }}
  <button type="submit">تسجيل</button>
</form>
{% for msg in messages %}
  <p>{{ msg }}</p>
{% endfor %}
```

```
      from django.shortcuts import render, redirect, get_object_or_404
from .models import Student, Address, Student2, Address2, Gallery
from .forms import StudentForm, AddressForm, Student2Form, Address2Form, GalleryForm
from django.contrib.auth.decorators import login_required
from django.contrib.auth.decorators import login_required
@login_required
def student_list(request):
    students = Student.objects.all()
    return render(request, 'lab11/student_list.html', {'students': students})
@login_required
def student_add(request):
    if request.method == 'POST':
        form = StudentForm(request.POST)
        if form.is_valid():
            student = form.save()
            # إضافة العنوان مباشرة بعد إضافة الطالب
            address_form = AddressForm(request.POST)
            if address_form.is_valid():
                address = address_form.save(commit=False)
                address.student = student  # ربط العنوان بالطالب
                address.save()
            return redirect('student_list')
    else:
        form = StudentForm()
        address_form = AddressForm()
    return render(request, 'lab11/student_form.html', {'form': form, 'address_form': address_form})

@login_required
def student_update(request, pk):
    student = get_object_or_404(Student, pk=pk)
    if request.method == 'POST':
        form = StudentForm(request.POST, instance=student)
        if form.is_valid():
            form.save()
            return redirect('student_list')
    else:
        form = StudentForm(instance=student)
    return render(request, 'lab11/student_form.html', {'form': form})
@login_required
def student_delete(request, pk):
    student = get_object_or_404(Student, pk=pk)
    if request.method == 'POST':
        student.delete()
        return redirect('student_list')
    return render(request, 'lab11/student_confirm_delete.html', {'student': student})

# Task 2 views
@login_required
def student2_list(request):
    students = Student2.objects.all()
    return render(request, 'lab11/student2_list.html', {'students': students})
@login_required
def student2_add(request):
    if request.method == 'POST':
        form = Student2Form(request.POST)
        if form.is_valid():
            student = form.save()
            # إضافة العناوين المرتبطة
            address_form = Address2Form(request.POST)
            if address_form.is_valid():
                address = address_form.save(commit=False)
                address.students.add(student)  # ربط الطالب بالعناوين
                address.save()
            return redirect('student2_list')
    else:
        form = Student2Form()
        address_form = Address2Form()
    return render(request, 'lab11/student2_form.html', {'form': form, 'address_form': 
address_form})


# Task 3 views (Gallery)
@login_required
def gallery_add(request):
    if request.method == 'POST':
        form = GalleryForm(request.POST, request.FILES)
        if form.is_valid():
            form.save()
            return redirect('gallery_list')
    else:
        form = GalleryForm()
    return render(request, 'lab11/gallery_form.html', {'form': form})
@login_required
def gallery_list(request):
    galleries = Gallery.objects.all()
    return render(request, 'lab11/gallery_list.html', {'galleries': galleries})

from django.shortcuts import render, redirect
from .forms import StudentImageForm
from .models import StudentImage
@login_required
def upload_image(request):
    if request.method == 'POST':
        form = StudentImageForm(request.POST, request.FILES)
        if form.is_valid():
            form.save()
            return redirect('image_list')
    else:
        form = StudentImageForm()
    return render(request, 'lab11/upload_image.html', {'form': form})
@login_required
def image_list(request):
    images = StudentImage.objects.all()
    return render(request, 'lab11/image_list.html', {'images': images})
```