**Django project Version 3 Notes**

Creating superuser, Admin panel, user authorization

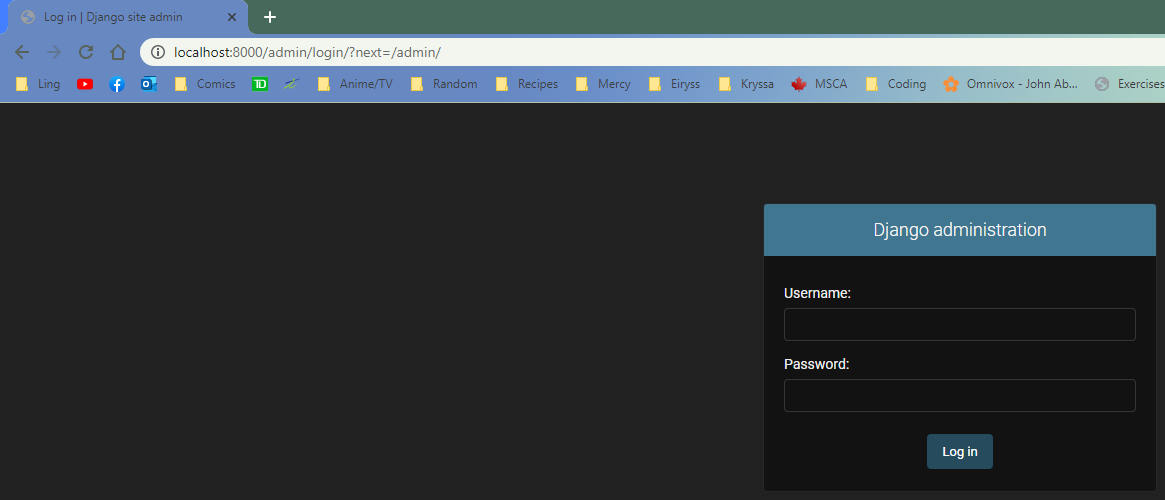
---------------------------------------------------------------------------------------------

**STEP 1 - Admin Panel**

1. Go to the django project’s Admin panel
   1. <http://localhost:8000/admin/>

* You will be redirected to this URL:
  + <http://localhost:8000/admin/login/?next=/admin/>
* To login to the admin panel, need to create Superuser

What you should see in your browser

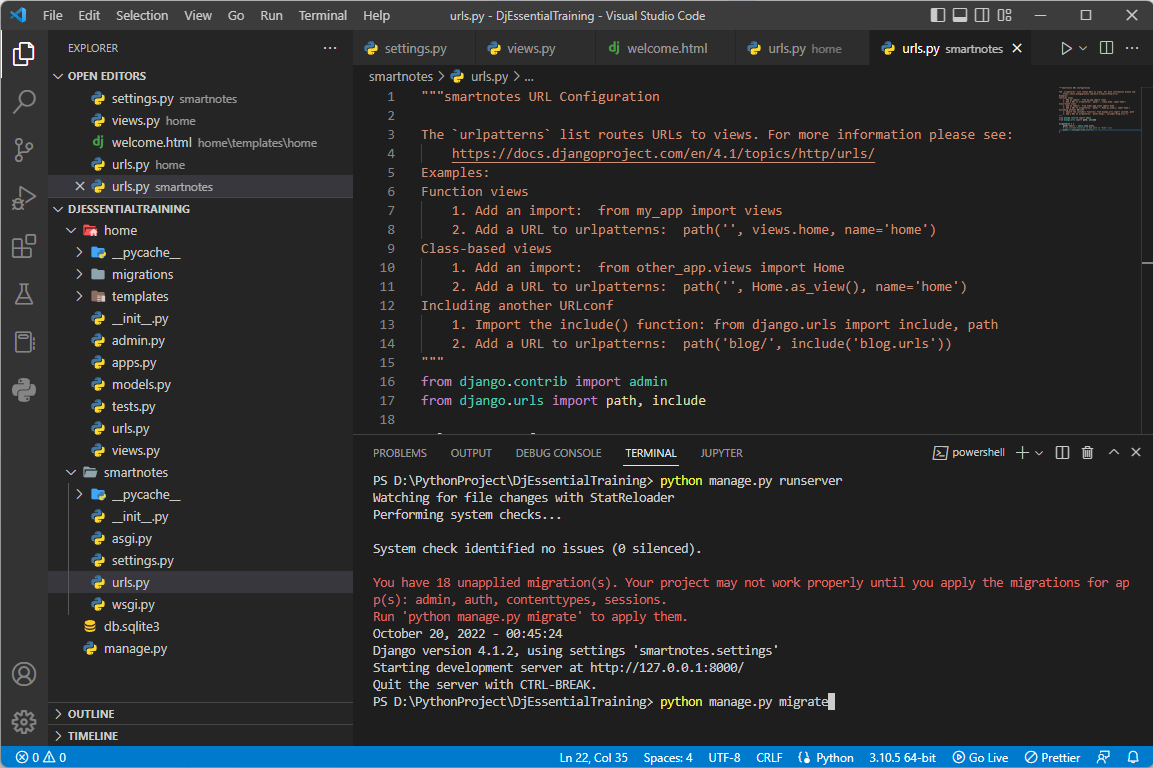


**STEP 2 - Migrate database changes**

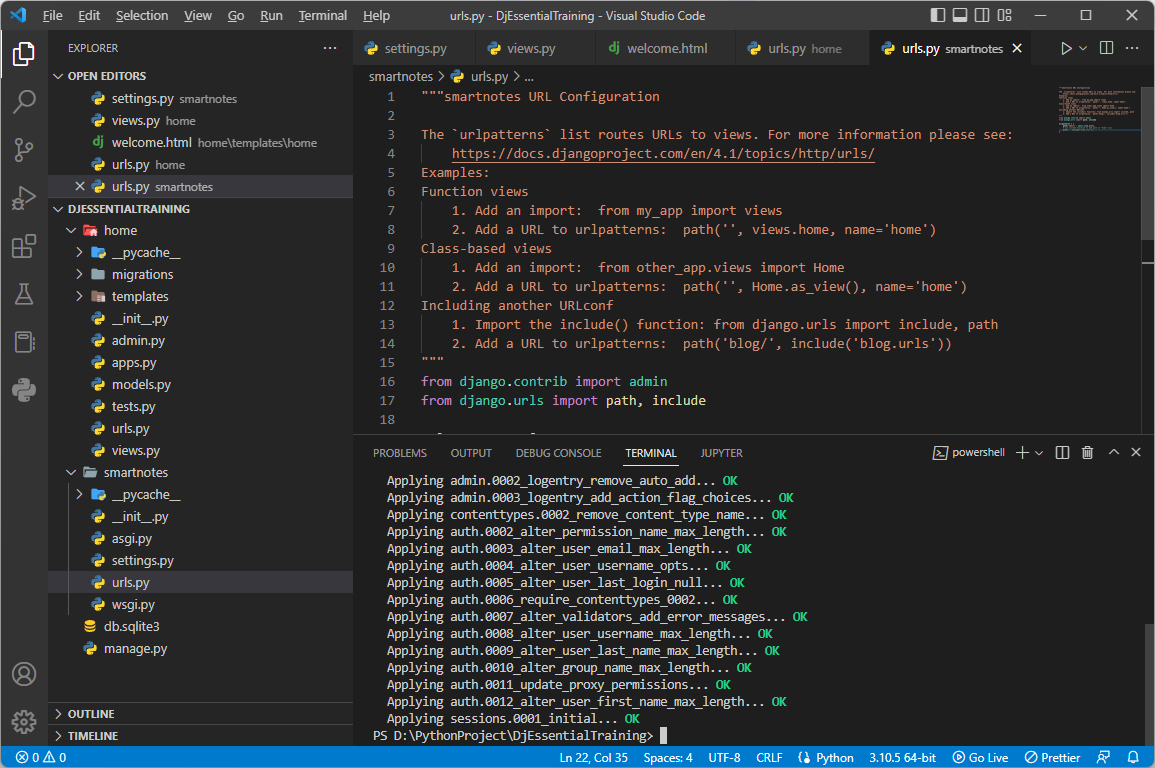
1. In terminal, type the following:
   1. python manage.py migrate

* Migrates all unapplied migrations to database
* Acts like git commit

Before Migrating to Database



After Migrating to Database

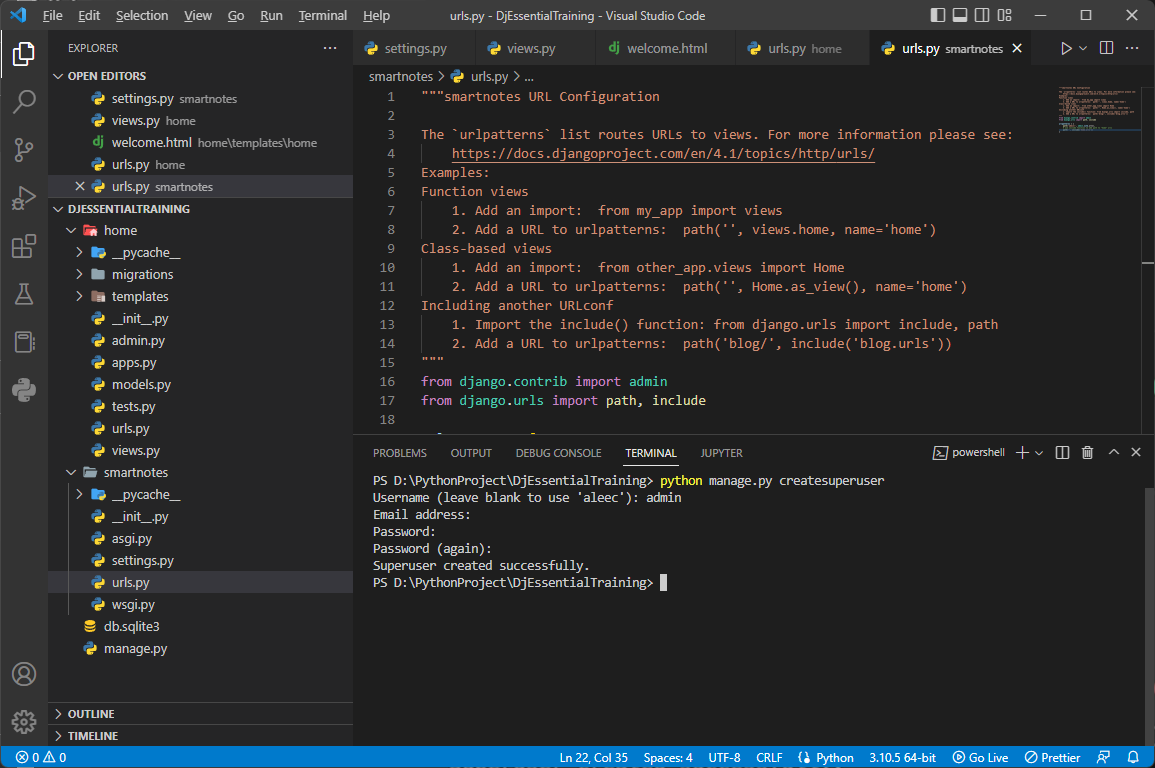


**STEP 3 - Create Superuser**

1. In terminal, type the following:
   1. python manage.py createsuperuser
      1. Type a username (admin)
         1. By default, will select computer user
      2. (Optional) type an email
      3. Type a password (projectpassword123)
      4. Type password again (projectpassword123)

* Remember what you typed for your password, it won’t show up in the terminal
* Can’t use common passwords like password123
* Can bypass adding a password, but not recommended due to security concerns

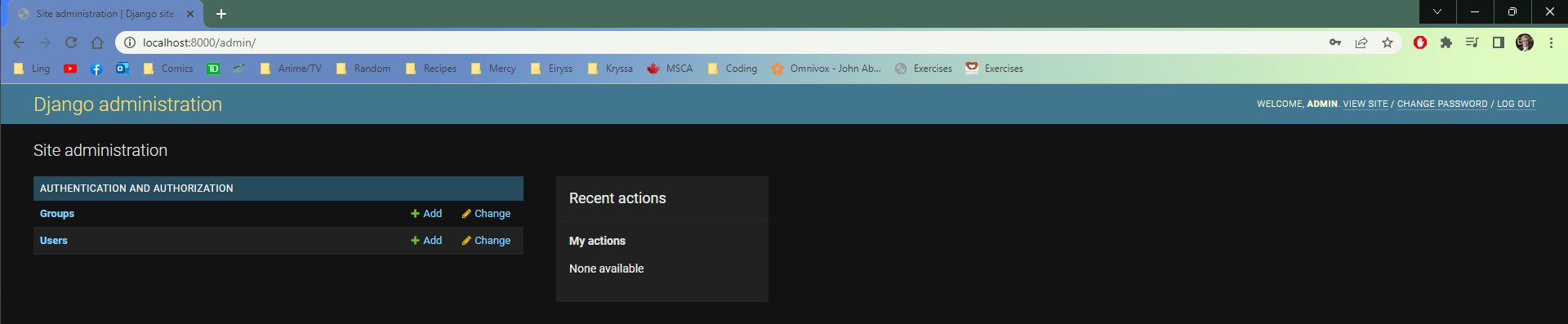
What your terminal should look like after creating superuser



**STEP 4 - Login to Admin Panel**

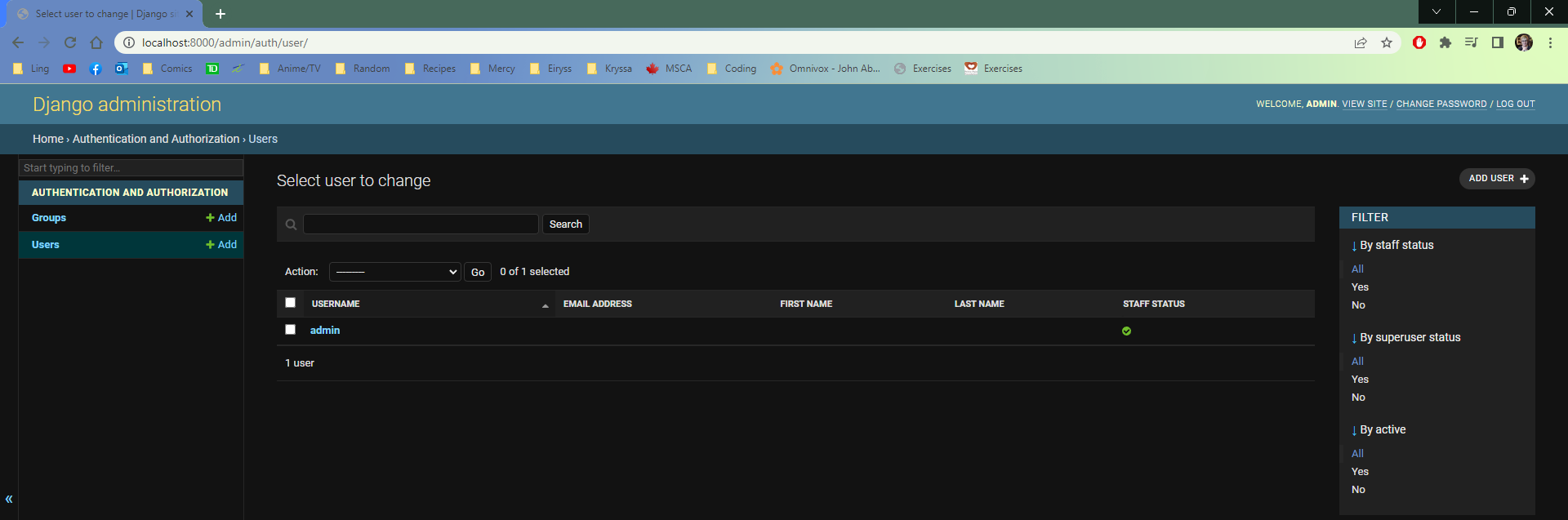
1. Run server
2. Go to <http://localhost:8000/admin/>
3. Login with superuser username (admin) and password (projectpassword123)

What your browser should show after logging into Admin Panel



* Admin Panel can be used to view database tables of users and models
* Can filter or search for specific users
* Can create or delete users

What Admin Panel user table should look like

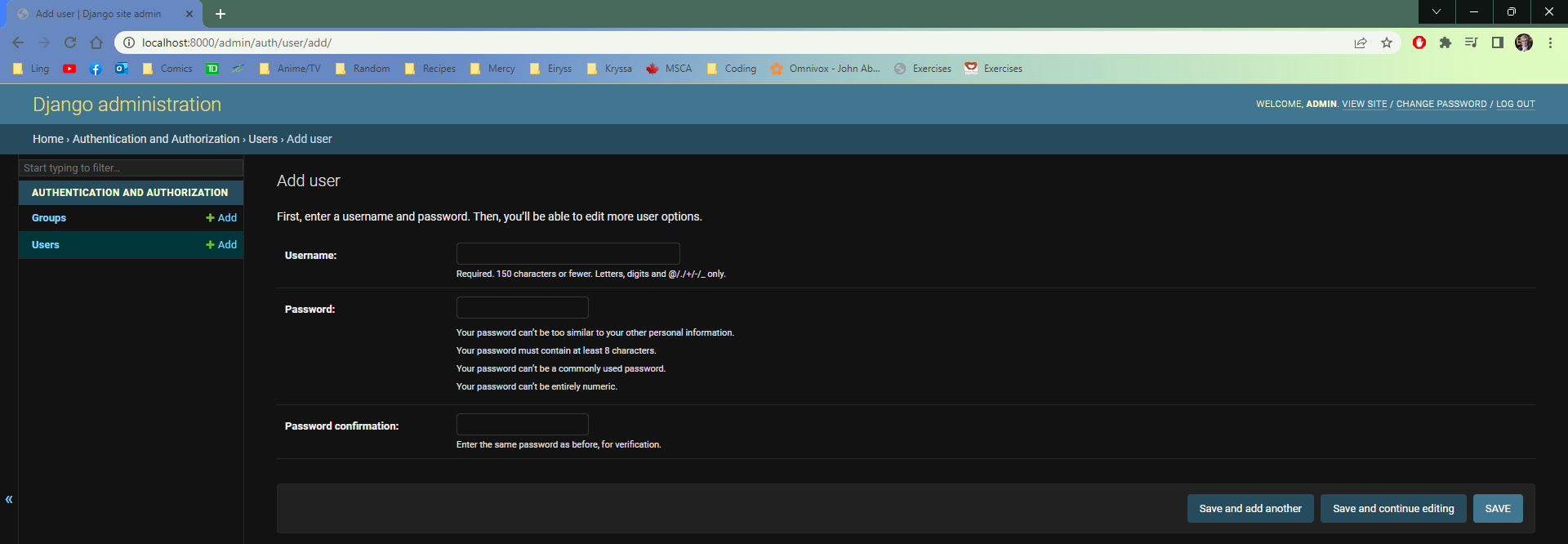


**STEP 5 - Create a new user**

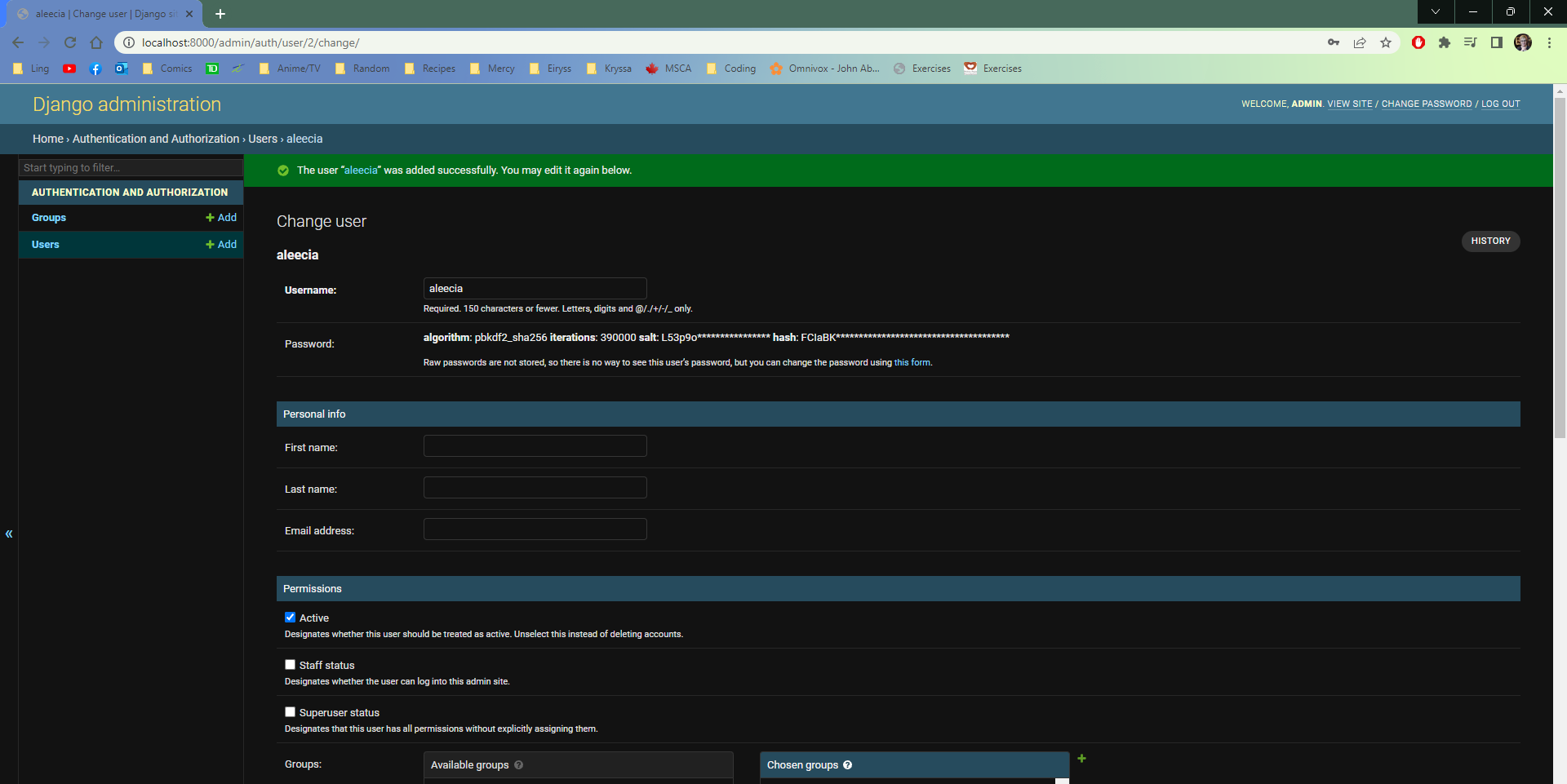
1. Go to user table
2. Click “ADD USER”
3. Fill in username, password, and password confirmation for new user
4. Click “SAVE”

* Can add other things to created user
  + Personal information, permissions, groups
  + User passwords are encrypted by Django

What Add User page should look like



What created user page should look like

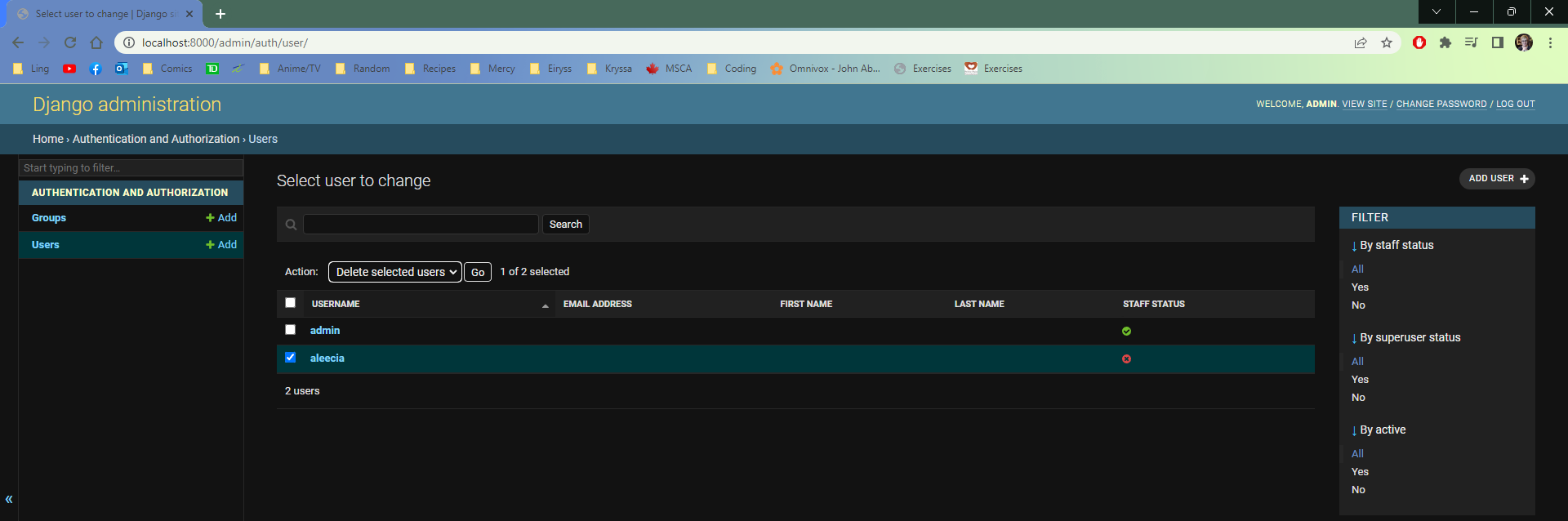


**STEP 6 - Delete User**

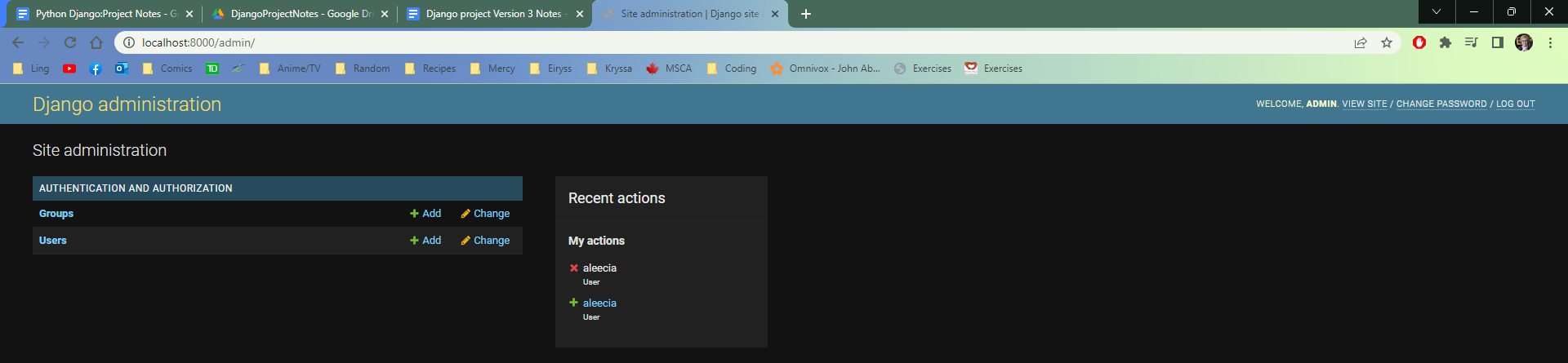
1. Go to Users table
2. Check user you want to delete
3. Select “Delete selected users” in dropdown menu
4. Click Go
5. On next screen, confirm you want to delete user

* Admin Panel shows recent actions

User table when deleting user



Admin Panel Recent Actions



**STEP 7 - User Authentication**

1. Create new file “authorized.html” in templates folder > “home” folder
2. Add basic HTML to authorized.html
   1. <html><h1>You are in a restricted area</h1></html>
3. Import Login\_required from django
   1. from django.contrib.auth.decorators import login\_required

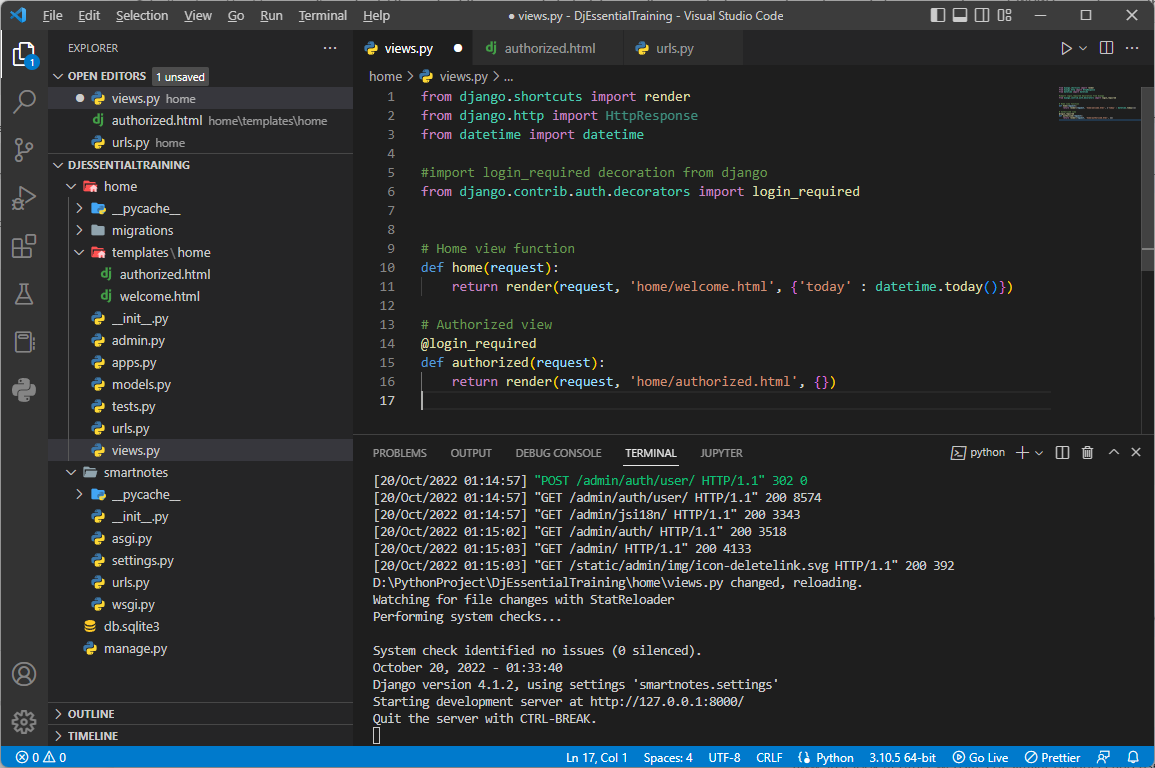
* login\_required is a decorated added to view functions that requires the user to be logged-in to view the webpage

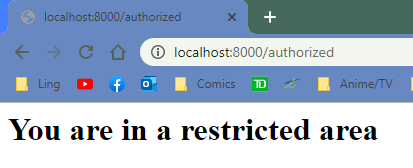
1. Add view for authorized in app folder (“home”) > views.py
   1. @login\_required
   2. def authorized(request):
      1. return render(request, ‘home/authorized.html’, {})
2. Add url path for authorized in app folder (“home”) > urls.py
   1. path(‘authorized’, views.authorized),

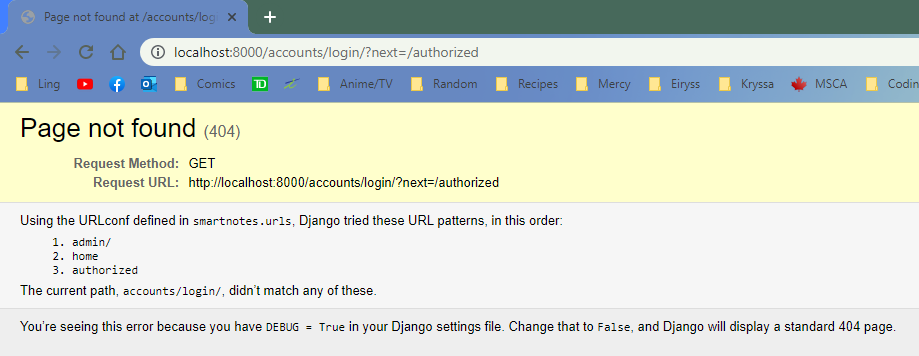
* Can add a redirect to login if no user is logged-in

1. Add parameter to @login\_required for authorized in app folder (“home”) > views.py
   1. @login\_required(login\_url=’/admin‘)

What View should look like after importing login\_required and adding authorized view



What the browser shows if user is logged in  
  
What the browser shows if no user is logged in



What the browser shows after login\_required redirect

