МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ

І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ

КАФЕДРА КОМП’ЮТЕРНИХ НАУК

**ЛАБОРАТОРНА РОБОТА №8**

З дисципліні "Програмування на мові Python"

Виконав студент 4-го курсу

Гарбич Анатолій

Група КН-21002б,

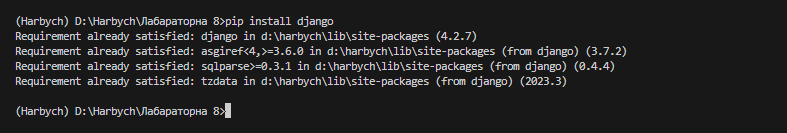
**Варіант № 3**

Перевірив:

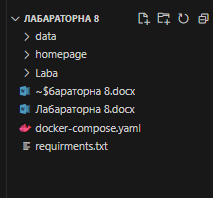
к.ф.-м.н. доцент кафедри КН

*Кириченко Віктор Вікторович*

Установили django



Створили проект Laba та програму homepage



Наший докер файл

version: '3.12.0'

services:

  db\_lb8:

    container\_name: db\_lb8

    restart: always

    image: postgres:16.0-alpine3.18

    environment:

      - POSTGRES\_USER=admin

      - POSTGRES\_PASSWORD=root

      - POSTGRES\_DB=db\_lb8

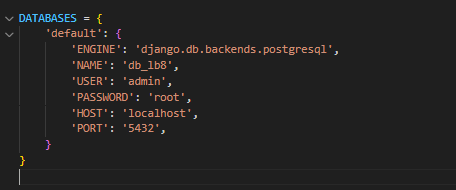
    volumes:

      - ./data:/var/lib/postgresql/data

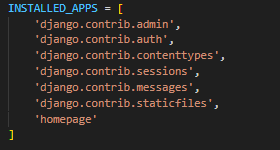
    ports:

      - "5432:5432"

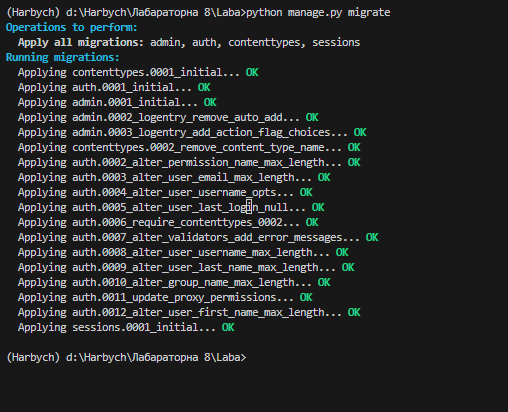
Переходимо в settings.py та підключаємо нашу бд



Додаємо наший додаток в список програм



Робимо міграції



Створюємо наші моделі бд

from django.db import models

class Department(models.Model):

    department\_id = models.AutoField(primary\_key=True)

    department\_name = models.CharField(max\_length=255)

    phone\_number = models.CharField(max\_length=255, null=True, blank=True)

    room\_number = models.IntegerField()

class Position(models.Model):

    position\_id = models.AutoField(primary\_key=True)

    position\_name = models.CharField(max\_length=255)

    salary = models.IntegerField()

    bonus\_percentage = models.IntegerField()

class Project(models.Model):

    project\_id = models.AutoField(primary\_key=True)

    project\_name = models.CharField(max\_length=255)

    deadline = models.DateField()

    funding\_size = models.IntegerField()

class ProjectExecution(models.Model):

    execution\_id = models.AutoField(primary\_key=True)

    project = models.ForeignKey(Project, on\_delete=models.CASCADE)

    department = models.ForeignKey(Department, on\_delete=models.CASCADE)

    start\_date = models.DateField()

class Employee(models.Model):

    employee\_id = models.AutoField(primary\_key=True)

    last\_name = models.CharField(max\_length=255)

    first\_name = models.CharField(max\_length=255)

    middle\_name = models.CharField(max\_length=255)

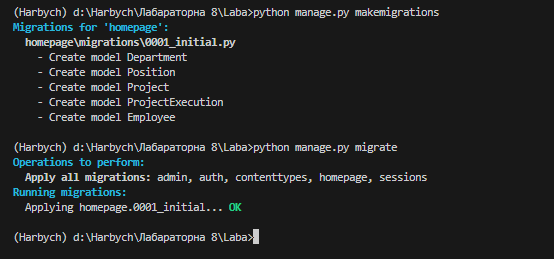
    address = models.CharField(max\_length=255)

    phone\_number = models.CharField(max\_length=255, null=True, blank=True)

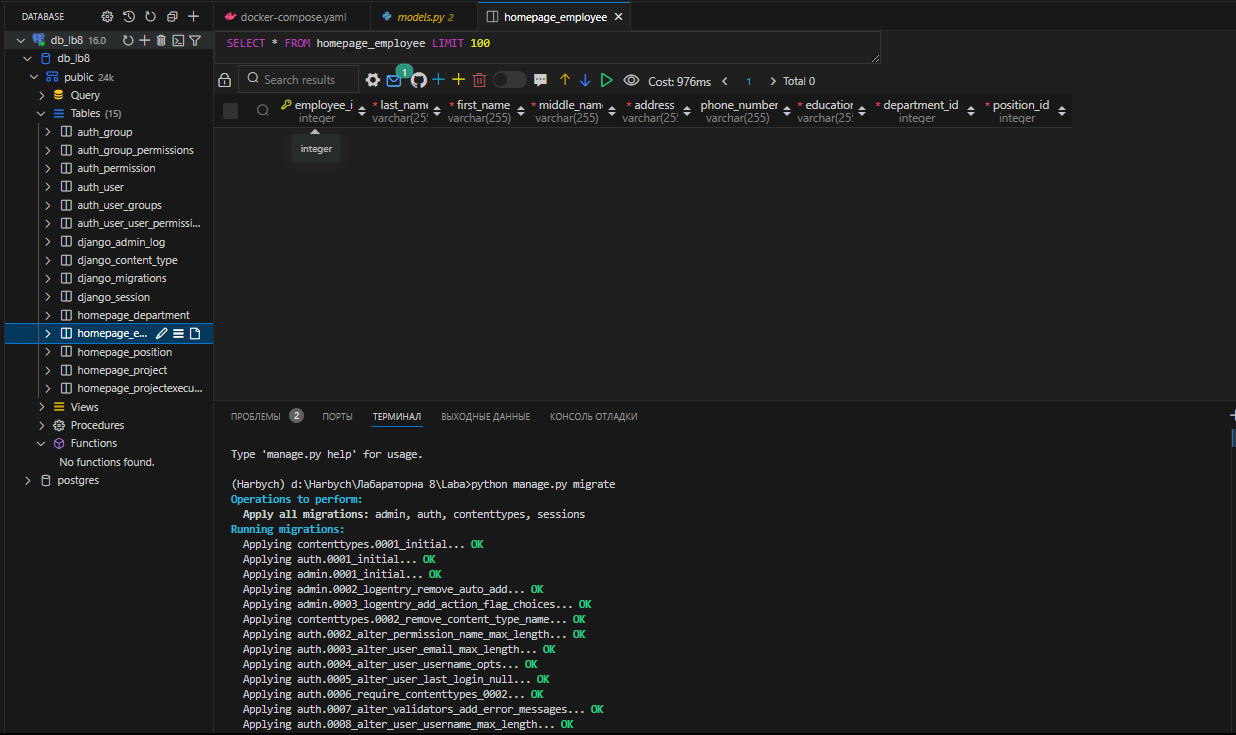
    education = models.CharField(max\_length=255)

    department = models.ForeignKey(Department, on\_delete=models.CASCADE)

    position = models.ForeignKey(Position, on\_delete=models.CASCADE)

Робимо зміни та повторюємо міграцію 

Наші таблиці створені



Добавляю до urls.py наш додаток

from django.contrib import admin

from django.urls import path, include

urlpatterns = [

    path('admin/', admin.site.urls),

    path('', include('homepage.urls')),

]

Створюємо urls.py в додатку home

from django.urls import path

from .views import home

urlpatterns = [

    path('', home, name='index.html'),

]

Views.py

from django.shortcuts import render

from .models import Employee, Project, ProjectExecution, Position, Department

def home(request):

    employees = Employee.objects.all()

    projects = Project.objects.all()

    project\_executions = ProjectExecution.objects.all()

    position = Position.objects.all()

    departament = Department.objects.all()

    context = {

        'employees': employees,

        'projects': projects,

        'project\_executions': project\_executions,

        'position' : position,

        'departament': departament

    }

    return render(request, 'index.html', context)

admin.py

from django.contrib import admin

from .models import Department, Employee, Position, Project,ProjectExecution

class DepartmentAdmin(admin.ModelAdmin):

    list\_display = ('department\_id', 'department\_name', 'phone\_number', 'room\_number')

class EmployeeAdmin(admin.ModelAdmin):

    list\_display = ('employee\_id', 'last\_name', 'first\_name', 'middle\_name', 'address','phone\_number','education','department','position')

class PositionAdmin(admin.ModelAdmin):

    list\_display = ('position\_id', 'position\_name', 'salary', 'bonus\_percentage')

class ProjectAdmin(admin.ModelAdmin):

    list\_display = ('project\_id', 'project\_name', 'deadline', 'funding\_size')

class ProjectExecutionAdmin(admin.ModelAdmin):

    list\_display = ('execution\_id', 'project', 'department', 'start\_date')

admin.site.register(Department, DepartmentAdmin)

admin.site.register(Employee, EmployeeAdmin)

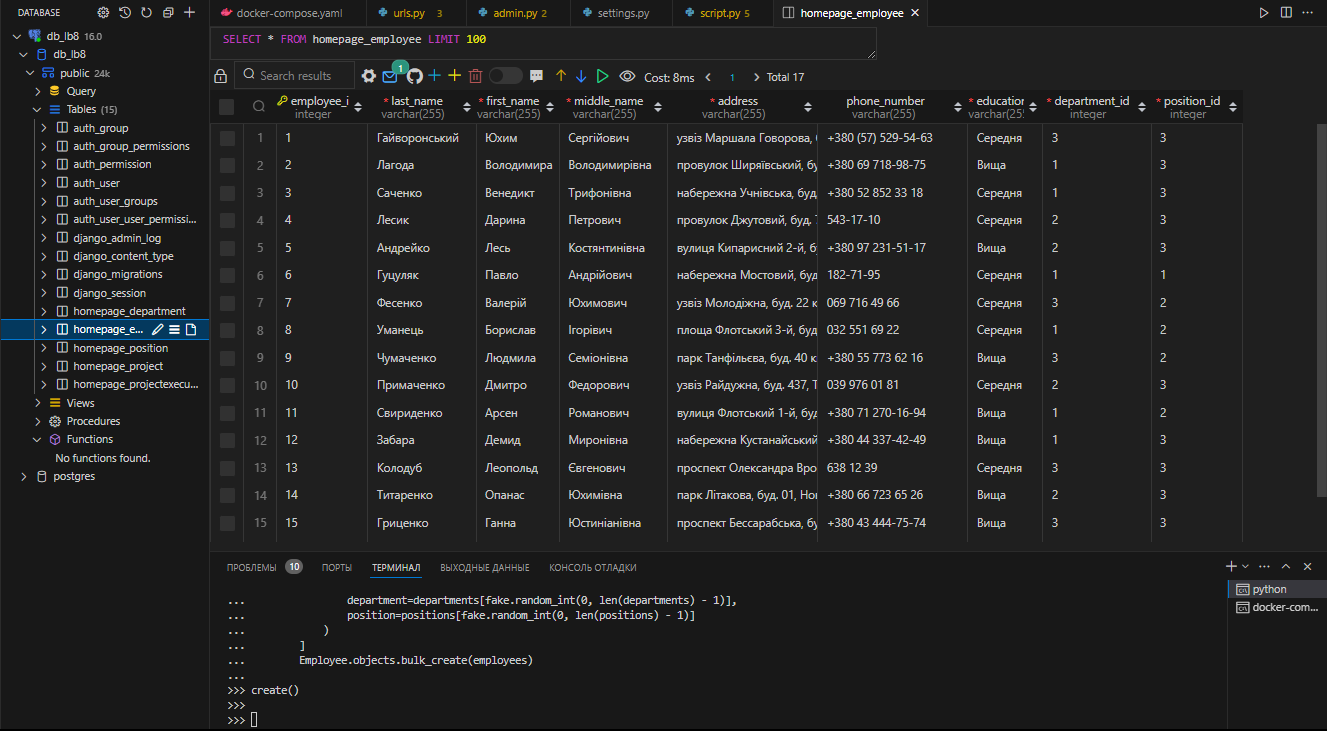
admin.site.register(Position, PositionAdmin)

admin.site.register(Project, ProjectAdmin)

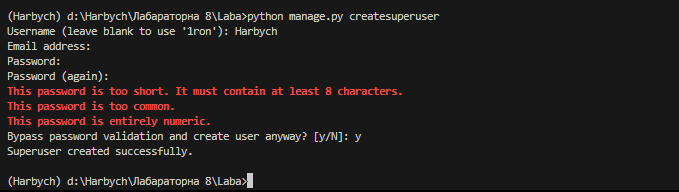
admin.site.register(ProjectExecution, ProjectExecutionAdmin)

Через команду python manage.py shell\_plus

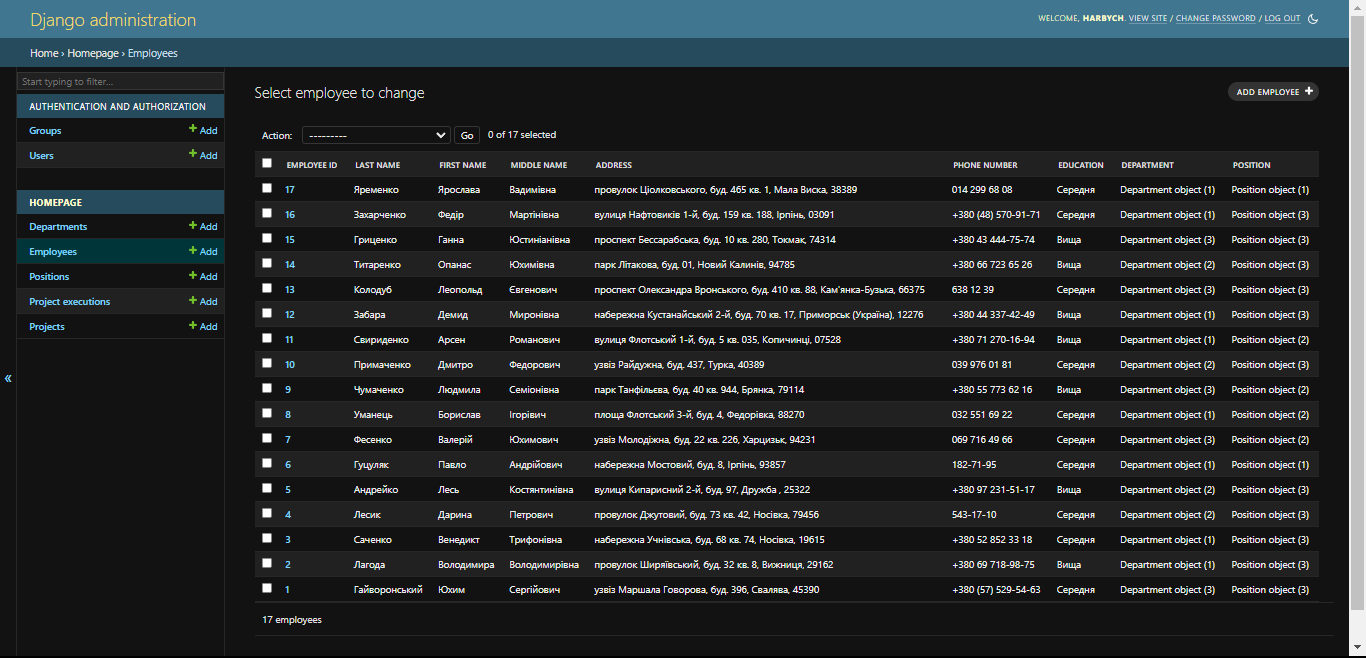
Заповняємо нашу бд кодом із файлу scripts.py



Створюємо адміністратора



Перевіряємо наші данні в панелі Адміністратора за адресою http://127.0.0.1:8000/admin/

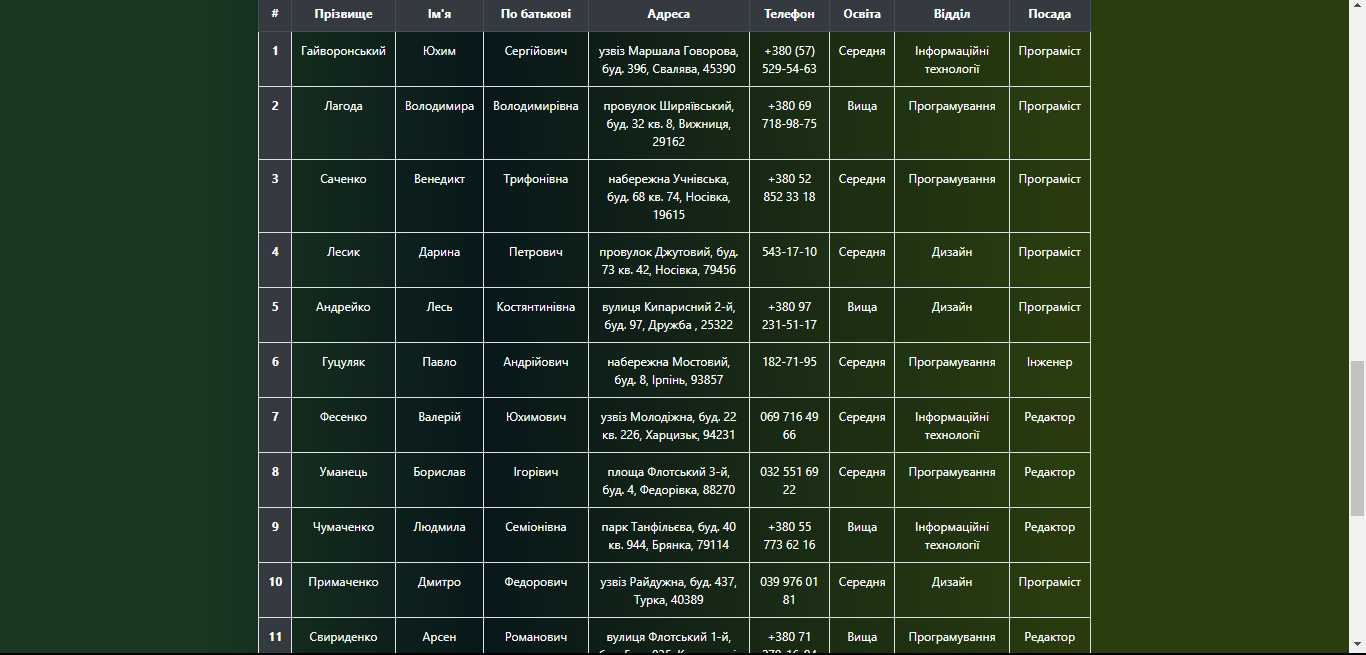


Усе працює створюмо папку templates в якій створюємо наш html



Наша веб сторінка





Силка на гіт