НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

Кафедра інформаційних систем та мереж

**Звіт**

З дисципліни «Екстримальне програмування»

Лабораторної робота №1

**Основи з Django**

Варіант №21

Виконав:

Студент гр. КН-311

Нанівський О.І.

Прийняв:

Щербак С.С.

Львів – 2020

Django — високорівневий відкритий Python-фреймворк для розробки веб-систем. Названо його було на честь джазмена Джанго Рейнхардта. Сайт на Django будується з однієї або декількох частин, які рекомендується робити модульними. Це одна з істотних архітектурних відмінностей цього фреймворку від деяких інших.

Для створення проекту необхідно спочатку його завантажити. Один з варіантів команда:

pip install Django.

Створемо проект:

django-admin startproject mysite

Структура проекту:

mysite/

manage.py

mysite/

\_\_init\_\_.py

settings.py

urls.py

asgi.py

wsgi.py

Ці файли:

Зовнішній каталог mysite / root є контейнером для вашого проекту. Його ім'я не має значення для Джанго; ви можете перейменувати його на все, що завгодно.

Manag.py: утиліта командного рядка, яка дозволяє взаємодіяти з цим проектом Django різними способами.

Внутрішній каталог mysite / фактичний пакет Python для вашого проекту. Його ім'я - ім'я пакету Python, яке потрібно використовувати для імпортування будь-чого всередині нього (наприклад, mysite.urls).

mysite / \_\_ init\_\_.py: порожній файл, який повідомляє Python, що цей каталог слід вважати пакетом Python.

mysite / settings.py: налаштування / конфігурація для цього проекту Django.

mysite / urls.py: декларації URL для проекту Django; "зміст" вашого веб-сайту, що працює на Django.

mysite / asgi.py: вхідна точка для ASGI-сумісних веб-серверів.

mysite / wsgi.py: точка входу для веб-серверів, сумісних з WSGI.

Для перевірки коректності встановлення можна запустити сервер розробки:

python manage.py runserver

Для створення нового додатку використовується наступна команда:

python manage.py startapp polls

Структура додатку:

polls/

\_\_init\_\_.py

admin.py

apps.py

migrations/

\_\_init\_\_.py

models.py

tests.py

views.py

Для робати додатку його необхідно записати в файл налаштувань. Додати в список INSTALLED\_APPS.

Деякі з включених додатків можуть використовувати таблиці бази даних. Для їх створення потрібно прописати команду:

python manage.py migrate

По замовчуванню Django використовує sqlite.

Моделі бази даних пишуться за допомогою python в фацлі models.py

Приклад моделей:

**from** **django.db** **import** models

**class** **Question**(models.Model):

question\_text = models.CharField(max\_length=200)

pub\_date = models.DateTimeField('date published')

**class** **Choice**(models.Model):

question = models.ForeignKey(Question, on\_delete=models.CASCADE)

choice\_text = models.CharField(max\_length=200)

votes = models.IntegerField(default=0)

Щоб зберегти зміни для майбутньогї міграції викоритсовується команда:

python manage.py makemigrations polls

Також в Django присутня панель адміна за посиланням «/admin».

В якій зручно можна створювати або видаляти дані в таблиці. Але попередньо дану можель потріно зареєструвати в файлі admin.py відповідного додатка.

Приклад реєстрації:

**from** **django.contrib** **import** admin

**from** **.models** **import** Question

admin.site.register(Question)

Акаунт адміна створюється наступною командою:

python manage.py createsuperuser