**ЛАБОРАТОРНА РОБОТА № 1**

**Тема:** Повторення основних понять мови Python

**Мета роботи**: полягає у вивченні та застосуванні на практиці основного функціоналу Python

**Обладнання та програмне забезпечення*:*** ПК, Visual Studio Code, MS Word.

**Короткі теоретичні відомості**

Django — Python-фреймворк з відкритим кодом для швидкої розробки веб-систем. Спочатку технологію розробляли як засіб для керування сайтами новин LJWorld.com, lawrence.com та Kusports.com компанії The World Company і це значно вплинуло на її архітектуру, оскільки реалізовано цілий ряд функціональних можливостей, які допомагають у швидкій розробці веб-сайтів інформаційного характеру. Щоб мати уявлення про те, що ви можете зробити з Django, добре було б дізнатися, хто його використовує.

Серед найбільших компаній, що використовують Django, ми маємо: Instagram, Disqus, Mozilla, Bitbucket, Last.fm, National Geographic. Сайт з використанням Django будується з однієї або декількох частин, які рекомендовано робити модульними (аплікації, app). Архітектура схожа на «Модель-Вид-Контролер» (MVC). Однак, тут роль «контролера» класичної моделі MVC виконує «вид» (view), а «видом» називається «шаблон» (template). Таким чином, MVC розробники Django називають MTV («МодельШаблон-Вид»). Однією з основних переваг для розробника є відсутність потреби створювати контролери та сторінки для адміністративної частини сайту, в збірці є вбудований модуль для керування вмістом, який можна долучити до будь-якого сайту, написаний на Django, і який може керувати відразу декількома сайтами на одному сервері.

Ще однією значною перевагою Django є адміністративний модуль, який дає змогу створювати, змінювати і вилучати будь-які об’єкти наповнення сайту, фіксуючи всі дії та надає інтерфейс для керування обліковими записами користувачів і групами (з призначенням прав). У збірку також внесені засоби для системи коментарів і «статичні сторінки», які можна використовувати без необхідності писати додаткові контролери та відображення.

До базових функцій Django належать: Об’єктно-реляційне відображення (ORM), яке допомагає суттєво спростити роботу з базою даних. Об’єкти БД в термінології Django іменуються «моделями». Розробнику не потрібно писати SQL-запити (але така можливість є), бо під час виконання синхронізації проекту з БД автоматично будуть створені усі таблиці з полями, які відповідають властивостям (properties) описаних моделей. Зручним також є синтаксис url-адрес, який побудований на регулярних виразах. Розробник не обмежений у використанні певної схеми посилань. Посилання можуть групуватися за кожним модулем проекту в окремий файл. Крім того, можна використовувати багато інших способів групування url-адрес, як стосовно конкретного модуля, так і стосовно усього проекту. Зручна система шаблонів, яка передбачає наявність окремої мови для їх опису. Вона є достатньо простою, містить оператори циклу, умови, засоби форматування даних. Мова шаблонів виконує функцію відображення даних.

Гнучка підсистема кешування дає змогу дуже швидко налаштувати Django-проект для роботи з Memcached чи будьякою іншою надбудовою. Інструменти Django дають змогу кешувати SQL-вибірки, шаблони та їх частини і просто окремі змінні. Простою є інтернаціоналізація, що базується на концепції «лінивого» перекладу. Це означає, що якщо певний рядок тексту не має перекладу, то буде використано базовий текст і не буде показано повідомлення про помилку. Також можна використовувати спеціальні функції для контролю перекладу рядкових даних.

У збірці Django є власний веб-сервер для розробки і налагоджування. Він автоматично відслідковує зміни у файлах програмного коду і перезапускається, що дуже зручно при розробці проекту. Для розробки різного рівня веб-застосунків можна використовувати вже готові модулі чи надбудови. На сайті djangopackages.com дуже легко відшукати пакет, який найкраще підійде для вирішення конкретної задачі. Всі ці застосунки розповсюджуються вільно і кожен охочий може їх завантажити з репозиторію чи приєднатися до команди розробників. Для роботи з Python існує багато спеціальних середовищ чи надбудов, та все ж найпопулярнішим вважається PyCharm. Одночасно розвиваються дві окремі гілки цього проекту: комерційна (розробляється компанією JetBreins) і відкрита (розвивається спільнотою).

Установіть django командою

**pip install django**

І створіть проект (назва проекту mysite, може бути ваша власна) командою

**django-admin startproject mysite**

Тепер в каталозі myblog поруч з myvenv зявилась папка mysite. Вона міститиме всі вихідні файли майбутнього сайту. Команда django-admin (а також скрипт manage.py) мають цілий набір підготованих команд, які дозволяють працювати із базою даних, запускати міграції, запускати розробницький Django сервер і ще дуже багато інших речей.

Робиться це доволі просто командою скрипта manage.py:

- створення початкової схеми бази даних:

**python mysite/manage.py migrate**

- додавання суперкористувача, username i password будуть використовуватись для входу в адмінку, можна створювати довільну кількість суперкористувачів:

**python mysite/manage.py createsuperuser**

**A blue screen with white text

Description automatically generated**

**Завдання для самостійної роботи:**

1. Установити Python Django
2. Створити Django проект та ознайомитись з структурою папок всередині проєкта
3. Створити початкову схему бд і створити суперкористувача
4. Запустити сервер

### Контрольні запитання:

1. В чому відмінності Django від JavaScript?
2. Яка структура папок в створеному проекті Django і за що вони відповідають?
3. Напишіть яким чином можна контролювати версії встановлений пакетів(бібліотек)?
4. Напишіть переваги використання Python Django