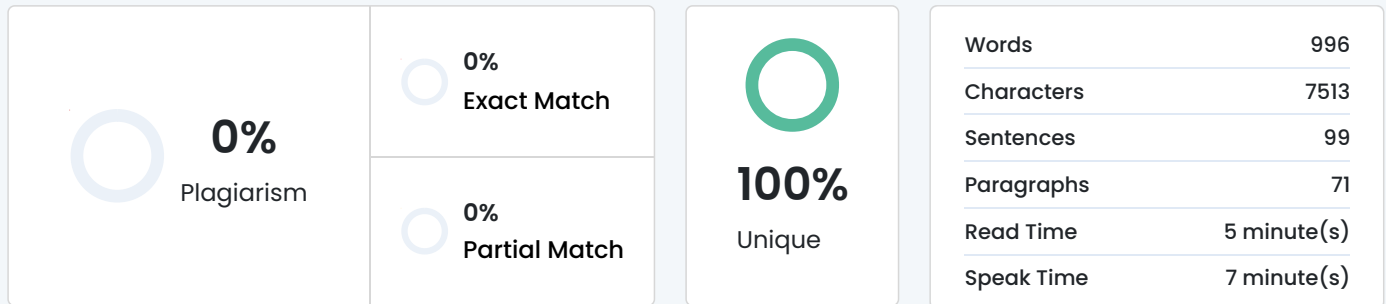


Plagiarism Scan Report



Content Checked For Plagiarism

Початок розробки:

У 1989 році Гвідо ван Россум почав працювати над новою мовою у вільний час, використовуючи напрацювання проєкту ABC та додаючи до них такі важливі функції, як робота з винятками, модульність та взаємодію з операційною системою.

Назву «Python» він обрав не на честь змії, а через своє захоплення британським комедійним шоу Monty Python's Flying Circus. Він прагнув, щоб нова мова була легкою та зрозумілою, а її використання приносило задоволення.

Офіційний випуск:

Перша версія Python 0.9.0 була випущена в 1991 році. Вона вже мала ключові особливості, які визначають мову до сьогодні:

- Динамічну типізацію
- Автоматичне керування пам'яттю
- Підтримку об'єктно-орієнтованого програмування
- Вбудовані типи даних (списки, словники тощо)

У 1994 році вийшла Python 1.0, яка стала популярною серед розробників завдяки своїй простоті та потужності.

Подальший розвиток:

- Python 2.0 (2000 р.) — додано збірку сміття та підтримку спискових включень (list comprehensions).
- Python 3.0 (2008 р.) — значне оновлення, яке принесло покращення в синтаксисі та управлінні пам'яттю, проте не зберегло зворотну сумісність із Python 2.x.

На сьогодні Python є однією з найпопулярніших мов програмування у світі, активно використовується в розробці веб-додатків, наукових дослідженнях, аналізі даних, машинному навчанні та багатьох інших сферах.

Рис.2.2.1 – Середовище розробки Visual Studio

2.3 База даних SQL Server

SQL Server — це потужна реляційна система керування базами даних (РСКБД), розроблена компанією Microsoft. Вона забезпечує зберігання, обробку та управління великими обсягами даних, а також підтримує мову структурованих запитів SQL (Structured Query Language) для взаємодії з даними.

Історія та розвиток:

Перша версія SQL Server була випущена в 1989 році як спільний продукт компаній Microsoft, Sybase та Ashton-Tate. Однак у подальшому Microsoft самостійно розвивала систему, зробивши її ключовим компонентом екосистеми своїх корпоративних рішень.

З роками SQL Server зазнав численних удосконалень, серед яких підтримка хмарних технологій, розширені засоби безпеки, покращена продуктивність та оптимізація запитів.

Основні можливості SQL Server:

1. Реляційна модель даних – SQL Server використовує таблиці, що мають зв'язки між собою, що дозволяє ефективно зберігати й обробляти великі обсяги структурованої інформації.

2. Підтримка транзакцій – забезпечує цілісність даних завдяки використанню механізмів ACID (атомарність, узгодженість, ізоляція, надійність).
3. Розширена система безпеки – містить механізми аутентифікації, авторизації, шифрування та аудитів для захисту даних.
4. Оптимізація продуктивності – завдяки кешуванню запитів, індексуванню та аналітичним інструментам SQL Server дозволяє швидко обробляти складні запити.
5. Підтримка хмарних технологій – інтеграція з Microsoft Azure дає змогу працювати з базами даних у хмарному середовищі.
6. Вбудовані засоби резервного копіювання – дозволяють автоматично створювати резервні копії для захисту від втрати даних.

Використання SQL Server у проекті:

У цьому проекті SQL Server використовується для зберігання інформації про книги, їхні характеристики (автор, жанр, рейтинг), а також управління даними користувачів та рекомендаціями. Основні завдання, які виконує база даних у програмі:

- Збереження списку книг із відповідними атрибутами.
- Фільтрація та пошук книг за жанром, рейтингом та іншими параметрами.
- Зберігання інформації про користувачів і їхні вподобання.
- Формування рекомендацій на основі вподобань користувачів.

Для взаємодії Python з SQL Server у проекті використовується бібліотека pyodbc, яка забезпечує виконання SQL-запитів та отримання даних з бази.

SQL Server є оптимальним вибором для реалізації цього програмного додатку, оскільки забезпечує високу продуктивність, безпеку та надійність збереження інформації.

Рис.2.3.1 – База даних BooksDB в SQL Server

3. ПРОЕКТУВАННЯ ЗАГАЛЬНОГО АЛГОРИТМУ РОБОТИ ПРОГРАМИ

3.1 Загальна структура програми

Програмний додаток для рекомендації книг має модульну структуру, що забезпечує зручність у розробці, підтримці та розширенні функціоналу. Основні компоненти програми включають:

- Графічний інтерфейс користувача (GUI) – реалізований за допомогою Tkinter, забезпечує взаємодію користувача з додатком.
- База даних (SQL Server) – використовується для зберігання інформації про книги, користувачів та рекомендації.
- Модуль управління базою даних – клас для взаємодії з SQL Server через бібліотеку pyodbc.
- Модуль обробки рекомендацій – алгоритм вибору книг на основі жанру, рейтингу та вподобань користувача.
- Модуль керування користувачами та книгами – реалізує додавання, редагування та видалення книг, а також управління профілями користувачів.

3.2 Алгоритм роботи програми

- Запуск програми – відкривається головне вікно з інтерфейсом.
- Завантаження даних – підключення до SQL Server, отримання списку книг.
- Додавання, редагування та видалення книг – адміністратор може змінювати базу книг.
- Вибір параметрів користувачем – користувач обирає жанр, рейтинг тощо.
- Генерація рекомендацій – алгоритм аналізує вибрані параметри та пропонує відповідні книги.
- Відображення результату – користувач отримує список рекомендованих книг.

Ця структура дозволяє зробити програму зручною, гнучкою та легкою у підтримці, а використання SQL Server та Python забезпечує надійну та швидку роботу додатка.

Рис.3.2.1 – Алгоритм роботи застосунку від початку до дій користувача

Рис.3.2.2 – Алгоритм роботи завдання “Додати книгу”

Рис.3.2.3 – Алгоритм роботи завдання “Редагувати книгу”

Рис.3.2.4 – Алгоритм роботи завдання “Видалити книгу”

Рис.3.2.5 – Алгоритм роботи завдання “Отримати рекомендації”

Рис.3.2.6 – Алгоритм роботи завдання “Отримати список книг”

Проектування алгоритму є важливим етапом розробки програмного забезпечення, що дозволяє структурувати логіку роботи та оптимізувати процеси. Запропонована схема забезпечує гнучкість, надійність та зручність у використанні, а також гарантує ефективний підбір та рекомендацію книг відповідно до вподобань користувача. Використання бази даних дозволяє зберігати та обробляти інформацію про книги, жанри, рейтинги та користувачів, що сприяє точності та персоналізації рекомендацій.

4. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Розробка програмного забезпечення для рекомендації книг включає кілька основних етапів: створення бази даних, реалізацію основних модулів та розробку графічного інтерфейсу. База даних містить інформацію про книги, жанри, рейтинги та вподобання користувачів. Основні модулі забезпечують обробку даних та генерацію рекомендацій, а графічний інтерфейс надає користувачам зручний доступ до функціоналу додатку.

4.1 Створення бази даних

Створення бази даних є одним із ключових етапів розробки програмного додатку для рекомендації книг. База даних забезпечує збереження та управління інформацією про книги, жанри, рейтинги, відгуки користувачів та їхні вподобання.

Процес розробки бази даних включає:

- Проектування структури даних та взаємозв'язків між таблицями.
- Використання SQL Server для зберігання даних.
- Реалізацію взаємодії додатку з базою даних через мову Python та бібліотеку pyodbc.

Правильна організація бази даних дозволяє швидко обробляти інформацію, створювати

Для роботи з базою даних використовується бібліотека pyodbc, яка дозволяє взаємодіяти з SQL Server через ODBC (Open Database Connectivity).

ODBC – це стандартний інтерфейс програмування, який дозволяє Python підключатися до різних СУБД (систем управління базами даних), зокрема Microsoft SQL Server.

Рис.4.1.1 – Імпорт pyodbc

Matched Source

No plagiarism found