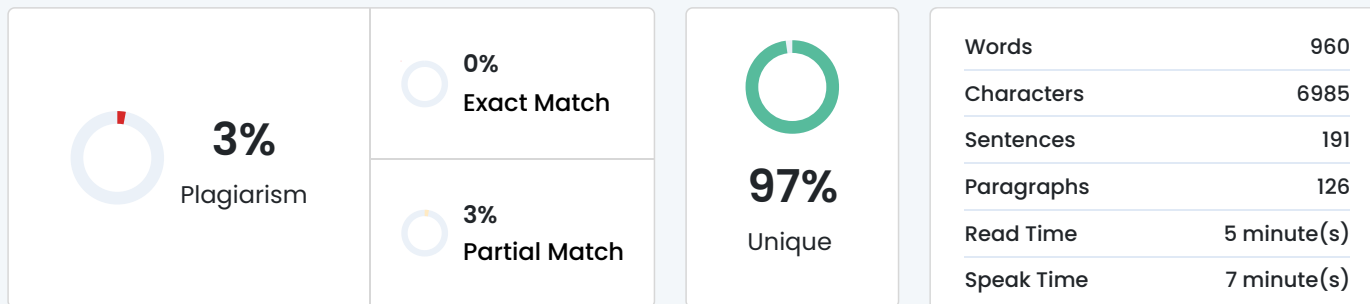


Plagiarism Scan Report



Content Checked For Plagiarism

4.9 Візуальна частина

Імпорт бібліотек:

- tkinter – основна бібліотека для створення GUI в Python.
- messagebox – модуль для відображення повідомлень (підтвердження успіху, помилки тощо).
- ttk – модуль для розширених віджетів інтерфейсу.
- database_connection – імпорт функцій для роботи з базою даних:
 - o add_book_to_db() – додає книгу до бази даних.
 - o get_recommendations_from_db() – отримує рекомендації книг.
 - o update_book_in_db() – оновлює дані книги в базі.
 - o delete_book_from_db() – видаляє книгу з бази.

Рис.4.9.1 – Імпорт бібліотек

4.10 Функція для додавання книги

Призначення: отримує дані з полів вводу та передає їх у базу.

Алгоритм роботи:

1. Отримує назву, автора, жанр, рік та рейтинг книги.
2. Перевіряє, чи всі поля заповнені.
3. Пробоє конвертувати рік у int, а рейтинг у float.
4. Викликає функцію add_book_to_db() для додавання книги.
5. Відображає повідомлення про успішне додавання або помилку.
6. Очищає поля вводу після додавання.

Рис.4.10.1 – Функція для додавання книги

4.11 Функція для редагування книги

Призначення: оновлює дані книги у базі даних.

Алгоритм роботи:

1. Отримує ID книги та нові значення полів.
2. Перевіряє, чи всі поля заповнені.
3. Конвертує рік і рейтинг у відповідні числові формати.
4. Викликає update_book_in_db() для оновлення запису в базі.
5. Показує повідомлення про успіх або помилку.
6. Очищає поля вводу.

Рис.4.11.1 – Функція для редагування книги

4.12 Функція для видалення книги

Призначення: видаляє книгу з бази даних за її ID.

Алгоритм роботи:

1. Отримує ID книги.
2. Перевіряє, чи введено ID.
3. Викликає `delete_book_from_db()` для видалення книги.
4. Показує повідомлення про успіх або помилку.
5. Очищає поля вводу.

Рис.4.12.1 – Функція для видалення книги

4.13 Функція очищення форми

Функція `clear_add_book_fields()` потрібна для того, щоб після виконання операцій з книгами всі поля ставали порожніми та були готові до введення нових даних.

Коли ця функція викликається, вона:

1. Очищає поле ID книги → `entry_book_id.delete(0, tk.END)`
2. Очищає поле назви книги → `entry_title.delete(0, tk.END)`
3. Очищає поле автора → `entry_author.delete(0, tk.END)`
4. Очищає поле жанру → `entry_genre.delete(0, tk.END)`
5. Очищає поле року видання → `entry_year.delete(0, tk.END)`
6. Очищає поле рейтингу → `entry_rating.delete(0, tk.END)`

Після додавання, редагування або видалення книги потрібно очистити поля вводу, щоб користувач міг вводити нові дані без необхідності видаляти старі вручну.

Рис.4.13.1 – Функція очищення форми

4.14 Функція для отримання рекомендацій

Призначення: відображає список рекомендованих книг за жанром та рейтингом.

Алгоритм роботи:

1. Отримує вибраний жанр і мінімальний рейтинг.
2. Конвертує рейтинг у число або залишає `None`.
3. Викликає `get_recommendations_from_db()` для отримання списку книг.
4. Відображає список рекомендацій у текстовому полі.

Рис.4.14.1 – Метод `send_reminder`

4.15 Ініціалізація головного вікна

1. Ініціалізація головного вікна:

- `root = tk.Tk()` – створює головне вікно додатка.
- `root.title("Програмний додаток для рекомендацій книг")` – задає заголовок вікна.
- `root.geometry("650x750")` – встановлює розмір вікна (ширина 650 пікселів, висота 750 пікселів).
- `root.config(bg="#e0f7fa")` – задає фон вікна (блідо-блакитний колір).

2. Налаштування стилю кнопок:

- Використовується `ttk.Style()` для зміни стилю кнопок (`TButton`).
- `style.configure("TButton", font=("Arial", 12), padding=10, relief="flat", background="#4CAF50", foreground="red")` – змінює вигляд кнопок (шрифт, колір, розмір).
- `style.map("TButton", background=[('active', '#388E3C')])` – змінює колір кнопки при наведенні.

3. Фрейм для роботи з книгами:

- `frame_add_book=tk.LabelFrame(root,text="Додати/Редагувати/Видалити книгу", ...)` – створює контейнер для віджетів, пов'язаних з додаванням, редагуванням і видаленням книг.
- Додаються поля вводу:
 - o `entry_book_id` – для введення ID книги при редагуванні або видаленні.
 - o `entry_title` – для введення назви книги.
 - o `entry_author` – для введення імені автора.
 - o `entry_genre` – для введення жанру книги.
 - o `entry_year` – для введення року видання.
 - o `entry_rating` – для введення рейтингу книги.
- Додаються кнопки:
 - o Додати книгу (викликає `add_book`).
 - o Редагувати книгу (викликає `edit_book`).

- о Видалити книгу (викликає delete_book).

4. Фрейм для рекомендацій книг:

- frame_recommend = tk.LabelFrame(root, text="Рекомендації", ...) – створює контейнер для рекомендацій.
- combo_genre – випадаючий список жанрів.
- entry_recommend_rating – поле для введення мінімального рейтингу.
- Отримати рекомендації – кнопка, яка викликає recommend_books.
- result_text – текстове поле для відображення списку рекомендованих книг.

5. Запуск головного циклу програми:

- root.mainloop() – запускає головний цикл обробки подій, щоб вікно програми залишалося відкритим та реагувало на дії користувача.

Рис.4.15.1 – Ініціалізація головного вікна

Рис.4.15.2 – Налаштування фреймів та кнопок

Рис.4.15.3 – Налаштування рекомендацій

5. КЕРІВНИЦТВО КОРИСТУВАЧА

Програмний додаток для рекомендацій книг має зручний графічний інтерфейс, що дозволяє користувачам легко керувати базою даних книг та отримувати рекомендації.

Запуск програми:

Щоб розпочати роботу з додатком, необхідно запустити файл book_recommendation_app.py. Після запуску відкриється головне вікно, яке містить основні інструменти для керування книгами та отримання рекомендацій.

Головне вікно містить два основні розділи:

- Блок для управління книгами (додавання, редагування, видалення записів).
- Блок рекомендацій (отримання списку рекомендованих книг).

Рис.5.1 – Загальний вигляд програмного додатку

Функціональні можливості програми:

1) Додавання нової книги:

Щоб додати книгу до бази даних, користувачеві необхідно:

1. Ввести назву книги у відповідне поле.
2. Вказати автора книги.
3. Обрати або ввести жанр книги.
4. Ввести рік видання книги.
5. Вказати рейтинг (оцінку книги за шкалою, наприклад, від 1 до 10).
6. Натиснути кнопку "Додати книгу".

Після натискання кнопки книга буде збережена в базі даних, і користувач отримає повідомлення про успішне додавання.

Рис.5.2 – Додавання нової книги

Рис.5.3 – Повідомлення про успішне додавання нової книги до бази даних

2) Редагування книги:

Щоб змінити інформацію про вже існуючу книгу, необхідно:

1. Ввести ID книги, яку потрібно відредагувати (ID можна знайти у списку або при перегляді даних).

2. Ввести нові значення для змінюваних полів (назва, автор, жанр, рік видання, рейтинг).
3. Натиснути кнопку "Редагувати книгу".

Якщо книга з вказаним ID є в базі даних, програма оновить її дані та покаже повідомлення про успішне редагування. Якщо ID не знайдено, з'явиться відповідне повідомлення про помилку.

Рис.5.4 – Редагування існуючої книги

Рис.5.5 – Повідомлення про успішне оновлення книги

3) Видалення книги:

Для видалення книги з бази даних потрібно:

1. Ввести ID книги, яку потрібно видалити.
2. Натиснути кнопку "Видалити книгу".

Програма видалить запис із бази даних та виведе повідомлення про успішне видалення. Якщо книга з таким ID відсутня, програма повідомить про помилку.

Matched Source

Similarity 15%

Title: Python. Програми з графічним інтерфейсом — урок. Інформатика, 8 клас.

```
root = Tk() root.title('Графічна програма') # Встановити заголовок вікна root.geometry('250x150') # Встановити розміри вікна (в пікселях) root.mainloop() # команда запуску головного циклу обробки подій
```

<https://www.miyklas.com.ua/p/informatica/8-klas/algoritmi-ta-programi-394917/programi-z-grafichnim-interfeisom-395197/re-6038e80b-3201-49cd-a207-7518a6aab2c0>
