🎮 Movie Search Web App (Flask + MySQL)

🔖 Project Description

This project is a web application built with **Flask** that allows users to search for movies in the **Sakila** database using filters such as **keyword**, **genre**, and **release year**. It records successful user queries in a separate database and displays the **Top 10 most popular searches**.

It demonstrates work with:

- SQL joins and filtering
- Web development using Flask
- Data logging and basic analytics

🔧 Application Structure

- Flask app initialization and route handling
- Route `/` for movie search
- Route `/top-queries` for viewing top 10 most frequent queries
- A logging function that stores successful search queries into a MySQL table
- Auto-launch in browser on application start

✅ This project was created as the **final assignment** for the course *Python Fundamentals*.

🚀 Features

- 🔎 Movie search by keyword, genre, and year
- 💃 SQL-based filtering using joins in the `sakila` database
- 📟 Logs successful search queries to a user activity database
- 📊 Displays the 10 most frequently searched combinations

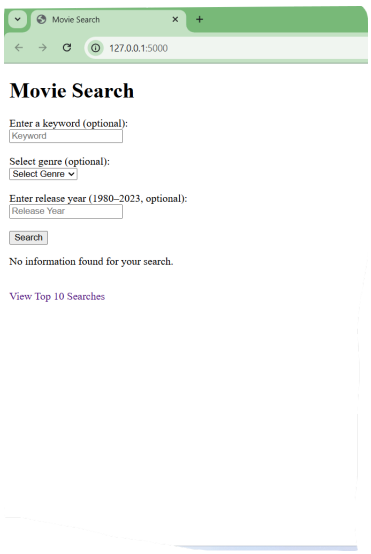
- 🌐 Clean web interface using Flask and Jinja2 templates
- 🌍 Auto-launch in browser on startup

🛠️ Tech Stack

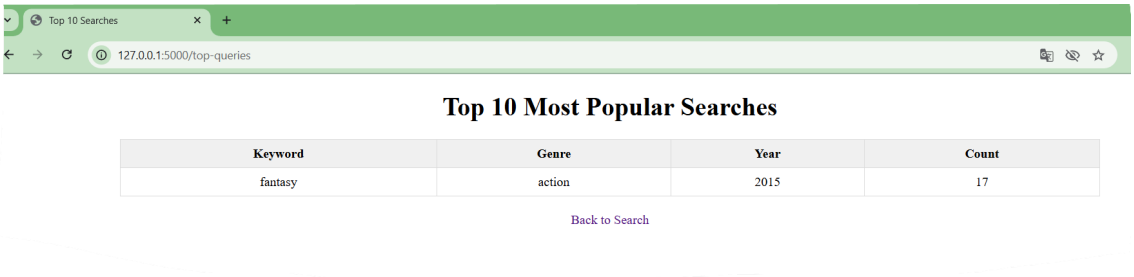
- Python 3.x
- Flask
- MySQL (Sakila dataset)
- HTML/CSS (Jinja2)
- Optional: dotenv for local config

🗄️ Screenshots

Search Page



Top 10 User Queries



⚙️ How to Run Locally

Clone the repository:

```
git clone https://github.com/your-username/movie-search-app.git
cd movie-search-app
```

1.

Install the dependencies:

```
pip install -r requirements.txt
```

2.
3. Configure the database connection:
    - Edit `DBMC.py` and replace the placeholder values (`your_host`, `your_user`, `your_password`) with your actual MySQL credentials.

Start the application:

```
python app.py
```

4. The app will open at: http://127.0.0.1:5000

📂 Project Structure

```
├── app.py              # Main Flask application
├── DBMC.py             # Database connection manager
├── db_film_templates/  # HTML templates (search + top queries)
```

```
├── static/              # (Optional) CSS / images
├── screenshots/         # Images for README
├── Описание проекта.odt   # Full project description in Russian
└── requirements.txt       # Python dependencies
```

📌 Notes

- This app uses the sample **sakila** film database.
- Logging is stored in a second database (e.g. `Yuzhnyi`) in the `search_queries` table.
- If the database is not connected, a placeholder or local SQLite version may be used for demo purposes.

📜 Full Technical Description

The full project structure, SQL queries, and code explanations are documented in Russian in:

📄 Описание проекта.odt

👤 Author

Yevhen Yuzhnyi
LinkedIn

📄 License