CS 5200 Project
City Living Quality Evaluation System

Peiyuan Sun | Xinzhi Zhang

README

Step 1: Required Python Modules

pip install flask
pip install pymysql

Step 2: Import the Database

Download the self-contained dump file and open it in MySQL Workbench. Execute the file and ensure `city_evaluation` database has been created.

Note: Once the import has completed, the tables in `city_evaluation` should contain schemas, data and procedures.

Step 3: Run the Project

Once you have cloned this repository and navigated to the directory in your terminal, simply run

python run.py

Enter your MySQL username and password,

MySQL Username: <username>
MySQL Password: <password>

and then navigate to `localhost:9494` in your browser.

Project Description

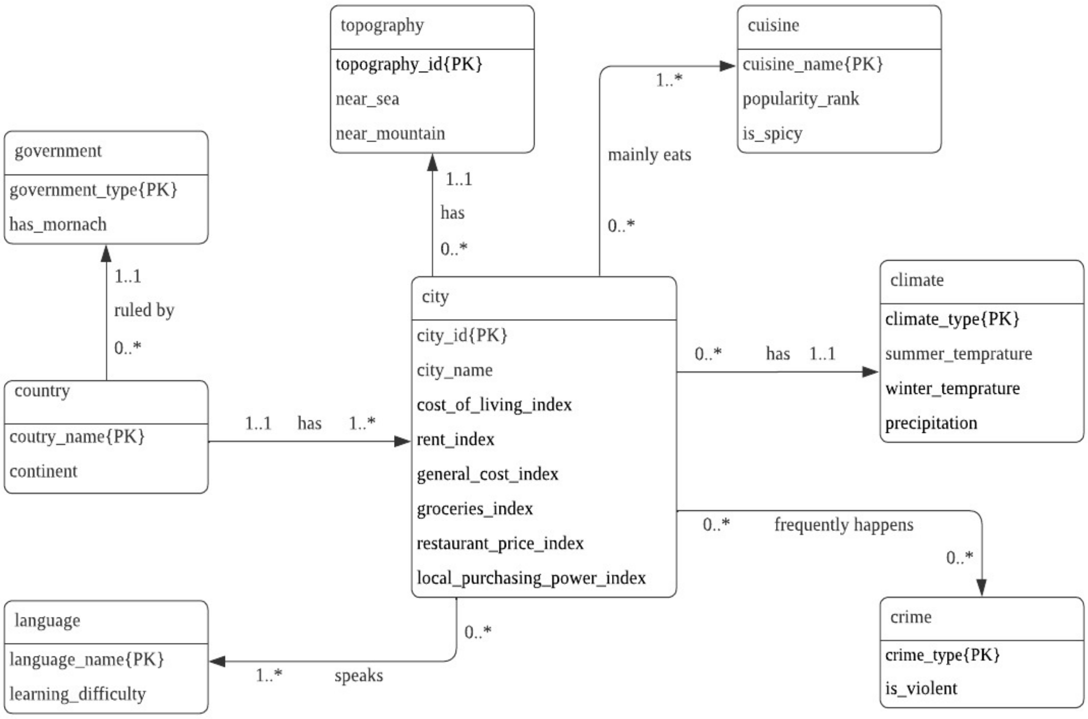
This system provides a platform to make reference for users to settle down in terms of topography, language, climate, cost of living and crime rate, it will also function as an excellent user travel guide. The users can log in and do CRUD(create, read, update, delete) operations to the city's data.

The indexes of different cities are calculated by setting the New York's to 100, and the data is aquired from numbeo.

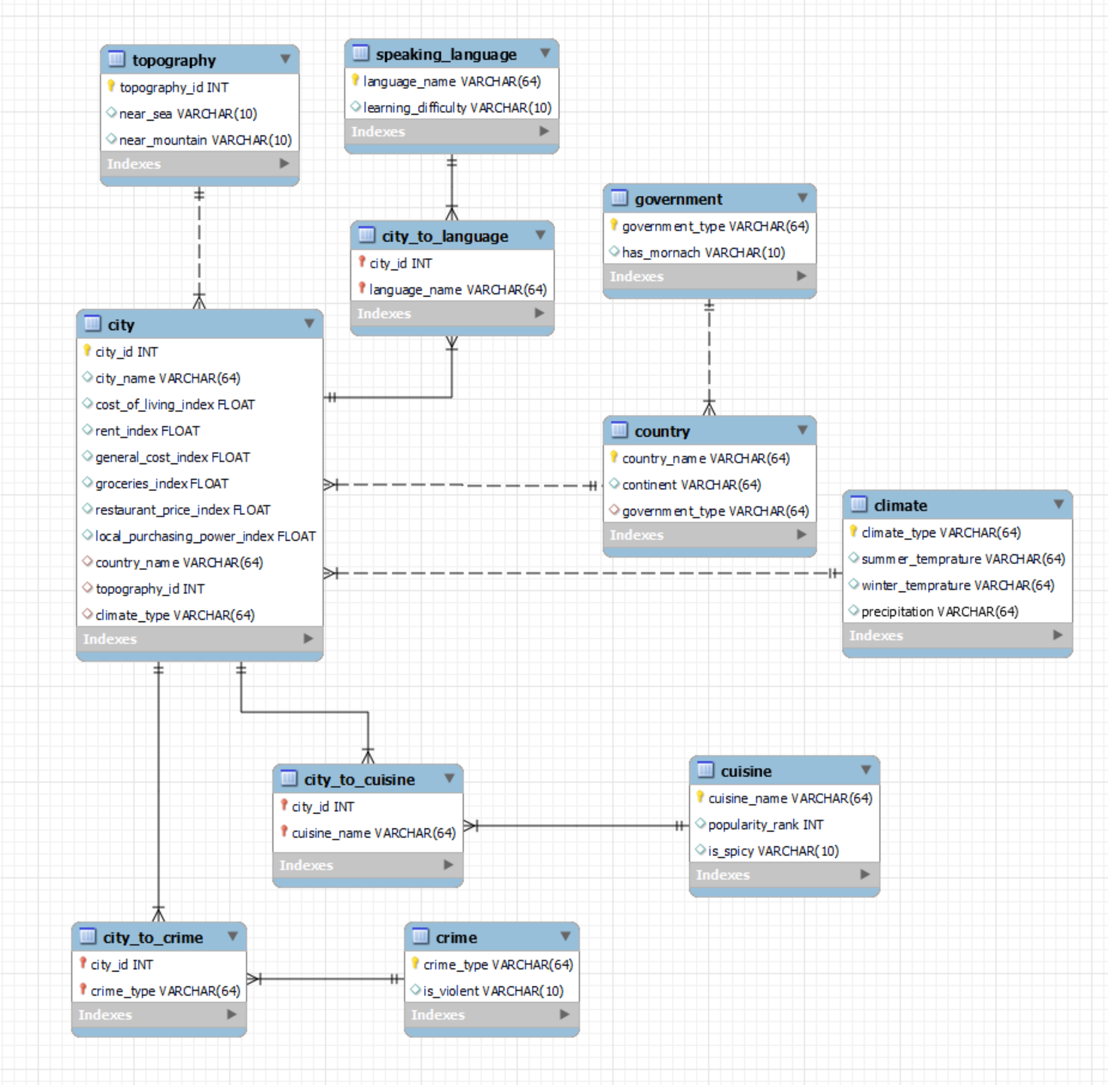
Technical Specifications

Database: MySql
Back End: Python
Front End: HTML, CSS, Javascript
Libraries: flask, pymysql, render_template, request, jsonify, url_for

UML Diagram

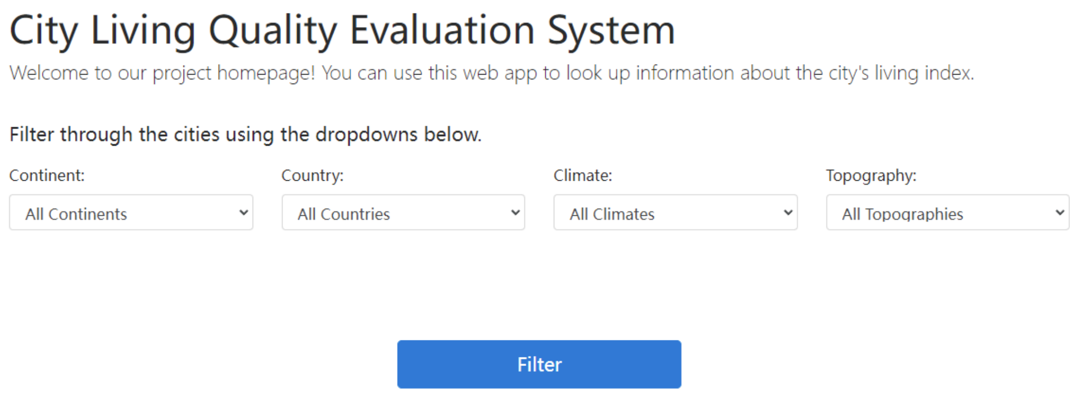

topography
topography_id{PK}
near_sea
near_mountain
cuisine
cuisine_name{PK}
popularity_rank
is_spicy
1..*
mainly eats
0..*
government
government_type{PK}
has_mornach
1..1
has
0..*
1..1
ruled by
0..*
city
city_id{PK}
city_name
cost_of_living_index
rent_index
general_cost_index
groceries_index
restaurant_price_index
local_purchasing_power_index
climate
climate_type{PK}
summer_temprature
winter_temprature
precipitation
0..*
has
1..1
country
coutry_name{PK}
continent
1..1
has
1..*
0..*
frequently happens
0..*
crime
crime_type{PK}
is_violent
language
language_name{PK}
learning_difficulty
0..*
1..*
speaks

Logic Design



User Flow

1. Homepage:



City Living Quality Evaluation System

Welcome to our project homepage! You can use this web app to look up information about the city's living index.

Filter through the cities using the dropdowns below.

Continent: All Continents

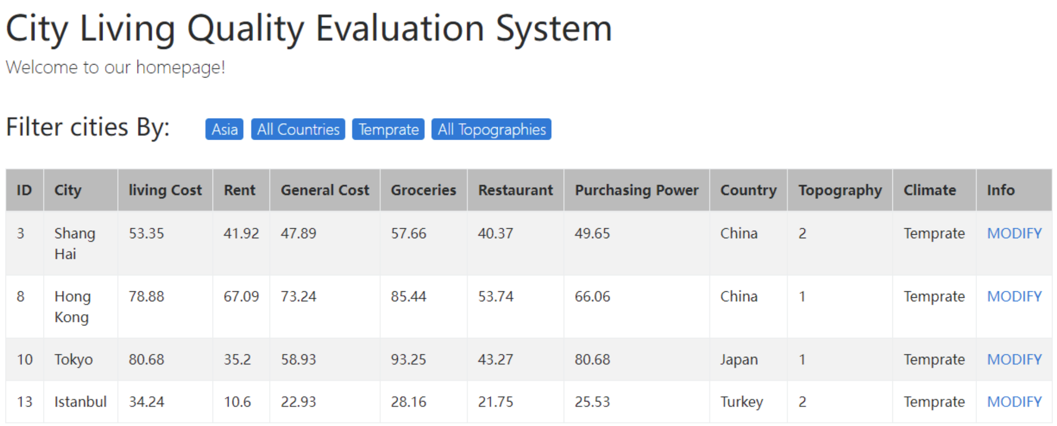
Country: All Countries

Climate: All Climates

Topography: All Topographies

Users can read the data of cities by selecting 4 filters, continent, country, climate and topography, if they don’t select any filter and click the filter buttom, the web page will output the whole city table.

Result(select continent: Asia and climate: temprate):



City Living Quality Evaluation System

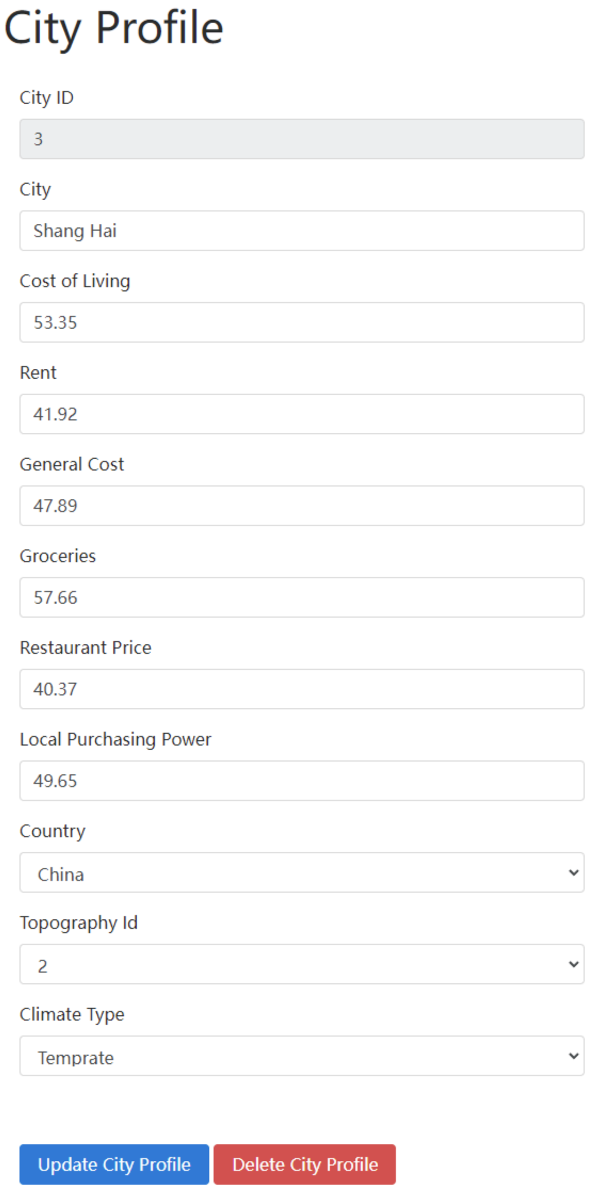
Welcome to our homepage!

Filter cities By:

ID	City	living Cost	Rent	General Cost	Groceries	Restaurant	Purchasing Power	Country	Topography	Climate	Info
3	Shang Hai	53.35	41.92	47.89	57.66	40.37	49.65	China	2	Temprate	MODIFY
8	Hong Kong	78.88	67.09	73.24	85.44	53.74	66.06	China	1	Temprate	MODIFY
10	Tokyo	80.68	35.2	58.93	93.25	43.27	80.68	Japan	1	Temprate	MODIFY
13	Istanbul	34.24	10.6	22.93	28.16	21.75	25.53	Turkey	2	Temprate	MODIFY

2. Update and delete

By clicking the MODIFY buttom on the right, the users can update or delete the city data they choose.

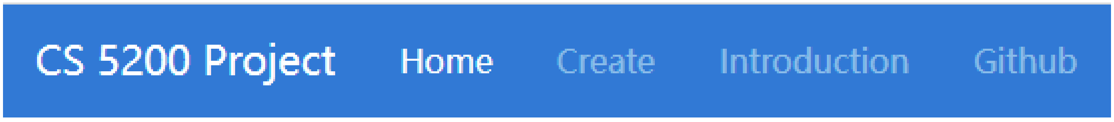


City Profile

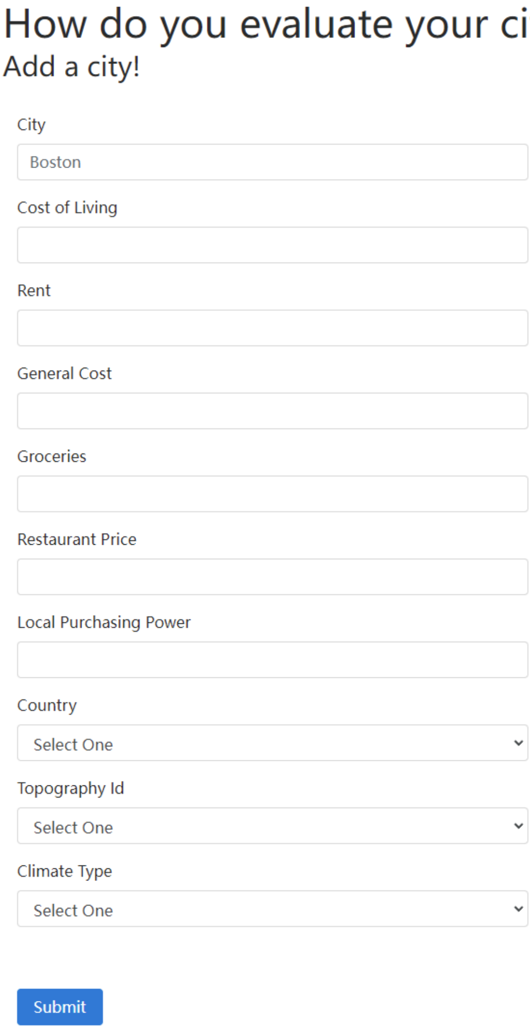
City ID	3
City	Shang Hai
Cost of Living	53.35
Rent	41.92
General Cost	47.89
Groceries	57.66
Restaurant Price	40.37
Local Purchasing Power	49.65
Country	China
Topography Id	2
Climate Type	Temprate

3. Create

By clicking the Create on the headline, we can get into the create page.



Then, users can type in the data of city that they want to add, and the city_id will be automatically generated.



How do you evaluate your cities?

Add a city!

City

Boston

Cost of Living

Rent

General Cost

Groceries

Restaurant Price

Local Purchasing Power

Country

Select One

Topography Id

Select One

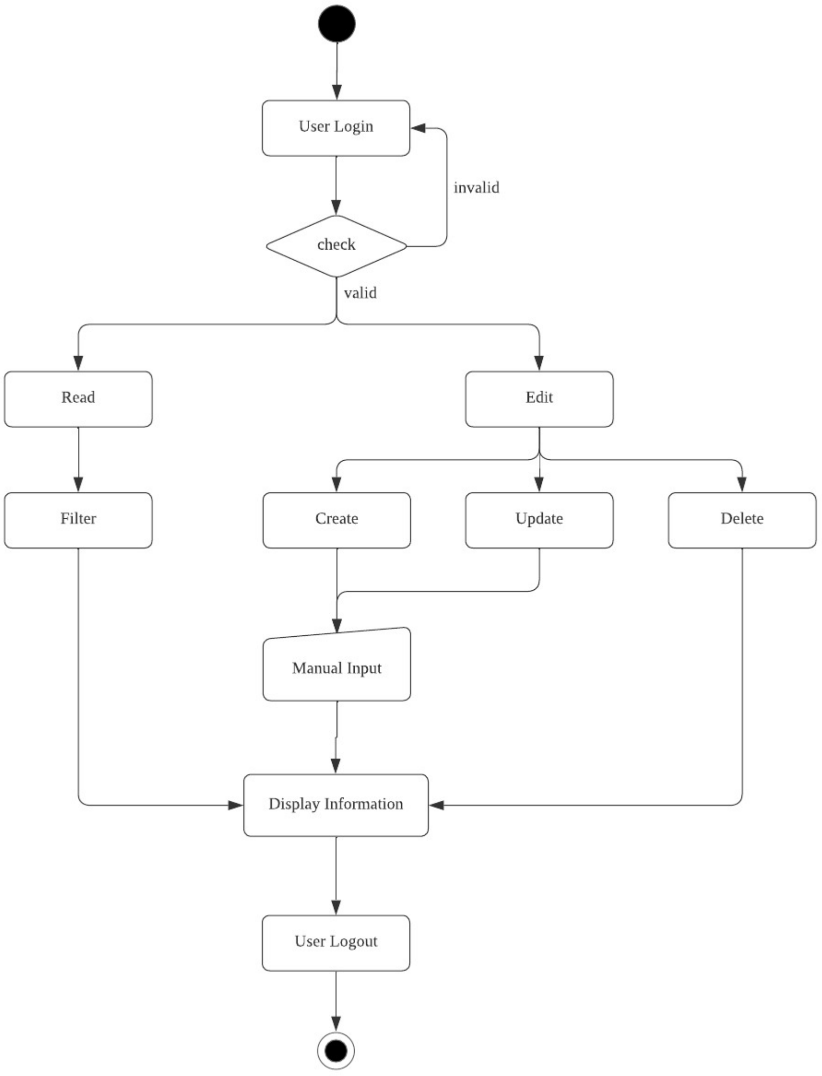
Climate Type

Select One

4. Introduction and github

On the headline, there are also two buttoms: introduction and github, users can get the description of this project, technical specifications, UML graph and flowchart by clicking introduction, and github buttom navigates to our github repositories of this project, users can check code and further information in it.

5. Flowchart



Lesson we learned

1. Technical expertise gained

Used MySQL workbench to design the database, write procedures that realize CRUD operations, and error handling;
Used Python libraries pymysql and flask to connect MySQL with host language;
Used HTML, CSS and Javasrcript to construct the website;

2. Insights

Created seperated tables to describe many-to-many relationship in MySQL databse;
Realized four filters to select cities in home page;
Realized auto increment in city_id when creating a new record;
Made clear link 'MODIFY' to update and delete the city profile;

3. Alternative design to the project

We can also use commands to reach to the CRUD operations to the cities, for example, we can just type 'localhost:9494/profile/<city_id>' to get into the page of modifying the city data. However, this method is not simple and intuitive enough.

4. Code lessons

In our process of coding, it always happened that we obfuscated the name of SQL attributes, SQL tables, python variables and HTML text, and it cost us a lot of time to proofread these names in different platforms. It taught us to have a good naming notifications and always add notes to our codes.

Future work

1. Add functions of ranking and comparing.
2. Add more cities to our database.
3. Beautify our webpage with GUI.
4. Add a world map which can visualize the indexes.