Minh Vu

CSCI 220

Final Project Report

I. Main Idea:

I will create a database schema from topic 2: Railway System Database. The main user of this database will be the employees of the railway company. They will use this database to keep track of all the essential information like train schedules, employees, customers, and ticket purchases.

II. User Stories:

Since this is just a demo of the application, users will use the terminal to interact with the database.

III. User Cases:

Just like the last question in the last quiz, users can do a handful of commands:

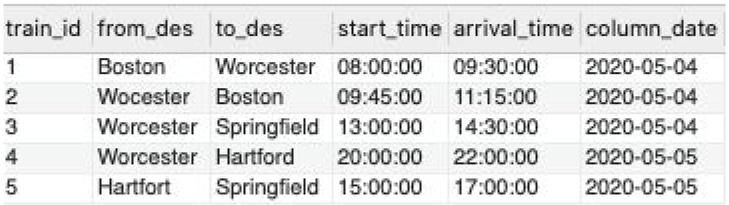
1. L – List: lists all records in a specified table. The user will be asked for the name of the table.
2. A – Add: add a record to a specified table. The user will be asked for the name of the table and the value of each column.
3. D – Delete: deletes a record in a specified table. The user will be asked for the name of the table.
4. S – Search: search from a specified table
5. V – View: create a new view
6. T – Trigger: create a new trigger
7. X – Exit: exit application

The original idea is that an employee can use this application to store and manipulate the data of the railway company. Every day, there are thousands of tickets get purchased, hundreds of new train schedules, and employees working, so it is important to store those data safely.

IV. Data Requirement:

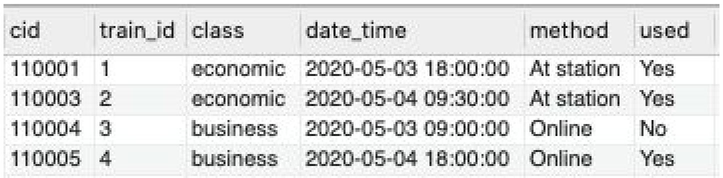
Here is the sample data I came up for the project:

Train_schedule:



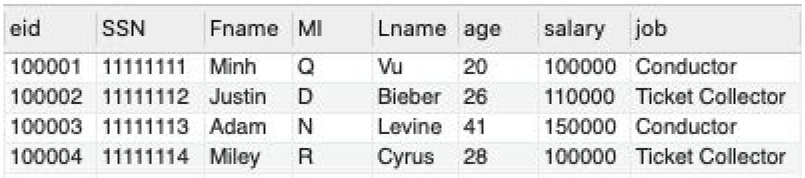
train_id	from_des	to_des	start_time	arrival_time	column_date
1	Boston	Worcester	08:00:00	09:30:00	2020-05-04
2	Wocester	Boston	09:45:00	11:15:00	2020-05-04
3	Worcester	Springfield	13:00:00	14:30:00	2020-05-04
4	Worcester	Hartford	20:00:00	22:00:00	2020-05-05
5	Hartfort	Springfield	15:00:00	17:00:00	2020-05-05

Ticket_purchase:

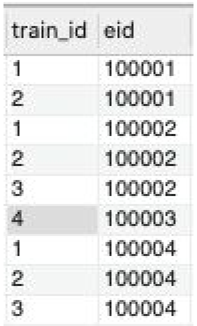


cid	train_id	class	date_time	method	used
110001	1	economic	2020-05-03 18:00:00	At station	Yes
110003	2	economic	2020-05-04 09:30:00	At station	Yes
110004	3	business	2020-05-03 09:00:00	Online	No
110005	4	business	2020-05-04 18:00:00	Online	Yes

Employee and Work_on:

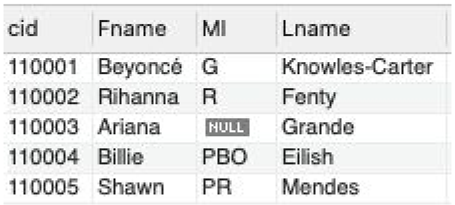


eid	SSN	Fname	MI	Lname	age	salary	job
100001	11111111	Minh	Q	Vu	20	100000	Conductor
100002	11111112	Justin	D	Bieber	26	110000	Ticket Collector
100003	11111113	Adam	N	Levine	41	150000	Conductor
100004	11111114	Miley	R	Cyrus	28	100000	Ticket Collector



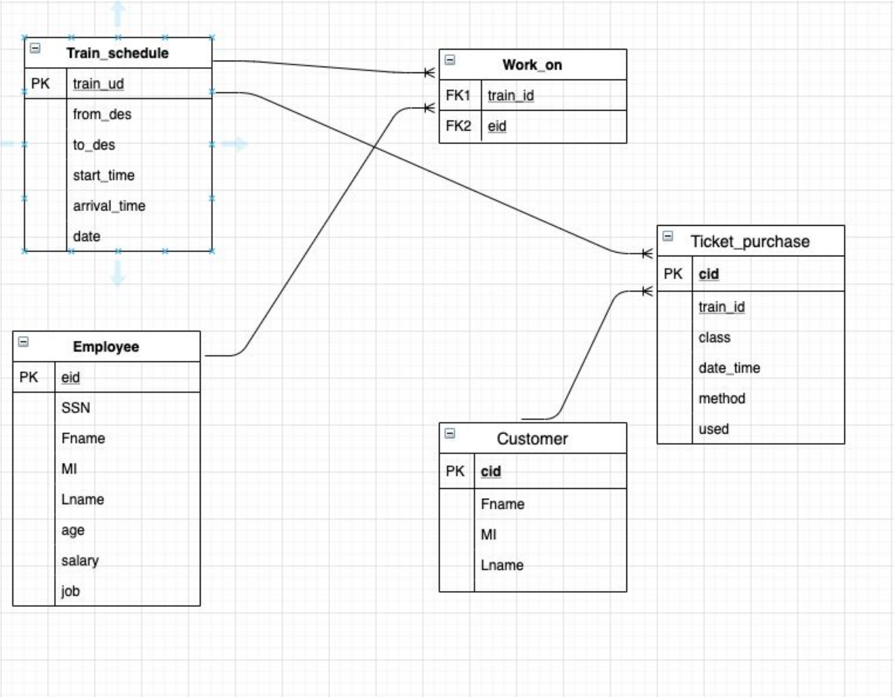
train_id	eid
1	100001
2	100001
1	100002
2	100002
3	100002
4	100003
1	100004
2	100004
3	100004

Customer:



cid	Fname	MI	Lname
110001	Beyoncé	G	Knowles-Carter
110002	Rihanna	R	Fenty
110003	Ariana	NULL	Grande
110004	Billie	PBO	Eilish
110005	Shawn	PR	Mendes

Here is the relational model:



V. Demo:

The application is pretty straightforward. What I learned during this project is that it can be very tricky to design a good database. We will have to use a lot of Algorithms and Software Engineering knowledge to design a user-friendly application. For example, when a user wants to add a record to a table, we cannot know for sure which table the user will pick. Right now, since my demo database only has 5 tables, I chose the brute force method, which is to have an if-else statement for each case. This is definitely not ideal since it takes a lot of space and time, especially if we have more tables. Using a good algorithm will not only fix this problem but it will also make our code more readable.

VI. Lessons:

I think the biggest thing I learned from this project is how to integrate MySQL with another programming language (Java, Python). I also learned that there is a big difference between executeQuery() and executeUpdate().

If I were to do this project again, the first thing I would is to get a partner. The coding part of a database application was actually much longer and harder than I thought so with another person, we could have divided the work more evenly. I could not really finish the project because I underestimated the complicity of the project.

I would have also focused more on the user interface if I were to do this project again. The demo was solely for programmers so the application was not very applicable.

If I have more time, I will definitely focus on more advanced algorithms to make the code more efficient and more readable.