DBMS Project

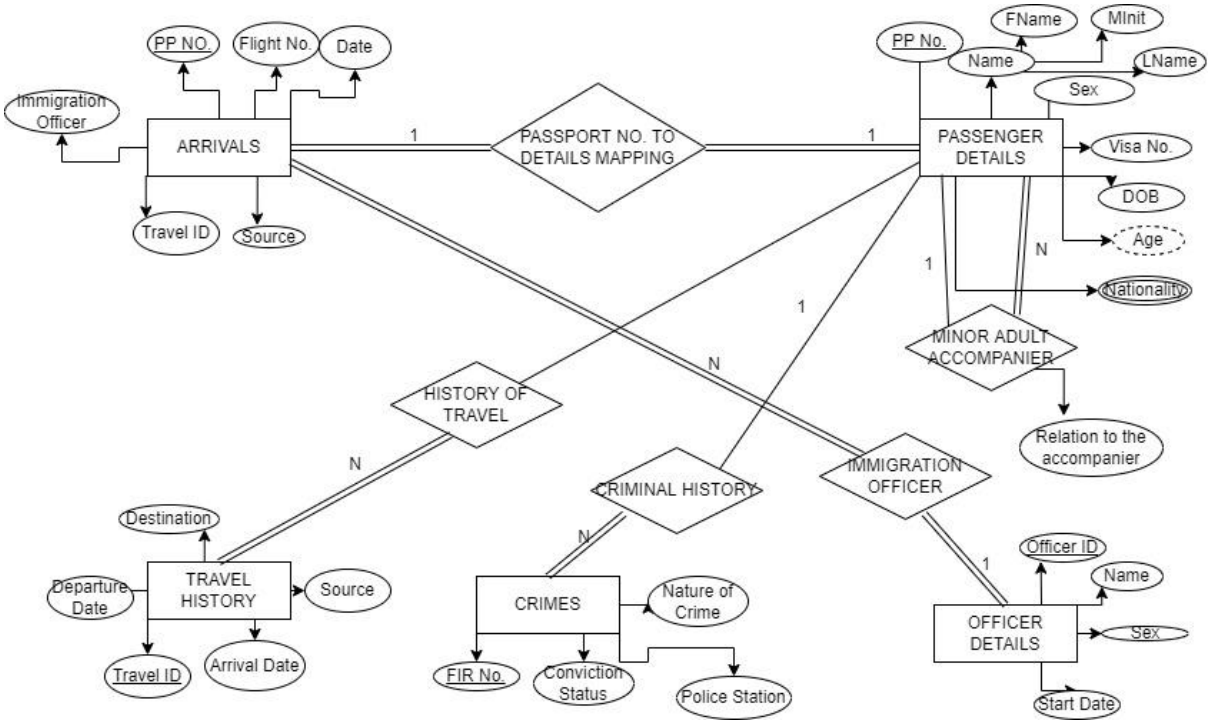
Airport Immigration Management System

NAME	***ATHARVA S GADAD***
SRN	***PES1UG20CS088***
SEM	***V***
SEC	***B***

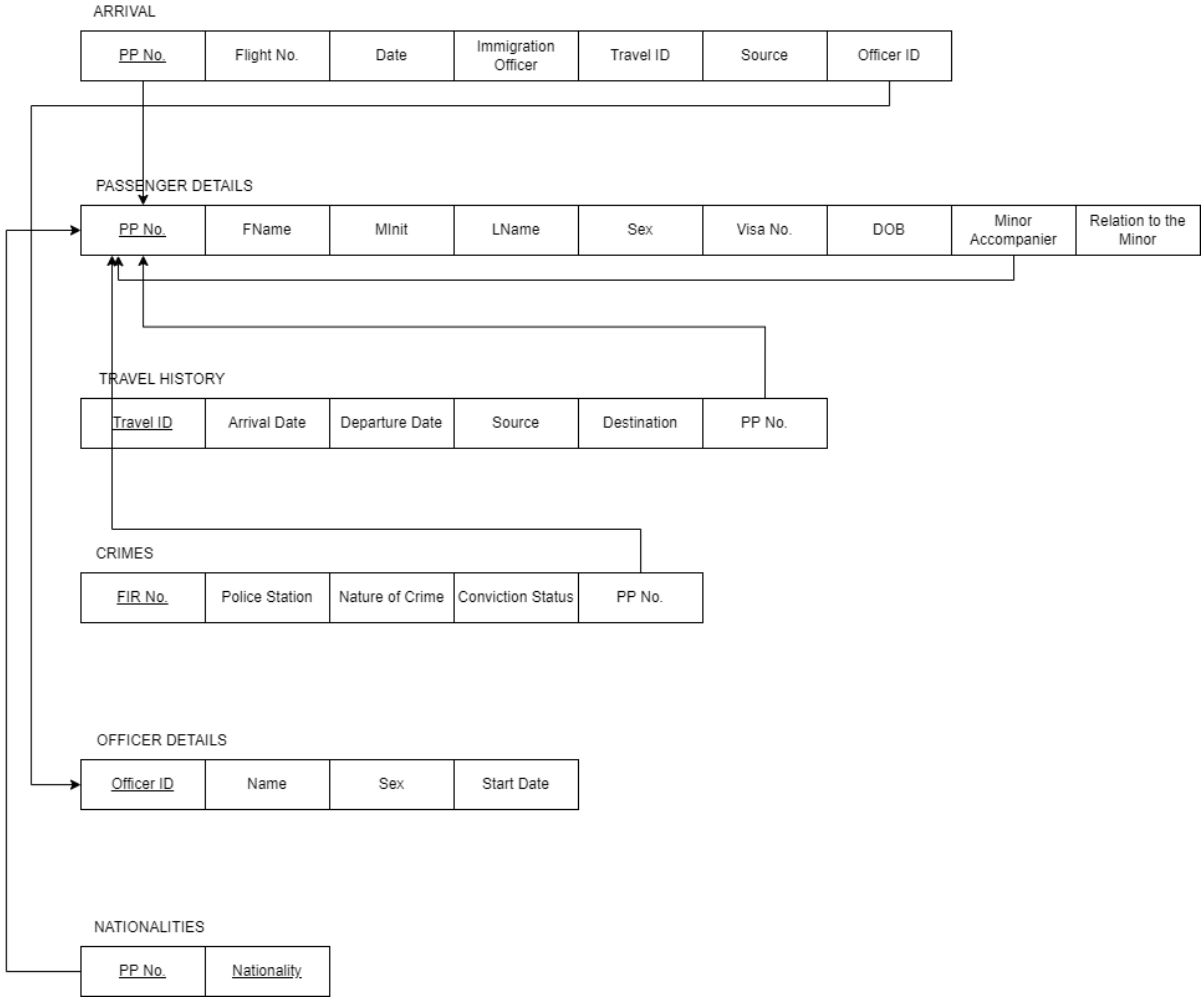
Project Description and Scope of Project

The objective of the project was to create an Airport Immigration Management System. This system keeps track of all arrivals into the country, their travel history, i.e., their previous visits to India, the details of all passengers and their criminal histories. The motive of the project was to create a system to make the process of background checks more streamlined and efficient, allowing immigration officers to make informed decisions. The application also keeps track of the various immigration officers and their associated details.

ER Diagram


PP NO.
Flight No.
Date
Immigration Officer
ARRIVALS
Travel ID
Source
1
PASSPORT NO. TO DETAILS MAPPING
1
PP No.
FName
MInit
Name
LName
Sex
PASSENGER DETAILS
Visa No.
DOB
Age
Nationality
1
N
1
N
MINOR ADULT ACCOMPANIER
Relation to the accompanier
HISTORY OF TRAVEL
N
CRIMINAL HISTORY
IMMIGRATION OFFICER
Destination
Departure Date
TRAVEL HISTORY
Source
Travel ID
Arrival Date
N
CRIMES
Nature of Crime
FIR No.
Conviction Status
Police Station
1
Officer ID
Name
OFFICER DETAILS
Sex
Start Date

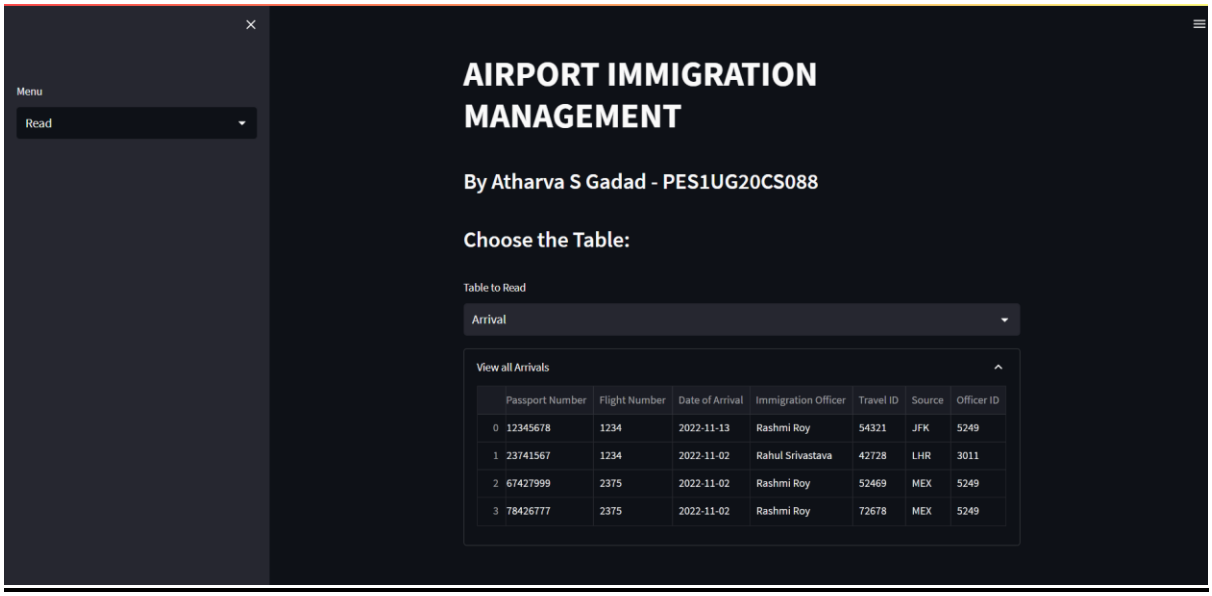
Relational Schema



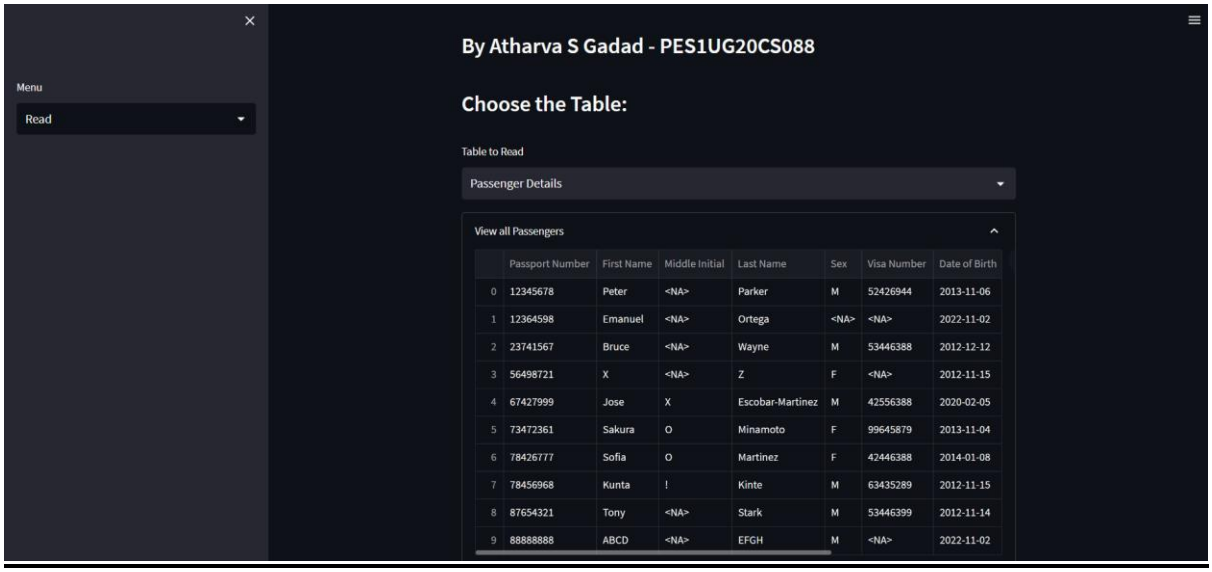
DDL Statements

```
create table IF NOT EXISTS passenger_details(pp_no varchar(8) not null primary
key, fname varchar(255) not null, minit varchar(1),
lname varchar(255) not null, sex varchar(1), visa_no varchar(8), dob date,
minor_accompanier varchar(8), relation_to_minor varchar(255), foreign key
(minor_accompanier)
references passenger_details(pp_no) on delete cascade);

create table IF NOT EXISTS travel_history(travel_id varchar(5) primary key not
null, date_of_arrival date not null,
date_of_departure date not null, source varchar(255) not null, destination
varchar(255) not null,
pp_no varchar(8) not null, foreign key (pp_no) references
passenger_details(pp_no) on delete cascade);

create table IF NOT EXISTS crime (fir_no varchar(6) not null primary key,
police_station varchar(255), nature_of_crime varchar(255) not null,
conviction_status varchar(255) not null, pp_no varchar(8) not null, foreign
key (pp_no) references passenger_details(pp_no) on delete cascade);

create table IF NOT EXISTS officer_details(officer_id varchar(4) not null
primary key, name varchar(255) not null, sex varchar(1),
join_date date not null);

create table IF NOT EXISTS arrival(pp_no varchar(8) not null primary key,
flight_no varchar(4), date_of_arrival date,
immigration_officer varchar(255), travel_id varchar(5), source varchar(255),
officer_id varchar(4),
foreign key (pp_no) references passenger_details(pp_no) on delete cascade,
foreign key (officer_id) references officer_details(officer_id) on delete
cascade);

create table IF NOT EXISTS nationalities(pp_no varchar(8) not null,
nationality varchar(100) not null,
foreign key (pp_no) references passenger_details(pp_no) on delete cascade,
primary key(pp_no, nationality));
```

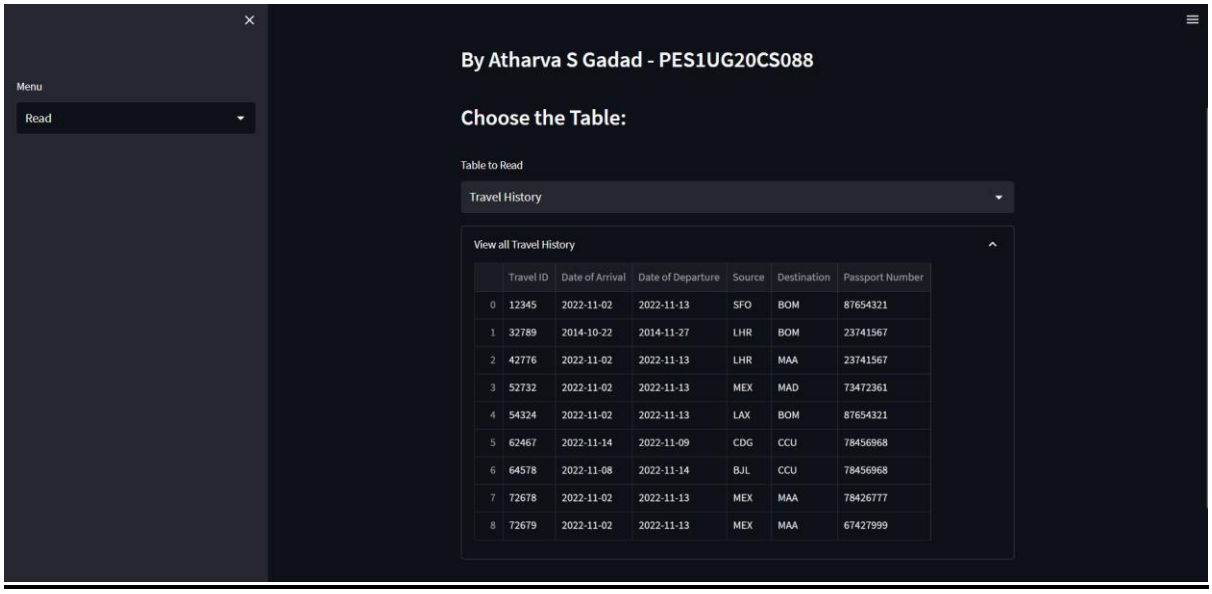
Populating Database:



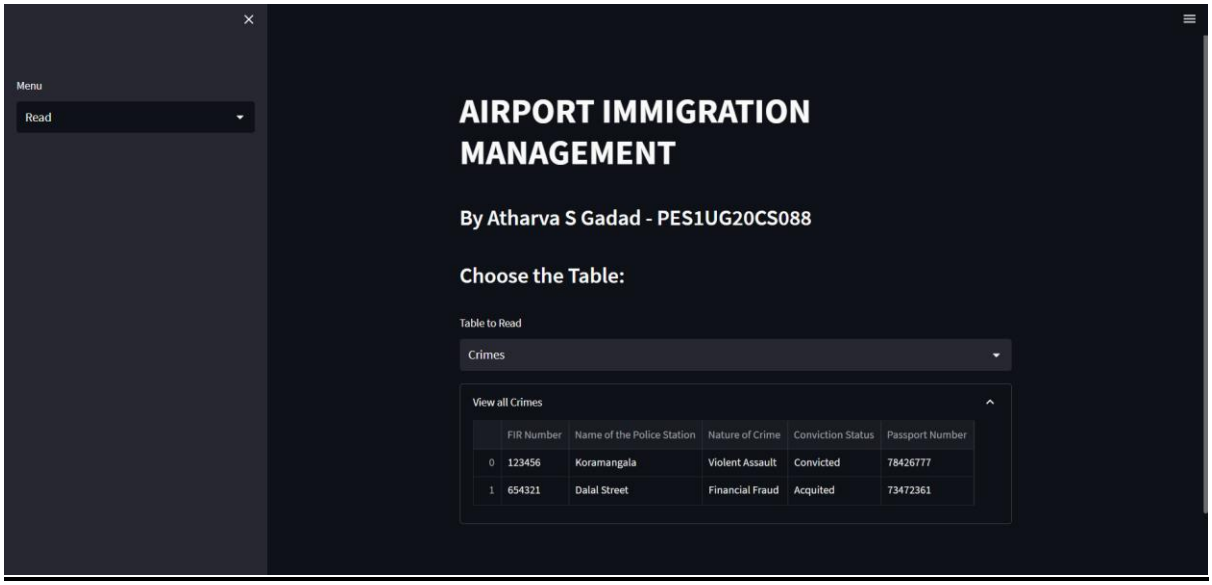
	Passport Number	Flight Number	Date of Arrival	Immigration Officer	Travel ID	Source	Officer ID
0	12345678	1234	2022-11-13	Rashmi Roy	54321	JFK	5249
1	23741567	1234	2022-11-02	Rahul Srivastava	42728	LHR	3011
2	67427999	2375	2022-11-02	Rashmi Roy	52469	MEX	5249
3	78426777	2375	2022-11-02	Rashmi Roy	72678	MEX	5249

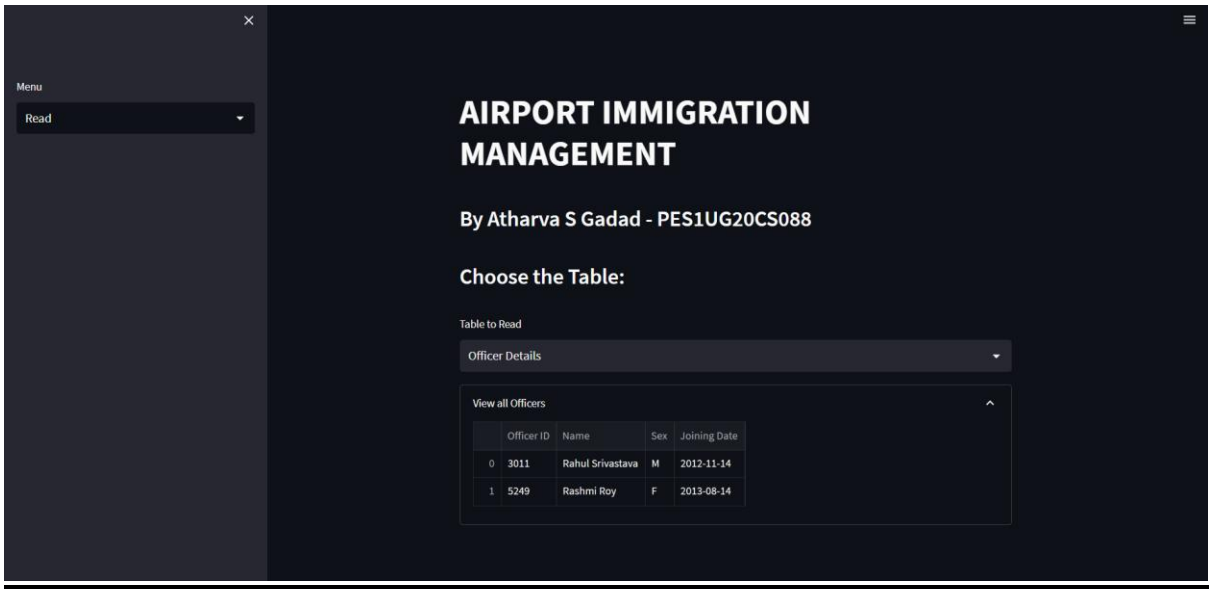


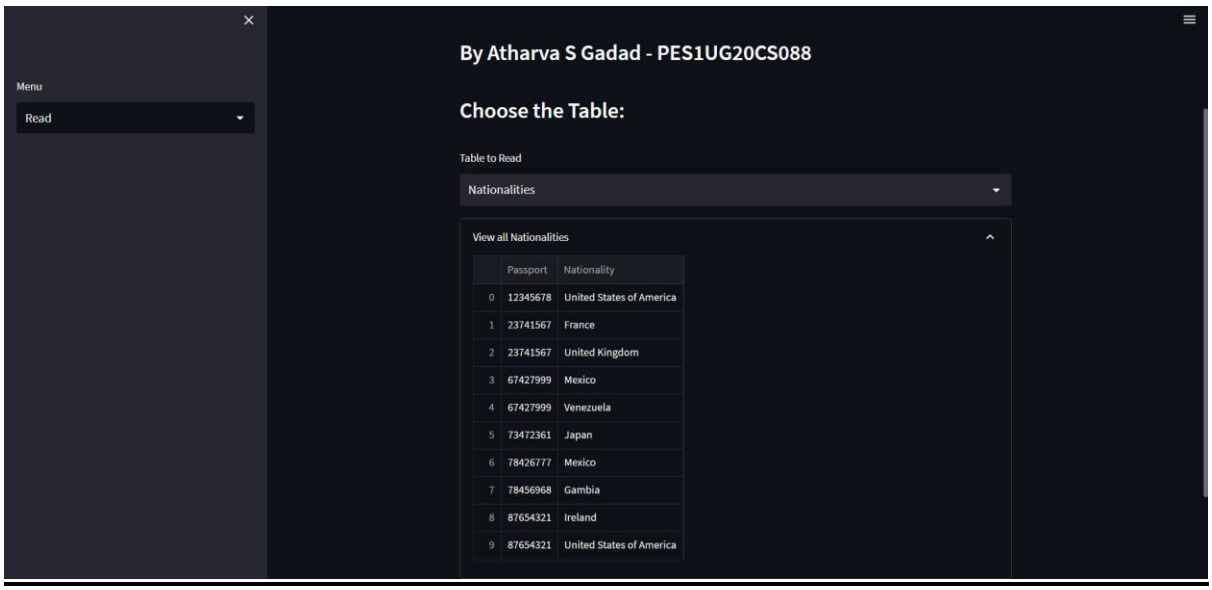
	Passport Number	First Name	Middle Initial	Last Name	Sex	Visa Number	Date of Birth
0	12345678	Peter	<NA>	Parker	M	52426944	2013-11-06
1	12364598	Emanuel	<NA>	Ortega	<NA>	<NA>	2022-11-02
2	23741567	Bruce	<NA>	Wayne	M	53446388	2012-12-12
3	56498721	X	<NA>	Z	F	<NA>	2012-11-15
4	67427999	Jose	X	Escobar-Martinez	M	42556388	2020-02-05
5	73472361	Sakura	O	Minamoto	F	99645879	2013-11-04
6	78426777	Sofia	O	Martinez	F	42446388	2014-01-08
7	78456968	Kunta	!	Kinte	M	63435289	2012-11-15
8	87654321	Tony	<NA>	Stark	M	53446399	2012-11-14
9	88888888	ABCD	<NA>	EFGH	M	<NA>	2022-11-02



	Travel ID	Date of Arrival	Date of Departure	Source	Destination	Passport Number
0	12345	2022-11-02	2022-11-13	SFO	BOM	87654321
1	32789	2014-10-22	2014-11-27	LHR	BOM	23741567
2	42776	2022-11-02	2022-11-13	LHR	MAA	23741567
3	52732	2022-11-02	2022-11-13	MEX	MAD	73472361
4	54324	2022-11-02	2022-11-13	LAX	BOM	87654321
5	62467	2022-11-14	2022-11-09	CDG	CCU	78456968
6	64578	2022-11-08	2022-11-14	BJL	CCU	78456968
7	72678	2022-11-02	2022-11-13	MEX	MAA	78426777
8	72679	2022-11-02	2022-11-13	MEX	MAA	67427999


Menu
Read
AIRPORT IMMIGRATION MANAGEMENT
By Atharva S Gadad - PES1UG20CS088
Choose the Table:
Table to Read
Crimes
View all Crimes
FIR Number Name of the Police Station Nature of Crime Conviction Status Passport Number
0 123456 Koramangala Violent Assault Convicted 78426777
1 654321 Dalal Street Financial Fraud Acquited 73472361

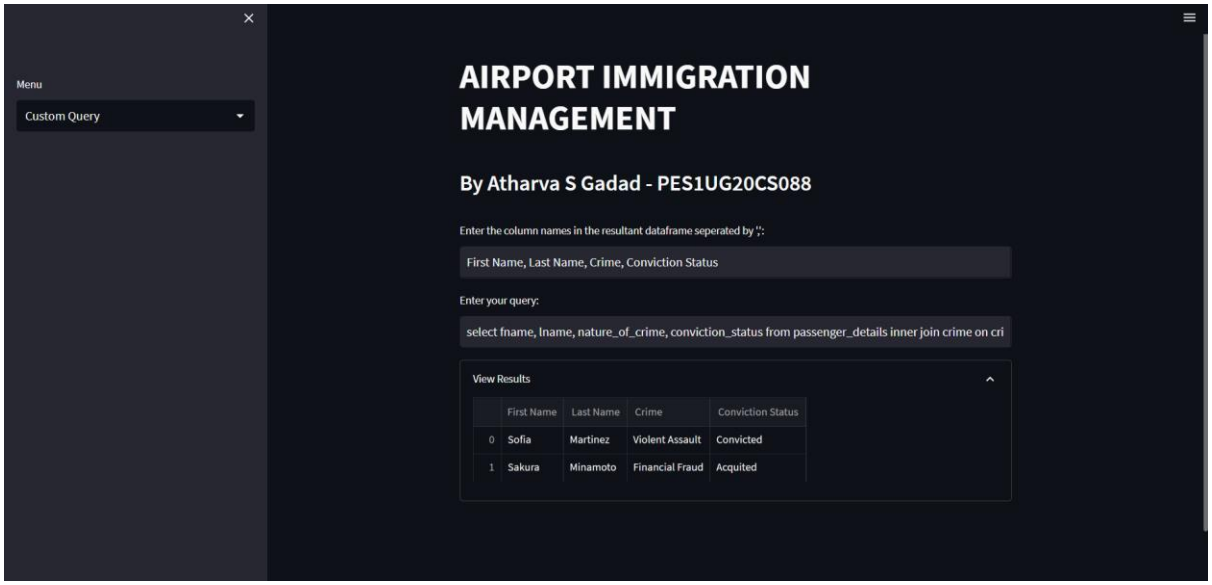

Menu
Read
AIRPORT IMMIGRATION MANAGEMENT
By Atharva S Gadad - PES1UG20CS088
Choose the Table:
Table to Read
Officer Details
View all Officers
Officer ID Name Sex Joining Date
0 3011 Rahul Srivastava M 2012-11-14
1 5249 Rashmi Roy F 2013-08-14


Menu
Read
By Atharva S Gadad - PES1UG20CS088
Choose the Table:
Table to Read
Nationalities
View all Nationalities
Passport Nationality
0 12345678 United States of America
1 23741567 France
2 23741567 United Kingdom
3 67427999 Mexico
4 67427999 Venezuela
5 73472361 Japan
6 78426777 Mexico
7 78456968 Gambia
8 87654321 Ireland
9 87654321 United States of America

Join Queries

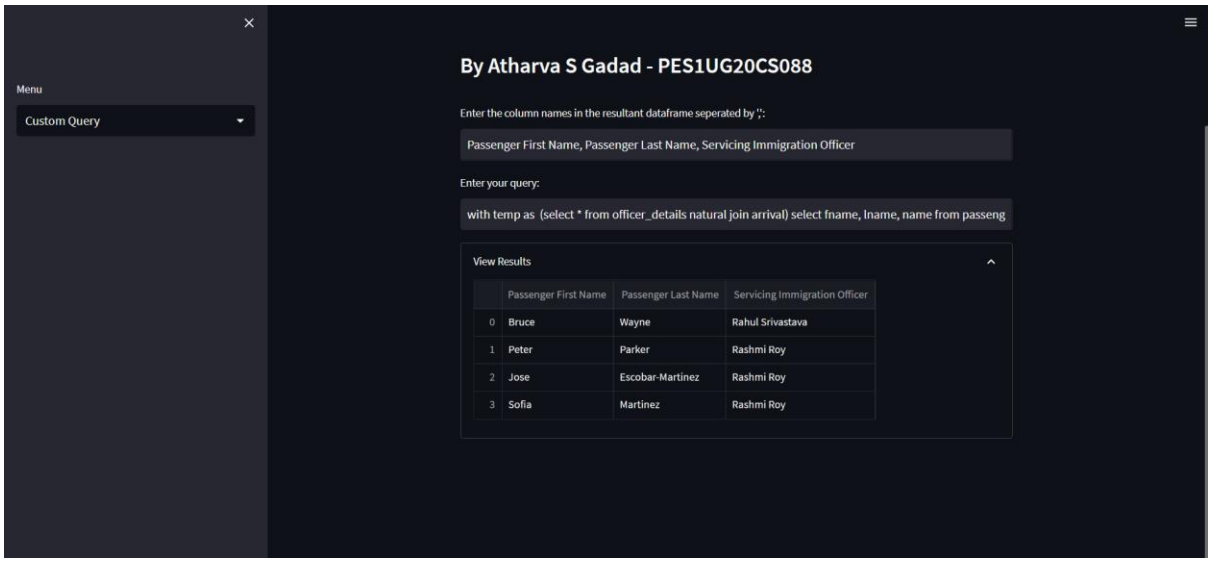
1) Selecting the first name, last name, nature of crime, and conviction status of all criminals

```
select fname, lname, nature_of_crime, conviction_status from
passenger_details inner join crime on
crime.pp_no=passenger_details.pp_no;
```



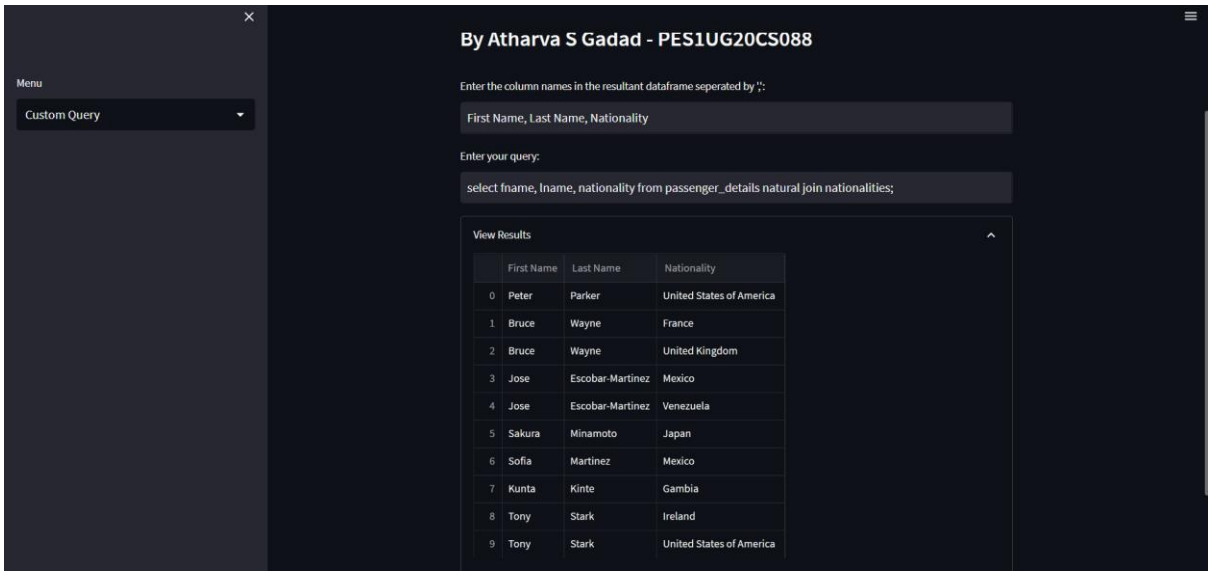
2) Selecting each arrival and servicing immigration officer

```
with temp as  (select * from officer_details natural join arrival)
select fname, lname, name from passenger_details inner join temp on
temp.pp_no=passenger_details.pp_no;
```



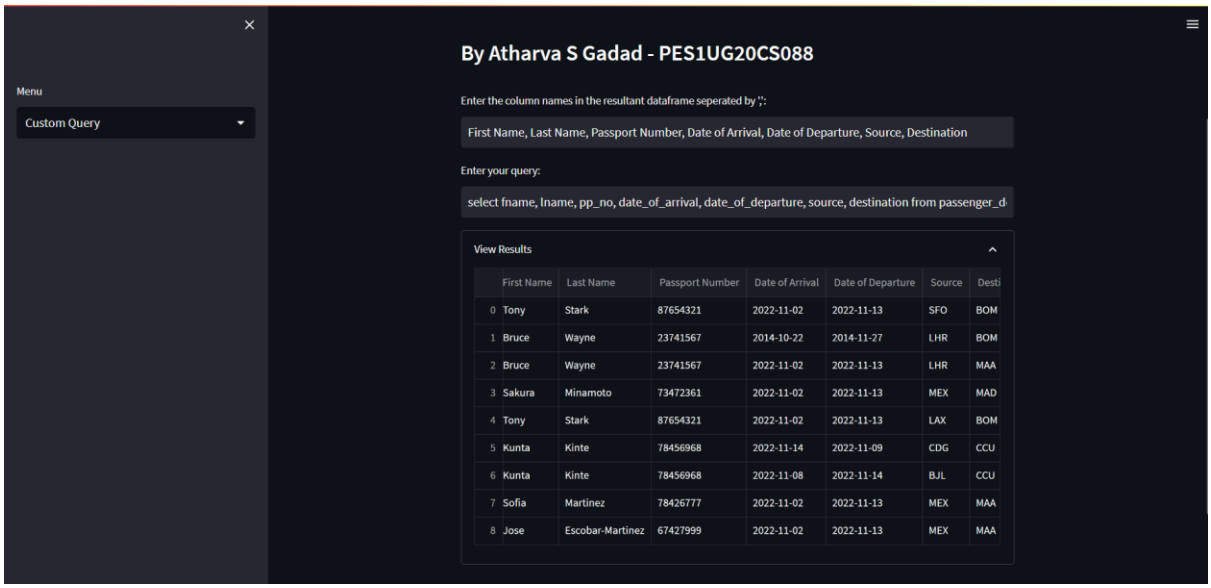
3) Mapping all passengers to their various nationalities

```
select fname, lname, nationality from passenger_details natural join
nationalities;
```



4) Linking each visit in travel history with the names of the passengers

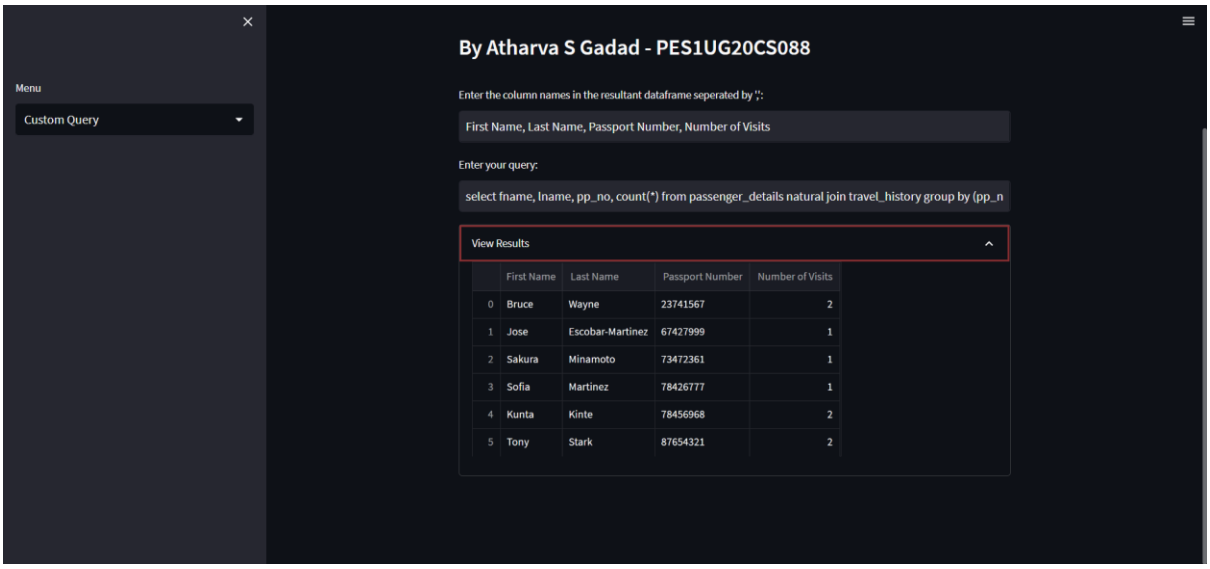
```
select fname, lname, pp_no, date_of_arrival, date_of_departure, source,
destination from passenger_details natural join travel_history;
```



Aggregate Functions

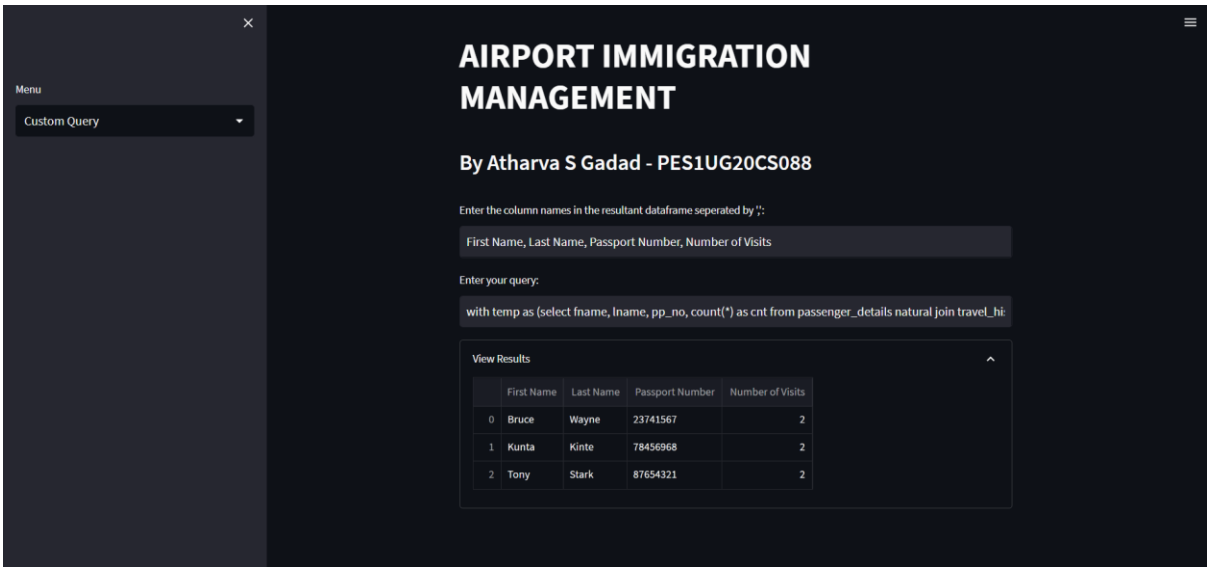
1) Finding number of visits by each traveler

```
select fname, lname, pp_no, count(*) from passenger_details natural
join travel_history group by (pp_no);
```



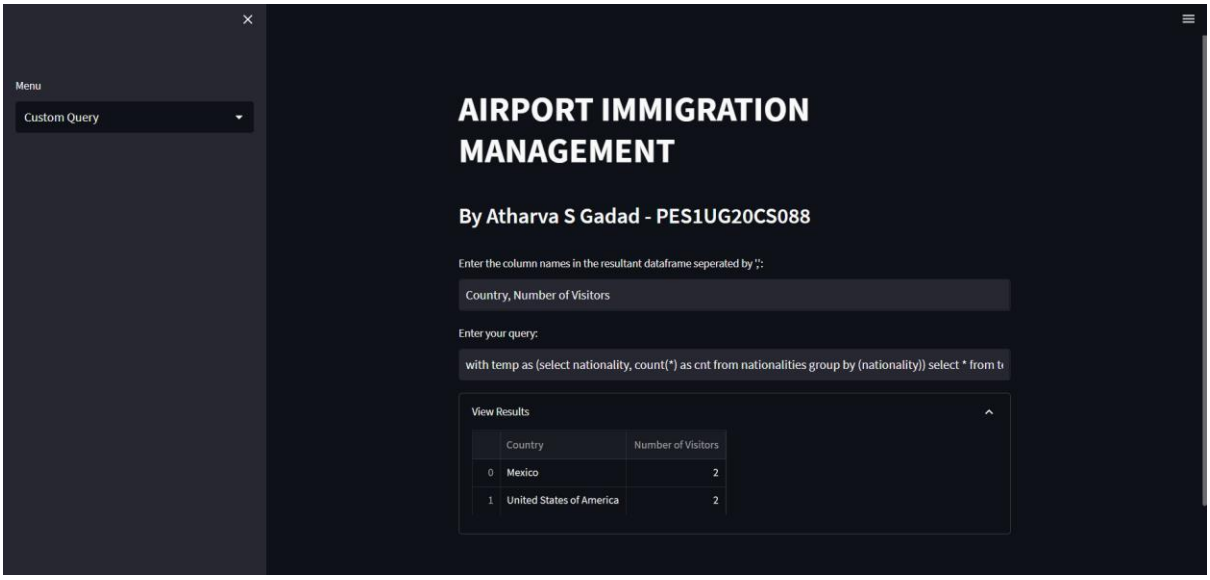
2) Finding travelers with maximum visits

```
with temp as (select fname, lname, pp_no, count(*) as cnt from
passenger_details natural join travel_history group by (pp_no))
select * from temp where cnt=(select max(cnt) from temp);
```



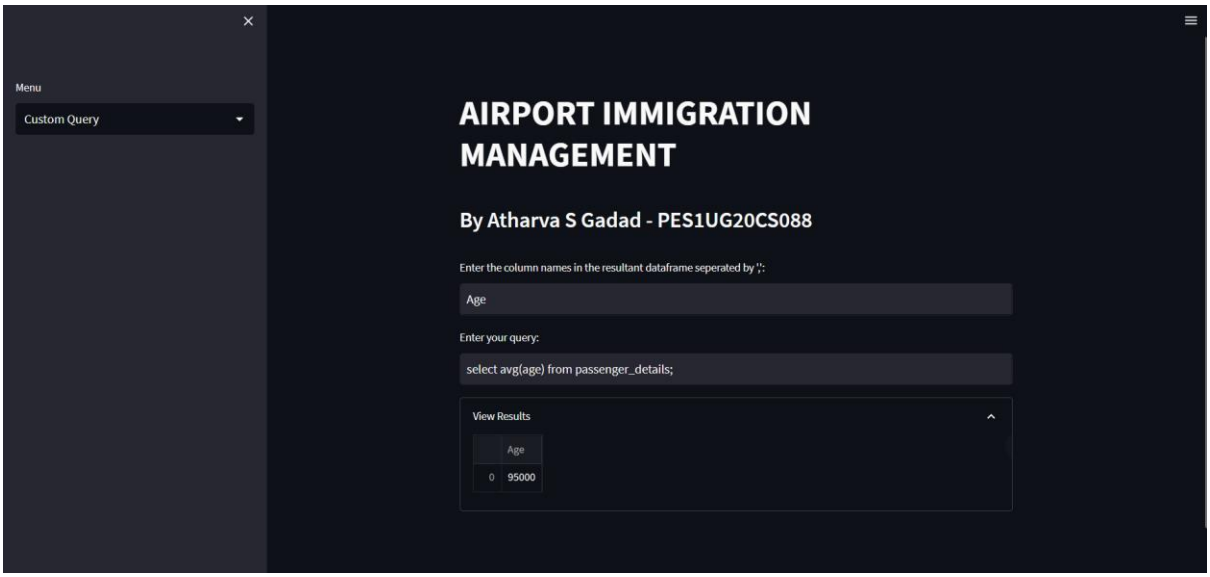
3) Finding countries from which most visitors hail

```
with temp as (select nationality, count(*) as cnt from nationalities
group by (nationality)) select * from temp where cnt=(select
max(cnt) from temp);
```



4) Calculating average age of passengers

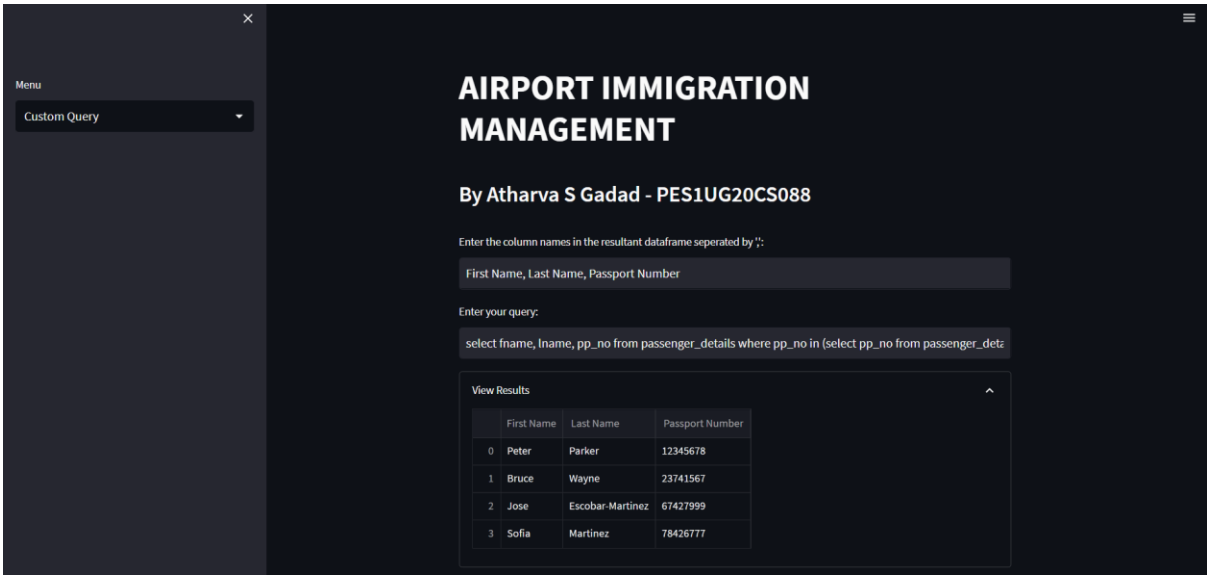
```
select avg(age) from passenger_details;
```



Set Operations

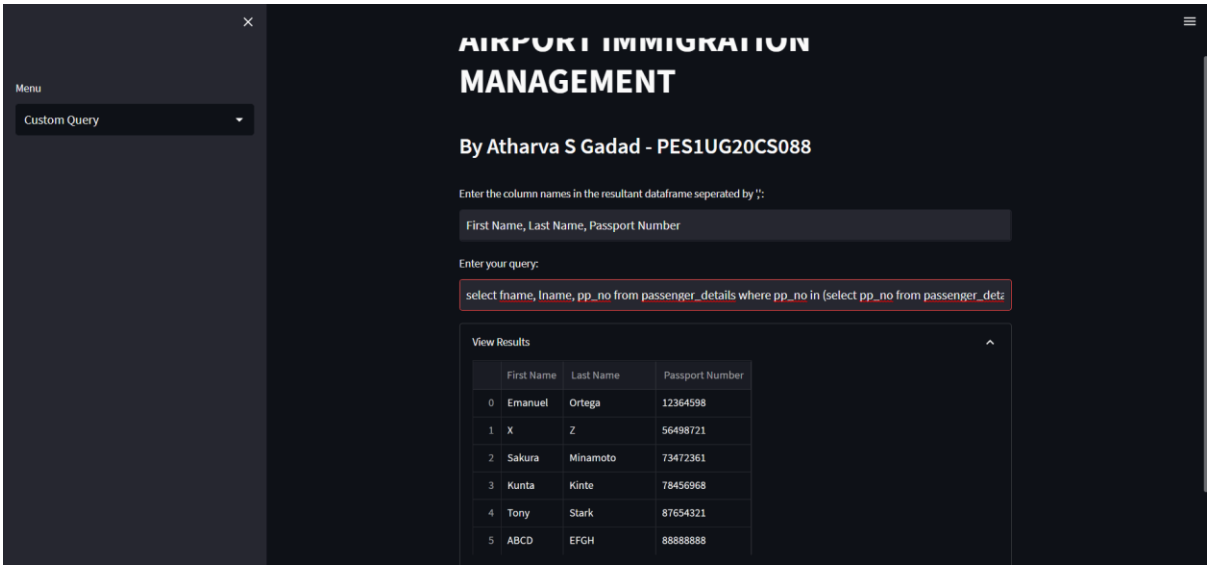
1) Finding all passengers details who have arrived recently

```
select fname, lname, pp_no from passenger_details where pp_no in
(select pp_no from passenger_details intersect select pp_no from
arrival);
```



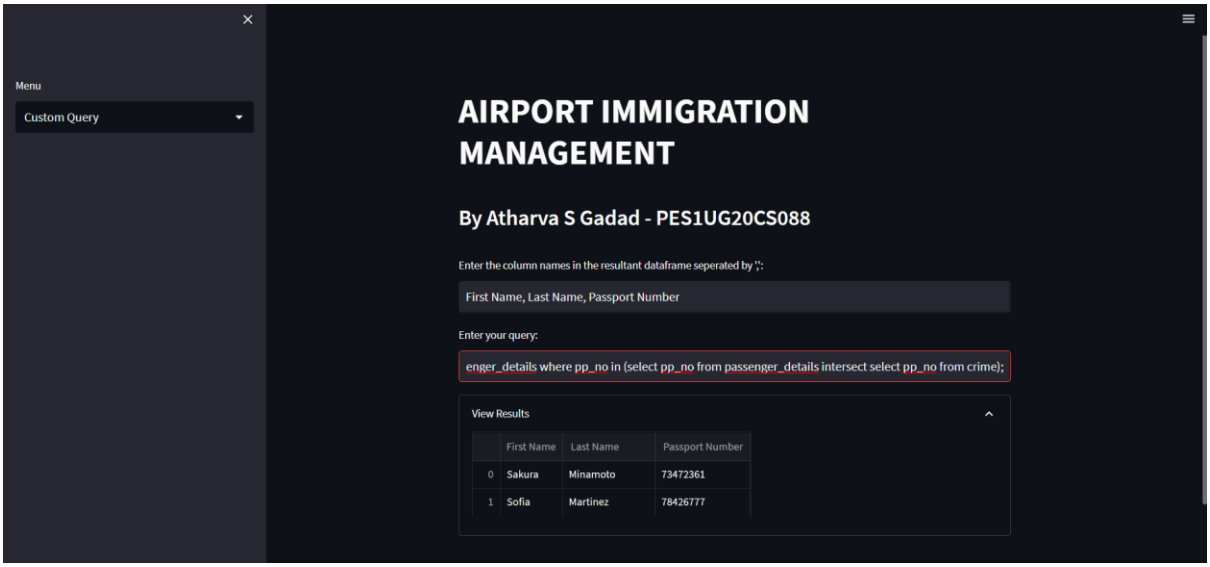
2) Finding all passengers who have not travelled recently

```
select fname, lname, pp_no from passenger_details where pp_no in
(select pp_no from passenger_details except select pp_no from
arrival);
```



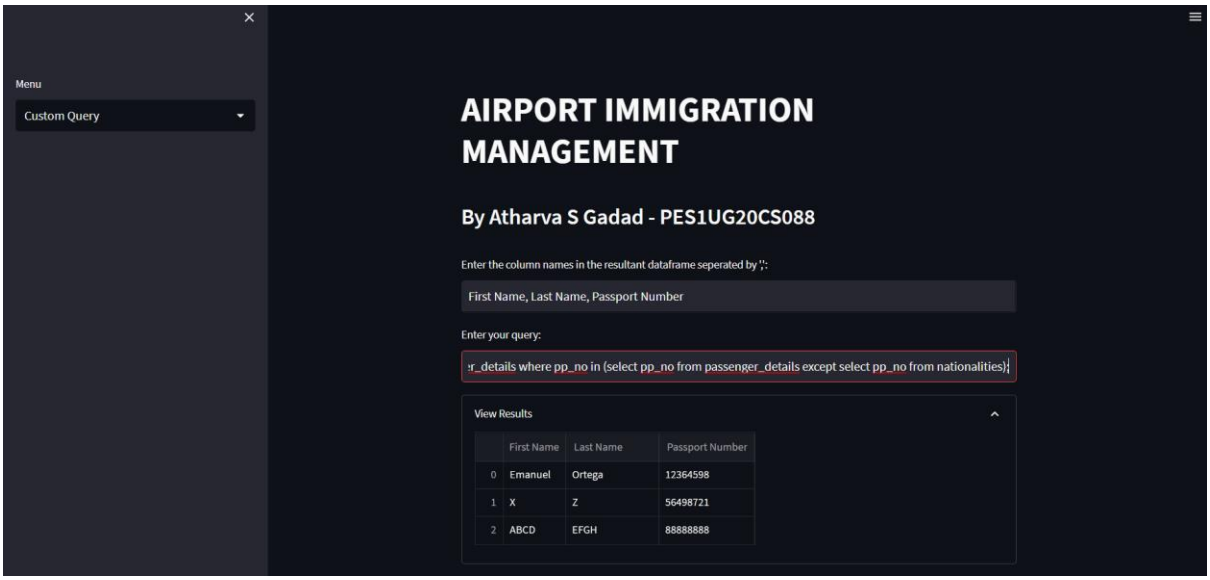
3) Selecting details of criminals

```
select fname, lname, pp_no from passenger_details where pp_no in
(select pp_no from passenger_details intersect select pp_no from
crime);
```



4) Selecting all passengers whose nationality information is unavailable

```
select fname, lname, pp_no from passenger_details where pp_no in
(select pp_no from passenger_details except select pp_no from
nationalities);
```



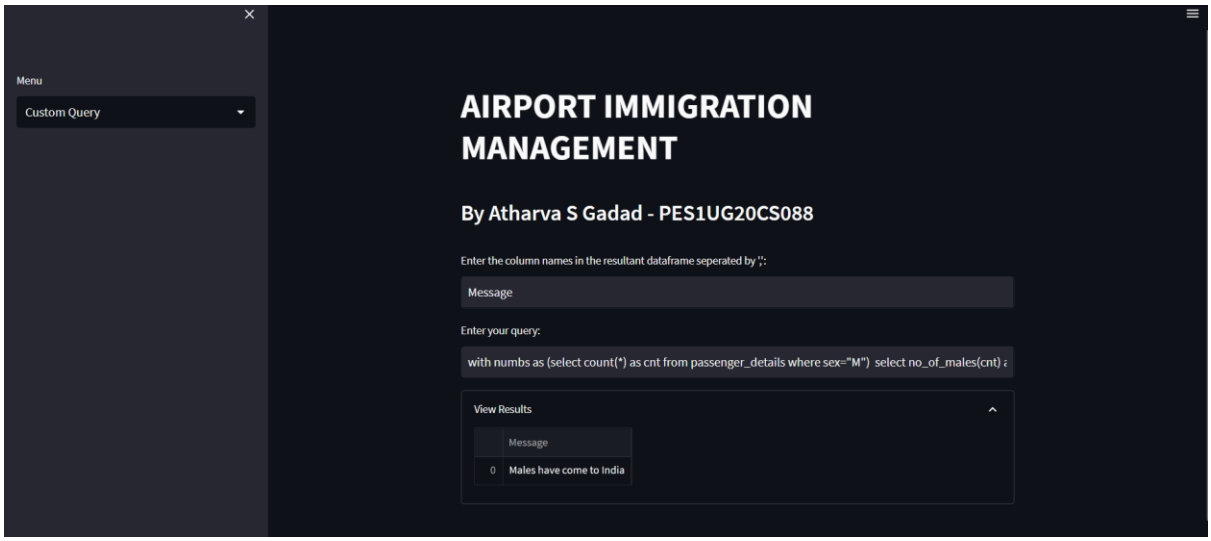
Function

The function is to check if number of males who have come to India is greater than 1

```
DELIMITER $$
CREATE FUNCTION no_of_males(males int)
returns varchar(50)
deterministic
begin
declare return_value varchar(50);
if males>1 then
    set return_value="Males have come to India";
else
    set return_value="Males have not come to India";
end if;
return return_value;
end $$
DELIMITER ;
```

We will call the function as follows:

```
with numbs as (select count(*) as cnt from passenger_details where sex="M")
select no_of_males(cnt) as male_count from numbs;
```



We see that number of males in India is greater than 1

Procedure

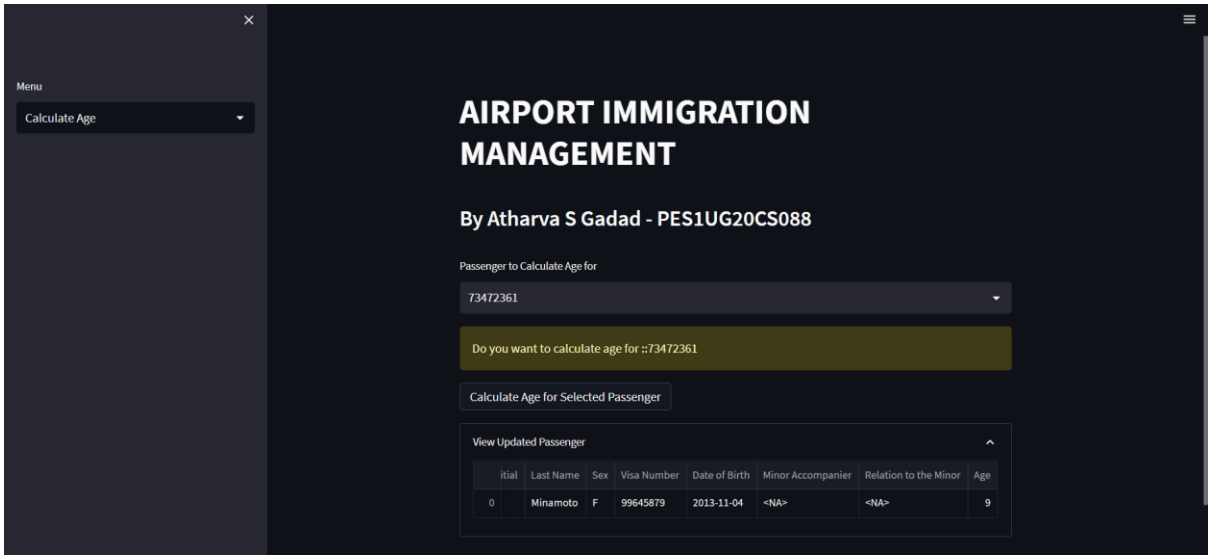
My procedure calculates the age of a selected passenger

```
DELIMITER $$
create procedure dob_age(in pp_no varchar(20), in dob date, out age int)
begin
update passenger_details set
passenger_details.age =DATE_FORMAT(FROM_DAYS(DATEDIFF(NOW(),dob)), '%Y') + 0
where passenger_details.pp_no=pp_no;
end;
$$
DELIMITER ;
```

In the back-end, it is called as follows

```
set @p0='<selected_pp_no>';
set @p1=(select passenger_details.dob from passenger_details where
passenger_details.pp_no=@p0);
set @p2=0;
call dob_age(@p0, @p1, @p2);
select * from passenger_details where passenger_details.pp_no=@p0;
```

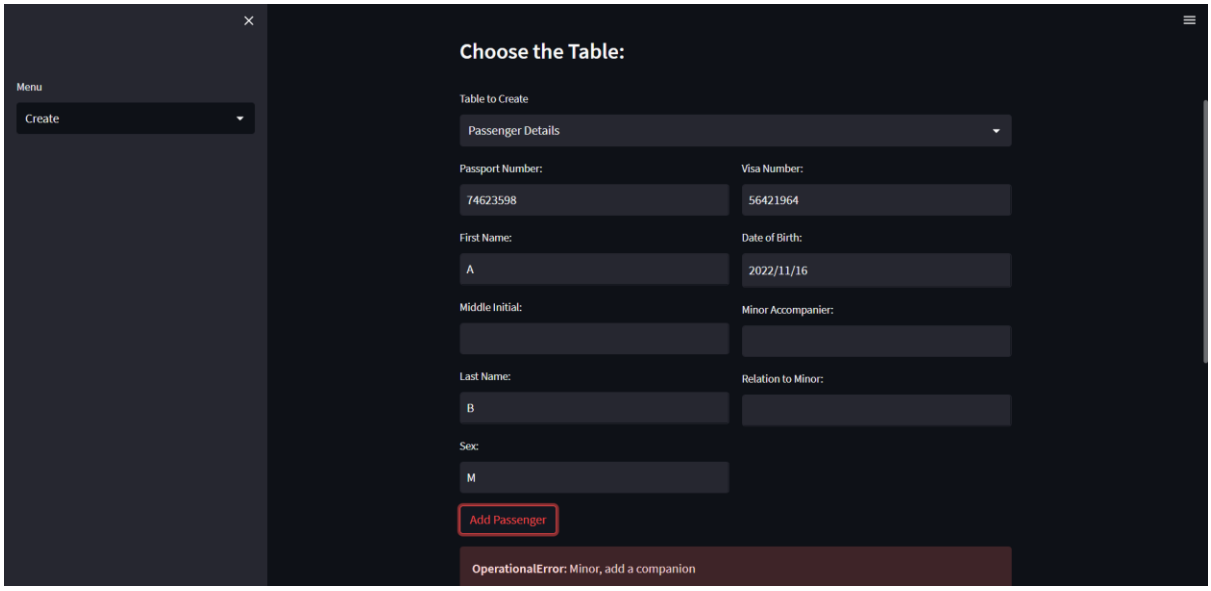
In the front end, we choose any passport number from the drop-down menu and proceed as shown



Trigger

My trigger checks passengers' age, and if they are below a certain age, they must have a companion accompanying them. Else, during insert, an error is thrown

```
DELIMITER $$
CREATE TRIGGER check_birth_date
BEFORE INSERT
ON passenger_details FOR EACH ROW
BEGIN
DECLARE error_msg VARCHAR(255);
declare age int;
set age=DATE_FORMAT(FROM_DAYS(DATEDIFF(NOW(),new.dob)), '%Y') + 0;
SET error_msg = ('Minor, add a companion');
IF (age<3 and new.minor_accompanier is null) THEN
SIGNAL SQLSTATE '45000'
SET MESSAGE_TEXT = error_msg;
END IF;
END $$
DELIMITER ;
```

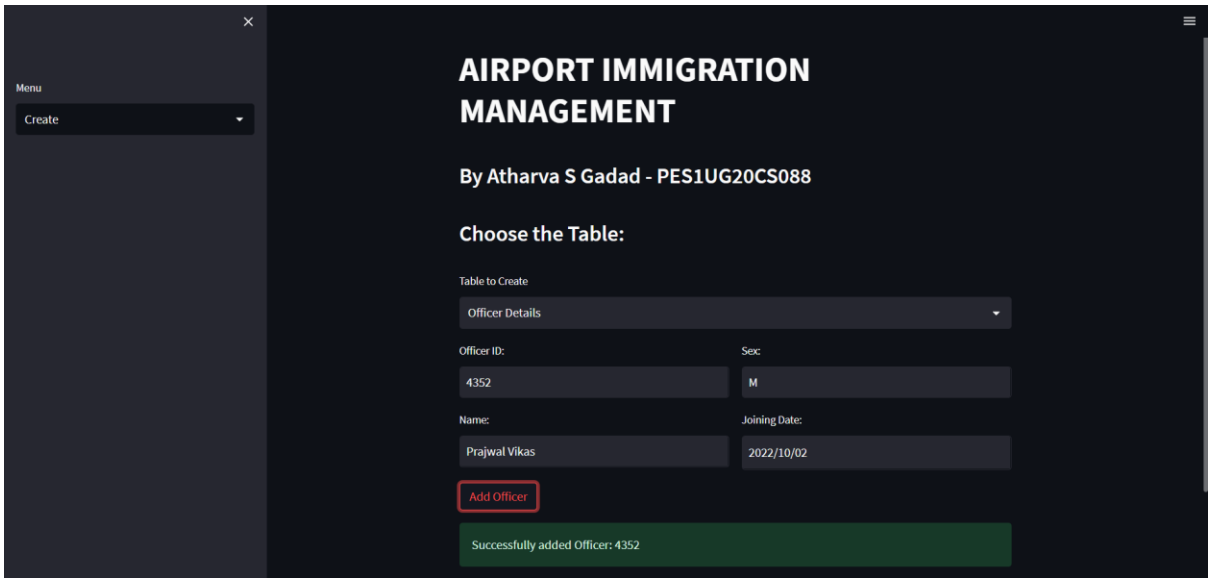


Cursor

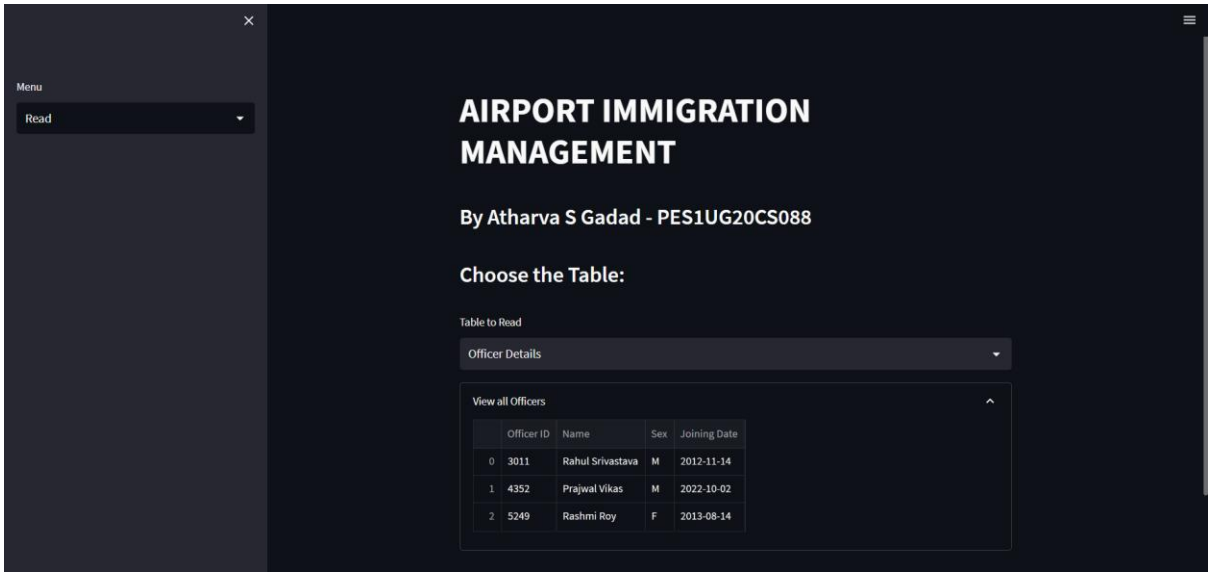
```
import mariadb
mydb = mariadb.connect(
user="atharva",
password="my_password",
host="127.0.0.1",
port=3306,
database="airport_immi_man"
)
c = mydb.cursor()
c.execute('source c:/users/shaileshgadad/desktop/dbms lab/project/tables
creation/tablesCreation.sql')
```

CRUD Front-end

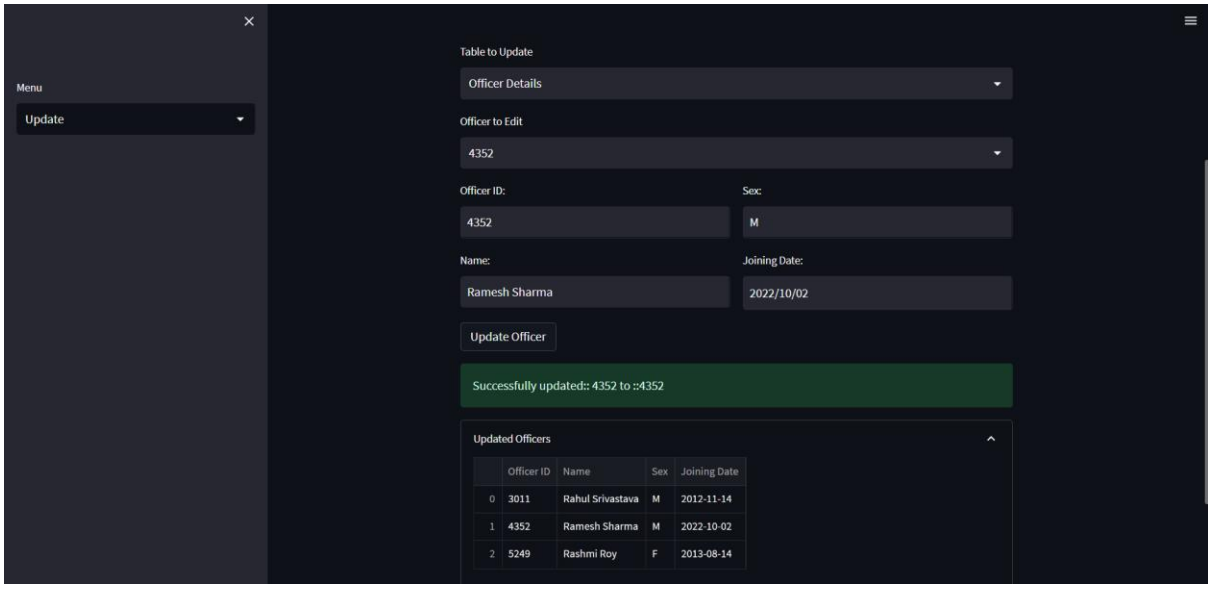
1) Create



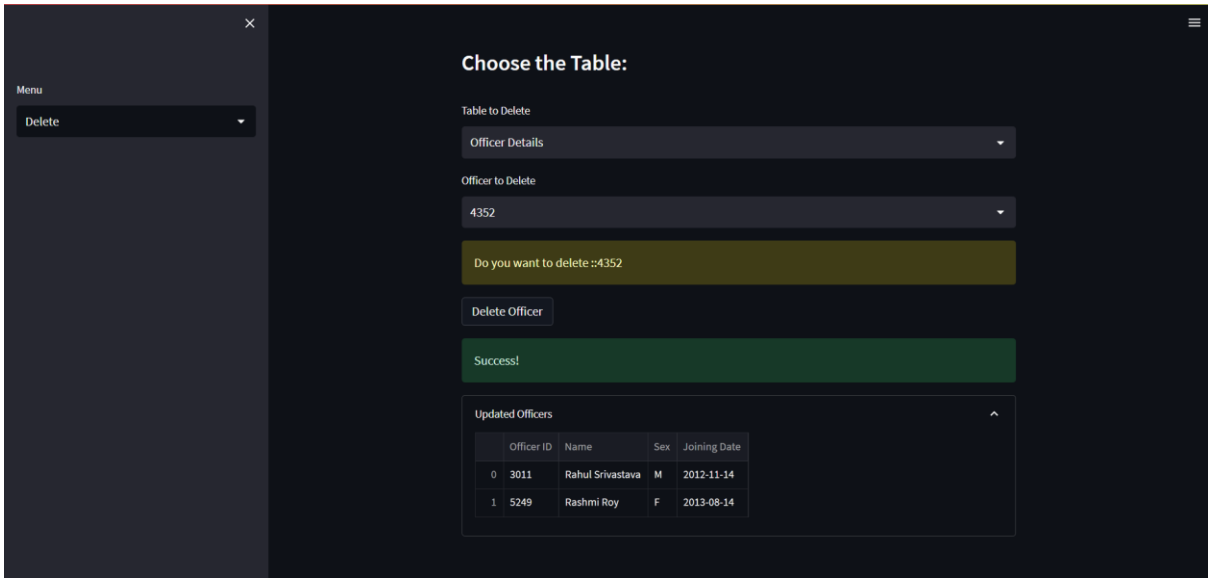
2) Read



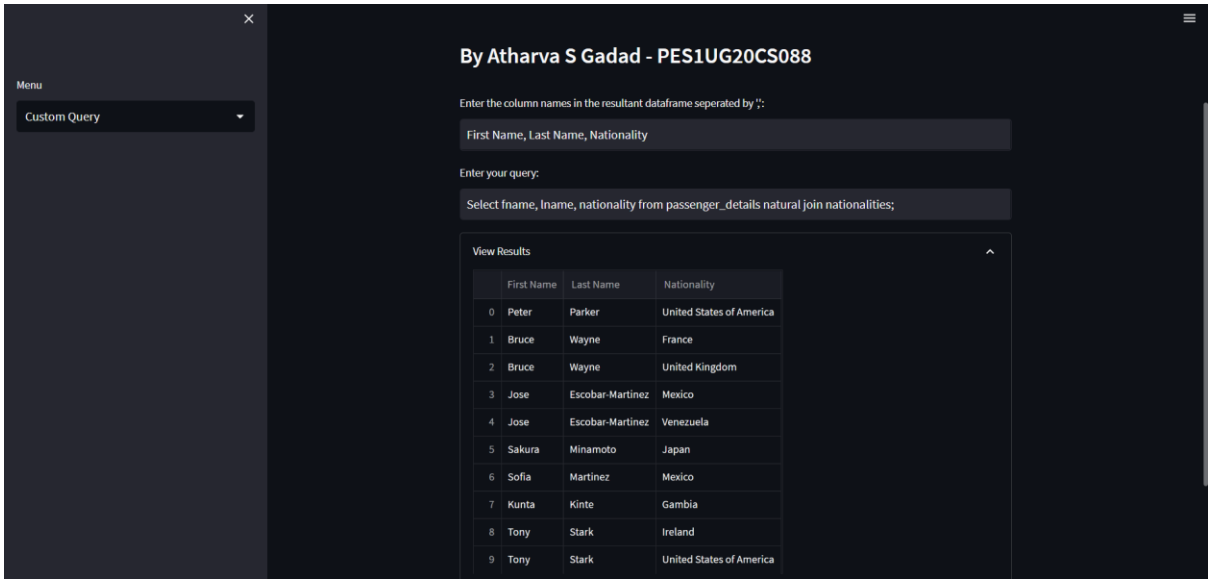
3) Update



4) Delete

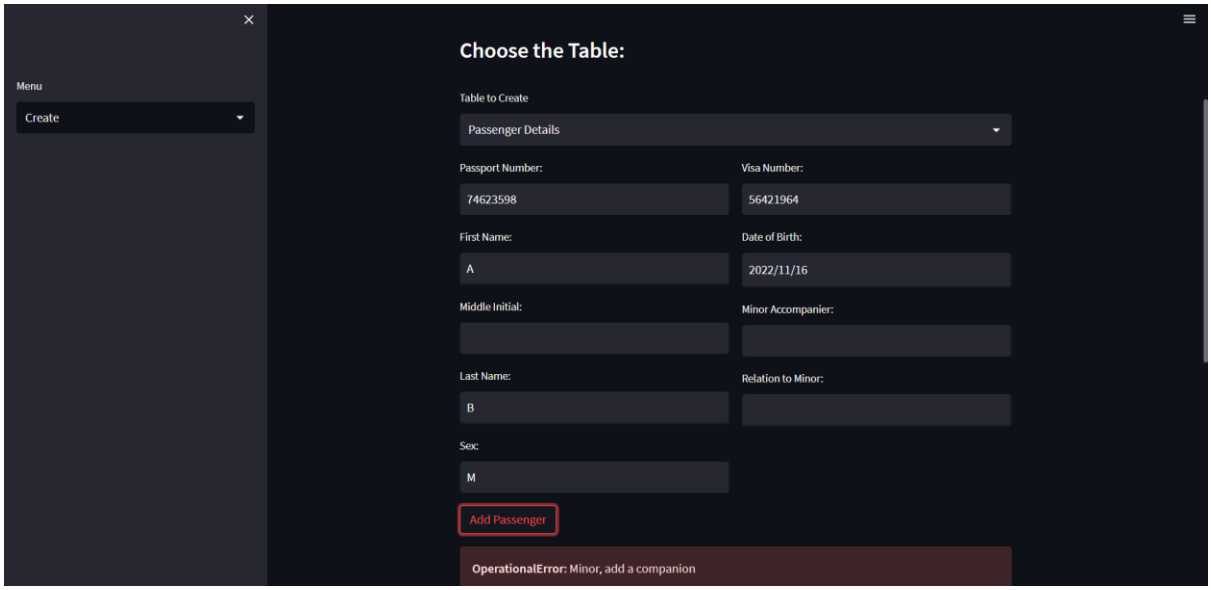


Custom SQL Statement in the Front-end



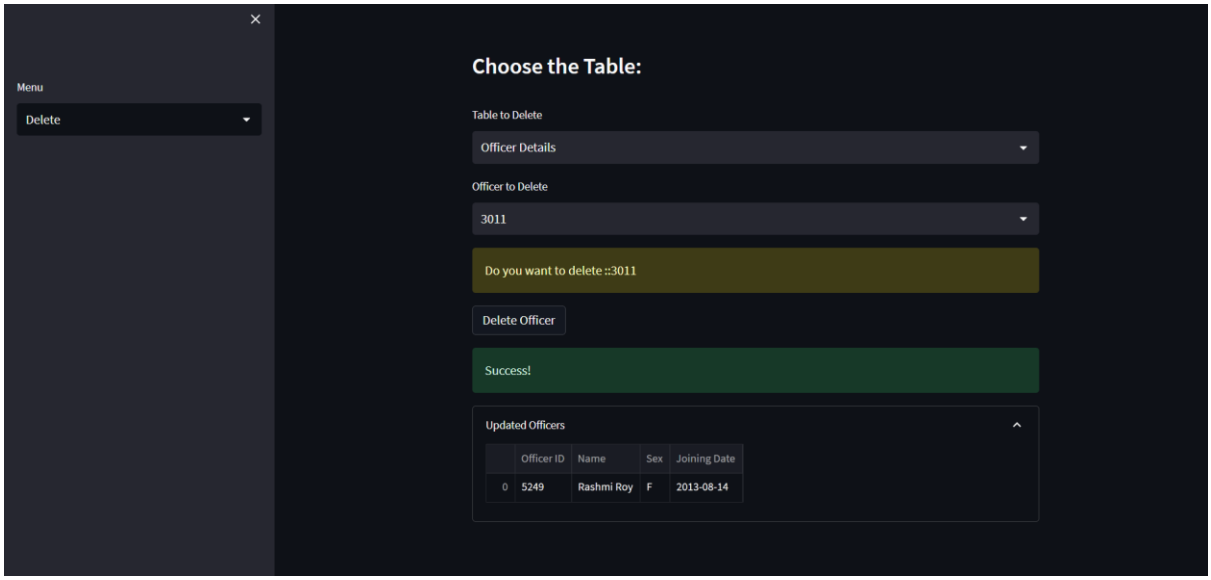
Modifications

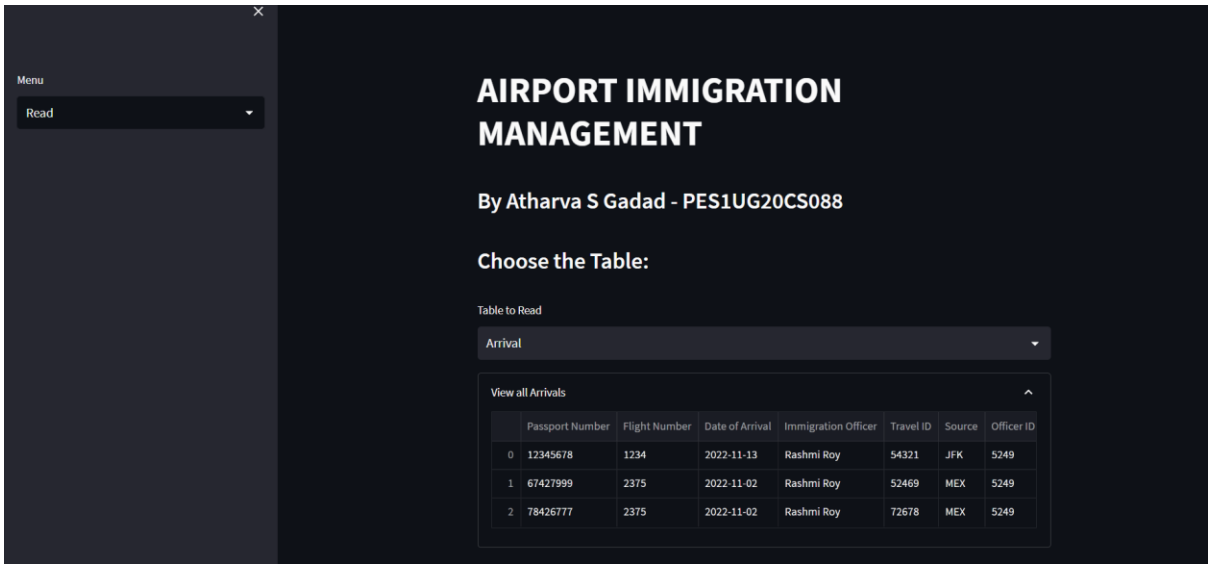
I was asked to add a trigger to ensure that people below a certain age are not allowed to travel. The results for the same are as shown below.



Another modification was adding on delete cascade constraint to all tables

	Passport Number	Flight Number	Date of Arrival	Immigration Officer	Travel ID	Source	Officer ID
0	12345678	1234	2022-11-13	Rashmi Roy	54321	JFK	5249
1	23741567	1234	2022-11-02	Rahul Srivastava	42728	LHR	3011
2	67427999	2375	2022-11-02	Rashmi Roy	52469	MEX	5249
3	78426777	2375	2022-11-02	Rashmi Roy	72678	MEX	5249


Menu
Delete
Choose the Table:
Table to Delete
Officer Details
Officer to Delete
3011
Do you want to delete ::3011
Delete Officer
Success!
Updated Officers
Officer ID
Name
Sex
Joining Date
0
5249
Rashmi Roy
F
2013-08-14


Menu
Read
AIRPORT IMMIGRATION MANAGEMENT
By Atharva S Gadad - PES1UG20CS088
Choose the Table:
Table to Read
Arrival
View all Arrivals
Passport Number
Flight Number
Date of Arrival
Immigration Officer
Travel ID
Source
Officer ID
0
12345678
1234
2022-11-13
Rashmi Roy
54321
JFK
5249
1
67427999
2375
2022-11-02
Rashmi Roy
52469
MEX
5249
2
78426777
2375
2022-11-02
Rashmi Roy
72678
MEX
5249