

Flight Reservation

CS 157A

Introduction to Databases

Team Members

Amir Radman

Sophia Tan

David Ruvalcaba

♠ TEAM ACE ♠

Database Schema

```
drop database if exists flightReservation;
```

```
create database flightReservation;
```

```
use flightReservation;
```

```
drop table if exists user;
```

```
create table user
```

```
(uID INT AUTO_INCREMENT Primary Key,
```

```
  uNAME VARCHAR(30) NOT NULL,
```

```
  age INT);
```

```
ALTER Table user AUTO_INCREMENT = 100;
```

```
drop table if exists flightList;
```

```
create table flightList
```

```
(fID INT AUTO_INCREMENT Primary Key,
```

```
  aName VARCHAR(30) NOT NULL,
```

```
  numSeats INT NOT NULL);
```

```
ALTER Table flightList AUTO_INCREMENT = 200;
```

```
create table reservation
```

```
(uID INT,
```

```
fID INT,
```

```
reservedDate Date NOT NULL Default '0000-00-00' ,
```

```
updatedAt timestamp NOT NULL on update current_timestamp default  
current_timestamp,
```

```
PRIMARY KEY (uID,fID,reservedDate),
```

```
FOREIGN KEY (uID) references user (uID) on delete cascade,
```

```
FOREIGN KEY (fID) references flightList (fID) on delete cascade);
```

```
drop table if exists seat;
```

```
create table seat
```

```
(uID INT,
```

```
fID INT,
```

```
seatNumber INT NOT NULL auto_increment UNIQUE,
```

```
PRIMARY KEY (uID,fID,seatNumber),
```

```
FOREIGN KEY (uID) references user (uID) on delete cascade,
```

```
FOREIGN KEY (fID) references flightList (fID) on delete cascade);
```

```
ALTER Table seat AUTO_INCREMENT = 300;
```

```
drop table if exists canceledReservation;  
  
create table canceledReservation  
(uID INT,  
fID INT,  
canceledDate Date NOT NULL Default '0000-00-00',  
reservedDate Date NOT NULL Default '0000-00-00',  
PRIMARY KEY (uID,fID,canceledDate)  
);
```

```
drop table if exists Archive;  
  
create table Archive  
(uID INT,  
fID INT,  
updatedAt timestamp NOT NULL);
```

```
DROP TRIGGER IF EXISTS INSERTTORES;  
  
delimiter //  
  
CREATE TRIGGER INSERTTORES
```

AFTER INSERT ON RESERVATION

FOR EACH ROW

BEGIN

insert into seat(uid,fID) values(new.uid,new.fID);

END; //

delimiter ;

DROP TRIGGER IF EXISTS INSERTTOCANCEL;

delimiter //

CREATE TRIGGER INSERTTOCANCEL

AFTER delete ON reservation

FOR EACH ROW

BEGIN

insert into canceledreservation(uid,fid,canceledDate,reservedDate) values
(old.uid,old.fid,current_date(),old.reservedDate);

delete from seat where uid = uid and fID = fID;

END; //

delimiter ;

drop PROCEDURE if exists archivedReservation;

delimiter //

```
create PROCEDURE archivedReservation(IN value varchar(55))
```

```
BEGIN
```

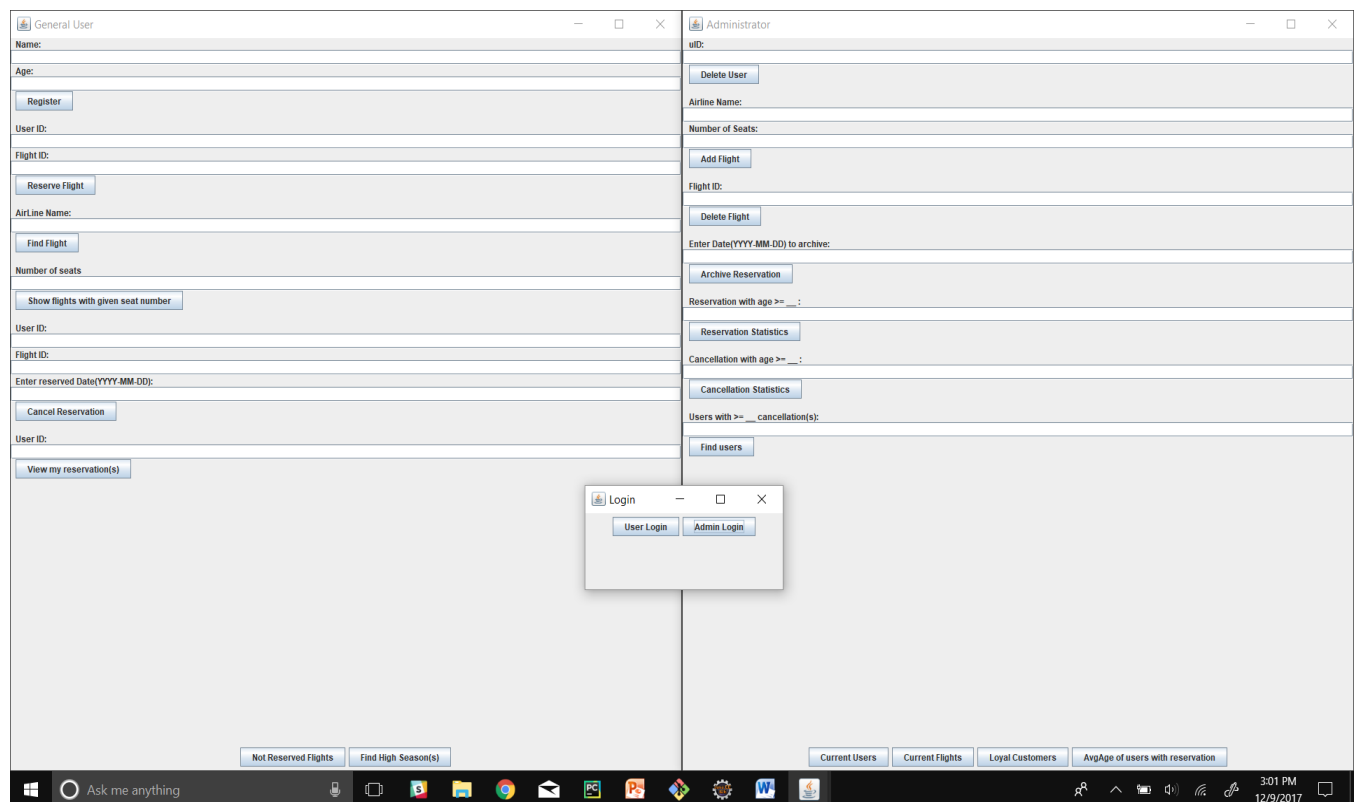
```
insert into Archive select uid,fid,updatedAt from reservation where  
date(updatedAt) < value ;
```

```
delete from reservation where date(updatedAt) < value;
```

```
END; //
```

```
delimiter ;
```

Main GUI Application (LoginPopUp.java)



All Select Statements

General User Commands

- 1- `Select uid,fid,reservedDate from reservation where uid = ?`
- 2- `Select * from flightList where aName = ?`
- 3- `select FL.fid,aName,numSeats,uid,reservedDate from flightList FL left Outer JOIN reservation R on FL.fid = R.fid`
- 4- `select distinct MonthName(reservedDate) as 'High Season Month(s)' from reservation group by Month(reservedDate) having(count(*)>=3)`
- 5- `Select * from flightList where numSeats = ?`

Administrator Commands

- 1- `select * from user JOIN reservation using(uID) group by uid having(age >= ?)`
- 2- `select * from user JOIN canceledReservation using(uID) group by uid having(age >= ?)`
- 3- `Select uid,uname,age from user u1 where (select count(*) from canceledreservation where u1.uid = uid group by uid)>= ?`
- 4- `Select * from user`
- 5- `Select * from flightList`
- 6- `select distinct uid from reservation where uid NOT IN (select uid from canceledreservation)`
- 7- `select avg(age) as 'averageAge' from user U Natural join (select distinct uid from reservation group by uid) T`

All Update Statements

*** There were no update statements ***

All Delete Statements

General User Commands

1- delete from Reservation where uid= ? and fid=? and reservedDate=?

Administrator Commands

- 1- delete from user where uid = ?
- 2- Delete from flightList where fid = ?

All Insert Statements

General User Commands

- 1- insert into user(uName,age) values(?,?)
- 2- insert into reservation(uid,fid,reservedDate) values(?,?,current_Date())

Administrator Commands

- 1- insert into flightList(aName,numSeats) values(?,?)

15 Functional Requirements

Functional Requirements	Corresponding SQL Query
Adding Users	Insert into user(uName,age) values(?,?)
Deleting Users	Delete from user where uID = ?
Adding Reservation	Insert into reservation (uID , fID,reservedDate) values(?,?,current_Date())
Canceling Reservation	Delete from reservation where uid = ? and fid = ?
Users with reservation who have not canceled any flights AT ALL (LOYAL CUSTOMERS)	select distinct uid from reservation where uid NOT IN (select uid from canceledreservation);
Adding Flights	Insert into flightList(aName, numSeats) values(?,?)
Deleting Flights	Delete from flightList where fid = ?
User's reservation with age >= ?	select * from user JOIN reservation using(uID) group by uid having(age >= ?);
User's cancellation with age >= ?	select * from user JOIN canceledReservation using(uID) group by uid having(age >= ?);
Finding flights using airline name	Select aName from flightList group by aName having (aName = ?)
Finding flights with ? # seats	Select fid,aName from flightList where numSeats = ?
Flights that have not been reserved at all	select * from flightList FL left Outer JOIN reservation R on FL.fid = R.fid;
High Season Month(s): Months with more than 3 reservations	select distinct MonthName(reservedDate) as 'High Season Month(s)' from reservation group by Month(reservedDate) having(count(*)>=3);
Users who have canceled more than ? reservation	Select uid,uname from user u1 where ? < (select count(*) from canceledreservation where u1.uid = uid group by uid);
Average age of users who reserved	Select avg(age) as 'averageAge' from user U NATURAL JOIN (select distinct uid from reservation group by uid) T;

Relations Screenshots with Populated Data

User

uID	uNAME	age
100	Amir	20
101	Kim	30
102	David	40
NULL	NULL	NULL

FlightList

fID	aName	numSeats
200	united	130
201	american	140
202	southwest	150
NULL	NULL	NULL

Seat

uID	fID	seatNumber
100	200	300
101	201	301
102	202	302
NULL	NULL	NULL

Reservation

uID	fID	reservedDate	updatedAt
100	200	2017-12-09	2017-12-09 18:30:56
101	201	2017-12-09	2017-12-09 18:31:06
102	202	2017-12-09	2017-12-09 18:31:35
NULL	NULL	NULL	NULL

CanceledReservation

uID	fID	canceledDate	reservedDate
100	200	2017-12-09	2017-12-09
101	201	2017-12-09	2017-12-09
102	202	2017-12-09	2017-12-09
NULL	NULL	NULL	NULL

Reservation Before Archive

uID	fID	reservedDate	updatedAt
100	200	2017-11-15	2017-12-08 18:42:46
100	200	2017-11-16	2017-12-09 18:42:46
100	200	2017-11-17	2017-12-11 18:42:46
101	201	2017-10-08	2017-12-12 18:42:46
102	201	2017-10-09	2017-12-13 18:42:46
102	201	2017-10-10	2017-12-14 18:42:46
NULL	NULL	NULL	NULL

Archive(Empty)

Reservation After Archive

uID	fID	reservedDate	updatedAt
101	201	2017-10-08	2017-12-12 18:42:46
102	201	2017-10-09	2017-12-13 18:42:46
102	201	2017-10-10	2017-12-14 18:42:46
NULL	NULL	NULL	NULL

Archive

uID	fID	updatedAt
100	200	2017-12-08 18:42:46
100	200	2017-12-09 18:42:46
100	200	2017-12-11 18:42:46

Key Constraints & Foreign Key Constraint

Foreign Key Violation

Reservation Using a Non-Existent Flight ID

es Mailings

Name:

Age:

User ID:

Flight ID:

AirLine Name:

Number of seats

User ID:

✕

error occurred. Error: -> Cannot add or update a child row: a foreign key constraint fails ('flightreservation`.`reservation`, CONSTRAINT `reservation_ibfk_2` FOREIGN KEY (`fid`) REFERENCES `flightlist` (`fid`) ON DELETE CASCADE)

Key Constraint Violation

Same Day Reservation Using Same User ID & Flight ID

The screenshot shows a flight reservation system interface. The form includes fields for Flight ID (100), AirLine Name (200), Number of seats, User ID, Flight ID, and Enter reserved Date(YYYY-MM-DD). Buttons include Reserve Flight, Find Flight, Show flights with given seat number, Cancel Reservation, and View my reservation(s). A modal message box is displayed in the center, stating: "An error occured. Error: => Duplicate entry '100-200-2017-12-09' for key 'PRIMARY'".

15 Functionality Screenshots

Adding User

The screenshot shows the 'General User' registration form. The Name field contains 'Amir' and the Age field contains '20'. A 'Register' button is visible. A modal message box is displayed, stating: "Registered Successfully. Your id is: 100".

Delete User

The screenshot shows the 'Administrator' interface. The ulD field contains '100'. A 'Delete User' button is visible. A modal message box is displayed, stating: "User deleted Successfully!".

Adding Flight

Airline Name: Southwest
Number of Seats: 160
Add Flight

Message
Flight added Successfully.
Flight id is: 202
OK

Delete Flight

Flight ID: 200
Delete Flight

Flight deleted Successfully
OK

Flight Reservation

User ID:
Flight ID:
Reserve Flight

Message
Reservation successful !
OK

High Season

Find High Season(s)
High Season Mo...
High Season Month => December

Airline Name Search

AirLine Name: american
Find Flight

Locate flights with airline name
Flight ID | Airline Name | Seats => 201 | american | 140

Airline Search by Seat

Find Flight
Number of seats: 150
Show flights with given seat number

Flights with 150 Seats
Flight ID | Airline Name | Seats => 202 | southwest | 150

User Reservations with Age >= __

The screenshot shows a software interface with a modal window titled "User's reservation with age >= 39". The modal contains a table with one row: "User ID | User Name | Age => 103 | John | 50". In the background, a form has a button "Archive Reservation" and a label "Reservation with age >= : 39".

User Cancellation with Age >= __

The screenshot shows a software interface with a modal window titled "User's cancellation with age >= 20". The modal contains a table with one row: "User ID | User Name | Age => 101 | Kim | 30". In the background, a form has a button "Reservation Statistics" and a label "Cancellation with age >= : 20".

Delete Flight

The screenshot shows a software interface with a modal window titled "Message". The modal contains the text "Flight deleted Successfully" and an "OK" button. In the background, a form has a button "Add Flight", a label "Flight ID:", a text input field with "200", and a button "Delete Flight".

Flight Cancellation

The screenshot shows a software interface with a modal window titled "Message". The modal contains the text "Reservation cancelled successfully !" and an "OK" button. In the background, a form has a label "User ID:" with "101", a label "Flight ID:" with "202", a label "Enter reserved Date(YYYY-MM-DD):" with "2017-12-09", and a button "Cancel Reservation".

Users with \geq __ Cancellations

Users with \geq __ cancellation(s):
1

Find users

Users with more than or equal to 1 cancellations

User ID	User Name	Age
101	Kim	30

Loyal Customers

Customers with reservation and NO Cancellation

user ID
103

Not Reserved Flights (Outer Join)

Flight ID:

Reserve Flight

AirLine Name:

Find Flight

Number of seats

Show flights with given s

User ID:

Flight ID:

Enter reserved Date(YYYY-MM-DD)

Cancel Reservation

User ID:

View my reservation(s)

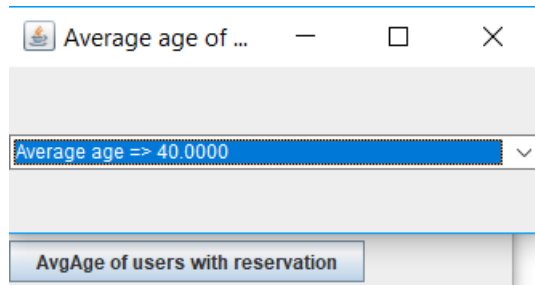
Flights with 0 reservations

ID	Airline	Seats	Reservation Date
201	american	140	2017-12-09
201	american	140	2017-12-09
202	southwest	150	null
203	iranian	160	2017-12-09
203	iranian	160	2017-12-09

Not Reserved Flights

Find High Season(s)

Average Age of Users with Reservation



END OF REPORT
