**RAILWAY RESERVATION SYSTEM**

**Project submitted to the**

**SRM University – AP, Andhra Pradesh**

**for the partial fulfillment of the requirements to award the degree of**

**Bachelor of Technology in**

**Computer Science and Engineering**

**School of Engineering and Sciences**

**Submitted by**

**Vulchi Devi Chinmayi**

**AP20110010405**

****

**Under the Guidance of**

**Dr.Rajiv Senapati**

**SRM University–AP**

**Neerukonda, Mangalagiri, Guntur**

**Andhra Pradesh – 522 240**

**12-2022**

**CERTIFICATE**

Date: 12/12/2022

This is to certify that the work present in this Project entitled “RAILWAY RESERVATION SYSTEM ” has been carried out by Vulchi Devi Chinmayi under my/our supervision. The work is genuine, original, and suitable for submission to the SRM University – AP for the award of Bachelor of Technology/Master of Technology in School of Engineering and Sciences.

Supervisor

(Signature)

Dr. Rajiv Senapati

Associative Professor of Computer Science Department

**ACKNOWLEDGEMENT**

I would like to express my special thanks of gratitude to my DBMS sir who gave me the golden opportunity to do this wonderful project on the topic Railway Reservation System

Secondly , I would like to thank my parents and friends who helped me a lot in finalizing this project with in the limited time frame

Lastly I would like to thank all my supporters who have motivated me to fulfill this project before the time line.I also want to thank all my friends who displayed appreciation for my work and motivated me to continue my work.

Vulchi Devi Chinmayi

AP20110010405

DATE

7/12/2022

**TABLE OF CONTENTS**

Certificate ii

Acknowledgement iii

Table of contents iv

Abstract v

Introduction vi

Project Description vii

Entities viii

ER Model x

Relations xi

ER to Relational Model xii

Relational model before Normalization xiii

Normalization xiv

Relational model after Normalization xviii

Tables xx

SQL Queries xxviii

Future Work & References xxx

**ABSTRACT**

This project is all about creating database for railway reservation system. In this present situation where people are travelling from one place to another for business, entertainment, education etc railway reservation system is very important. If there is no database for railway reservation system then it will be a tedious work to identify what seats are booked and how many seats are available . Railway reservation system helps passengers to enquiry about booking tickets ,cancelling tickets , and enquiry about trains. The main aim of this project is to store details of passengers and also help them to book tickets. It also helps to cancel the tickets. Passenger can book their tickets easily by knowing the seats available, train availability etc. So there is a great need of database for railway reservation system. In this railway reservation system there are many entities and relationships exist between those entities.

**INTRODUCTION**

DBMS is an organized form of collection of data. By using DBMS it is easy for users ,administrators to update ,delete ,insert information . In this project I have created a database on railway reservation system. The main purpose of this database is to use it for booking and cancelling tickets and also enquiry about the trains. This project is free from anamolies like insert anamoly,delete anamoly,update anamoly**.** It helpsadministrator to record the passenger details and also users can easily know about status of various trains. The creation of database reduces the manual errors.

If there is no database then there may be chances of errors, loss of data , redundancy etc. All these can be eliminated using database. Maintaining database for every system is crucial in the present scenario. Sometimes there may be issues like two people book the ticket for same seat and there may be disputes and fight among passengers. Such situations can be eliminated using database . So in order to reduce these errors I have made a railway reservation system database.

**PROJECT DESCRIPTION**

This project is all about building a database for railway reservation system.

This project is used to book ,cancel tickets ,enquiry about trains. This includes many entities such as passenger, train, payment, ticket, manager, station,cancellation.

Whenever a passenger wants to book tickets he she should give basic information such as name, identity proof like aadhar, age, gender ,mobile number.

The record of train consists of train number, train name, arriival time, departure time ,source and destination of train and also seats available. Whenever the passenger wants to book tickets he or she should look into the availability of seats and book them .As there is a database one can easily know the seats that are booked. It makes work easy and simple.

For the passenger to travel he must book a ticket . Each record of ticket entity consists of ticket id, train number , source, dest, date&time and seat number, pnr number,type of class he or she wants to travel. Each ticket has a unique ticket id so ticket id is used to distinguish each and every record. Train number is to describe in which train he wants to travel . Based on seats availability seat is booked for the passenger on the date and time he wished to travel.

Each ticket consists of a payment. So there is a seperate entity created as payment. This includes payment id,payment mode, fare,security pin.Payment mode can be anything among google pay, phone pay, cash .

There are some hundreds of trains and all these trains should pass by stations. So I included a seperate entity named station. Each record of station has station id,station name,trains count that are passing through station and location of station.

As there are hundreds of trains all these should be managed properly in such a way that there will no inconvience for passengers. So a seperate entity manger is created who manages all trains. Each record in manager contains manager id,manager name, experience,mobile . Each station has only one manager ..Sometimes there is a possibility that passenger may wish to cancel the ticket.So a new entity named cancellation is created. It contains cancellation\_id ,fare,date of issue as attributes.

**ENTITIES**

* passenger
* train
* ticket
* payment
* manager
* cancel
* station

**STEP2: Identify the attributes of each entity and the type of the attribute**

* passenger(p\_name,aadhar,gender,age,contact)

**Attribute Data type Constraint**

p\_name varchar(30) not null

aadhar varchar(30) primary key

gender char not null

age int not null

contact varchar(50) multivalued attribute

* train(t\_num,t\_name,arrival\_time,dept\_time,source\_loc,seats\_avail,des\_loc)

**Attribute Data type Constraint**

t\_num int primary key

t\_name varchar(30) not null

arrival\_time double not null

dept\_time double not null

source\_loc varchar(30) not null

dest\_loc varchar(30) not null

seats\_avail int

* ticket(ticket\_id, pnr\_number, source\_name,dest,date\_time,t\_num,seat\_num,class)

**Attribute Data type Constraint**

ticket\_id int primary key

pnr\_number int not null

source\_name varchar(30) not null

dest varchar(30) not null

date\_time varchar(30) not null

t\_num int foreign key

seat\_num int not null

class varchar(30) not null

* payment(p\_id,p\_mode,fare,security\_pin)

**Attribute Data type Constraint**

p\_id int primary key

p\_mode varchar(30) not null

fare double not null

security\_pin varchar(40) not null

* station(st\_id,st\_name,location,t\_count)

**Attribute Data type Constraint**

st\_id int primary key

st\_name varchar(30) not null

location varchar(30) not null

t\_count int not null

* manager(m\_name,m\_id,conatct,exp,location,email)

**Attribute Data type Constraint**

m\_name varchar(30) not null

m\_id int primary key

conatct varchar(30) multivalued attribute

exp int

location varchar(30) not null

email varchar(30) not null

* cancellation(c\_id,fare,date\_issue)

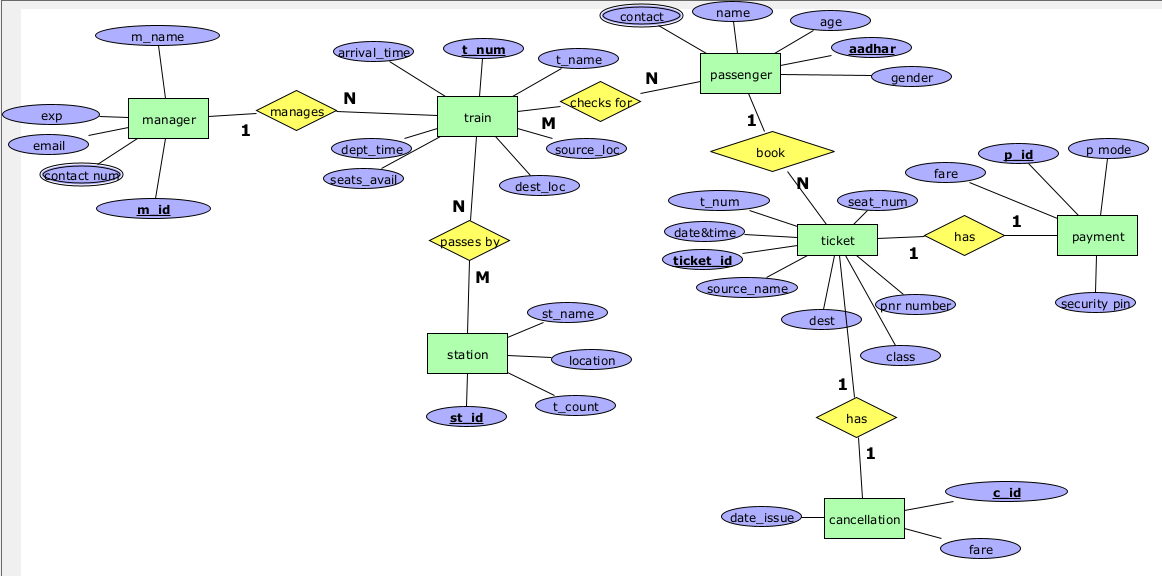
**Attribute Data type Constraint**

c\_id int primary key

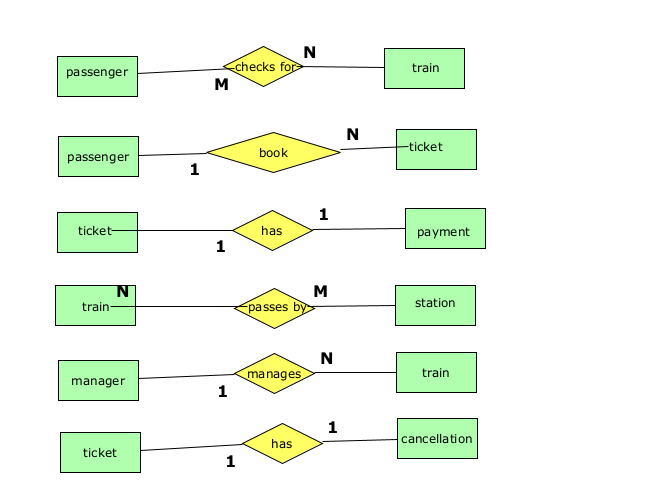
fare double not null

date\_issue varchar(30) not null

**ER MODEL**

****

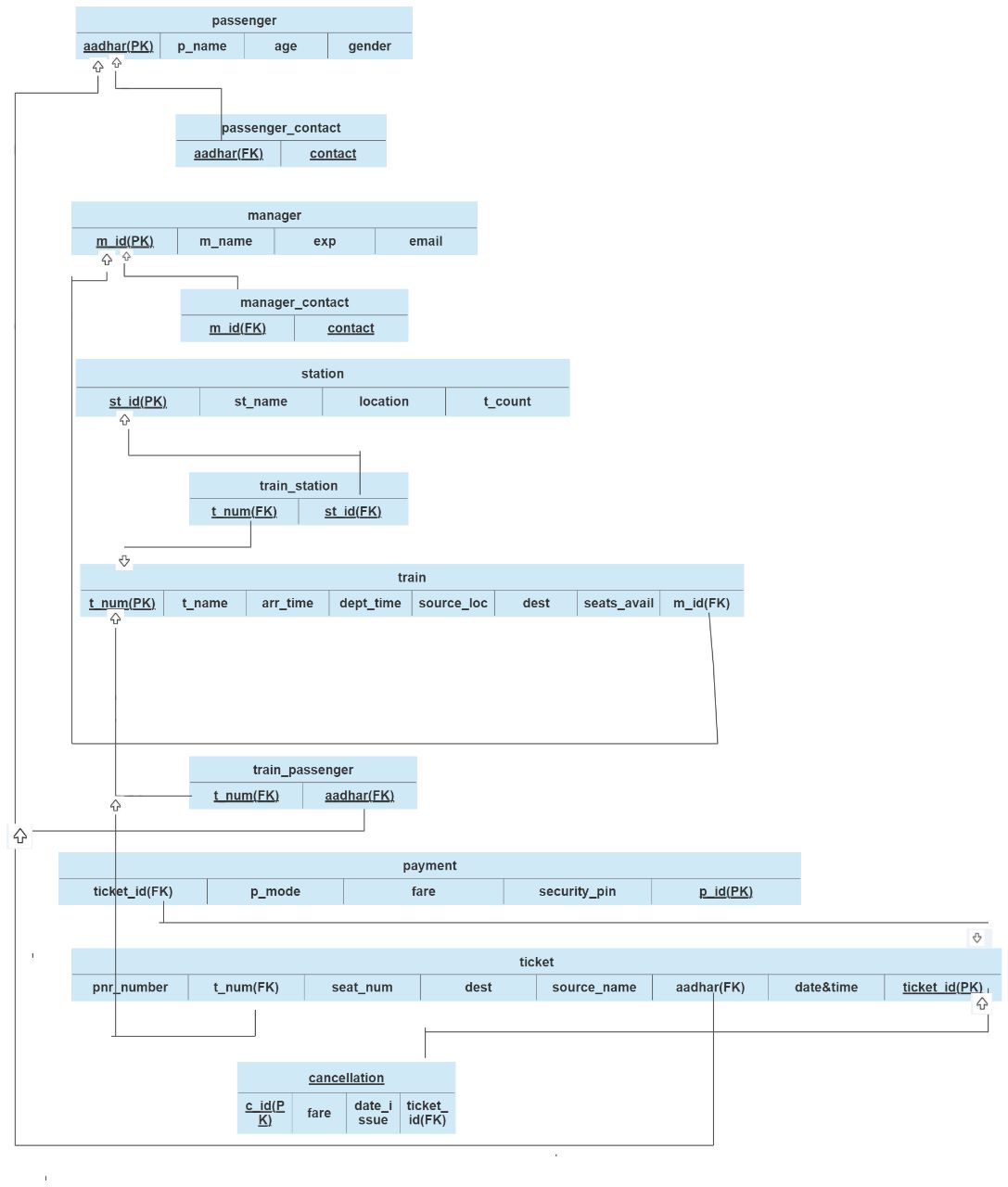
**RELATIONS**

****

**ER TO RELATIONAL MODEL**

* The entity "passenger" contains multivalued attribute contact.So primary key(aadhar) and contact is kept in a seperate table "passenger\_contact"
* The entity "manager" contains multivalued attribute contact.So primary key(m\_id) and contact is kept in a seperate table "manager\_contact"
* The relation between passenger and train is M:M .So a new table train\_passenger is created with the attributes t\_num, aadhar as foreign keys.
* The relation between station and train is M:M .So a new table train\_station is created with the attributes t\_num, st\_id as foreign keys.
* The relation between passenger and ticket is 1:N.So the primary key of passenger i.e aadhar is kept as foreign key in ticket table.
* The relation between payment and ticket is 1:1.So the primary key of ticket i.e ticket\_id is kept as foreign key in payment table.
* The relation between train and manager is 1:N.So the primary key of ticket i.e m\_id is kept as foreign key in train table.
* The entity "cancel" has a 1:1 relation with ticket. So the primary key of ticket should be kept as foreign key in cancel.

**RELATIONAL MODEL BEFORE NORMALIZATION**



**NORMALIZATION**

* Passenger(p\_name,aadhar,age,gender,class,contact)
* The relation "passengers" is in first normal form since all attributes are atomic
* The relation "passengers " is in second normal form as it satisfies the first condition of 2 NF i.e primary key(aadhar) consists of only one attribute
* The relation "passengers" satisfies third normal form as there are no transitive FD

aadhar->p\_name

aadhar->age

aadhar->gender

* The relation "passengers" is in BCNF as aadhar is a super key
* The relation "passengers" doesnot satisfy 4NF as we have a multivalued attributes contact

So I made a new table "passenger\_contact" in such a way it satisfies all FD's.

* passenger\_contact(aadhar,contact)
* The relation "passenger\_contact" is in first normal form since all attributes are atomic
* The relation "passenger\_contact " is in second normal form as it satisfies the second condition of 2 NF i.e there is no non key attribute .
* The relation "passenger\_contact" satisfies third normal form as there are no transitive FD
* manager(m\_id,m\_name,email,exp)
* The relation "manager" is in first normal form since all attributes are atomic
* The relation "manager " is in second normal form as it satisfies the first condition of 2 NF i.e primary key(m\_id) consists of only one attribute
* The relation "manager" satisfies third normal form as there are no transitive FD

m\_id->m\_name

m\_id->email

m\_id->exp

* The relation "manager" is in BCNF as m\_id is a super key
* The relation "manager" satisfy 4NF as there is no multivalued attributes
* manager\_contact(aadhar,contact)
* The relation "manager\_contact" is in first normal form since all attributes are atomic
* The relation "manager\_contact " is in second normal form as it satisfies the second condition of 2 NF i.e there is no non key attribute .
* The relation "manager\_contact" satisfies third normal form as there are no transitive FD
* station(st\_id,st\_name,loc,t\_count)
* The relation "station" is in first normal form since all attributes are atomic
* The relation "station " is in second normal form as it satisfies the first condition 2NF i.e primary key(aadhar) consists of only one attribute
* The relation "station" satisfies third normal form as there are no transitive FD

st\_id->st\_name

st\_id->t\_count

st\_id->loc

* The relation "station" is in BCNF as st\_id is a super key
* The relation "station" satisfy 4NF as it doesnot has a multivalued attributes
* train1(t\_num,t\_name,arr\_time,source\_loc,dest,dept\_time,seats\_avail)
* The relation "train1" is in first normal form since all attributes are atomic
* The relation "train1 " is in second normal form as it satisfies the first condition 2NF i.e primary key(t\_num) consists of only one attribute
* The relation "train1" satisfies third normal form as there are no transitive FD

t\_num->t\_name

t\_num->source

t\_num->dest

t\_num->arr\_time

t\_num->seats\_avail

t\_num->dept\_time

* The relation "train1" is in BCNF as t\_num is a super key
* The relation "train1" satisfy 4NF as it doesnot has a multivalued attributes
* train2(m\_id,t\_num)
* The relation "train2" is in first normal form since all attributes are atomic
* The relation "train2 " is in second normal form as it satisfies the first condition 2NF NF i.e there is no non key attribute
* The relation "train2" satisfies third normal form as there are no transitive FD
* The relation "train2" satisfy 4NF as it doesnot has a multivalued attributes
* train\_station(st\_id,t\_num)
* The relation "train\_station" is in first normal form since all attributes are atomic
* The relation "train\_station" is in second normal form as it satisfies the second condition 2NF NF i.e there is no non key attribute
* The relation "train\_station" satisfies third normal form as there are no transitive FD
* The relation "train\_station" satisfy 4NF as it doesnot has a multivalued attributes
* train\_passenger(aadhar,t\_num)
* The relation "train\_passenger" is in first normal form since all attributes are atomic
* The relation "train\_passenger" is in second normal form as it satisfies the second condition 2NF NF i.e there is no non key attribute
* The relation "train\_passenger" satisfies third normal form as there are no transitive FD
* The relation "train\_passenger" satisfy 4NF as it doesnot has a multivalued attributes

payment\_p1(ticket\_id,p\_mode,fare,security\_pin)

* The relation "payment\_p1" is in first normal form since all attributes are atomic
* The relation " payment\_p1" is in second normal form as it satisfies the second condition 2NF i.e primary key(ticket\_id) consists of only one attribute
* The relation "payment\_p1" satisfies third normal form as there are no transitive FD
* The relation "payment\_p1" satisfies BCNF as ticket\_id is the super key
* The relation "payment\_p1" satisfy 4NF as it doesnot has a multivalued attributes
* payment\_p2(ticket\_id,p\_id)
* The relation "payment\_p2" is in first normal form since all attributes are atomic
* The relation " payment\_p2" is in second normal form as it satisfies the second condition 2NF i.e primary key(p\_id) consists of only one attribute
* The relation "payment\_p2" satisfies third normal form as there are no transitive FD

p\_id->ticket\_id

* The relation "payment\_p2" satisfies BCNF as p\_id is the super key
* The relation "payment\_p2" satisfy 4NF as it doesnot has a multivalued attributes

ticket1(pnr\_number,t\_num,seat\_num,dest,source\_name,class)

* The relation "ticket1" is in first normal form since all attributes are atomic
* The relation " ticket1" is in second normal form as it satisfies the second condition 2NF i.e primary key(pnr\_number) consists of only one attribute
* The relation "ticket1" satisfies third normal form as there are no transitive FD

pnr\_number->t\_num

pnr\_number->seat\_num

pnr\_number->source\_name

pnr\_number->dest

pnr\_number->class

* The relation "ticket1" satisfies BCNF as pnr\_number is the super key
* The relation "ticket1" satisfy 4NF as it doesnot has a multivalued attributes
* ticket2(aadhar,date&time,ticket\_id)
* The relation "ticket2" is in first normal form since all attributes are atomic
* The relation " ticket2" is in second normal form as it satisfies the third condition i.e the non key attributes are fully functionally dependent on primary key
* The relation "ticket2" satisfies third normal form as there are no transitive FD

(aadhar,date&time)->ticketid

* The relation "ticket2" satisfies BCNF as (aadhar,date&time) is the super key
* The relation "ticket2" satisfy 4NF as it doesnot has a multivalued attributes
* ticket3(pnr\_number,ticket\_id)
* The relation "ticket3" is in first normal form since all attributes are atomic
* The relation " ticket3" is in second normal form as it satisfies the first condition i.e there is only one attribute in primary key.
* The relation "ticket3" satisfies third normal form as there are no transitive FD

ticket\_id->pnr\_number

* The relation "ticket3" satisfies BCNF as (ticket\_id) is the super key
* The relation "ticket3" satisfy 4NF as it doesnot has a multivalued attributes
* cancellation1(c\_id,amount,date\_time)
* The relation "cancellation1" is in first normal form since all attributes are atomic
* The relation " cancellation1" is in second normal form as it satisfies the first condition i.e there is only one attribute in primary key.
* The relation "cancellation1" satisfies third normal form as there are no transitive FD

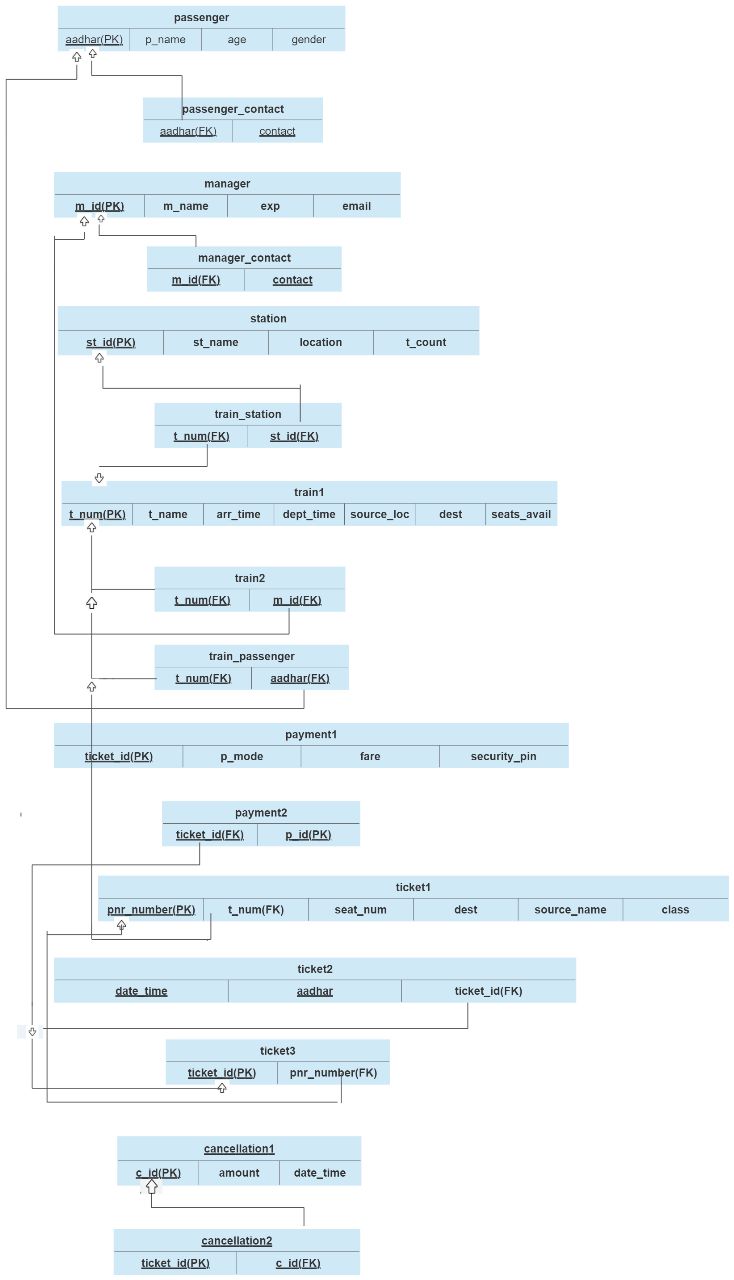
c\_id->amount

c\_id->date\_time

* The relation "cancellation" satisfies BCNF as (c\_id) is the super key
* The relation "cancellation" satisfy 4NF as it doesnot has a multivalued attributes
* cancellation2(c\_id,ticket\_id)
* The relation "cancellation2" is in first normal form since all attributes are atomic
* The relation " cancellation2" is in second normal form as it satisfies the first condition i.e there is only one attribute in primary key.
* The relation "cancellation2" satisfies third normal form as there are no transitive FD

ticket\_id->c\_id

* The relation "cancellation2" satisfies BCNF as (c\_id) is the super key
* The relation "cancellation2" satisfy 4NF as it doesnot has a multivalued attributes.

****

**TABLES**

create database dbms\_project;

use dbms\_project;

create table passenger

( p\_name varchar(30) not null,

aadhar varchar(30),

age int not null,

gender char not null,

primary key(aadhar)

);

create table passenger\_contact

(

aadhar varchar(30),

contact varchar(50) ,

primary key(aadhar,contact),

foreign key(aadhar) references passenger(aadhar)

);

create table station

(

st\_name varchar(50) not null,

location varchar(50) not null,

t\_count int not null ,

st\_id int ,

primary key(st\_id)

);

create table manager

(

m\_id int ,

m\_name varchar(30) not null,

email varchar(40) not null,

exp int not null,

primary key(m\_id)

);

create table manager\_contact

(

m\_id int,

contact varchar(50) ,

primary key(m\_id,contact),

foreign key(m\_id) references manager (m\_id)

);

create table train1

(

t\_num int,

t\_name varchar(30) not null,

arr\_time double not null,

seats\_avail int not null,

dept\_time double not null,

source\_name varchar(40) not null,

destination varchar(40) not null,

primary key(t\_num)

);

create table train2

(

m\_id int,

t\_num int,

primary key(m\_id,t\_num),

foreign key(t\_num) references train1(t\_num),

foreign key(m\_id) references manager(m\_id)

);

create table train\_station

(

st\_id int,

t\_num int,

primary key(st\_id,t\_num),

foreign key(st\_id) references station(st\_id),

foreign key(t\_num) references train1(t\_num)

);

create table train\_passenger

(

aadhar varchar(30),

t\_num int,

primary key(aadhar,t\_num),

foreign key(aadhar) references passenger(aadhar),

foreign key(t\_num) references train1(t\_num)

);

create table payment\_p1

(

ticket\_id int,

fare double not null,

p\_mode varchar(30) not null,

security\_pin varchar(40) not null,

primary key(ticket\_id)

);

create table payment\_p2

(

ticket\_id int not null,

p\_id int auto\_increment,

primary key(p\_id),

foreign key(ticket\_id) references ticket3(ticket\_id)

);

create table ticket1

(

pnr\_number int,

t\_num int not null,

seat\_num int not null,

dest varchar(30) not null,

source\_name varchar(30) not null,

class varchar(30) not null,

primary key(pnr\_number),

foreign key(t\_num) references train1(t\_num)

);

create table ticket3

(

ticket\_id int,

pnr\_number int not null,

primary key(ticket\_id),

foreign key(pnr\_number) references ticket1(pnr\_number)

);

create table ticket2

(

aadhar varchar(30),

date\_time varchar(30),

ticket\_id int not null,

primary key(aadhar,date\_time),

foreign key(ticket\_id) references ticket3(ticket\_id)

);

create table cancellation1(

c\_id int,

date\_time varchar(30) not null,

amount double not null,

primary key(c\_id)

);

create table cancellation2

(

ticket\_id int ,

c\_id int not null,

primary key(ticket\_id),

foreign key (c\_id) references cancellation1(c\_id)

);

**INSERT VALUES**

insert into passenger

values

('devi','9812445',20,'f'),

('chinmayi','678543',19,'f'),

('sai','4321678',22,'m'),

('latha','456782347',42,'f'),

('srinu','3456789876',49,'m'),

('sanvi','45672387',18,'f'),

('ram','2765890',37,'m'),

('sita','523456787',29,'f');

insert into passenger\_contact

values

('9812445','9849803404'),

('9812445','8919217663'),

('678543','8296115490'),

('3456789876','7093425416'),

('3456789876','7556943419'),

('2765890','9247661834'),

('523456787','8790452311'),

('45672387','7909654325');

insert into station

values

('Vijayawada Junction','Vijayawada',54,12345),

('Tirupati Junction','Tirupathi',28,43256),

('Rajamundry railway station','rajamandry',43,67543),

('Palasa','kakinada',32,78965),

('tenali railway','tenali',89,675908),

('eluru junction','eluru',61,876590),

('chennai railway','chennai',49,4356218);

insert into manager

values

(123,'santhan','santhan@gmail.com',4),

(345,'rani','rani@gmail.com',8),

(567,'raj','raj@gmail.com',6),

(890,'lakshmi','lakshmi@gmail.com',4),

(760,'venkat','venkat@gmail.com',8),

(654,'sreeja','sreeja@gmail.com',7),

(991,'nandan','nandan@gmail.com',5);

insert into manager\_contact

values

(123,'9876546789'),

(123,'7891234789'),

(991,'9716728906'),

(654,'7891265076'),

(567,'8675490172'),

(567,'7659016555'),

(760,'9210628909'),

(890,'7568764390');

insert into train1

values

(65565,'rajamandry experess',3.40,12,4.15,'vij','chennai'),

(89890,'tirupati express',8.30,21,8.45,'hyd','delhi'),

(97654,'chennai express',9.0,9,9.45,'vij','eluru'),

(41234,'kakinada express',7.15,17,7.20,'mumbai','tirupati'),

(59087,'delhi\_passenger',4.20,15,4.30,'kolkata','warangal'),

(87690,'kanyakumari',8.45,8,9.10,'chennai','hyderabad'),

(76907,'mumbai\_super\_fast',1.15,26,1.25,'guntur','banglore');

insert into train2

values

(123,65565),

(991,76907),

(991,87690),

(654,59087),

(890,41234),

(890,97654),

(760,89890);

insert into train\_station

values

(12345,65565),

(43256,89890),

(67543,41234),

(67543,59087),

(675908,87690),

(876590,76907),

(4356218,97654);

insert into train\_passenger

values

('9812445',97654),

('678543',97654),

('4321678',76907),

('456782347',87690),

('3456789876',87690),

('45672387',41234),

('2765890',59087),

('523456787',89890);

insert into payment\_p1

values

(452389,890,'cash',11111),

(22223,780.5,'google pay',32132),

(67676,1020,'cash',54545),

(98989,750,'phone pay',25252),

(78787,690,'cash',554589),

(25231,1290,'phone pay',765678),

(47474,875,'google pay',88888);

insert into payment\_p2

values

(452389,38989),

(22223,90909),

(67676,10293),

(78787,934122),

(25231,379087),

(47474,599990);

insert into ticket1

values

(9999999,65565,24,'hyderabad','vijayawada','first class'),

(8888888,89890,45,'delhi','chennai','second class'),

(7777777,97654,76,'vizag','hyderabad','first class'),

(6666666,97654,43,'tirupati','vijayawada','sleeper'),

(5555555,41234,35,'chennai','mumbai','third class'),

(4444444,41234,89,'palasa','hyderabad','first class'),

(3333333,59087,55,'eluru','guntur','second class');

insert into ticket2

values

('9812445','23 sep 3.15 pm',452389),

('678543','14 nov 5.am',22223),

('4321678','29 march 4.30 pm',67676),

('3456789876','8 feb 7.am',78787),

('45672387','14 nov 5 am',25231),

('2765890','6 april 8 pm',47474);

insert into ticket3

values

(452389,9999999),

(67676,8888888),

(22223,7777777),

(25231,5555555),

(47474,6666666),

(78787,4444444);

insert into cancellation1

values

(5321,'12 dec 12.pm',230),

(7890,'23 nov 3.pm',540);

insert into cancellation2

values

(22223,5321),

(47474,7890);

**QUERIES**

* select contact from passenger\_contact inner join passenger where passenger.aadhar=678543 and passenger.aadhar=passenger\_contact.aadhar;



* **Expected output:**

8289115490

* **My output:**



**2) **

* **Expected output:**

123

* **My output:**

****

**3) **

* **Expected output:**

No output

* **My output:**

** **

* **Expected output:**

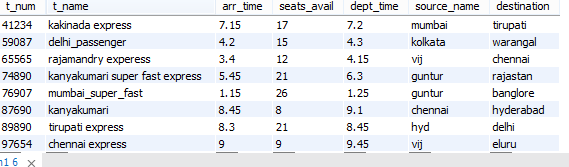
**45**2389

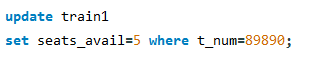
* **My output:**

****

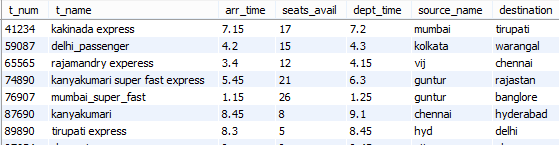
**5 ) **

* **Output:**

****

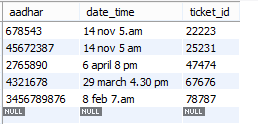
**6) **

* **Output:**

****

**7) **

* **Output:**

****

**FUTURE WORK**

My future scope is to do front end of railway reservation system. As knowing about the status of trains is important I would like to add some more features like updates about the stations, live locations of trains and also to add details of employees.

**REFERENCES**

[**https://www.slideshare.net/shashankkarnati/railway-management-system-database-mini-project**](https://www.slideshare.net/shashankkarnati/railway-management-system-database-mini-project)

Gautam, Krishna Kant. "Railway Reservation System." (2021).

CHAUDHARY, SUNNY. "RAILWAY RESERVATION SYSTEM." (2021).