Project Report

On

“Railway Management System”

By

Priyanshu Mangal

**Overview of Project**

This project is about creating a database about the Railway Management System. The railway management system facilitates the passengers to enquire about the trains available on the basis of source and destination, booking and cancellation of tickets, enquire about the status of the booked ticket, etc. The aim of case study is to design and develop a database maintaining the records of different trains, stations, and passengers. The record of the train includes its number, name, days on which it is available etc. Passengers can book their tickets for the train in which seats are available. For this, passengers have to provide the desired train number and the date for which ticket is to be booked. Before booking a ticket for a passenger, the validity of train number and booking date is checked. Once the train number and booking date are validated, it is checked whether the seat is available. If yes, the ticket is booked with confirm status and corresponding ticket No is generated which is stored along with other details of the passenger. The ticket once booked can be cancelled at any time. For this, the passenger has to provide the ticket ID (the unique key). The ticket ID is searched and the corresponding record is deleted.

**Table of Contents**

1.Introduction 3-5

2.ER Model 6-7

3.FD and Normalization 8-9

4.DDL 10-14

5. Triggers 14-15

6.SQL Queries 15-16

7.Conclusion 17

# 

# **Introduction**

Following are the entities along with their attributes.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| `Account`  ( |  | | | |
|  | `Username` | varchar(15) | NOT | NULL, |
|  | `Password` | varchar(20) | NOT | NULL, |
|  | `Email\_Id` | varchar(35) | NOT | NULL, |

`Address` varchar(50) DEFAULT NULL,

PRIMARY KEY (`Username`)

)

‘Contact ‘’

‘Passenger’

(

`Username` varchar(15) NOT NULL DEFAULT '',

`Phone\_No` char(10) NOT NULL DEFAULT '', PRIMARY KEY (`Username`,`Phone\_No`),

CONSTRAINT `Contact\_ibfk\_1` FOREIGN KEY (`Username`) REFERENCES

`Account` (`Username`) ON DELETE CASCADE

)

(

`Passenger\_Id` int(11) NOT NULL AUTO\_INCREMENT,

`First\_Name` varchar(20) NOT NULL,

`Last\_Name` varchar(20) NOT NULL,

`Gender` char(1) NOT NULL,

`Phone\_No` char(10) DEFAULT NULL,

`Ticket\_No` int(10) NOT NULL,

`Age` int(11) NOT NULL,

`Class` varchar(20) NOT NULL, PRIMARY KEY (`Passenger\_Id`), KEY `Ticket\_No` (`Ticket\_No`),

CONSTRAINT `Passenger\_ibfk\_1` FOREIGN KEY (`Ticket\_No`) REFERENCES

`Ticket` (`Ticket\_No`) ON DELETE CASCADE

)

`Station`

‘Stoppage’

(

(

`Station\_Code` char(5) NOT NULL DEFAULT

`Station\_Name` varchar(25) NOT NULL, PRIMARY KEY (`Station\_Code`)

)

`Train\_No` int(6) NOT NULL DEFAULT '0',

`Station\_Code` char(5) NOT NULL DEFAULT '',

`Arrival\_Time` time DEFAULT NULL,

`Departure\_Time` time DEFAULT NULL,

PRIMARY KEY (`Train\_No`,`Station\_Code`), KEY `Station\_Code` (`Station\_Code`),

CONSTRAINT `Stoppage\_ibfk\_1` FOREIGN KEY (`Train\_No`) REFERENCES `Train` (`Train\_No`) ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT `Stoppage\_ibfk\_2` FOREIGN KEY (`Station\_Code`) REFERENCES `Station` (`Station\_Code`) ON DELETE CASCADE ON UPDATE CASCADE

)

**‘Ticket’**

(

`Ticket\_No` int(10) NOT NULL AUTO\_INCREMENT,

`Train\_No` int(6) NOT NULL,

`Date\_of\_Journey` date NOT NULL,

`Username` varchar(15) NOT NULL, PRIMARY KEY (`Ticket\_No`),

KEY `Username` (`Username`), KEY `Train\_No` (`Train\_No`),

CONSTRAINT `Ticket\_ibfk\_1` FOREIGN KEY (`Username`) REFERENCES

`Account` (`Username`) ON DELETE CASCADE)

)

‘Train’

(

`Train\_No` int(6) NOT NULL DEFAULT '0',

`Name` varchar(25) NOT NULL,

`Seat\_Sleeper` int(4) NOT NULL,

`Seat\_First\_Class\_AC` int(4) NOT NULL,

`Seat\_Second\_Class\_AC` int(4) NOT NULL,

`Seat\_Third\_Class\_AC` int(4) NOT NULL,

`Wifi` char(1) NOT NULL,

`Food` char(1) NOT NULL,

`Run\_On\_Sunday` char(1) NOT NULL,

`Run\_On\_Monday` char(1) NOT NULL,

`Run\_On\_Tuesday` char(1) NOT NULL,

`Run\_On\_Wednesday` char(1) NOT NULL,

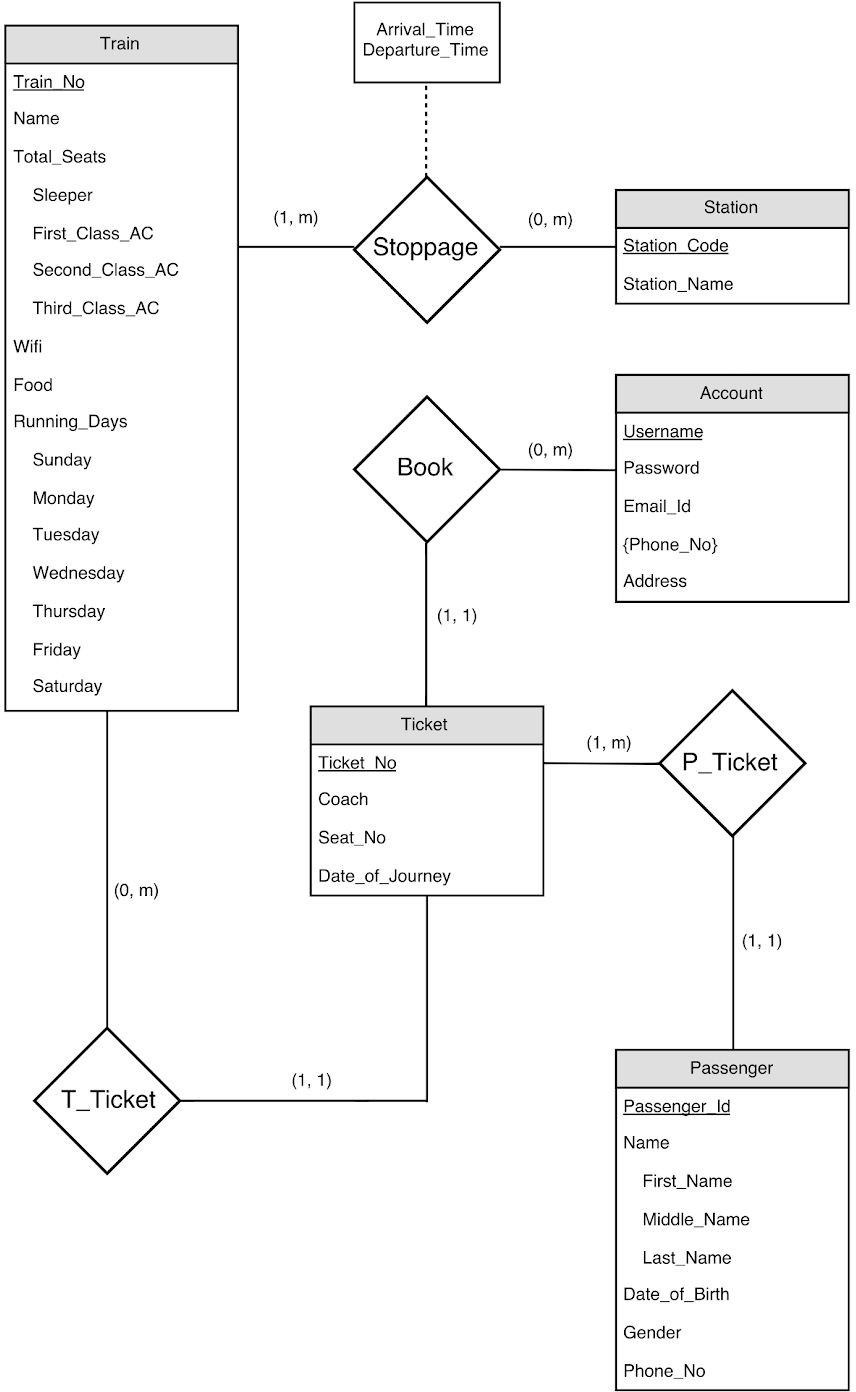
`Run\_On\_Thursday` char(1) NOT NULL,

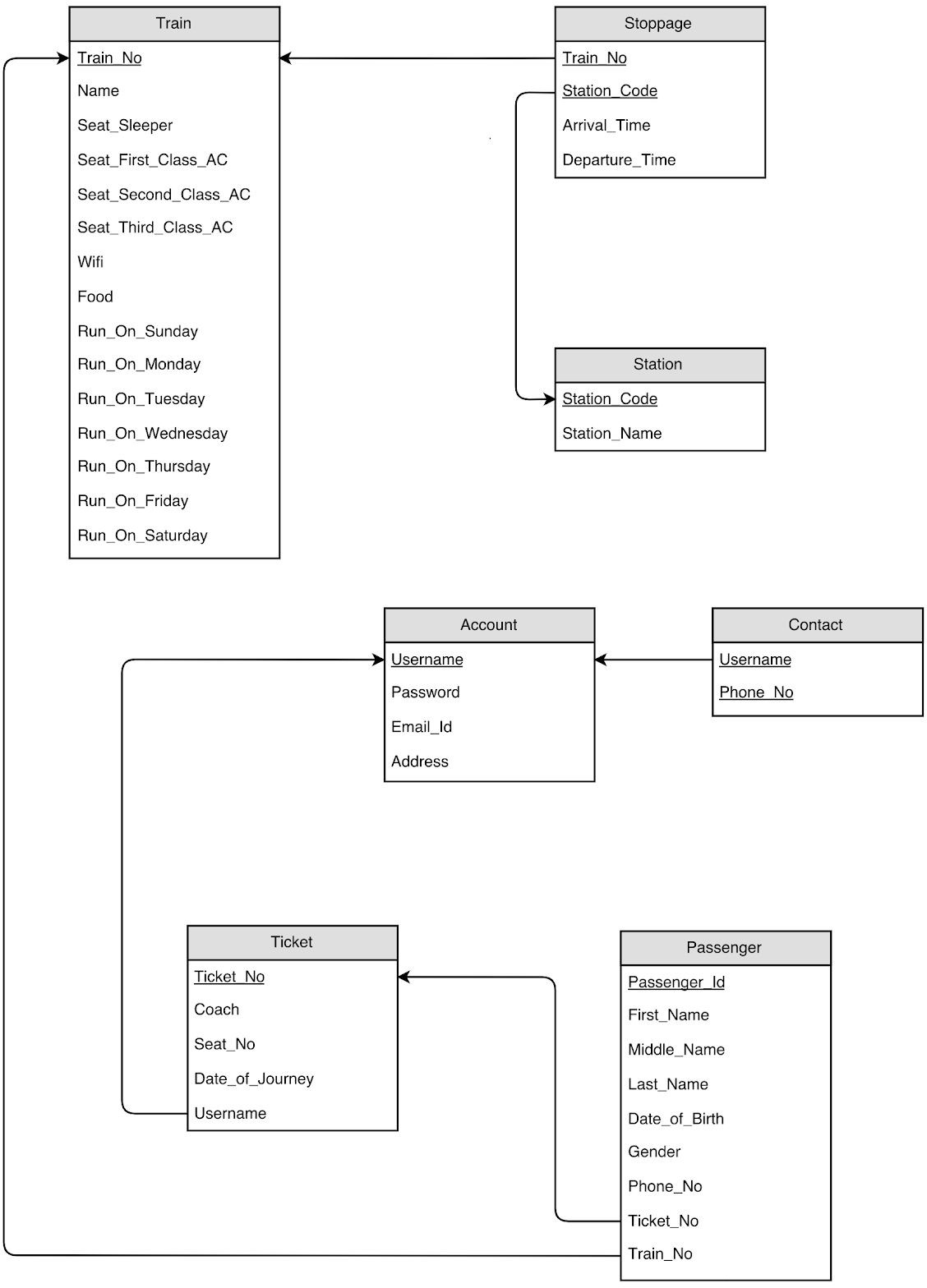
`Run\_On\_Friday` char(1) NOT NULL,

`Run\_On\_Saturday` char(1) NOT NULL, PRIMARY KEY (`Train\_No`)

)

**ER Model**





# 

# **FD and Normalization**

**Functional dependencies**

**TRAIN**

`Train\_No`->( `Name`, `Seat\_Sleeper` , `Seat\_First\_Class\_AC`, `Seat\_Second\_Class\_AC`,

Seat\_Third\_Class\_AC`,`Wifi`,`Food`,`Run\_On\_Sunday``Run\_On\_Monday``Run\_On\_Tuesd

ay` `Run\_On\_Wednesday` `Run\_On\_Thursday` `Run\_On\_Friday`,`Run\_On\_Saturday` )

**STOPPAGE**

(`Train\_No`, `Station\_Code`) -> (`Arrival\_Time` , `Departure\_Time`) **TICKET**

`Ticket\_No`-> (`Train\_No` `Date\_of\_Journey` `Username`) **STATION**

`Station\_Code` -> `Station\_Name`

**PASSENGER**

`Passenger\_Id`-> (`First\_Name` `Last\_Name` `Gender` `Phone\_No` `Ticket\_No` `Age`

`Class`)

**ACCOUNT**

`Username`->( `Password` `Email\_Id` `Address` )

### FIRST NORMAL FORM:

As per the rule of first normal form, an attribute (column) of a table cannot hold multiple values. It should hold only atomic values.

The above schema is in 1NF since all the attributes are atomic and not multivalued.

Since a passenger could have multiple phone numbers,it would violate the 1NF rules.Hence we have created a separate table called contact to handle this.

### SECOND NORMAL FORM:

A table is said to be in 2NF if both the following conditions hold:

-Table is in 1NF (First normal form)

-No non-prime attribute is dependent on the proper subset of any candidate key of table.

If in Passenger table we consider ticket\_no and first\_name as the candidate key,then date\_of\_birth would depend only on the name and it would violate the 2NF.

### THIRD NORMAL FORM:

A table design is said to be in 3NF if both the following conditions hold:

-Table must be in 2NF

-Transitive functional dependency of non-prime attribute on any super key should be removed.

Our schema follows the above rules and hence is in 3NF.

# DDL

**CREATE DATABASE railways;**

**USE railways;**

**CREATE TABLE account**

(

`Username` VARCHAR(50) PRIMARY KEY ,

`Password` VARCHAR(10) NOT NULL,

`Email\_Id` VARCHAR(40) NOT NULL,

`Address`VARCHAR(60) DEFAULT NULL

);

INSERT INTO account

VALUES('Priyanshu01',12345678,'pm01@gmail.com','Old airport road,bangalore'),

('Mangal02',34578321,'mp02@gmail.com','Jaipur'),

('Som2003',789564321,'som2003@gmail.com','Dholpur'),

('Dishu450',543789034,'dishu.dm@gmail.com','Agra'),

('Aditya685',789654098,'adi1820@gmail.com','Morena'),

('Pawan123',765439087,'pawan45@gmail.com','Bharatpur'),

('Ajay34567',890675476,'tyagi390@gmail.com','Gwalior'),

('Dhairya237',908765442,'soni678@gmail.com','Indore'),

('Shourya896',907654876,'agwl567@gmail.com','Ahmedabad'),

('Samyak452',674590875,'kaloo789@gmail.com','Kolkata');

**CREATE TABLE contact**

(

Username VARCHAR(50),FOREIGN KEY(username) REFERENCES account(username),

Phone\_No CHAR(10) NOT NULL

);

INSERT INTO contact

VALUES('Priyanshu01',8949967310),

('Mangal02',7452885312),

('Som2003',9352874214),

('Dishu450',9530404364),

('Aditya685',7727817094),

('Pawan123',8502979140),

('Ajay34567',9119177656),

('Dhairya237',8955386632),

('Shourya896',8306198068),

('Samyak452',8432056756);

**CREATE TABLE passenger**

(

`Passenger\_Id` INT NOT NULL AUTO\_INCREMENT,

`First\_Name` VARCHAR(30) NOT NULL,

`Last\_Name` VARCHAR(30) NOT NULL,

`Gender` VARCHAR(10) NOT NULL,

`Phone\_No` CHAR(10) DEFAULT NULL,

`Ticket\_No` INT NOT NULL,FOREIGN KEY (`Ticket\_No`) REFERENCES `Ticket`

(`Ticket\_No`) ON DELETE CASCADE,

`Age` INT NOT NULL,

`Class` VARCHAR(20) NOT NULL,

PRIMARY KEY (`Passenger\_Id`),

KEY `Ticket\_No` (`Ticket\_No`)

);

**CREATE TABLE station**

(

`Station\_Code` CHAR(10) NOT NULL PRIMARY KEY,

`Station\_Name` VARCHAR(50) NOT NULL

);

INSERT INTO station

VALUES ('JP','JAIPUR JUNCTION'),

('DHO','DHOLPUR JUNCTION'),

('AGC','AGRA CANTT'),

('GWL','GWALIOR JUNCTION'),

('MTJ','MATHURA JUNCTION'),

('NDLS','NEW DELHI'),

('NZM','HAZRAT NIZAMUDDIN'),

('CNB','KANPUR CENTRAL'),

('VGLJ','VIRANGANA LAKSHMIBAI JHANSI'),

('BPL','BHOPAL JUNCTION');

**CREATE TABLE train**

(

`Train\_No` INT NOT NULL PRIMARY KEY ,

`Name` VARCHAR(40) NOT NULL,

`Second\_Sitting` VARCHAR(1500) NOT NULL,

`Seat\_Sleeper` VARCHAR(1500) NOT NULL,

`Seat\_First\_Class\_AC` VARCHAR(1500) NOT NULL,

`Seat\_Second\_Class\_AC` VARCHAR(1500) NOT NULL,

`Seat\_Third\_Class\_AC` VARCHAR(1500) NOT NULL,

`Executive\_Chair\_Car` VARCHAR(1500) NOT NULL,

`Ac\_Chair\_Car` VARCHAR(1500) NOT NULL,

`Wifi` CHAR(10) NOT NULL,

`Food` CHAR(10) NOT NULL,

`Run\_On\_Sunday` CHAR(10) NOT NULL,

`Run\_On\_Monday` CHAR(10) NOT NULL,

`Run\_On\_Tuesday` CHAR(10) NOT NULL,

`Run\_On\_Wednesday` CHAR(10) NOT NULL,

`Run\_On\_Thursday`CHAR(10)NOT NULL,

`Run\_On\_Friday` CHAR(10) NOT NULL,

`Run\_On\_Saturday` CHAR(10) NOT NULL

);

INSERT INTO train

VALUES (12001,'HBJ SHATABDI EXP','N','N','N','N','N',112,1092,'Y','Y','Y','Y','Y','Y','Y','Y','Y'),

(12279,'VGLJ NDLS TAJ EXP',1484,'N','N','N','N','N',234,'N','N','Y','Y','Y','Y','Y','Y','Y'),

(15046,'GKP OKHA EXP','N',560,108,216,216,'N','N','N','Y','Y','Y','Y','N','N','N','N'),

(16317,'CAPE JAT HIMSAGAR EXP','N',480,'N',360,216,'N','N','N','Y','Y','Y','N','N','N','Y','Y'),

(12137,'CSMT FZR PUNJAB MAIL','N',480,138,216,216,'N','N','N','Y','Y','Y','Y','Y','Y','Y','Y'),

(19665,'KURJ UDZ EXP','N',400,130,216,360,'N','N','N','Y','Y','Y','Y','Y','Y','Y','Y');

**CREATE TABLE ticket**

(

`Ticket\_No` INT NOT NULL AUTO\_INCREMENT PRIMARY KEY,

`Train\_No` INT NOT NULL,FOREIGN KEY (Train\_No) REFERENCES train (Train\_No) ON UPDATE CASCADE,

`Date\_of\_Journey` DATE NOT NULL,

`Username` VARCHAR(50) NOT NULL,FOREIGN KEY (Username) REFERENCES account (Username) ON DELETE CASCADE

);

**CREATE TABLE stoppage**

(

`Train\_No` INT NOT NULL,FOREIGN KEY (Train\_No) REFERENCES train (Train\_No) ON DELETE CASCADE ON UPDATE CASCADE,

`Station\_Code` CHAR(10) NOT NULL ,FOREIGN KEY (Station\_Code) REFERENCES station (Station\_Code) ON DELETE CASCADE ON UPDATE CASCADE,

`Arrival\_Time` TIME DEFAULT NULL,

`Departure\_Time` TIME DEFAULT NULL

);

-- alter table stoppage ADD CHECK (EXTRACT(HOUR FROM Arrival\_Time) <24 AND EXTRACT(HOUR FROM Departure\_Time) <24);

INSERT INTO stoppage

VALUES(12001,'BPL','15:27:00','15:30:00'),

(12001,'VGLJ','18:37:00','18:42:00'),

(12001,'GWL','19:40:00','19:45:00'),

(12001,'AGC','21:20:00','21:25:00'),

(12279,'VGLJ',NULL,'15:20:00'),

(12279,'GWL','16:33:00','16:35:00'),

(12279,'MTJ','19:00:00','19:02:00'),

(12279,'NDLS','21:35:00',NULL),

(12137,'DHO','17:20:00','17:22:00'),

(12137,'NZM','20:55:00','20:57:00'),

(15046,'AGC','6:35:00','6:40:00'),

(15046,'CNB','11:30:00','11:35:00'),

(16317,'BPL','9:05:00','9:10:00'),

(16317,'DHO','16:10:00','16:12:00'),

(16317,'NZM','20:16:00','20:21:00'),

(19665,'VGLJ','13:40:00','13:48:00'),

(19665,'JP','22:30:00','22:40:00');

alter table stoppage ADD CHECK (EXTRACT(HOUR FROM Arrival\_Time) <24 AND EXTRACT(HOUR FROM Departure\_Time) <24);

INSERT INTO account

VALUES ('admin','admin@1234','adm19@gmail.com','Old airport road,bangalore');

SELECT a.Train\_No

FROM stoppage as a join stoppage as b on a.Train\_No = b.Train\_No

WHERE a.Station\_Code = "VGLJ" and b.Station\_Code = "NDLS";

DROP TRIGGER cancellation;

drop table passenger;

INSERT INTO passenger

VALUES ('1','priyanshu','mangal','male','8949967310','1','22','first class ac');

# Triggers

A trigger has been created which is invoked each time a ticket is cancelled.The trigger helps in increasing the number of seats in a coach after cancellation.

delimiter //

create trigger cancellation before delete on ticket for each row

BEGIN

set @trainno=old.train\_no; set @ticketno=old.ticket\_no;

SET @class = (SELECT p.class FROM PASSENGER p

WHERE p.ticket\_no = @ticketno); if @class='first class ac' then

UPDATE Train set Seat\_First\_Class\_AC = Seat\_First\_Class\_AC+1 WHERE Train\_No = @trainno ;

elseif @class='sleeper' then

UPDATE Train set Seat\_Sleeper = Seat\_Sleeper+1 WHERE Train\_No = @trainno ; elseif @class='second class ac' then

UPDATE Train set Seat\_Second\_Class\_AC = Seat\_Second\_Class\_AC+1 WHERE Train\_No = @trainno ;

elseif @class='third class ac' then

UPDATE Train set Third\_Class\_AC = Seat\_Third\_Class\_AC+1 WHERE Train\_No = @trainno ;

end if; END// delimiter ;

# SQL Queries

### /\* Find total number of first class seats available on any train that reaches bangalore before 7pm on Monday .\*/

CREATE VIEW c(Station\_code,Train\_no,Arrival\_Time)AS

SELECT Stoppage.Station\_code,Train\_no,Arrival\_Time

FROM Station INNER JOIN Stoppage ON station.Station\_code=Stoppage.Station\_code

WHERE station.Station\_name='VIRANGANA LAKSHMIBAI JHANSI';

SELECT \*

FROM a;

CREATE VIEW b(Station\_code,Train\_no,Arrival\_Time)AS

SELECT Station\_code,Train\_no,Arrival\_Time

FROM a

WHERE EXTRACT(HOUR FROM Arrival\_Time)<19;

SELECT \*

FROM b;

CREATE VIEW c(Station\_code,Train\_no,Arrival\_Time,First\_Class\_seats,Run\_on\_monday)AS

SELECT Station\_code,train.Train\_no,Arrival\_Time,Seat\_First\_Class\_AC,Run\_on\_monday

FROM train INNER JOIN b ON train.Train\_No=b.Train\_No

WHERE train.Run\_On\_Monday='Y' AND train.Seat\_First\_Class\_AC >0;

SELECT \*

FROM c;

SELECT SUM(First\_class\_seats)

FROM c;

### /\* Find the time at which last train leaves New delhi station \*/

**Create View f**(Departure\_time)AS

SELECT Departure\_time

FROM Stoppage

WHERE Station\_Code IN

(SELECT Station\_code

FROM station

WHERE Station\_Name='New Delhi') ; SELECT \*

FROM f;

SELECT **MAX**(Departure\_time)

FROM f**;**

### /\* Find the phone number of the user whose email id is [ajitesh@pes.edu](mailto:ajitesh@pes.edu) \*/

SELECT phone\_no

FROM contact

WHERE

username **IN** (SELECT username FROM account WHERE email\_id='pm01@gmail.com)**;**

Conclusion and Future Scope

Our system can successfully give information on any train, find trains running between two stations, book tickets and cancel tickets. This system could be used for official train booking. However, several other features could be added like booking meals on trains etc. Also payment gateways have to be implemented to make sure the transactions happen securely.