

A Project Report On
Railway Reservation System

Developed By:
DIRGH PATEL (IT-086)
DHRUV PATEL (IT-085)

Department of Information Technology
Faculty of Technology
DDU University



Department of Information Technology
Faculty of Technology, Dharmsinh Desai University
College Road, Nadiad – 387001

DHARMSINH DESAI UNIVERSITY

NADIAD-387001, GUJARAT



CERTIFICATE

This is to certify that the project entitled “**Railway Reservation System**” is a bonafide report of the work carried out by

1) **Mr.Dirgh patel**, Student ID No : **18ITUOS123**

2) **Mr.Dhruv patel**, Student ID No : **18ITUON122**

of the Department of Information Technology, semester V, under the guidance and supervision for the subject Database Management System. They were involved in Project training during the academic year 2020-2021.

Prof. Archana N. Vyas

(Project Guide)

Department of Information Technology,

Faculty of Technology,

Dharmsinh Desai University, Nadiad Date:

31/11/20

TABLE OF CONTENTS

1. SYSTEM OVERVIEW	2
1.1 Current system	2
1.2 Problems	3
1.3 Advantages of the Proposed system (over current)	3
2. E-R DIAGRAM	4
2.1 Entities	4
2.2 Relationships	4
2.3 ER Diagram	5
3. DATA DICTIONARY	6
4. SCHEMA DIAGRAM	9
5. DATABASE IMPLEMENTATION	10
5.1 Create Schema	11
5.2 Insert Data values	15
5.3 Queries	18
5.4 PL/SQL Blocks (Procedures and Functions)	19
5.5 Function	20
5.6 Cursors	21
5.7 Trigger	23
6. BIBLIOGRAPHY	24

1. SYSTEM OVERVIEW

1.1 CURRENT SYSTEM

A Railway Reservation System is one which helps us to reserve or cancel seats in train. When we want to travel from one place to another place. Since it is difficult to maintain a record of all information manually, we have created it with the help of an E-R diagram, many tables. It consists of several steps to be followed and many forms to be maintained i.e for the convenience of the passengers.

1.2 PROBLEMS

- Maintenance of a huge amount of Database manually is a very tough due to which passengers are not getting the proper information.
- Passengers had to wait for a long time standing in a queue for the confirmation of the ticket.
- Passengers can't be able to know the proper arrival and departure time which is really very inconvenient.
- It takes a lot of work and time for cancellation.

1.3 ADVANTAGES OF THE PROPOSED SYSTEM

Through this system, we try to achieve the below objectives:

- To inform users about their reservation status.
- Available seats and price of ticket.
- Schedules of different trains.
- Provides the details of the various stations from the source to the destination to the passenger.
- To Reduce cost, occur on management.

- To prevent users from getting Invalid Information.
- Train Fare
- Travel Distance.

2. E-R DIAGRAM

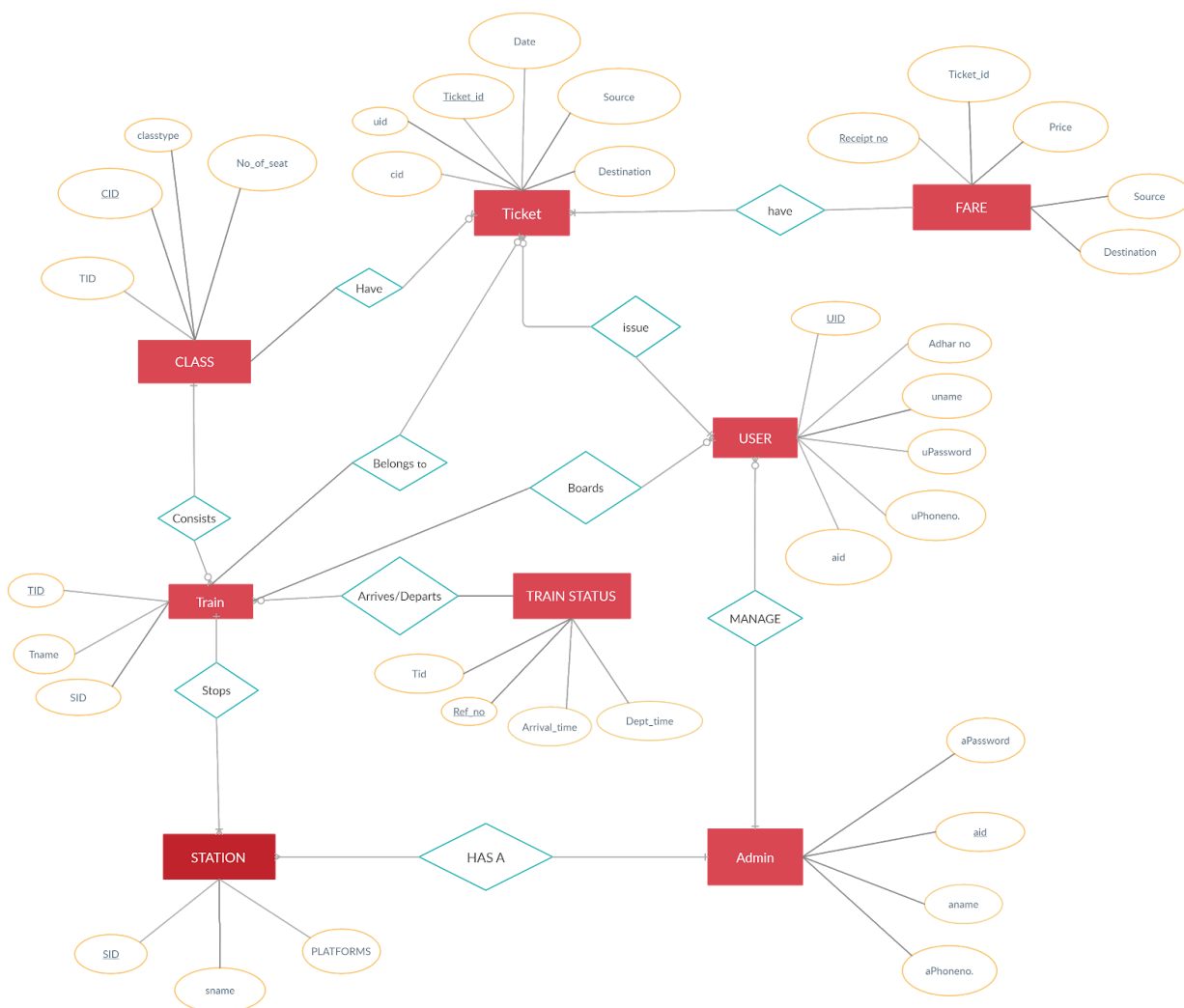
2.1 ENTITIES & ATTRIBUTES

- EMPLOYEE: Phone number, Salary, Employee_id, Name
- USER: Email, Password, Phone number, Name.
- TIME: Arrival and Depart time, Reference number.
- TICKET: Source and Destination, Ticket type, Date, Ticket number,uid
- FARE: Price, Ticket_id, Receipt number.
- STATION: Station number, id, platform.
- TRAIN: Start and stop station, Train_no, Train_name.
- CLASS: Train_id, type, Seats,classtype.

2.2 RELATIONSHIP

- User: Books: Ticket
- Train: Arrives/Departs: Time
- Train: Consists: Class
- Employee: Helps: Users
- Train : Stops: Station
- Ticket: Belongs to: Train

2.3 ER DIAGRAM



3. DATA DICTIONARY

USER1

	Field	Type	Null	Key	Default	Extra
►	uid	varchar(5)	NO	PRI	NULL	
	uname	varchar(10)	NO		NULL	
	uphoneno	decimal(10,0)	NO		NULL	
	adharno	decimal(12,0)	NO		NULL	
	aid	varchar(5)	NO	MUL	NULL	

USER2

	Field	Type	Null	Key	Default	Extra
►	uid	varchar(5)	NO	PRI	NULL	
	upassword	varchar(10)	NO		NULL	

ADMIN1

	Field	Type	Null	Key	Default	Extra
►	aid	varchar(5)	NO	PRI	NULL	
	aname	varchar(5)	NO		NULL	
	aphoneno	int	NO		NULL	

ADMIN2

	Field	Type	Null	Key	Default	Extra
►	aid	varchar(5)	NO	PRI	NULL	
	apassword	varchar(10)	NO		NULL	

FARE

	Field	Type	Null	Key	Default	Extra
►	receipt_no	varchar(5)	NO	PRI	NULL	
	ticket_id	int	NO	MUL	NULL	
	price	int	NO		NULL	

STATION

	Field	Type	Null	Key	Default	Extra
►	sid	int	NO	PRI	NULL	
	sname	varchar(10)	NO		NULL	
	platform	int	NO		NULL	

CLASS

	Field	Type	Null	Key	Default	Extra
►	cid	int	NO	PRI	NULL	
	tid	int	NO	MUL	NULL	
	seats	int	NO		NULL	
	classtype	varchar(20)	NO		NULL	

TICKET

	Field	Type	Null	Key	Default	Extra
►	ticket_id	int	NO	PRI	NULL	
	cid	int	NO	MUL	NULL	
	ticket_date	date	NO		NULL	
	source	varchar(10)	NO		NULL	
	destination	varchar(10)	NO		NULL	
	uid	varchar(5)	YES	MUL	NULL	

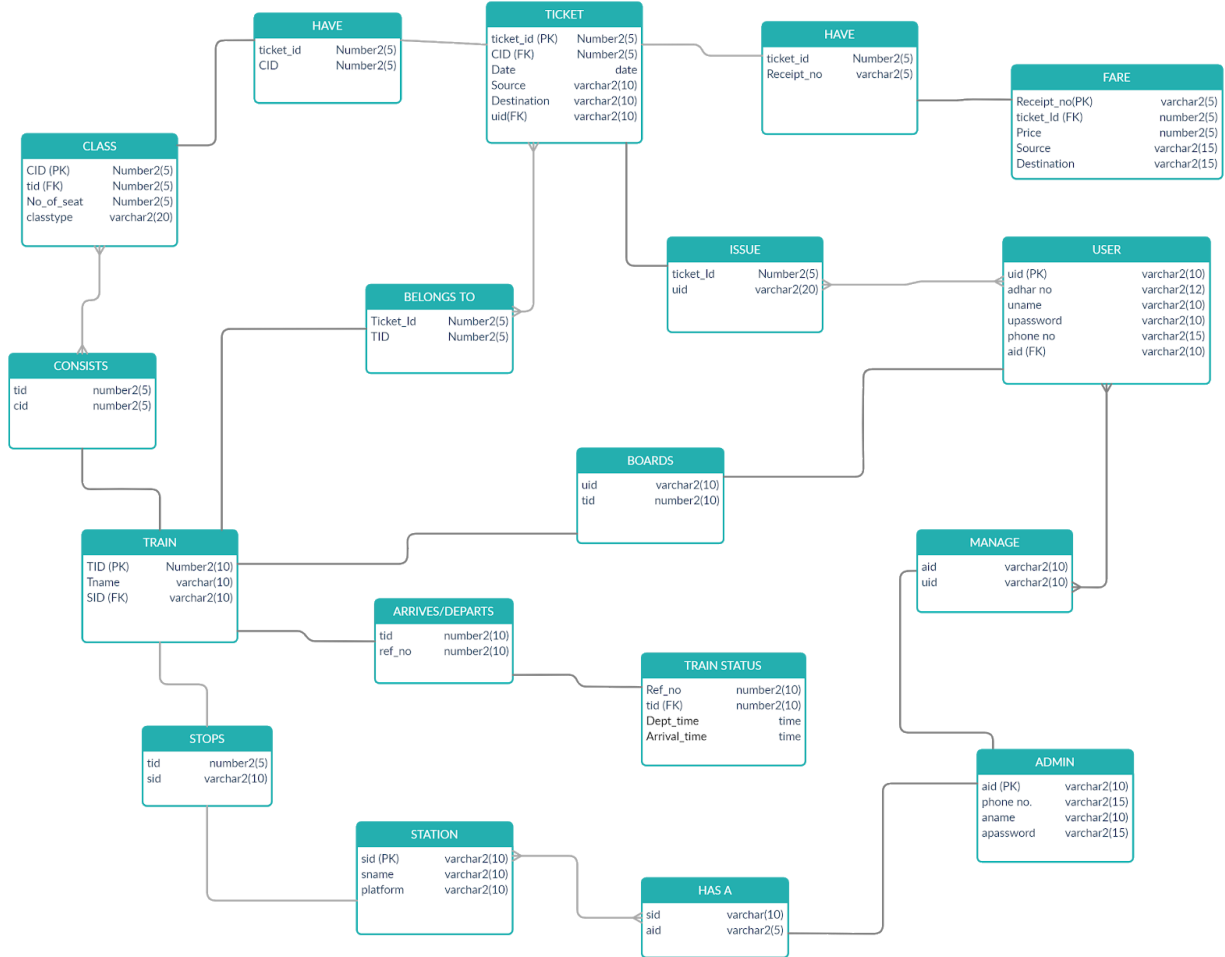
TRAIN

	Field	Type	Null	Key	Default	Extra
►	tid	int	NO	PRI	NULL	
	tname	varchar(30)	NO		NULL	
	sid	int	NO	MUL	NULL	

TRAIN STATUS

	Field	Type	Null	Key	Default	Extra
►	ref_no	varchar(5)	NO	PRI	NULL	
	trainid	int	NO	MUL	NULL	
	dept_time	datetime	NO		NULL	
	arrival_time	datetime	NO		NULL	

4. SCHEMA DIAGRAM



5. DATABASE IMPLEMENTATION

5.1 CREATE SCHEMA

1.ADMIN1

```
CREATE TABLE `railway`.`admin1` (  
  `aid` VARCHAR(5) NOT NULL,  
  `aname` VARCHAR(5) NOT NULL,  
  `aphoneno` INT(10) NOT NULL,  
  PRIMARY KEY (`aid`));
```

	aid	aname	aphoneno
▶	A101	RAJA	1234567890
	A102	Dev	987654321
	A103	Akash	456789123
	A104	Dirgh	605432198
	A105	Dhruv	908745632
*	NULL	NULL	NULL

2. ADMIN2

```
CREATE TABLE `railway`.`admin2` (  
  `aid` VARCHAR(5) NOT NULL,  
  `password` VARCHAR(10) NOT NULL,  
  PRIMARY KEY (`aid`));
```

	aid	apassword
▶	A101	password
	A102	123456
	A103	football
	A104	qwerty
	A105	welcome
*	NULL	NULL

3.USER1

```
CREATE TABLE `railway`.`user1` (  
  `uid` VARCHAR(5) NOT NULL,  
  `uname` VARCHAR(10) NOT NULL,  
  `uphoneno` INT(10) NOT NULL,  
  `adharno` INT(10) NOT NULL,
```

```

`aid` VARCHAR(5) NOT NULL,
PRIMARY KEY (`uid`),
INDEX `aid_idx` (`aid` ASC) VISIBLE,
CONSTRAINT `aid`
FOREIGN KEY (`aid`)
REFERENCES `railway`.`admin1` (`aid`)
ON DELETE NO ACTION
ON UPDATE NO ACTION);

```

	uid	uname	uphoneno	adharno	aid
►	U101	Ebrahim	8888888888	123456789987	A101
	U102	Arjun	9537678894	857694523641	A102
	U103	Bheem	8140656906	147258369963	A102
	U104	Krishna	9054203318	741258963258	A105
	U105	Hanuman	6355649653	963258741258	A103
	U106	Ram	8522589631	753159842684	A104
	U107	Kumbhkaran	7532648922	987456258963	A105
	U108	kiran	9636985258	222333456789	A104
*	NULL	NULL	NULL	NULL	NULL

4. USER2

```

CREATE TABLE `railway`.`user2` (
  `uid` INT NOT NULL,
  `upassword` VARCHAR(10) NOT NULL,
  PRIMARY KEY (`uid`));

```

	uid	upassword
►	U101	dragon
	U102	starwars
	U103	solo
	U104	monkey
	U105	welcome
*	NULL	NULL

5. FARE

```

CREATE TABLE `railway`.`fare` (
  `receipt_no` VARCHAR(5) NOT NULL,
  `ticket_id` INT(5) NOT NULL,

```

```

`price` INT(5) NOT NULL,
PRIMARY KEY (`receipt_no`),
INDEX `ticket_id_idx` (`ticket_id` ASC) VISIBLE,
CONSTRAINT `ticket_id`
  FOREIGN KEY (`ticket_id`)
    REFERENCES `railway`.`ticket` (`ticket_id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION);

```

	receipt_no	ticket_id	price
►	F101	20101	50
	F102	20102	75
	F103	20103	75
	F104	20104	60
	F105	20105	100
	F106	20106	100
	F107	20107	80
	F108	20108	80
	F109	20109	90
	F110	20110	40
•	NULL	NULL	NULL

6. TICKET

```

CREATE TABLE `railway`.`ticket` (
  `ticket_id` INT(5) NOT NULL,
  `cid` INT(5) NOT NULL,
  `ticket_date` DATE NOT NULL,
  `source` VARCHAR(10) NOT NULL,
  `destination` VARCHAR(10) NOT NULL,
  PRIMARY KEY (`ticket_id`),
  INDEX `cid_idx` (`cid` ASC) VISIBLE,
  CONSTRAINT `cid`
    FOREIGN KEY (`cid`)
      REFERENCES `railway`.`class` (`cid`)

```

	ticket_id	cid	ticket_date	source	destination	uid
▶	20101	30101	2001-01-23	Vadodara	Ahmedabad	U101
	20102	30103	2001-06-15	Nadiad	Ahmedabad	U102
	20103	30103	2001-06-15	Nadiad	Ahmedabad	U102
	20104	30102	2001-07-26	Vadodara	Surat	U103
	20105	30101	2001-11-06	Anand	Surat	U104
	20106	30101	2001-11-06	Anand	Surat	U104
	20107	30104	2001-05-07	Nadiad	Anand	U105
	20108	30104	2001-04-08	Anand	Nadiad	U106
	20109	30103	2001-03-15	Anand	Ahmedabad	U107
	20110	30102	2001-02-16	Surat	Nadiad	U108
•	NULL	NULL	NULL	NULL	NULL	NULL

7. CLASS

```
CREATE TABLE `railway`.`class` (
  `cid` INT(5) NOT NULL,
  `tid` INT(5) NOT NULL,
  `seats` INT(5) NOT NULL,
  PRIMARY KEY (`cid`),
  INDEX `tid_idx` (`tid` ASC) VISIBLE,
  CONSTRAINT `tid`
    FOREIGN KEY (`tid`)
      REFERENCES `railway`.`train` (`tid`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION);
```

	cid	tid	seats	classtype
▶	30101	5050	15	First Class
	30102	5052	20	Second Class
	30103	5053	25	First Class
	30104	5051	35	First Class
	30105	5054	30	Second Class
•	NULL	NULL	NULL	NULL

8. TRAIN

```
CREATE TABLE `railway`.`train` (
  `tid` INT(5) NOT NULL,
  `tname` VARCHAR(10) NOT NULL,
```

```

`sid` INT(5) NOT NULL,
PRIMARY KEY (`tid`),
INDEX `sid_idx` (`sid` ASC) VISIBLE,
CONSTRAINT `sid`
  FOREIGN KEY (`sid`)
    REFERENCES `railway`.`station` (`sid`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION);

```

	tid	tname	sid
▶	5050	Jan Shatabdi Express	10102
	5051	Udyan Express	10101
	5052	Surat Mail	10104
	5053	Intercity Express	10103
	5054	Memoo	10105
★	NULL	NULL	NULL

9. STATION

```

CREATE TABLE `railway`.`station` (
  `sid` INT(5) NOT NULL,
  `sname` VARCHAR(10) NOT NULL,
  `platform` INT NOT NULL,
  PRIMARY KEY (`sid`));

```

	sid	sname	platform
▶	10101	Nadiad	3
	10102	Ahmedabad	12
	10103	Vadodara	10
	10104	Surat	8
	10105	Anand	5
★	NULL	NULL	NULL

10. TRAIN STATUS

```

CREATE TABLE `railway`.`train_status` (
  `ref_no` INT NOT NULL,
  `trainid` INT(5) NOT NULL,
  `dept_time` DATETIME NOT NULL,
  `arrival_time` DATETIME NOT NULL,
  PRIMARY KEY (`ref_no`),
  INDEX `trainid_idx` (`trainid` ASC) VISIBLE,
  CONSTRAINT `trainid`
    FOREIGN KEY (`trainid`)
      REFERENCES `railway`.`train` (`tid`)

```

ON DELETE NO ACTION
ON UPDATE NO ACTION);

	ref_no	trainid	dept_time	arrival_time
▶	R 101	5050	2020-01-05 05:05:05	2020-01-05 05:05:05
	R 102	5051	2001-10-05 04:05:10	2020-09-05 05:05:05
	R 103	5052	2001-12-05 06:05:04	2020-10-05 05:05:05
	R 104	5053	2001-11-05 07:05:06	2020-10-05 05:05:05
	R 105	5054	2001-09-05 11:11:11	2020-08-05 10:10:10
*	NULL	NULL	NULL	NULL

5.2 INSERT DATA VALUES

ADMIN1 - ADMIN2:

```
insert into admin1 values("A101","RAJA",1234567890);
insert into admin2 values("A101","password");
insert into admin1 values("A102","Dev",987654321);
insert into admin2 values("A102","123456");
insert into admin1 values("A103","Akash",456789123);
insert into admin2 values("A103","football");
insert into admin1 values("A104","Dirgh",605432198);
insert into admin2 values("A104","qwerty");
insert into admin1 values("A105","Dhruv",908745632);
insert into admin2 values("A105","welcome");
select * from admin1;
select * from admin2;
delete from admin1 where aname = "RAJA";
delete from admin1 where aname = "RAJA";
```

USER1:

```
insert into user1 values("U101","Ebrahim","8888888888","123456789987","A101");
insert into user1 values("U102","Arjun","9537678894","857694523641","A102");
insert into user1 values("U103","Bheem","8140656906","147258369963","A102");
insert into user1 values("U104","Krishna","9054203318","741258963258","A105");
insert into user1 values("U105","Hanuman","6355649653","963258741258","A103");
```

```
insert into user1 values("U106","Ram","8522589631","753159842684","A104");
insert into user1 values("U107","Kumbhkaran","7532648922","987456258963","A105");
insert into user1 values("U108","kiran","9636985258","222333456789","A104");
```

USER2:

```
insert into user2 values("U101","dragon");
insert into user2 values("U102","starwars");
insert into user2 values("U103","solo");
insert into user2 values("U104","monkey");
insert into user2 values("U105","welcome");
select * from user2;
```

STATION:

```
insert into station values(10101,"Nadiad",3);
insert into station values(10102,"Ahmedabad",12);
insert into station values(10103,"Vadodara",10);
insert into station values(10104,"Surat",8);
insert into station values(10105,"Anand",5);
select * from station;
```

TRAIN:

```
insert into train values(5050,"Jan Shatabdi Express",10102);
insert into train values(5051,"Udyan Express",10101);
insert into train values(5052,"Surat Mail",10104);
insert into train values(5053,"Intercity Express",10103);
insert into train values(5054,"Memoo",10105);
insert into train values(5055,"Chennai Express",10101);
insert into train values(5056,"Madras Express",10102);
```

```
select * from train;
```

CLASS:

```
insert into class values(30101,5050,15,"First Class");
insert into class values(30102,5052,20,"Second Class");
insert into class values(30103,5053,25,"First Class");
insert into class values(30104,5051,35,"First Class");
```



```
insert into class values(30105,5054,30,"Second Class");
```

TICKET:

```
insert into ticket values(20101,30101,'2001-01-23',"Vadodara","Ahmedabad","U101");
insert into ticket values(20102,30103,'2001-06-15',"Nadiad","Ahmedabad","U102");
insert into ticket values(20103,30103,'2001-06-15',"Nadiad","Ahmedabad","U102");
insert into ticket values(20104,30102,'2001-07-26',"Vadodara","Surat","U103");
insert into ticket values(20105,30101,'2001-11-06',"Anand","Surat","U104");
insert into ticket values(20106,30101,'2001-11-06',"Anand","Surat","U104");
insert into ticket values(20107,30104,'2001-05-07',"Nadiad","Anand","U105");
insert into ticket values(20108,30104,'2001-04-08',"Anand","Nadiad","U106");
insert into ticket values(20109,30103,'2001-03-15',"Anand","Ahmedabad","U107");
insert into ticket values(20110,30102,'2001-02-16',"Surat","Nadiad","U108");
select * from ticket;
```

FARE:

```
insert into fare values("F101",20101,50);
insert into fare values("F102",20102,75);
insert into fare values("F103",20103,75);
insert into fare values("F104",20104,60);
insert into fare values("F105",20105,100);
insert into fare values("F106",20106,100);
insert into fare values("F107",20107,80);
insert into fare values("F108",20108,80);
insert into fare values("F109",20109,90);
insert into fare values("F110",20110,40);
select * from fare;
```

TRAIN_STATUS:

```
insert into train_status values ("R101",5050,'2020-01-05 05-05-05','2020-01-05 05-05-05');
insert into train_status values ("R102",5051,'2001-10-05 04-05-10','2020-09-05 05-05-05');
insert into train_status values ("R103",5052,'2001-12-05 06-05-04','2020-10-05 05-05-05');
insert into train_status values ("R104",5053,'2001-11-05 07-05-06','2020-10-05 05-05-05');
insert into train_status values ("R105",5054,'2001-09-05 11-11-11','2020-08-05 10-10-10');
```

5.3 QUERIES

1. List name of users whose name start with

select uname from user1 where uname like 'd%';

	uname
▶	Krishna
	Kumbhkaran
	kiran

2. List name of train status in descending order.

select * from train_status order by arrival_time desc;

	ref_no	trainid	dept_time	arrival_time
▶	R103	5052	2001-12-05 06:05:04	2020-10-05 05:05:05
	R104	5053	2001-11-05 07:05:06	2020-10-05 05:05:05
	R102	5051	2001-10-05 04:05:10	2020-09-05 05:05:05
	R105	5054	2001-09-05 11:11:11	2020-08-05 10:10:10
	R101	5050	2020-01-05 05:05:05	2020-01-05 05:05:05
*	NULL	NULL	NULL	NULL

3. Count total no of seats from a train having first-class as class type.

select sum(seats) as total seats from class where classtype="first-class";

	sum(seats)
▶	75

4. List all prices ranging from

select * from fare where price between 80 and 100;

	receipt_no	ticket_id	price
▶	F105	20105	100
	F106	20106	100
	F107	20107	80
	F108	20108	80
	F109	20109	90
*	NULL	NULL	NULL

5. Find the minimum and maximum cost of a ticket.

select min(price) as "Min_price", max(price) as "Max_price" from fare;

	Min_price	Max_price
▶	40	100

6. Display all admin which works for appropriate users.

SELECT B.ANAME AS "ADMIN WORKS FOR -" ,A.UNAME AS "USER " FROM
USERS A INNER JOIN ADMIN B ON A.UID=B.AID;

	ADMIN WORKS FOR -	USER
▶	RAJA	Ebrahim
	Dev	Arjun
	Dev	Bheem
	Akash	Hanuman
	Dirgh	Ram
	Dirgh	kiran
	Dhruv	Krishna
	Dhruv	Kumbhakaran

7. List all Train name and its arrival and departure time by using train & status table.

SELECT T.tid as "Train no" , T.Tname, TS.Dept_time, TS.Arrival_Time FROM
Train T left JOIN TRAIN_STATUS TS ON T.TID = TS.trainid ORDER BY T.Tid;

	Train no	Tname	Dept_time	Arrival_Time
▶	5050	Jan Shatabdi Express	2020-01-05 05:05:05	2020-01-05 05:05:05
	5051	Udyan Express	2001-10-05 04:05:10	2020-09-05 05:05:05
	5052	Surat Mail	2001-12-05 06:05:04	2020-10-05 05:05:05
	5053	Intercity Express	2001-11-05 07:05:06	2020-10-05 05:05:05
	5054	Memoo	2001-09-05 11:11:11	2020-08-05 10:10:10
	5055	Chennai Express	NULL	NULL
	5056	Madras Express	NULL	NULL

8. Details of Trains Which can come at the same station.

SELECT A.SID,A.TID AS Train1 FROM Train A, Train B WHERE A.TID != B.TID
AND A.SID = B.SID ORDER BY A.SID;

	SID	Train1
▶	10101	5051
	10101	5055
	10102	5050
	10102	5056

9. Trains having second class compartment.

select train.tname as "Train Name",train.tid as "Train No" from train,class where class.classtype = "Second Class" and train.tid = class.tid

	Train Name	Train No
▶	Surat Mail	5052
	Memoo	5054

10. Display ticket id whose price is the maximum among all.

select ticket_id as "Ticket Id",price as "Max price" from fare f1 where price = (select max(price) from fare f2);

	Ticket Id	Max price
▶	20105	100
	20106	100

5.4 PL/SQL

THIS PL/SQL GIVE USER ID, ADMIN ID,ADHAR NO AND PHONE NUMBER OF KRISHNA.

declare

v_user user1 % rowtype;

begin

select username,userid,uphoneno,adharno,aid into

v_user.username,v_user.userid,v_user.uphoneno,v_user.adharno,v_user.aid from user1 where username='Krishna';

dbms_output.put_line(v_user.username||' '||v_user.userid||' '||v_user.uphoneno||' '||v_user.adharno||' '||v_user.aid);

End;

```
Statement processed.  
Krishna U104 9054203318 741258963258 A105
```

5.5 FUNCTION

A function which counts the total number of seats.

```
CREATE OR REPLACE FUNCTION totalSeats RETURN  
number IS  
total number(3) := 0;  
BEGIN  
    SELECT sum(seats) into total FROM class;  
    RETURN total;  
END;  
  
BEGIN  
    dbms_output.put_line('Total seats =' || totalSeats);  
END;  
/
```

```
Statement processed.  
Total Seats = 125
```

5.6 CURSORS

This Cursor Gives The Information About People AdharCard number And Their PHONE NUMBER whose NAME starts with either 'A' or 'K' .

```
declare  
    d1 user1.adharno % type;  
    d2 user1.uphoneno % type;
```

```

cursor c1 is select adhar no, uphone no from user1 where
                username like 'A%'
                or username like 'K%';
begin
dbms_output.put_line('Adhar no'||      '||'Phone Number');
    open c1;
    loop
    fetch c1 into d1,d2;
    exit when c1 % notfound;
    dbms_output.put_line(d1||      '||d2);
    end loop;
    close c1;
End;

```

```

statement processed.
Adhar no      Phone Number
857694523641  9537678894
741258963258  9054203318
987456258963  7532648922
889875523510  9824262642
222333456789  9636985258

```

5.7 TRIGGERS

1.This trigger updates price value by 200 whenever new values is Inserted into fare.

```

create trigger farex
before insert
on fare
for each row
set new.price = new.price + 200;

```

BEFORE:

	receipt_no	ticket_id	price
▶	F101	20101	50
	F102	20102	75
	F103	20103	75
	F104	20104	60
	F105	20105	100
	F106	20106	100
	F107	20107	80
	F108	20108	80
	F109	20109	90
	F110	20110	40
*	NULL	NULL	NULL

insert into ticket values(20111,30103,'2001-02-16','Surat','Nadiad','U108');
insert into fare values('F111',20111,500);

AFTER:

	receipt_no	ticket_id	price
	F101	20101	50
	F102	20102	75
	F103	20103	75
	F104	20104	60
	F105	20105	100
	F106	20106	100
	F107	20107	80
	F108	20108	80
	F109	20109	90
	F110	20110	40
▶	F111	20111	700
*	NULL	NULL	NULL

2. This updates the arrival time of train and creates a new table which consists of train's previous and current arrival time.

```
CREATE TABLE Timechanges (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    trainid INT,  
    previoustime datetime,  
    currenttime datetime,  
    changedAt TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP  
);
```

```

DROP TRIGGER IF EXISTS `railway`.`train_status_AFTER_UPDATE`;

DELIMITER $$
USE `railway`$$
CREATE DEFINER=`root`@`localhost` TRIGGER `train_status_AFTER_UPDATE` AFTER
UPDATE ON `train_status` FOR EACH ROW BEGIN
    IF OLD.arrival_time <> new.arrival_time THEN
        INSERT INTO Timechanges(trainid, previoustime, currenttime)
        VALUES(old.trainid, old.arrival_time, new.arrival_time);
    END IF;
END$$
DELIMITER ;

```

update train_status SET arrival_time='2020-01-06 12:05:47' where trainid=5051;

	id	trainid	previoustime	currenttime	changedAt
▶	1	5051	2020-09-05 05:05:05	2020-01-06 12:05:47	2020-10-31 00:56:38
*	NULL	NULL	NULL	NULL	NULL

6. BIBLIOGRAPHY

- ❖ For the implementation of this Railway Reservation System project we referred to many websites and books.
- ❖ We created the ER Diagram and Schema Diagram on "<https://creately.com/>".

Reference Website:

1. <https://www.w3schools.com>
2. <https://www.mysqltutorial.org>
3. <https://stackoverflow.com/>
4. <https://www.w3resource.com>

Submitted By,

IT085- Dhruv Patel

IT086- Dirgh Patel