“South Central Railway Database Project”

(A MySQL Relational Database Management Project)

Presented By:

R.YERRISWAMI
G. BRAMHA TEJA REDDY
G. VARAKRISHNA YADAV
P. DATTATREYA
M.SREELEKHA
M.NEERAJA

(Team members)

ABSTRACT

This project was built to create a structured database system for the South Central Railway. It contains ten interconnected tables covering districts, stations, departments, employees, platforms, trains, schedules, tickets, and complaints.

The database ensures accurate storage of information, avoids duplication, and makes it easy to run queries for reports and analysis. With this system, railway authorities can better manage staff, monitor train operations, handle passenger bookings, and resolve complaints more effectively.

Overall, the project demonstrates how databases improve efficiency, support decision-making, and provide a foundation for future digital railway services.

INTRODUCTION

Railways play a vital role in connecting people and supporting economic growth in India. Managing stations, trains, employees, tickets, and complaints requires a structured and reliable system.

The South Central Railway Database Project was developed using MySQL to organize and manage railway operations in Andhra Pradesh. It ensures accurate data storage, easy retrieval, and smooth coordination between different departments and stations.

Entity-Relationship Diagram:

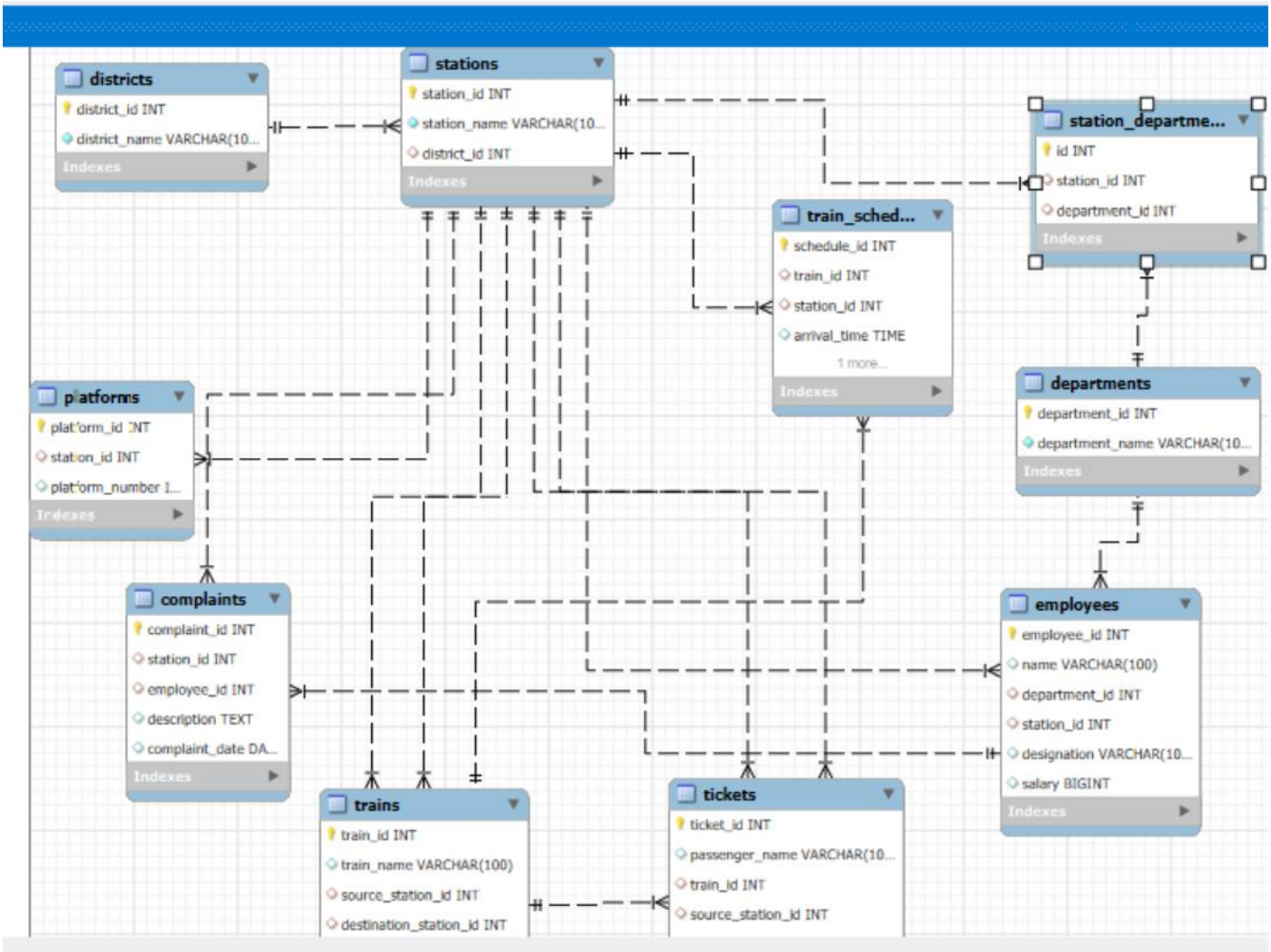

Table 1: districts

Master list of districts in Andhra Pradesh.

Column	Type	Constraints
district_id	INT	PRIMARY KEY
district_name	VARCHAR(100)	NOT NULL

Complete Sample Data

district_id	district_name
1	Guntur
2	Krishna
3	Nellore
4	Prakasam
5	Visakhapatnam
6	Vijayanagaram
7	Kurnool
8	Chittoor
9	Anantapur
10	Kadapa

```
                        -- TABLE:1--
-- to create a table with help of DDL commands--syntax: create table table_name (col1,co12..)---

create table districts (district_id int primary key,district_name varchar(100) not null);

-- to insert values in a table with help of DML commands--syntax: insert into table_name(col1,col2..) values (val1,val2..)--

insert into districts (district_id, district_name) values
(1, 'guntur'),
(2, 'krishna'),
(3, 'nellore'),
(4, 'prakasam'),
(5, 'visakhapatnam'),
(6, 'vijayanagaram'),
(7, 'kurnool'),
(8, 'chittoor'),
(9, 'anantapur'),
(10, 'kadapa');
```

Table 2: stations

Station catalog. (Note: sample insert omitted district_id; populate as needed.)

Column	Type	Constraints
station_id	INT	PRIMARY KEY
station_name	VARCHAR(100)	NOT NULL
district_id	INT	FOREIGN KEY → districts(district_id)

Complete Sample Data

station_id	station_name
1	guntur junction
2	vijayawada junction
3	nellore
4	ongole
5	visakhapatnam
6	vizianagaram
7	kurnool city
8	tirupati
9	anantapur
10	Kadapa

```
-- TABLE:2--
-- create a table with table name is "stations"--
-- A foreign key is a column in one table that links to the primary key of another table--

create table stations (station_id int primary key,station_name varchar(100) not null,district_id int,
    foreign key (district_id) references districts(district_id));

-- insert the values ina stations table--
insert into stations (station_id, station_name) values
(1, 'guntur junction'),
(2, 'vijayawada junction'),
(3, 'nellore'),
(4, 'ongole'),
(5, 'visakhapatnam'),
(6, 'vizianagaram'),
(7, 'kurnool city'),
(8, 'tirupati'),
(9, 'anantapur'),
(10, 'kadapa');
```

Table 3: departments

Functional departments within the railway organization.

Column	Type	Constraints
department_id	INT	PRIMARY KEY
department_name	VARCHAR(100)	NOT NULL

Complete Sample Data

department_id	department_name
1	operations
2	maintenance
3	safety
4	signaling
5	electrical
6	mechanical
7	commercial
8	medical
9	personnel
10	security

```
-- TABLE:3--
create table departments (department_id int primary key,department_name varchar(100) not null);

insert into departments (department_id, department_name) values
(1, 'operations'),
(2, 'maintenance'),
(3, 'safety'),
(4, 'signaling'),
(5, 'electrical'),
(6, 'mechanical'),
(7, 'commercial'),
(8, 'medical'),
(9, 'personnel'),
(10, 'security');
```

Table 4: station_departments

Mapping of which departments operate at which stations.

Column	Type	Constraints
id	INT	PRIMARY KEY
station_id	INT	FOREIGN KEY → stations(station_id)
department_id	INT	FOREIGN KEY → departments(department_id)

Complete Sample Data

id	station_id	department_id
1	1	1
2	1	2
3	2	1
4	2	3
5	3	1
6	3	4
7	4	1
8	5	2
9	6	5
10	7	6
11	8	7
12	9	8
13	10	9
14	10	10

```
-- TABLE:4--

create table station_departments (id int primary key,station_id int,department_id int,
    foreign key (station_id) references stations(station_id),
    foreign key (department_id) references departments(department_id));

insert into station_departments (id, station_id, department_id) values
(1, 1, 1),(2, 1, 2),(3, 2, 1),(4, 2, 3),(5, 3, 1),(6, 3, 4),(7, 4, 1),(8, 5, 2),(9, 6, 5),
(10, 7, 6),(11, 8, 7),(12, 9, 8),(13, 10, 9),(14, 10, 10);
```

Table 5: employees

Employee details with departments, postings, roles, and salaries.

Column	Type	Constraints
employee_id	INT	PRIMARY KEY
name	VARCHAR(100)	–
department_id	INT	FOREIGN KEY → departments(department_id)
station_id	INT	FOREIGN KEY → stations(station_id)
designation	VARCHAR(100)	–
salary	BIGINT	–

Complete Sample Data

employee_id	name	department_id	station_id	designation	salary
1	raju	1	1	station master	55000
2	sita	2	1	engineer	60000
3	kiran	3	2	safety officer	52000
4	mohan	4	3	signaler	48000
5	anita	5	4	electrician	45000
6	suresh	6	5	mechanic	47000
7	reena	7	6	ticket checker	40000
8	vijay	8	7	doctor	75000
9	latha	9	8	hr officer	50000
10	arun	10	10	rpf officer	48000

```
-- TABLE:5--

create table employees (employee_id int primary key,name varchar(100),department_id int,station_id int,designation varchar(100),
salary bigint,
    foreign key (department_id) references departments(department_id),
    foreign key (station_id) references stations(station_id));

insert into employees (employee_id, name, department_id, station_id, designation, salary) values
(1, 'raju', 1, 1, 'station master', 55000),
(2, 'sita', 2, 1, 'engineer', 60000),
(3, 'kiran', 3, 2, 'safety officer', 52000),
(4, 'mohan', 4, 3, 'signaler', 48000),
(5, 'anita', 5, 4, 'electrician', 45000),
(6, 'suresh', 6, 5, 'mechanic', 47000),
(7, 'reena', 7, 6, 'ticket checker', 40000),
(8, 'vijay', 8, 7, 'doctor', 75000),
(9, 'latha', 9, 8, 'hr officer', 50000),
(10, 'arun', 10, 10, 'rpf officer', 48000);
```

Table 6: platforms

Platforms available at each station.

Column	Type	Constraints
platform_id	INT	PRIMARY KEY
station_id	INT	FOREIGN KEY → stations(station_id)
platform_number	INT	

Complete Sample Data

platform_id	station_id	platform_number
1	1	1
2	1	2
3	2	1
4	2	2
5	3	1
6	4	1
7	5	1
8	6	1
9	7	1
10	8	1
11	9	1
12	10	1

```
-- TABLE:6--

create table platforms (platform_id int primary key,station_id int,platform_number int,
    foreign key (station_id) references stations(station_id));

insert into platforms (platform_id, station_id, platform_number) values
(1, 1, 1),(2, 1, 2),(3, 2, 1),(4, 2, 2),(5, 3, 1),(6, 4, 1),(7, 5, 1),
(8, 6, 1),(9, 7, 1),(10, 8, 1),(11, 9, 1),(12, 10, 1);
```

Table 7: trains

Train roster and origin/destination stations.

Column	Type	Constraints
train_id	INT	PRIMARY KEY
train_name	VARCHAR(100)	
source_station_id	INT	FOREIGN KEY → stations(station_id)
destination_station_id	INT	FOREIGN KEY → stations(station_id)

Complete Sample Data

train_id	train_name	source_station_id	destination_station_id
1	guntur express	1	2
2	coastal runner	2	5
3	tirupati intercity	8	1
4	vizag special	5	3
5	kurnool fast	7	4

```
-- TABLE:7--

create table trains (train_id int primary key,train_name varchar(100),source_station_id int,destination_station_id int,
    foreign key (source_station_id) references stations(station_id),
    foreign key (destination_station_id) references stations(station_id));

    select * from trains;

insert into trains (train_id, train_name, source_station_id, destination_station_id) values
(1, 'guntur express', 1, 2),
(2, 'coastal runner', 2, 5),
(3, 'tirupati intercity', 8, 1),
(4, 'vizag special', 5, 3),
(5, 'kurnool fast', 7, 4);
```

Table 8: train_schedule

Stop-wise arrival and departure times for trains.

Column	Type	Constraints
schedule_id	INT	PRIMARY KEY
train_id	INT	FOREIGN KEY → trains(train_id)
station_id	INT	FOREIGN KEY → stations(station_id)
arrival_time	TIME	
departure_time	TIME	

Complete Sample Data

schedule_id	train_id	station_id	arrival_time	departure_time
1	1	1	08:00:00	08:10:00
2	1	2	10:00:00	10:10:00
3	2	2	09:00:00	09:05:00
4	2	5	12:00:00	12:05:00

```
-- TABLE:8--

create table train_schedule (schedule_id int primary key,train_id int,station_id int,arrival_time time,departure_time time,
    foreign key (train_id) references trains(train_id),
    foreign key (station_id) references stations(station_id));

insert into train_schedule (schedule_id, train_id, station_id, arrival_time, departure_time) values
(1, 1, 1, '08:00:00', '08:10:00'),
(2, 1, 2, '10:00:00', '10:10:00'),
(3, 2, 2, '09:00:00', '09:05:00'),
(4, 2, 5, '12:00:00', '12:05:00');
```

Table 9: tickets

Passenger bookings and journey dates.

Column	Type	Constraints
ticket_id	INT	PRIMARY KEY
passenger_name	VARCHAR(100)	
train_id	INT	FOREIGN KEY → trains(train_id)
source_station_id	INT	FOREIGN KEY → stations(station_id)
destination_station_id	INT	FOREIGN KEY → stations(station_id)
date_of_journey	DATE	

Complete Sample Data

ticket_id	passenger_name	train_id	source_station_id	destination_station_id	date_of_journey
1	ravi	1	1	2	2025-08-08
2	meena	2	2	5	2025-08-08

```
-- TABLE:9--

create table tickets (ticket_id int primary key,passenger_name varchar(100),train_id int,source_station_id int,
    destination_station_id int,date_of_journey date,
    foreign key (train_id) references trains(train_id),
    foreign key (source_station_id) references stations(station_id),
    foreign key (destination_station_id) references stations(station_id));

insert into tickets (ticket_id, passenger_name, train_id, source_station_id, destination_station_id, date_of_journey) values
(1, 'ravi', 1, 1, 2, '2025-08-08'),
(2, 'meena', 2, 2, 5, '2025-08-08');
```

Table 10: complaints

Issue logging by station and employee.

Column	Type	Constraints
complaint_id	INT	PRIMARY KEY
station_id	INT	FOREIGN KEY → stations(station_id)
employee_id	INT	FOREIGN KEY → employees(employee_id)
description	TEXT	–
complaint_date	DATE	–

Complete sample data:

complaint_id	station_id	employee_id	description	complaint_date
1	1	1	delay in train arrival	2025-08-06
2	5	5	electrical issue at platform	2025-08-07

```
-- TABLE:10--

create table complaints (complaint_id int primary key,station_id int,employee_id int,description text,complaint_date date,
    foreign key (station_id) references stations(station_id),
    foreign key (employee_id) references employees(employee_id));

insert into complaints (complaint_id, station_id, employee_id, description, complaint_date) values
(1, 1, 1, 'delay in train arrival', '2025-08-06'),
(2, 5, 5, 'electrical issue at platform', '2025-08-07');
```

Conclusion

The South Central Railway Database Project shows how a relational database can effectively manage railway operations. By connecting districts, stations, departments, employees, trains, schedules, tickets, and complaints, the system keeps information organized and accurate.

This project ensures that railway data can be stored without duplication, retrieved quickly, and used for decision-making. It also helps in monitoring staff, tracking trains, managing passenger bookings, and handling complaints in a systematic way.

Overall, the project proves that databases are a powerful tool for improving efficiency, reducing errors, and supporting better services in the railway sector.