**ELECTION MANAGEMENT SYSTEM**

**A Case Study Submitted to**

**DEPARTMENT**

**OF**

**COMPUTER SCIENCE AND SYSTEMS ENGINEERING**

***Submitted by***

**PAMISETTY.NAVYA 22125A1510**

**SHAIK.NAYAZ UR REHAMAN 22125A1514**

*Under the Guidance of*

**M. P. Yogendra Prasad**

Assistant Professor



**Department of Computer Science and Systems Engineering Sree Vidyanikethan Engineering College (Autonomous)**

Sree Sainath Nagar, Tirupati – 517 102

(2022-2023)

**SREE VIDYANIKETHAN ENGINEERING COLLEGE** 

**(AUTONOMOUS)**

**Sree Sainath Nagar, Tirupati**

**DEPARTMENT OF COMPUTER SCIENCE AND SYSTEMS ENGINEERING**

**CERTIFICATE**

This is to certify that the Case Study report entitled

**ELECTION MANAGEMENT SYSTEM**

is the Bonafide work done by

**PEMISETTY.NAVYA 22125A1510**

**SHAIK.NAYAZ UR REHAMAN 22125A1514**

in the Department of **Computer Science and Systems Engineering**, and submitted to Computer Science and Systems Engineering during the academic year 2022-2023. This work has been carried out under my supervision.

***Guide: Head:***

P. Yogendra Prasad Dr. K. Ramani

Assistant Professor Professor & Head

Dept. of CSSE Dept. of CSSE

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**DEPARTMENT OF COMPUTER SCIENCE AND SYSTEMS ENGINEERING**

**VISION**

To become a centre of excellence in Computer Sciences and Systems Engineering through teaching, training, research and innovation to create quality engineering professionals who can solve the growing complex problems of the society.

**MISSION**

* Established with the cause of development of technical education in advanced computer sciences and engineering with applications to systems there by serving the society and nation.
* Transfer of Knowledge through contemporary curriculum and fostering faculty and student development.
* Create keen interest for research and innovation among students and faculty by understanding the needs of the society and industry.
* Skill development among diversity of students in technical domains and profession for development of systems and processes to meet the demands of the industry and research.
* Imbibing values and ethics in students for prospective and promising engineering profession and develop a sense of respect for all.

**PROGRAM EDUCATIONAL OBJECTIVES**

1. Demonstrate competencies in the Computer Science domain and Management with an ability to comprehend, analyze, design and create software systems for pursuing advanced studies in the areas of interest.

2. Evolve as entrepreneurs or be employed by acquiring required skill sets for developing computer systems and solutions in multi-disciplinary areas.

3. Exhibit progression and professional skill development in Computer programming and systems development with ethical attitude through life-long learning.

**PROGRAM SPECIFIC OUTCOMES**

**PSO1:** Employ Systems Approach to model the solutions for real life problems, design and develop software systems by applying Modern Tools.

**PSO2:** Develop solutions using novel algorithms in High Performance Computing and Data Science.

**PSO3:** Use emerging technologies for providing security and privacy to design, deploy and manage network systems.

**PROGRAM OUTCOMES**

1. Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

6. Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

**II B. Tech. – II Semester**

**(20BT40531) DATABASE MANAGEMENT SYSTEMS LAB**

**COURSE OUTCOMES**

**CO1**. Analyze the requirements of a given database problem and design viable ER-Models for implementation of database.

**CO2**. Create database schemas, select and apply suitable integrity constraints for querying databases using SQL interface.

**CO3.** Develop and interpret PL/SQL blocks to centralize database applications for maintainability and reusability.

**CO4.** Develop database applications for societal applications such as ticket reservation system, employee payroll system using modern tools.

**CO5.** Work independently and communicate effectively in oral and written forms.

**DECLARATION**

We hereby declare that this project report titled “Title” is a genuine work carried out by us, in B.Tech (Computer Science and Systems Engineering) degree course of Jawaharlal Nehru Technological University Anantapur and has not been submitted to any other course or University for the award of any degree by us.

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea / data / fact / source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Signature of the students

1.

2.

**ABSTRACT**

The tours and travel industry is a vibrant and rapidly evolving sector that serves as a vital gateway to global tourism. Travel agencies play a pivotal role in facilitating travel experiences by acting as intermediaries between travelers and various travel service providers. These agencies offer a comprehensive range of services, including itinerary planning, ticketing, accommodation booking, transportation arrangements, and more. In recent years, the industry has undergone a significant transformation, driven largely by technological advancements and changing consumer preferences.

This abstract explores the key components of a tours and travel agency, including its business model, services, technologies, and the challenges and opportunities it faces in the digital age. It also examines the importance of embracing digital transformation, leveraging data analytics, and adopting sustainable practices to enhance operational efficiency and drive growth. Ultimately, success in the tours and travel industry hinges on the ability of travel agencies to innovate, adapt, and deliver exceptional value to their customers.

The advent of online travel agencies (OTAs) and mobile applications has revolutionized the way travelers plan and book their trips. These digital platforms provide travelers with greater flexibility, convenience, and access to a wealth of information and options. Furthermore, they have enabled travel agencies to reach a broader audience and offer more personalized services.

**Keywords**: SQL Server, Microsoft SQL server management studio.

**TABLE OF CONTENTS**

**ELECTION MANAGEMENT SYSTEM PageNo.**

**ABSTRACT…………….……………………..…………………**  i

**CHAPTER 1. INTRODUCTION**

1.1 Introduction to the topic…………………………………........ 1-1

1.2 Problem Statement……………………………………………. 1-2

1.3 Objectives…………………………………………………….. 2-2

**CHAPTER 2. DATABASE DESIGN**

2.1 List of Attributes, entities and relationship…………………… 3-6

2. E-R Diagram………………………………………………….. 7-7

**CHAPTER 3. RELATIONAL MODEL**

3.1 Database languages……………………………………………. 8-8

3.2 Table Description……………………………………………… 9-11

3.3 Relational Database Scheme………………………………….. 12-20

3.4 Relational Queries……………………………………………. 20-51

**CHAPTER 4. CONCLUSION AND FUTURE WORK**

4.1 Conclusion………………………………………………………. 52-52

4.2 Future Work…………………………………………………….. 53-53

# CHAPTER 1. INTRODUCTION

**1.1 Introduction to the topic**

Elections are a cornerstone of democratic societies, allowing citizens to choose their representatives and shape the direction of their governments. However, managing the complexities of electoral processes, such as voter registration, candidate nominations, polling station management, and result compilation, can be a daunting task for election authorities. To address these challenges and enhance the efficiency and transparency of elections, Election Management Systems (EMS) have emerged as invaluable tools.

An Election Management System is a comprehensive digital solution that leverages technology to streamline and automate various aspects of the electoral process. It encompasses a range of functionalities, from voter registration to result declaration, and provides a centralized platform for election officials, candidates, and voters to interact and access essential information.

The implementation of an EMS brings numerous benefits. It simplifies administrative tasks, reduces manual errors, enhances data accuracy, and ensures the integrity of the electoral process. By incorporating advanced technologies like electronic voting machines (EVMs), biometric authentication, and data analytics, an EMS can significantly improve the efficiency, transparency, and credibility of elections.

**1.2 Problem Statement**

Managing elections is a complex task that involves multiple stages, from voter registration to result declaration. Traditional manual processes used by election authorities often face challenges such as inefficiency, inaccuracies, and lack of transparency. These challenges can undermine the integrity of the electoral process and erode public trust in the democratic system.

The problem lies in the need for an effective and reliable system that can streamline and automate various aspects of the electoral process. This system should address the following key issues:

**Inefficient Voter Registration**: Manual voter registration processes are time-consuming and prone to errors. They often result in inaccuracies, including duplicate entries or exclusion of eligible voters. Such inefficiencies can lead to disenfranchisement and undermine the legitimacy of elections.

**Tedious Candidate Nomination Process**: Traditional methods of candidate nomination involve extensive paperwork and manual verification, leading to delays and potential errors. The lack of a streamlined and user-friendly system can discourage potential candidates from participating in the electoral process.

**Vulnerability to Electoral Fraud**: Manual vote counting processes are susceptible to human error and manipulation. This vulnerability raises concerns about the accuracy and transparency of election results, potentially undermining public confidence in the electoral system.

To address these issues, there is a critical need for an advanced Election Management System that can automate and streamline the electoral process, ensuring efficiency, accuracy, transparency, and inclusiveness. Such a system should leverage modern technologies to enable online voter registration, simplify candidate nominations, optimize polling station management, secure electronic voting, and provide robust data analytics and reporting capabilities.

By addressing the challenges and limitations of traditional electoral processes, an effective Election Management System can enhance the integrity and credibility of elections, empower citizens, and strengthen democratic practices.

**1.3 Objectives**

The primary objectives of implementing an Election Management System (EMS) are to address the challenges and shortcomings of traditional electoral processes and to enhance the efficiency, transparency, and integrity of elections.

An important objective of an EMS is to facilitate accurate and inclusive voter registration. The system should provide a user-friendly platform for online voter registration, enabling eligible citizens to enroll, update their information, and verify their registration status. By maintaining accurate voter rolls, the EMS minimizes the chances of duplicate entries and ensures that all eligible voters can exercise their democratic rights.

The EMS aims to generate comprehensive reports and analytics to provide valuable insights into the electoral process. By analyzing data on voter turnout, demographic patterns, and election results, the system helps election authorities and stakeholders identify trends, assess performance, and make data-driven decisions for future electoral planning and improvements.

By achieving these objectives, an effective Election Management System contributes to the integrity, transparency, and efficiency of elections, fostering public trust and confidence in the democratic process.

# CHAPTER 2. DATABASE DESIGN

**2.1 List of Attributes, entities and relationship**

## Entity Name: Election

| **Attributes** | **Type** |
| --- | --- |
| ***Electionid*** | int(10) |
| E\_name | varchar(20) |
| Start Date | Date |
| End Date | Date |

## Entity Name: Election\_voter

| **Attributes** | **Type** |
| --- | --- |
| ***Electionid*** | int(10) |
| Voter\_id | int(10) |
| vote\_timestamp | varchar(25) |

## Entity Name: Parties

| **Attributes** | **Type** |
| --- | --- |
| ***Party\_id*** | int(10) |
| Party\_name | varchar(25) |
| Party\_found | int |

## Entity Name: Voting\_booth

| **Attributes** | **Type** |
| --- | --- |
| ***Boothid*** | int(10) |
| Booth\_name | varchar(25) |
| Booth\_location | varchar(25) |
| Booth\_status | varchar(25) |

## Entity Name: Candidate\_history

| **Attributes** | **Type** |
| --- | --- |
| ***History\_id*** | int(10) |
| Candidate\_id | int(10) |
| Election\_id | int(10) |
| Votes\_received | int(20) |

## Entity Name: Votes

| **Attributes** | **Type** |
| --- | --- |
| ***Vote\_id*** | int(10) |
| Election\_id | int(10) |
| Voter\_id | Int(10) |
| Party\_id | int(10) |

## 

## Entity Name: Administrator

| **Attributes** | **Type** |
| --- | --- |
| ***Adminid*** | int(10) |
| Admin\_name | varchar(10) |
| Adimin\_email | varchar(30) |
| Admin\_password | varchar(20) |

## 

## Entity Name: Candidate\_details

| **Attributes** | **Type** |
| --- | --- |
| ***Candidateid*** | int(10) |
| C\_name | varchar(10) |
| C\_dob | varchar(20) |
| C\_gender | char(1) |
| P\_id | int(10) |
| party\_affilation | varchar(25) |
| C\_city | varchar(10) |
| C\_state | varchar(15) |
| C\_zipcode | varchar(10) |

## Entity Name: Voter\_registration

| **Attributes** | **Type** |
| --- | --- |
| ***Registrationid*** | int(10) |
| Voter\_id | int(10) |
| R\_Date | date |
| R\_status | varchar(30) |

## Entity Name: voter\_table

| **Attributes** | **Type** |
| --- | --- |
| ***voterid*** | int(10) |
| v\_name | varchar(10) |
| v\_city | varchar(10) |
| v\_state | varchar(20) |
| v\_zipcode | varchar(10) |

## Entity Name: voter\_booth

| **Attributes** | **Type** |
| --- | --- |
| ***ratingid*** | int(10) |
| ratingtype | varchar(25) |
| customerid | int(10) |
| companyid | int(10) |

## Entity Name: election\_results

| **Attributes** | **Type** |
| --- | --- |
| ***Resultid*** | int(10) |
| E\_id | int(10) |
| C\_id | int(10) |
| vote\_count | int(10) |

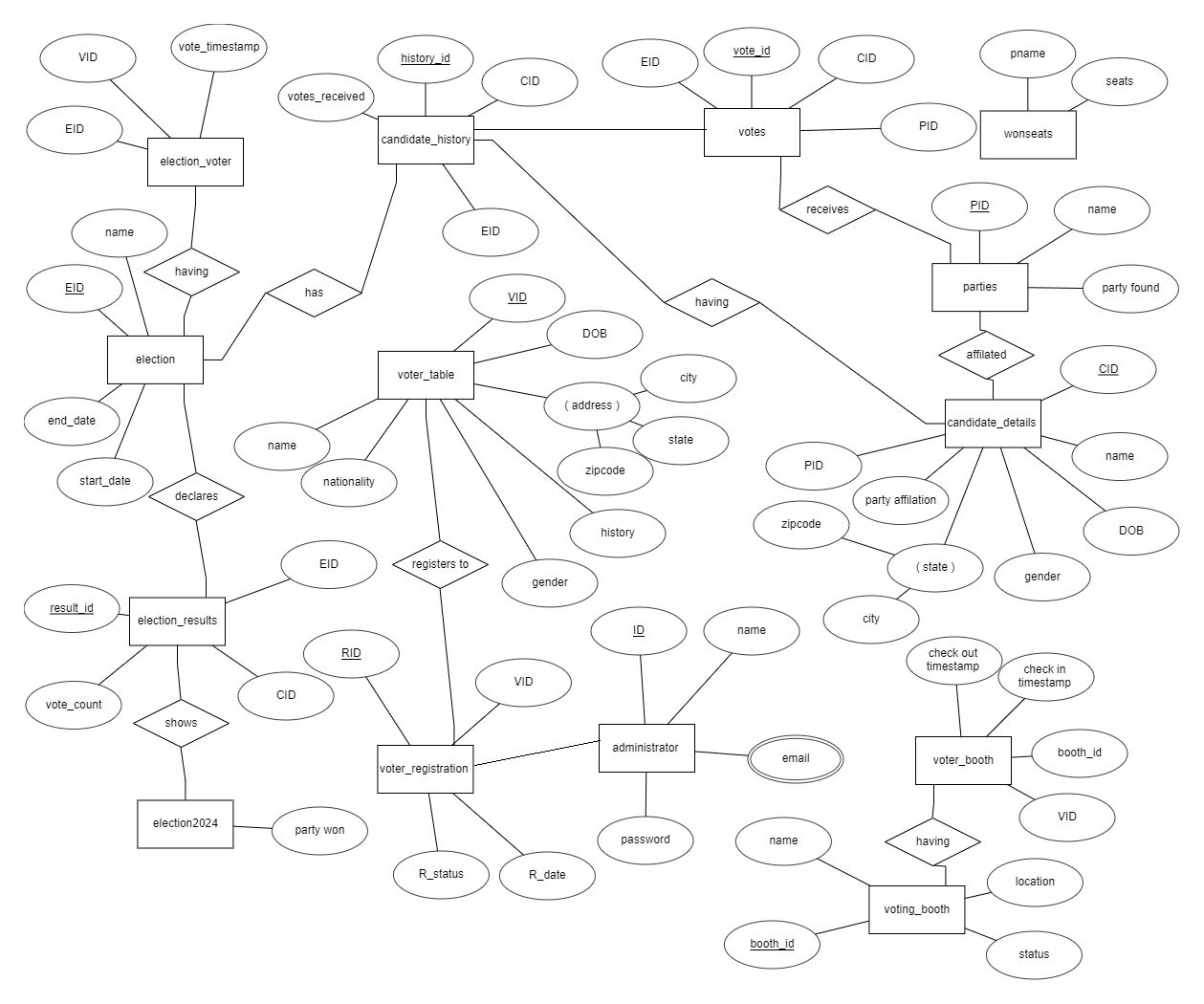
13.**Entity Name:** **wons\_seats**

| **Attributes** | **Type** |
| --- | --- |
| ***partyid*** | int(10) |
| pname | varchar(20) |
| wseats | int(10) |

14.**Entity Name:** **Election 2024**

| **Attributes** | **Type** |
| --- | --- |
| ***partyid*** | int(10) |
| pname | varchar(20) |
| partywon | varchar(20) |

# 2.2 ER DIAGRAM



# 2.2 E-R Diagram

**CHAPTER 3. RELATIONAL MODEL**

# 3.1 Database languages

Four categories of database languages :

**1.** **Data definition language (DDL)**

Data definition language (DDL) creates the framework of the database by specifying the database schema, which is the structure that represents the organization of data. Its common uses include the creation and alteration of tables, files, indexes and columns within the database. This language also allows users to rename or drop the existing database or its components.

Here's a list of DDL statements:

• CREATE: Creates a new database or object, such as a table, index or column.

• ALTER: Changes the structure of the database or object.

• DROP: Deletes the database or existing objects.

• RENAME: Renames the database or existing objects.

**2.** **Data manipulation language (DML)**

Data manipulation language (DML) provides operations that handle user requests, offering a way to access and manipulate the data that users store within a database. Its common functions include inserting, updating and retrieving data from the database.

Here's a list of DML statements:

• INSERT: Adds new data to the existing database table.

• UPDATE: Changes or updates values in the table.

• DELETE: Removes records or rows from the table.

• SELECT: Retrieves data from the table or multiple tables.

**3. Data control language (DCL)**

Data control language (DCL) controls access to the data that users store within a database. Essentially, this language controls the rights and permissions of the database system. It allows users to grant or revoke privileges to the database.

Here's a list of DCL statements:

• GRANT: Gives a user access to the database.

• REVOKE: Removes a user's access to the database.

**4. Transaction control language (TCL)**

Transaction control language (TCL) manages the transactions within a database. Transactions group a set of related tasks into a single, executable task. All the tasks must succeed in order for the transaction to work. Here's a list of TCL statements:

• COMMIT: Carries out a transaction.

• ROLLBACK: Restores a transaction if any tasks fail to execute.

**3.2 Table Description**

Following are the tables along with constraints used in All in one election

database.

1. Election: This table contains election details like election id, election name.

**Constraint:** election id should be provided if the election is linked up with this election data

2. Administrator: Administrator entity contains the details of the administrator like

admin id, admin name, login id, password.

**Constraint:** Here administrator id will not be accessed by any other relation because

all the details of the tables and their relations, tuples will be accessed by the admin.

3. Election voter: This table contains all the details of the voters such as voter id,

name, date of birth, gender, address, email, login, password and also the status of

the voter which described about the registration status of the voters.

**Constraint:** The election voter id will be as the primary key constraint for the voter

info relation in the database and also the voter id will be considered as the

foreign key constraint.

4. Parties: This entity contains the detailed description about the parties like party id,name, party type, number of parties count.

**Constraint:** considering party id as the main constraint of the election relation this id can

be accessed by the parties relation where it contains the details of the elections.

5. Voting booth: The voting booth table describes about the number of votings booths present through the particular point, it contains all the details of the votes.

**Constraint:** Here we can consider the primary key as the boarding voting booth id and the election id as the foreign key constraint.

6. votes: The votes table contains the details about the types of votes available, no. of voters available, voter id.

**Constraint:** votes id will be the primary key constraint of this relation and to know for

which election the votes belongs to the election id would be considered as the

foreign key.

7. Candidate history: The candidate history entity belongs to the history of the candidate provided.

**Constraint:** Candidate history id will be the primary key constraint of this relation and to know for which cid belongs to the eid would be considered as the foreign key.

8. Candidate details : The candidate details is entity of the database deals with the no. of candidates held on the particular election.

**Constraints:** candidate details id plays the role of primary key constraint in this relation and the election id would be referred as the foreign key constraint.

9. voter registration: Voter registration contains the details about the voters on the

election.

**Constraints:** The voter registration will be referred as the primary key, and cid will

be the foreign key.

10. voter table: Voter table contains the details of the voter like vid, rid,eid.

**Constraints:** Voter table is as the primary key would help in calculating the

votes details of the receipts generated to know this the election id will be

considered as the foreign key.

11. Election Results: Election results table will have the result of the elections to describe these details the attributes that are eid, cid, vid.

**Constraints:** Election result id will be considered as the primary key constraint and the

vid will be used as the reference and will be a foreign key constraint of the

relation.

The above descripted information is the brief detailing about the entities

and relations and their attributes.

**3.3 Relational Database Scheme**

The relational database schema for ***All in one travel booking system***database is as follows:

1. election(electionid, electionytype, electionname, electionplace)
2. administrator(adminid, adminname, loginid, password, status)
3. candidate details(candidateid, candidatename, dob, gender, address, contact, state, pincode, emailid, loginid, password, status)
4. voting booth(eid, cid, votingtype, address, availabality, status)
5. Election voter(voterid, eid, cid, votername, voter dob, gender)
6. voter regstration(registration id, voter tid, election id, address, v\_name, gender, contact, status)
7. candidate history(cid, eid, elactiontype, date, status )
8. voter table(vid, cid, date, voter name, gender, status )
9. voter booth(vid, cid, address, city, state, zipcode, no.of booths available)
10. Election result(election id, candidate id, voter id, no.of votes, election type, election place, city, state,zipcode)

**3.4 Relational Queries**

/\* create a table election\*/

create table election

(

EID int primary key,

E\_name varchar(30) not null,

startdate date not null,

enddate date not null,

);

select \* from election

insert into election values(1,'MP','2023-07-13','2023-07-13')

insert into election values(2,'MLC','2023-07-13','2023-07-13')

insert into election values(3,'MLA','2023-07-13','2023-07-13')

insert into election values(4,'MINISTER','2023-07-13','2023-07-13')

insert into election values(5,'CM','2023-07-13','2023-07-13')

insert into election values(6,'PM','2023-07-13','2023-07-13')

**OUTPUT**:

**EID E\_name startdate enddate**

1 MP 2023-07-13 2023-07-13

2 MLC 2023-07-13 2023-07-13

3 MLA 2023-07-13 2023-07-13

4 MINISTER 2023-07-13 2023-07-13

5 CM 2023-07-13 2023-07-13

6 PM 2023-07-13 2023-07-13

/\*create a table voter\*/

create table election\_voter

(

EID int foreign key references election(EID),

VID int foreign key references voter\_table(VID),

vote\_timestamp time not null,

);

select \* from election\_voter

insert into election\_voter values(3,52,'06:50:17')

insert into election\_voter values(2,50,'06:30:29')

insert into election\_voter values(6,61,'11:15:02')

insert into election\_voter values(1,63,'12:48:09')

insert into election\_voter values(2,56,'09:30:29')

insert into election\_voter values(4,70,'03:41:12')

insert into election\_voter values(6,62,'12:00:22')

insert into election\_voter values(2,53,'07:33:39')

insert into election\_voter values(5,55,'08:38:57')

insert into election\_voter values(3,64,'01:39:53')

insert into election\_voter values(4,68,'02:29:01')

insert into election\_voter values(5,66,'01:58:49')

**OUTPUT:**

**EID VID vote\_timestamp**

3 52 06:50:17

2 50 06:30:29

6 61 11:15:02

1 63 12:48:09

2 56 09:30:29

4 70 03:41:12

6 62 12:00:22

2 53 07:33:39

5 55 08:38:57

3 64 01:39:53

4 68 02:29:01

5 66 01:58:49

/\*create a table parties\*/

create table parties

(

PID int primary key,

Pname varchar(40) not null,

party\_found int null,

);

select \* from parties

insert into parties values(300,'TELUGU DESAM PARTY','1982-03-29')

insert into parties values(301,'JANASENA PARTY','2014-03-14')

insert into parties values(302,'YCP','1999-06-10')

insert into parties values(303,'BJP','1980-04-06')

insert into parties values(304,'CONGRESS','1885-12-28')

insert into parties values(305,'BRS','2023-04-11')

**Output:**

**PID Pname party\_found**

300 TELUGU DESAM PARTY 1982-03-29

301 JANASENA PARTY 2014-03-14

302 YCP 1999-06-10

303 BJP 1980-04-06

304 CONGRESS 1885-12-28

305 BRS 2023-04-11

/\* create a table voting\_booth\*/

create table voting\_booth

(

booth\_ID int primary key,

booth\_name varchar(30) not null,

booth\_location varchar(30) not null,

booth\_status varchar(30) check(booth\_status='available' or booth\_status='not available')

);

select \* from voting\_booth

insert into voting\_booth values(700,'SV NAGAR','RAJAMPETA','available')

insert into voting\_booth values(701,'RS ROAD','GUNTUR','available')

insert into voting\_booth values(702,'VIDYA NAGAR','KADAPA','available')

insert into voting\_booth values(703,'RS ROAD','KURNOOL','not available')

insert into voting\_booth values(704,'RAMAPUR','NANDYALA','available')

insert into voting\_booth values(705,'RS ROAD','KRISHNA','not available')

insert into voting\_booth values(706,'VIDYA NAGAR','RAJAMPETA','available')

insert into voting\_booth values(707,'RANGAMPETA','NELLORE','available')

insert into voting\_booth values(708,'RS ROAD','SRISAILAM','available')

insert into voting\_booth values(709,'RAMAPUR','DHARMAVARAM','available')

insert into voting\_booth values(710,'KUCHIVARIPALLI','KURNOOL','not available')

**OUTPUT**:

**booth\_ID booth\_name booth\_location booth\_status**

700 SV NAGAR RAJAMPETA available

701 RS ROAD GUNTUR available

702 VIDYA NAGAR KADAPA available

703 RS ROAD KURNOOL not available

704 RAMAPUR NANDYALA available

705 RS ROAD KRISHNA not available

706 VIDYA NAGAR RAJAMPETA available

707 RANGAMPETA NELLORE available

708 RS ROAD SRISAILAM available

709 RAMAPUR DHARMAVARAM available

710 KUCHIVARIPALLI KURNOOL not available

/\* create a table candidate\_history\*/

create table candidate\_history

(

history\_id int primary key,

CID int foreign key references candidate\_details(CID),

EID int foreign key references election(EID),

votes\_received int not null

);

select \* from candidate\_history

insert into candidate\_history values(1000,600,1,1300)

insert into candidate\_history values(1001,601,2,1250)

insert into candidate\_history values(1002,602,4,1698)

insert into candidate\_history values(1003,603,6,1382)

insert into candidate\_history values(1004,604,1,1248)

insert into candidate\_history values(1005,605,4,1249)

insert into candidate\_history values(1006,606,5,1987)

insert into candidate\_history values(1007,607,6,1099)

insert into candidate\_history values(1008,608,2,1234)

insert into candidate\_history values(1009,609,4,1344)

insert into candidate\_history values(1010,610,5,1325)

**OUTPUT:**

**history\_id CID EID votes\_received**

1000 600 1 1300

1001 601 2 1250

1002 602 4 1698

1003 603 6 1382

1004 604 1 1248

1005 605 4 1249

1006 606 5 1987

1007 607 6 1099

1008 608 2 1234

1009 609 4 1344

1010 610 5 1325

/\*create table votes\*/

create table votes

(

vote\_id int primary key,

EID int foreign key references election(EID),

VID int foreign key references voter\_table(VID),

PID int foreign key references parties(PID)

);

select \* from votes

insert into votes values(1200,2,50,301)

insert into votes values(1201,3,52,301)

insert into votes values(1202,1,56,300)

insert into votes values(1203,5,61,305)

insert into votes values(1204,6,62,301)

insert into votes values(1205,3,63,302)

insert into votes values(1206,4,64,303)

insert into votes values(1207,5,70,304)

insert into votes values(1208,1,53,301)

insert into votes values(1209,2,55,302)

insert into votes values(1210,6,68,303)

**Output:**

**vote\_id EID VID PID**

1200 2 50 301

1201 3 52 301

1202 1 56 300

1203 5 61 305

1204 6 62 301

1205 3 63 302

1206 4 64 303

1207 5 70 304

1208 1 53 301

1209 2 55 302

1210 6 68 303

1211 2 66 301

/\* create table administrator\*/

create table administrator

(

admin\_id int primary key,

admin\_name varchar(30) not null,

admin\_email varchar(30) not null,

admin\_password varchar(10) not null

);

select \* from administrator

insert into administrator values(100,'ROHIT','RO171@GMAIL.COM','ROHIT12')

insert into administrator values(101,'RAHUL','BULLET321@GMAIL.COM','BULLET')

insert nto administrator values(102,'VENKATESH','VENKY@GMAIL.COM','ROHIT12')

insert into administrator values(103,'NEHA','NEHA9817@GMAIL.COM','NEHA12')

insert into administrator values(104,'DEVARSHINI','DEV@GMAIL.COM','DEV')

insertntoadministratorvalues(105,'KEERTHI','KEERTHI@GMAIL.COM','KERTHI123')

insert into administrator values(106,'RAHANE','RAHA2321@GMAIL.COM','RAHAN')

insert into administrator values(107,'VIJAY','VIJU@GMAIL.COM','VIJU2324')

insert into administrator values(108,'ALLU ARJUN','ALLU@GMAIL.COM','ALLU722')

insertintoadministratorvalues(109,'PRASANTHI','PRASANTHI@GMAIL.COM','PRASANTHI4321')

**OUTPUT:**

**admin\_id admin\_name admin\_email admin\_password**

100 ROHIT [RO171@GMAIL.COM](mailto:RO171@GMAIL.COM) ROHIT12

101 RAHUL [BULLET321@GMAIL.COM](mailto:BULLET321@GMAIL.COM) BULLET

102 VENKATESH [VENKY@GMAIL.COM](mailto:VENKY@GMAIL.COM) ROHIT12

103 NEHA [NEHA9817@GMAIL.COM](mailto:NEHA9817@GMAIL.COM) NEHA12

104 DEVARSHINI [DEV@GMAIL.COM](mailto:DEV@GMAIL.COM) DEV

106 RAHANE [RAHA2321@GMAIL.COM](mailto:RAHA2321@GMAIL.COM) RAHAN

107 VIJAY [VIJU@GMAIL.COM](mailto:VIJU@GMAIL.COM) VIJU2324

108 ALLU ARJUN [ALLU@GMAIL.COM](mailto:ALLU@GMAIL.COM) ALLU722

110 SAILAJA [SAILU2325@GMAIL.COM](mailto:SAILU2325@GMAIL.COM) SAILU897

/\*create table election\_results\*/

create table election\_results

(

result\_id int primary key,

EID int foreign key references election(EID),

CID int foreign key references candidate\_details(CID),

vote\_count int not null,

);

select \* from election\_results

insert into election\_results values(1300,1,600,1568)

insert into election\_results values(1301,5,601,7832)

insert into election\_results values(1302,4,602,1568)

insert into election\_results values(1303,6,603,5490)

insert into election\_results values(1304,3,604,5568)

insert into election\_results values(1305,5,605,2590)

insert into election\_results values(1306,1,606,568)

insert into election\_results values(1307,4,607,2568)

insert into election\_results values(1308,6,608,1469)

insert into election\_results values(1309,2,609,1561)

insert into election\_results values(1310,5,610,962)

**OUTPUT:**

**result\_id EID CID vote\_count**

1300 1 600 1568

1301 5 601 7832

1302 4 602 1568

1303 6 603 5490

1304 3 604 5568

1305 5 605 2590

1306 1 606 568

1307 4 607 2568

1308 6 608 1469

1309 2 609 1561

1310 5 610 962

**SQL QUERIES:**

**1.find the name of the candidate whose name ending second leter is A.**

select \* from candidate\_details where c\_name like '%a\_'

**output:**

**CID C\_name C\_DOB C\_gender PID party\_affilation C\_city C\_state C\_zipcode**

604 SRINIVAS 1984-04-13 M 301 MLA GUNTUR AP 522001

605 VENKATESWAR 1983-06-10 M 302 MLA KURNOOL AP 518001

607 SATHYA SAI 1979-09-05 M 302 MLA DHARMAVARAM AP 515671

608 ANIL KUMAR 1972-02-13 M 302 MLC SRISAILAM AP 518101

609 ESWAR 1979-02-04 M 303 MLA SRISAILAM AP 518101

612 SUBBARAO 1974-08-29 M 305 MLA DHARMAVARAM AP 515671

613 PRABHAKAR 1974-03-30 M 301 MLC KURNOOL AP 518001

614 SURYAVARA PRASAD 1979-10-31 M 303 MLA KADAPA AP 516001

617 SRI RAM 1976-03-02 M 304 MLA GUNTUR AP 522001

619 BHUVANA KARUNAKAR 1985-06-13 M 302 MLA TIRUPATI AP 517501

**2.find the person whose name starts with s and from tirupati.**

select \* from candidate\_details

where c\_name like 's%'and

c\_city in('tirupati')

**output:**

**CID C\_name C\_DOB C\_gender PID party\_affilation C\_city C\_state C\_zipcode**

620 SUGUNA 1980-08-25 F 300 MLA TIRUPATI AP 517501

**3.find the maximum votes of the election results of each election.**

select max(vote\_count) as average\_votes

from election\_results group by eid

**output:**

**average\_votes**

1 2167

2 6587

3 5568

4 2568

5 7832

6 5490

**4.find the total no.of active booths from voting booth.**

select count(\*) as active\_booths from

voting\_booth where booth\_status='available'

**output:**

**active\_booths**

16

**5.find out the history id of candidate whose received maximum votes.**

select history\_id,cid from candidate\_history

where votes\_received=(select max(votes\_received) from candidate\_history)

**output:**

**history\_id cid**

1006 606

**6.find the registration details of eligible and active voters with their registration dates.**

select R\_date,RID,VID from voter\_registration

where R\_status='eligible'or R\_status='active' order by R\_Date

**output:**

**R\_date RID VID**

2022-01-25 511 61

2022-02-07 508 58

2022-02-12 514 64

2022-02-18 512 62

2022-04-06 505 55

2022-07-01 520 70

2022-07-12 500 50

2022-07-14 502 52

2022-08-22 513 63

2022-09-01 506 56

2022-12-19 503 53

2022-12-20 516 66

**7.create a temporary table for inactive voters.**

select \* into inactive\_voters from voter\_registration

where R\_status='inactive'

select \* from inactive\_voters

**output:**

**RID VID R\_Date R\_status**

504 54 2022-09-03 inactive

507 57 2010-01-08 inactive

509 59 2022-09-17 inactive

515 65 2022-10-11 inactive

517 67 2022-07-21 inactive

518 68 2009-04-08 inactive

**8.find out the unambiguous list of voters for who is eligible to vote.**

select v\_name,V\_dob,V\_history from voter\_table

where V\_DOB<'2003-01-01'order by V\_DOB,V\_history

**output:**

**v\_name V\_dob V\_history**

RAM CHARAN 1997-10-15 YES

PUNEETH 1998-06-29 YES

SURENDRA 2001-07-31 NO

ROSHINI 2001-10-18 NO

REVANTH 2002-01-24 NO

SWATHI 2002-02-11 NO

NAYAZ 2002-03-03 NO

NANDINI 2002-04-09 NO

SRINIVAS 2002-05-07 NO

SALVAR 2002-07-09 NO

DHARANI 2002-10-13 NO

PAWAN KALYAN 2002-11-29 NO

HEMA LATHA 2002-11-29 NO

KOHLI 2002-12-07 NO

ANUSHKA 2002-12-30 NO

NAVYA 2002-12-31 NO

**9.find the total no. of active ,inactive,eligible and not eligible voters.**

select count(\*) as list,R\_status

from voter\_registration group by R\_status

**output:**

**list R\_status**

4 active

8 eligible

6 inactive

3 noteleigible

**10.find the name and id of the candidate got min votes?**

SELECT candidate\_name, CID

FROM candidate\_details

WHERE votes = (SELECT MIN(votes) FROM candidate\_details);

**output:**

**history\_id CID EID votes\_received**

1007 607 6 1099

**11.find the vote count of each candidate from voters?**

SELECT v.vote\_id, v.EID, v.VID, v.PID, er.CID, er.vote\_count

FROM votes AS v

JOIN election\_results AS er ON v.EID = er.EID;

**output:**

**vote\_id EID VID PID CID vote\_count**

1200 2 50 300 609 1561

1200 2 50 300 611 3565

1200 2 50 300 613 6587

1200 2 50 300 615 2263

1200 2 50 300 619 1962

1201 3 52 301 604 5568

1201 3 52 301 616 1462

1201 3 52 301 617 1964

1202 1 56 300 600 1568

1202 1 56 300 606 568

1202 1 56 300 618 2167

1203 5 61 305 601 7832

1203 5 61 305 605 2590

1203 5 61 305 610 962

1203 5 61 305 614 2562

1203 5 61 305 620 3165

1204 6 62 301 603 5490

1204 6 62 301 608 1469

1204 6 62 301 621 4163

1205 3 63 302 604 5568

1205 3 63 302 616 1462

1205 3 63 302 617 1964

1206 4 64 303 602 1568

1206 4 64 303 607 2568

1206 4 64 303 612 1566

1206 4 64 303 622 3962

1207 5 70 304 601 7832

1207 5 70 304 605 2590

1207 5 70 304 610 962

1207 5 70 304 614 2562

1207 5 70 304 620 3165

1208 1 53 301 600 1568

1208 1 53 301 606 568

1208 1 53 301 618 2167

1209 2 55 302 609 1561

1209 2 55 302 611 3565

1209 2 55 302 613 6587

1209 2 55 302 615 2263

1209 2 55 302 619 1962

1210 6 68 303 603 5490

1210 6 68 303 608 1469

1210 6 68 303 621 4163

1211 2 66 301 609 1561

1211 2 66 301 611 3565

1211 2 66 301 613 6587

1211 2 66 301 615 2263

1211 2 66 301 619 1962

**12 find the total votes got from each election?**

SELECT EID, SUM(vote\_count) AS total\_votes

FROM election\_results

GROUP BY EID;

**ouput:**

**EID total\_votes**

1 4303

2 15938

3 8994

4 9664

5 17111

6 11122

**13 find the vote\_id,EID,VID and their vote count?**

SELECT v.vote\_id, v.EID, v.VID, er.vote\_count

FROM votes AS v

JOIN election\_results AS er ON v.EID = er.EID;

**output:**

**vote\_id EID VID vote\_count**

1200 2 50 1561

1200 2 50 3565

1200 2 50 6587

1200 2 50 2263

1200 2 50 1962

1201 3 52 5568

1201 3 52 1462

1201 3 52 1964

1202 1 56 1568

1202 1 56 568

1202 1 56 2167

1203 5 61 7832

1203 5 61 2590

1203 5 61 962

1203 5 61 2562

1203 5 61 3165

1204 6 62 5490

1204 6 62 1469

1204 6 62 4163

1205 3 63 5568

1205 3 63 1462

1205 3 63 1964

1206 4 64 1568

1206 4 64 2568

1206 4 64 1566

1206 4 64 3962

1207 5 70 7832

1207 5 70 2590

1207 5 70 962

1207 5 70 2562

1207 5 70 3165

1208 1 53 1568

1208 1 53 568

1208 1 53 2167

1209 2 55 1561

1209 2 55 3565

1209 2 55 6587

1209 2 55 2263

1209 2 55 1962

1210 6 68 5490

1210 6 68 1469

1210 6 68 4163

1211 2 66 1561

1211 2 66 3565

1211 2 66 6587

1211 2 66 2263

1211 2 66 1962

**14 find the total votes from candiadates details group by cid?**

SELECT c.CID, SUM(er.vote\_count) AS total\_votes

FROM candidate\_details c

INNER JOIN election\_results er ON c.CID = er.CID

GROUP BY c.CID

**output:**

**CID total\_votes**

600 1568

601 7832

602 1568

603 5490

604 5568

605 2590

606 568

608 1469

609 1561

610 962

611 3565

612 1566

613 6587

614 2562

615 2263

616 1462

617 1964

618 2167

619 1962

620 3165

621 4163

622 3962

**15.find the EID and CID of candidates whose got max vote count?**

SELECT e.EID, c.CID, er.vote\_count

FROM election\_results er

INNER JOIN candidate\_details c ON er.CID = c.CID

INNER JOIN election e ON er.EID = e.EID

WHERE er.vote\_count = (

SELECT MAX(er2.vote\_count)

FROM election\_results er2

WHERE er2.EID = er.EID

);

**output:**

**EID CID vote\_count**

6 603 5490

5 601 7832

4 622 3962

3 604 5568

2 613 6587

1 618 2167

**16.select maximum vote count from election results of each election?**

SELECT er.vote\_count

FROM election\_results er

WHERE er.vote\_count = (

SELECT MAX(er2.vote\_count)

FROM election\_results er2

WHERE er2.EID = er.EID

);

**output:**

**vote\_count**

5490

7832

3962

5568

6587

2167

**17 find the average votes from each party with party name and id?**

SELECT p.PID, p.Pname, AVG(er.vote\_count) AS average\_votes

FROM parties p

INNER JOIN candidate\_details c ON p.PID = c.PID

INNER JOIN election\_results er ON c.CID = er.CID

GROUP BY p.PID, p.Pname;

**output:**

**PID Pname average\_votes**

300 TELUGU DESAM PARTY 2090

301 JANASENA PARTY 6369

302 YCP 2054

303 BJP 2163

304 CONGRESS 3063

305 BRS 1566

**18 find the booth status is 'not available'?**

SELECT booth\_ID, booth\_name, booth\_location

FROM voting\_booth

WHERE booth\_status = 'not available';

**output:**

**booth\_ID booth\_name booth\_location**

703 RS ROAD KURNOOL

705 RS ROAD KRISHNA

710 KUCHIVARIPALLI KURNOOL

711 KUCHIVARIPALLI KODUR

714 AIR BYPASS GUNTUR

715 RAMAPUR NELLORE

716 SV NAGAR KRISHNA

719 VIDYA NAGAR SRISAILAM

724 AIR BYPASS RAJAMPETA

**19 create a view from parties?**

create view parties\_bkup as

(

select PID,pname,party\_found

FROM parties

)

**output:**

Commands completed successfully.

Completion time: 2023-06-08T19:55:28.3471105+05:30

**20 find the EID and CID,where EID from votes?**

select EID,CID from election\_results where EID in (select EID from VOTES)

**OUTPUT:**

**EID CID**

1 600

5 601

4 602

6 603

3 604

5 605

1 606

4 607

6 608

2 609

5 610

2 611

4 612

2 613

5 614

2 615

3 616

3 617

1 618

2 619

5 620

6 621

4 622

**21 find VID and booth\_id by using EXISTS clause?**

select VID,booth\_ID from voter\_booth as v WHERE EXISTS(SELECT \*from voter\_registration as r where v.VID=r.VID)

**output:**

**VID booth\_ID**

50 700

50 700

51 704

52 707

53 702

54 701

55 706

56 709

57 704

58 709

59 700

60 701

61 704

62 712

63 718

64 701

65 713

66 700

67 718

68 722

69 704

70 720

**22 find the CID and PID where PID from parties?**

select CID,PID from candidate\_details where PID in (select PID from parties)

**output:**

**CID PID**

600 302

601 301

602 300

603 301

604 301

605 302

606 303

608 302

609 303

610 302

611 302

612 305

613 301

614 303

615 302

616 300

617 304

618 300

619 302

620 300

621 304

622 303

**23 find the election got 2nd maximum votes?**

select result\_id,EID from election\_results where vote\_count=

(

select max(vote\_count)from election\_results where vote\_count<>

(

select max(vote\_count)from election\_results

)

)

**output:**

**result\_id EID**

1313 2

**24 select CID and votes received from candidate history by using EXISTS?**

select CID,votes\_received from candidate\_history as c WHERE EXISTS(SELECT \*from candidate\_details as d where c.CID=d.CID)

**output:**

**CID votes\_received**

600 1300

601 1250

602 1698

603 1382

604 1248

605 1249

606 1987

608 1234

609 1344

610 1325

611 1353

612 1325

613 1523

614 1533

615 1235

616 1252

617 1340

618 1320

619 1325

620 1874

621 1973

622 1562

**25 find parties\_bkps**

select \* from parties\_bkup

**output:**

**PID pname party\_found**

300 TELUGU DESAM PARTY 1982-03-29

301 JANASENA PARTY 2014-03-14

302 YCP 1999-06-10

303 BJP 1980-04-06

304 CONGRESS 1885-12-28

305 BRS 2023-04-11

**26 what is the defination or source code of 'parties\_bkup'view?**

SP\_helptext 'parties\_bkup'

**output:**

Text

create view parties\_bkup as

(

select PID,pname,party\_found

FROM parties

)

**27 what is the information does the 'sp\_help'stored procedure provide about the parties table?**

SP\_help 'parties\_bkup'

**output:**

Column\_name Type

PID int

pname varchar

party\_found date

**28 find the voters without having duplicates?**

SELECT \* FROM voter\_table

WHERE VID IN (

SELECT DISTINCT VID

FROM voter\_table

);

**output:**

**VID V\_name V\_DOB V\_nationality V\_gender V\_address V\_city V\_state V\_zipcode V\_history**

55 SALVAR 2002-07-09 INDIAN F NTR COLONY SRISAILAM AP 518101 NO

56 DIVYA 2003-09-13 INDIAN F LAKSHMI PURAM PRAKSAM AP 523001 NO

57 PUNEETH 1998-06-29 INDIAN M GHANDHI ROAD NANDYALA AP 518501 YES

58 REVANTH 2002-01-24 INDIAN M RAMAVARAM KURNOOL AP 518001 NO

59 SURENDRA 2001-07-31 INDIAN M RS ROAD URVA KONDA AP 515812 NO

60 SRINIVAS 2002-05-07 INDIAN M SV NAGAR KURNOOL AP 518001 NO

61 SWATHI 2002-02-11 INDIAN F VIDYA NAGAR RAJAMPETA AP 516115 NO

62 NANDINI 2002-04-09 INDIAN F RS ROAD SRISAILAM AP 518101 NO

63 KOHLI 2002-12-07 INDIAN M RS ROAD KADAPA AP 516001 NO

64 ANUSHKA SHARMA 2002-12-30 INDIAN F RANGAMPETA KAKINADA AP 518001 NO

65 HEMA LATHA 2002-11-29 INDIAN F RS ROAD GUNTUR AP 522001 NO

66 DHARANI 2002-10-13 INDIAN F RAJA COLONY CHITOOR AP 517001 NO

67 CHARAN 2003-08-17 INDIAN M RAMAPUR NELLORE AP 524002 NO

68 RAM CHARAN 1997-10-15 INDIAN M RS ROAD RAJAMPETA AP 516115 YES

69 PAWAN KALYAN 2002-11-29 INDIAN M KUCHIVARIPALLI TIRUPATI AP 517501 NO

70 ROSHINI 2001-10-18 INDIAN F AIR BYPASS TIRUPATI AP 517501 NO

**29 find the city and vote count using group by?**

SELECT V\_city, COUNT(\*) AS voter\_count

FROM voter\_table

GROUP BY V\_city;

**output:**

**V\_city voter\_count**

CHITOOR 1

DHARMAVARAM 1

GUNTUR 1

KADAPA 1

KAKINADA 1

KURNOOL 2

NANDYALA 1

NELLORE 3

PRAKSAM 1

RAJAMPETA 3

SRISAILAM 2

TIRUPATI 2

URVA KONDA 1

VISHAKAPATNAM 1

**30 perform inner join and find the total votes group by candidate id and name?**

SELECT C.CID, C.C\_name, COUNT(\*) AS total\_votes FROM candidate\_details AS C

INNER JOIN candidate\_details AS V ON C.CID = V.CID

GROUP BY C.CID, C.C\_name;

**output:**

**CID C\_name total\_votes**

600 MALLIKARJUNA 1

601 KUSUMA KUMARI 1

602 CHANGAL RAYUDU 1

603 PARTHA SARATHI 1

604 SRINIVAS 1

605 VENKATESWAR 1

606 PADMA 1

608 ANIL KUMAR 1

609 ESWAR 1

610 SAIBABU 1

611 SRIDEVI 1

612 SUBBARAO 1

613 PRABHAKAR 1

614 SURYAVARA PRASAD 1

615 RAGHUPATHI 1

616 ASHOK 1

617 SRI RAM 1

618 JAYLAKSHMAMMA 1

619 BHUVANA KARUNAKAR 1

620 SUGUNA 1

621 RAHUL 1

622 MAHESH 1

**31 find the no.of males and females?**

SELECT V\_gender AS gender

FROM voter\_table

GROUP BY V\_gender;

**output:**

**gender**

F

M

**32 Use subquery to find the candidates count of party affilation?**

SELECT party\_affilation, COUNT(\*) AS candidate\_count FROM candidate\_details

GROUP BY party\_affilation HAVING COUNT(\*) = (

SELECT MAX(candidate\_count) FROM (

SELECT party\_affilation, COUNT(\*) AS candidate\_count

FROM candidate\_details

GROUP BY party\_affilation

) AS subquery

);

**output:**

**party\_affilation candidate\_count**

MLA 13

**33 find the top 3 seats from wonseats?**

select pname,WSEATS from WONSEATS where WSEATS=(

select min(WSEATS) from WONSEATS where WSEATS in(

select distinct top 3 WSEATS from WONSEATS order by WSEATS desc))

**OUTPUT:**

**pname WSEATS**

YCP 32

**34 find the party name of maximum wonseats?**

SELECT Pname FROM WONSEATS

WHERE WSEATS = (

SELECT MAX(WSEATS)

FROM WONSEATS

);

**output:**

**Pname**

JANASENA

**35 find the party having top 2 wonseats?**

select pname,WSEATS from WONSEATS

WHERE WSEATS=(select min(WSEATS) FROM WONSEATS

where WSEATS in( select distinct top 2 WSEATS

from WONSEATS order by WSEATS desc))

**ouput:**

**pname WSEATS**

TELUGU DESAM PARTY 98

**36 Update registration status to eligible of RID is 502?**

update voter\_registration

set R\_status='eligible'

where RID=502

**output:**

(1 row affected)

Completion time: 2023-06-08T22:08:00.8716972+05:30

**37 delete votes of voter 1205?**

delete from votes where vote\_id=1205

**output:**

(1 row affected)

Completion time: 2023-06-08T22:11:24.8856020+05:30

**38 Update registration status to eligible of RID is 502?**

update voter\_registration

set R\_status= 'eligible'

where RID=502

**OUTPUT:**

(1 row affected)

Completion time: 2023-06-08T22:14:04.0224192+05:30

**39 find the person whose name starting with N belongs to dharmavaram?**

select \* from voter\_table

where v\_name like 'N%'and

v\_city in('DHARMAVARAM')

**output:**

**VID V\_name V\_DOB V\_nationality V\_gender V\_address V\_city V\_state V\_zipcode V\_history**

50 NAVYA 2002-12-31 INDIAN F RS ROAD DHARMAVARAM AP 515671 NO

**40 Add an attribute of table parties as OTHERS having varchar(30)?**

alter table parties

add OTHERS varchar(30) null

**OUTPUT:**

Commands completed successfully.

Completion time: 2023-06-08T22:23:28.0830616+05:30

**41 find the all details from parties?**

select \* from parties

**output:**

**PID Pname party\_found OTHERS**

300 TELUGU DESAM PARTY 1982-03-29 NULL

301 JANASENA PARTY 2014-03-14 NULL

302 YCP 1999-06-10 NULL

303 BJP 1980-04-06 NULL

304 CONGRESS 1885-12-28 NULL

305 BRS 2023-04-11 NULL

**42 find the name and id of parties using JOIN?**

SELECT P.PNAME, p.PID

FROM PARTIES AS P

JOIN votes AS v ON p.PID = v.PID;

**OUTPUT:**

**PNAME PID**

TELUGU DESAM PARTY 300

JANASENA PARTY 301

TELUGU DESAM PARTY 300

BRS 305

JANASENA PARTY 301

BJP 303

CONGRESS 304

JANASENA PARTY 301

YCP 302

BJP 303

JANASENA PARTY 301

**43Add an attribute to administrator as age ?**

alter table administrator

add age int null

**output:**

Commands completed successfully.

Completion time: 2023-06-08T22:32:42.9307748+05:30

**44 find the all details from administrator?**

select \* from administrator

**output:**

**admin\_id admin\_name admin\_email admin\_password age**

100 ROHIT RO171@GMAIL.COM ROHIT12 NULL

101 RAHUL BULLET321@GMAIL.COM BULLET NULL

102 VENKATESH VENKY@GMAIL.COM ROHIT12 NULL

103 NEHA NEHA9817@GMAIL.COM NEHA12 NULL

104 DEVARSHINI DEV@GMAIL.COM DEV NULL

106 RAHANE [RAHA2321@GMAIL.COM](mailto:RAHA2321@GMAIL.COM) RAHAN NULL

107 VIJAY [VIJU@GMAIL.COM](mailto:VIJU@GMAIL.COM) VIJU2324 NULL

108 ALLU ARJUN ALLU@GMAIL.COM ALLU722 NULL

110 SAILAJA [SAILU2325@GMAIL.COM](mailto:SAILU2325@GMAIL.COM) SAILU897 NULL

111 ANJALI [ANJU@GMAIL.COM](mailto:ANJU@GMAIL.COM) ANJALI23 NULL

112 SUKUMAR SUKU8797@GMAIL.COM SUKU@5412 NULL

**45 Update administrator age as 49 of person having password'ROHIT12'?**

update administrator

**set age=49**

**where admin\_password='ROHIT12'**

**46 Find the all details of both voter and voting booth?**

SELECT vb.\*, vb.booth\_ID, vb.booth\_status

FROM voting\_booth vb

LEFT JOIN voter\_booth vbo ON vb.booth\_ID = vbo.booth\_ID;

**OUTPUT:**

**booth\_ID booth\_name booth\_location booth\_status booth\_ID booth\_status**

700 SV NAGAR RAJAMPETA available 700 available

700 SV NAGAR RAJAMPETA available 700 available

700 SV NAGAR RAJAMPETA available 700 available

700 SV NAGAR RAJAMPETA available 700 available

701 RS ROAD GUNTUR available 701 available

701 RS ROAD GUNTUR available 701 available

701 RS ROAD GUNTUR available 701 available

702 VIDYA NAGAR KADAPA available 702 available

703 RS ROAD KURNOOL not available 703 not available

704 RAMAPUR NANDYALA available 704 available

704 RAMAPUR NANDYALA available 704 available

704 RAMAPUR NANDYALA available 704 available

704 RAMAPUR NANDYALA available 704 available

705 RS ROAD KRISHNA not available 705 not available

706 VIDYA NAGAR RAJAMPETA available 706 available

707 RANGAMPETA NELLORE available 707 available

708 RS ROAD SRISAILAM available 708 available

709 RAMAPUR DHARMAVARAM available 709 available

709 RAMAPUR DHARMAVARAM available 709 available

710 KUCHIVARIPALLI KURNOOL not available 710 not available

711 KUCHIVARIPALLI KODUR not available 711 not available

712 RS ROAD NANDALUR available 712 available

713 RAMAPUR KADAPA available 713 available

714 AIR BYPASS GUNTUR not available 714 not available

715 RAMAPUR NELLORE not available 715 not available

716 SV NAGAR KRISHNA not available 716 not available

717 AIR BYPASS DHARMAVARAM available 717 available

718 RAMAPUR KADAPA available 718 available

718 RAMAPUR KADAPA available 718 available

719 VIDYA NAGAR SRISAILAM not available 719 not available

720 KUCHIVARIPALLI ANANTHAPUR available 720 available

721 RAMAPUR CHITOOR available 721 available

722 VIDYA NAGAR NELLORE available 722 available

723 RAMAPUR KADAPA available 723 available

724 AIR BYPASS RAJAMPETA not available 724 not available

**47 find the all details of both voting and voter booth having status is 'not available'?**

SELECT vb.booth\_ID, vb.booth\_name, vb.booth\_status, vbo.check\_in\_timestamp, vbo.check\_out\_timestamp

FROM voting\_booth vb

LEFT JOIN voter\_booth vbo ON vb.booth\_ID = vbo.booth\_ID

WHERE vb.booth\_status = 'not available';

**OUTPUT:**

**booth\_ID booth\_name booth\_status check\_in\_timestamp check\_out\_timestamp**

703 RS ROAD not available NULL NULL

705 RS ROAD not available NULL NULL

710 KUCHIVARIPALLI not available NULL NULL

711 KUCHIVARIPALLI not available NULL NULL

714 AIR BYPASS not available NULL NULL

715 RAMAPUR not available NULL NULL

716 SV NAGAR not available NULL NULL

719 VIDYA NAGAR not available NULL NULL

724 AIR BYPASS not available NULL NULL

**48 find the details of administrator and voting booth?**

SELECT a.\*, vb.\*

FROM administrator a

LEFT JOIN voting\_booth vb ON a.admin\_id = vb.booth\_ID;

**OUTPUT:**

**admin\_id admin\_name admin\_email admin\_password age booth\_ID booth\_name booth\_location booth\_status**

100 ROHIT RO171@GMAIL.COM ROHIT12 49 NULL NULL NULL NULL

101 RAHUL BULLET321@GMAIL.COM BULLET NULL NULL NULL NULL NULL

102 VENKATESH VENKY@GMAIL.COM ROHIT12 49 NULL NULL NULL NULL

103 NEHA NEHA9817@GMAIL.COM NEHA12 NULL NULL NULL NULL NULL

104 DEVARSHINI DEV@GMAIL.COM DEV NULL NULL NULL NULL NULL

106 RAHANE RAHA2321@GMAIL.COM RAHAN NULL NULL NULL NULL NULL

107 VIJAY VIJU@GMAIL.COM VIJU2324 NULL NULL NULL NULL NULL

108 ALLU ARJUN ALLU@GMAIL.COM ALLU722 NULL NULL NULL NULL NULL

110 SAILAJA SAILU2325@GMAIL.COM SAILU897 NULL NULL NULL NULL NULL

111 ANJALI ANJU@GMAIL.COM ANJALI23 NULL NULL NULL NULL NULL

112 SUKUMAR SUKU8797@GMAIL.COM SUKU@5412 NULL NULL NULL NULL NULL

**49 find the details of candidates and election results using left join?**

SELECT er.\*, cd.\*

FROM election\_results er

LEFT JOIN candidate\_details cd ON er.CID = cd.CID;

1300 1 600 1568 600 MALLIKARJUNA 1980-06-23 M 302 MLA RAJAMPETA AP 516115

1301 5 601 7832 601 KUSUMA KUMARI 1984-10-18 F 301 MLA RAJAMPETA AP 516115

1302 4 602 1568 602 CHANGAL RAYUDU 1979-11-17 M 300 MLA RAJAMPETA AP 516115

1303 6 603 5490 603 PARTHA SARATHI 1991-06-12 M 301 MP VISHAKAPATNAM AP 500020

1304 3 604 5568 604 SRINIVAS 1984-04-13 M 301 MLA GUNTUR AP 522001

1305 5 605 2590 605 VENKATESWAR 1983-06-10 M 302 MLA KURNOOL AP 518001

1306 1 606 568 606 PADMA 1980-08-03 F 303 MP CHITOOR AP 517001

1307 4 607 2568 NULL NULL NULL NULL NULL NULL NULL NULL NULL

1308 6 608 1469 608 ANIL KUMAR 1972-02-13 M 302 MLC SRISAILAM AP 518101

1309 2 609 1561 609 ESWAR 1979-02-04 M 303 MLA SRISAILAM AP 518101

1310 5 610 962 610 SAIBABU 1989-05-20 M 302 MINISTER GUNTUR AP 522001

1311 2 611 3565 611 SRIDEVI 1980-06-21 F 302 MP NELLORE AP 524002

1312 4 612 1566 612 SUBBARAO 1974-08-29 M 305 MLA DHARMAVARAM AP 515671

1313 2 613 6587 613 PRABHAKAR 1974-03-30 M 301 MLC KURNOOL AP 518001

1314 5 614 2562 614 SURYAVARA PRASAD 1979-10-31 M 303 MLA KADAPA AP 516001

1315 2 615 2263 615 RAGHUPATHI 1972-11-02 M 302 MLC CHITOOR AP 517001

1316 3 616 1462 616 ASHOK 1978-12-09 M 300 MLA GUNTUR AP 522001

1317 3 617 1964 617 SRI RAM 1976-03-02 M 304 MLA GUNTUR AP 522001

1318 1 618 2167 618 JAYLAKSHMAMMA 1982-05-05 F 300 MLA DHARMAVARAM AP 515671

1319 2 619 1962 619 BHUVANA KARUNAKAR 1985-06-13 M 302 MLA TIRUPATI AP 517501

1320 5 620 3165 620 SUGUNA 1980-08-25 F 300 MLA TIRUPATI AP 517501

1321 6 621 4163 621 RAHUL 1972-09-07 M 304 MP RAJAMPETA AP 516115

1322 4 622 3962 622 MAHESH 1980-06-08 M 303 MP VISHAKAPATNAM AP 500020

**50 find the partywon in election2024?**

select PARTYWON from ELECTION2024

**OUTPUT:**

**PARTYWON**

JANASENA

**CHAPTER 4. CONCLUSION AND FUTUREWORK**

In conclusion, election management is a critical and complex process that plays a fundamental democratic societies. Effective election management ensures the integrity, transparency, and inclusivity of the electoral process, allowing citizens to freely express their will and choose their representatives. Here are the key points to consider:

1. **Importance of Election Management**: Elections serve as the foundation of a democratic system, giving citizens the power to elect their leaders and hold them accountable. Proper election management is essential to maintain public trust, promote political stability, and foster a fair and representative government.
2. **Electoral Laws and Regulations**: A robust legal framework is crucial for effective election management. Clear and comprehensive electoral laws should be in place to guide the conduct of elections, including provisions for candidate registration, campaign financing, voter registration, ballot counting, and dispute resolution.
3. **Voter Registration and Education**: Ensuring an accurate and inclusive voter registry is essential for a successful election. Election management bodies must implement effective voter registration systems and engage in voter education campaigns to inform citizens about their rights, the electoral process, and the importance of participation.
4. **Transparency and Accountability**: Transparency throughout the electoral process is crucial to building public trust. Election management bodies should provide regular updates on key activities, such as candidate registration, voter registration, ballot printing, and vote counting. Independent observation missions and media coverage also contribute to accountability and transparency.
5. **Electoral Security**: Maintaining a secure environment during elections is paramount. Measures should be taken to safeguard the integrity of the voting process, including securing polling stations, protecting voter information, and preventing any form of voter intimidation or coercion.

**4.2 Future Work**

Databases are used for **storing, maintaining and accessing any sort of data**. They collect information on people, places or things. That information is gathered in one place so that it can be observed and analyzed. Databases can be thought of as an organized collection of information and this helps us in creation of a website or a mobile app where it can be accessed by viewers globally.