ELECTION DATABASE

Introduction to the mini world:

This is a mini-world database for elections in a particular constituency. This structured database is designed to efficiently manage data related to electoral processes. It encompasses strong entities, like voters, election events, and candidates, as well as weak entities for campaign finance and historical election data. Relationships between these entities help maintain the integrity and transparency of election data, supporting processes such as voter registration, candidate tracking, and historical election result analysis.

Purpose of the Database:

The purpose of this database is to efficiently manage electoral data, enabling voter registration, election event management, candidate tracking, political party management, data analysis, government oversight, election observation, and public access. It serves as a central repository for election-related information, promoting transparency and facilitating fair and democratic elections.

Users of the Database:

Administrators

- **System Administrators**: Highest access, manage system settings and user roles. Observe and ensure fair elections.

Election Officials

- **Voter Registration Officials:** Manage voter records.
- **Election Event Managers:** Oversee election events.
- **Polling Station Supervisors:** Manage polling stations.

Candidates and Political Parties

- **Candidates:** Access campaign data.
- **Political Party Representatives:** Access party information.

Analysts and Researchers

- **Data Analysts:** Access data for research and analysis.

General Public

- **General Public:** Access for viewing election results, candidate profiles and party information.

Database Requirements

Strong Entity Types:

Entity Type	Attribute	Attribute Type	Sub Attributes	Data Type

Voters	Voter ID	Key Attribute	-	INT
	Name	Composite	First Name	VARCHAR
			Last Name	VARCHAR
	Address	Simple	-	VARCHAR
	Date of Birth	Simple	-	DATE
	Contact Information	Composite	Phone Number	INT
		Simple	E-mail ID	VARCHAR
Candidates	Candidate ID	Key Attribute	-	INT
	Name	Composite	First Name	VARCHAR
			Last Name	VARCHAR
	Party Affiliation	Simple	-	VARCHAR
	Contact Information	Composite	Phone Number	INT
			E-mail ID	VARCHAR
Election Events	Event ID	Key Attribute	-	INT
	Date	Simple	-	DATE
	Type	Simple	-	VARCHAR
	Election Type	Simple	-	VARCHAR
Polling Stations	Station ID	Key Attribute	-	INT
	Address	Simple	-	VARCHAR
	Contact Information	Composite	Phone Number	INT
			E-mail ID	VARCHAR
Political Parties	Party ID	Simple	-	INT
	Name	Simple	-	VARCHAR
	President	Composite	First Name	VARCHAR
			Last Name	VARCHAR
	Contestant	Composite	First Name	VARCHAR
			Last Name	VARCHAR
	Platform	Simple	-	VARCHAR
Ballots	Ballot ID	Key Attribute	-	INT
	Election Event	Simple	-	VARCHAR
	List of Candidates	Multi-Valued	-	VARCHAR
Election Officials	Official ID	Key Attribute	-	INT
	Name	Composite	First Name	VARCHAR
			Last Name	VARCHAR
	Role	Simple	-	VARCHAR
	Contact Information	Composite	Phone Number	INT
			E-mail ID	VARCHAR

Weak Entity Types:

Entity Type	Attribute	Attribute Type	Sub Attributes	Data Type
Campaign Finance	Transaction ID	Key Attribute	-	INT

	Candidate ID	Foreign Key	-	INT
	Date	Simple	-	DATE
	Amount	Simple	-	INT
Historical Data	Event ID	Foreign Key	-	INT
	Date	Simple	-	DATE
	Election Type	Simple	-	VARCHAR
	Total Votes	Simple	-	INT
	Winner	Composite	First Name Last name	VARCHAR
	Turnout Rate	Simple	-	INT
Vote Status	Voted On	Simple	-	DATE
	Status	Simple	-	VARCHAR
	Voter ID	Foreign Key	-	INT
Demographic information	Voter ID	Foreign Key	-	INT
	Age	Derived	-	INT
	Gender	Simple	-	VARCHAR
	Ethnicity	Simple	-	VARCHAR
	Education Level	Simple	-	VARCHAR
	Income Level	Simple	-	VARCHAR

RELATIONSHIPS:

BETWEEN STRONG ENTITIES:

1.**CONTESTS_IN** (Candidates (1, 1) and Election Events (1, N) [N:1]):

- A binary many-to-one relationship where many candidates participate in a single election event.

2. **STATIONS_FOR** (Polling Stations (1,1) and Election Events (1, N) [N:1]):

- A binary many-to-one relationship where many polling stations are associated with a single election event

3. **ASSIGNED_TO … IN** (Election Officials (1,1), Polling stations (1, M) and Election Event (1, N) [N:M:1]):

- A tertiary many-to-one relationship where many Election Officials are Assigned to each polling station in the election event.

4. **VOTES_IN … ALLOCATED_IN** (Voter (1,1) and Polling Stations (1,1) and Election Event (1, N) [N:1:1]):

- A tertiary many-to-one relationship where many voters are assigned to a single polling station to cast their vote in the election event.

5. **SENT_TO** (Ballots (1,1) and Polling Stations (1,1) [1:1]):

- A binary one-to-one relationship where a single ballot is sent to each polling station.

BETWEEN STRONG ENTITY AND WEAK ENTITY: (Identifying Relations)

1. **FINANCE_OF** (Candidates (1,1) and Campaign Finance (1,1) [1:1]):

- A binary one-to-one relationship where each candidate can have multiple campaign finance records, tracking contributions and expenditures.

2. **STATUS_OF** (Voters (1,1) and Vote Status (1,1) [1:1]):

- A binary one-to-one relationship where the voting status of each voter is stored.

3. **HISTORY_OF** (Election events (0,1) and Historical data (1,1) [1:1]):

- A binary one-to-one relationship election event and historical data.

4. **INFORMATION_OF** (Voters (1,1) and Demographic Information (1,1) [1:1]):

- A binary one-to-one relationship where each voter has one demographic information record, capturing their demographic details.

Functional Requirements:

MODIFICATIONS:

- **INSERT:**
  - **INSERT_VOTER**: The system should allow election officials to insert new voter records, capturing their voter ID, name, address, date of birth, phone number, and email ID.
  - **CREATE_EVENT:** The system should enable the insertion of new election events, specifying event details like date, type, location, election type, and status.
  - **INSERT_STATION:** Election officials should be able to insert new polling station records, specifying their station ID, location, and address.
  - **INSERT_PARTY:** Users should be able to insert political party records with details like party ID, name, President, and platform.
  - **INSERT_CANDIDATE:** The system should allow election officials to insert candidate records, capturing their name, election event, list of candidates and ballot format.
  - **INSERT_CAMPAIGN_FINANCE:** Candidates should be able to insert campaign finance records that track contributions and expenditures.
  - **INSERT_OFFICIAL:** Election officials should be able to insert new official records with their official ID, name, role, and contact information.
  - **INSERT_HISTORICAL_ELECTION_DATA:** The system should allow the insertion of historical election data, including event ID, date, election type, total votes, winner, turnout rate, and vote status.
  - **INSERT_RECORDS:** Users should be able to insert records for voter status, capturing the date and status.
  - **INSERT_DEMOGRAPHIC_DATA:** Users should be able to insert demographic information for voters, including age, gender, ethnicity, education level, and income level.
- **DELETE:**
  - **DELETE_VOTER**: The system should allow authorized users to delete voter records.
  - **DELETE_EVENT**: Authorized users should be able to delete election event records.
  - **DELETE_STATION**: Authorized users should be able to delete polling station records.
  - **DELETE_PARTY**: Authorized users should be able to delete political party records.
  - **DELETE_CANDIDATE**: Authorized users should be able to delete candidate records.
  - **DELETE_OFFICIAL**: Authorized users should be able to delete election-official records.
- **UPDATE:**
  - **UPDATE_VOTER_INFO**: Election officials should be able to update voter information as needed.
  - **UPDATE_EVENT**: Authorized users should be able to update event details.
  - **UPDATE_STATION**: Authorized users should be able to update polling station information.

  - **UPDATE_PARTY**: Authorized users should be able to update political party information.
  - **UPDATE_CANDIDATE**: Election officials should be able to update candidate information.
  - **UPDATE_OFFICIAL_INFO**: Authorized users should be able to update election official information.
  - **UPDATE_HISTORICAL_DATA**: Authorized users should be able to update historical election data.
  - **UPDATE_VOTER_STATUS**: Authorized users should be able to update the voting status of individual voters.
  - **UPDATE_DEMOGRAPHIC_DATA**: Authorized users should be able to update demographic information for voters.

RETRIEVALS:

- **SELECTION:**
  - **SORT_VOTERS_BY_AGE**: Sorts list of voters by their age range.
  - **SORT_CANDIDATES_BY_AGE:** Sorts list of candidates by age.
- **PROJECTION:**
  - **VIEW_VOTER_CONTACT_INFO:** Displays specific attributes from records, such as phone numbers and email addresses of voters for authorized users only.
  - **VIEW_CANDIDATE_INFO:** Displays specific attributes related to candidates, such as their names, party affiliations, and campaign expenditure. Contact details of Candidates are accessible only to Administrators.
  - **VIEW_STATIONS:** Displays polling stations and their contact information.
  - **GET_STATION:** Get polling station assigned to a voter based on Voter ID.
  - **GET_OFFICIALS:** Get List of Election Officials assigned to a polling station based on Station ID.
- **AGGREGATE:**
  - **TOTAL_VOTER_PARTICIPATION:** Perform an aggregate function to calculate the total number of voters who participated in the election event.
  - **AVERAGE_CAMPAIGN_FUND:** Perform an aggregate function to calculate the average campaign fund for a candidate.
- **SEARCH:**
  - **SEARCH_VOTER_BY_VOTER_ID:** Searches voters by Voter IDs.
  - **SEARCH_CANDIDATE_BY_PARTY:** Searches for the candidate based on the political party he/she is representing
  - **SEARCH_POLLING_STATION_BY_ADDRESS:** Searches for a polling station based on address.

Analysis:

- **GET_PERCENTAGE_POLLED_BASED_ON_SOCIAL_TENDENCIES:** Generates the percentage of votes polled by different age groups, gender and Income (Also, used to analyze the preferences of people from different demographic backgrounds).
- **GET_REPORT_BETWEEN_CAMPAGIN_FUND_AND_WINNER:** Gets a report showing relation between the campaign funds by a candidate and the final elected candidate. (Which on building further, can predict winner).
- **PREDICTING_WINNER:** This function does some math between the Previous Voter Turnout, This Voter Turnout and the previous leader and helps to predict the winner of the current election. (Professional analysts can predict the winner based on the poll percentage).

- **DISTANCE_BETWEEN_VOTER_LOC_AND_STATION_LOC:** This gets the distances between the polling station assigned to voters and their current address in the database, which is used to show that the voter prefers nearest polling station.

Applications of the Database:

1.Voter registration and management

- The database can be used to register new voters and maintain their records, including their voter ID, name, address, date of birth, contact information, and eligibility status.
- It can also be used to generate voter lists and polling station assignments.

2. Election event management

- The database can be used to create and manage election events, including their dates, locations, types, and candidate lists.
- It can also be used to generate ballots and other election materials.

3. Candidate tracking

- The database can be used to track candidate's participation in election events and their campaign finance information.

4. Political party management

- The database can be used to view political party information.

5. Data analysis

- The database can be used to analyze election data to identify trends and patterns. For example, analysts can use the database to study voter turnout, demographics, and voting preferences.

6. Government oversight and election observation

- The database can be used by government agencies and election observers to monitor and regulate elections.

7. Public access

- The database can be used to provide public access to information about elections and candidates. For example, voters can use the database to find their polling station or to learn more about the candidates running for office.

Assumptions:

- Candidates participating individually are listed as independent.
- Each polling station contains exactly one ballot.
- Each polling station has an administrator who has full access to the database.

- Ballot implies EVM, i.e., voting takes place only through EVMs and not by other means.
- A ballot is used only for one election event.

Uniqueness of the Database:

This database is a first of its kind, the already existing databases only show the poll percentages or Voter ID of that Voter or any other functionalities individually, this is an attempt to put it all in a single database so that it is user friendly as it is easy to manage and accessible to everyone. The database can be used by government agencies and election observers to monitor and regulate elections. For example, they can use the database to ensure that voter registration is accurate and that election results are fair and accurate.

Summary:

This database serves as a comprehensive system for managing all aspects of an electoral process within a specific constituency. It facilitates vital functions like voter registration, election event management, candidate tracking, political party oversight, data analysis, government regulation, and election observation. It also provides limited public access for voters to find information about elections and candidates. Ultimately, it plays a crucial role in ensuring transparent, fair, and well-organized democratic elections.