Case Study: 2024 Indian General Elections – Analyzing Constituency, Party, and State-Level Results

1. Introduction

Elections in India are one of the largest democratic exercises in the world. In this case study, we analyze the 2024 Indian General Elections, focusing on the Lok Sabha (House of the People) elections, using constituency-wise, party-wise, and state-wise results.

Objective of the Project

The objective of this project is to **build a structured database system** for analyzing **Indian General Election results** at the **constituency, party, and state levels**. The project aims to:

1. **Develop a Centralized Database** – Store election data in a well-organized SQL database for efficient querying and analysis.
2. **Standardize Election Data** – Define relationships between **candidates, constituencies, parties, and states** for accurate representation.

3. **Analyze Voting Trends** – Identify **winning margins, party dominance, and state-wise voting patterns**.
4. **Provide Data-Driven Insights** – Enable decision-makers, researchers, and journalists to analyze **voter behavior and political shifts**.

Tables :

- **states** – Stores state-wise information with unique state_id.
- **partywise_results** – Stores political party details, including the number of seats won (won).
- **constituencywise_results** – Stores results for each constituency, including the winning candidate, total votes, margin, and associated party (party_id).
- **Candidates** – Stores detailed voting data for each candidate, including EVM votes, postal votes, total votes, and % of votes.
- **statewise_results** – Stores state-wise election results, including leading and trailing candidates, margin, and status.

How the Schema Works:

- **The states table** ensures state-level classification.
- **The partywise_results** table tracks how many seats each party has won.
- The **constituencywise_results** table captures who won in each constituency, total votes, and margin.
- The **Candidates** table breaks down votes for each candidate, including EVM and postal votes.
- The **statewise_results** table gives a bigger picture at the state level, showing which candidates are leading and trailing.

SQL TABLE SCHEMAS:

```
CREATE TABLE states (
   state_id INT PRIMARY KEY,
   state_name VARCHAR(255) NOT NULL
);
```

```
CREATE TABLE partywise_results (
   party_id INT PRIMARY KEY,
   party_name VARCHAR(255) NOT NULL,
   won INT NOT NULL
);
```

```
CREATE TABLE constituencywise_results (
   constituency_id INT PRIMARY KEY,
   parliament_constituency VARCHAR(255) NOT NULL,
   constituency_name VARCHAR(255) NOT NULL,
   winning_candidate VARCHAR(255) NOT NULL,
   total_votes INT NOT NULL,
   margin INT NOT NULL,
   party_id INT,
   FOREIGN KEY (party_id) REFERENCES
partywise_results(party_id)
);
```

```
CREATE TABLE Candidates (
    candidate_id INT PRIMARY KEY AUTO_INCREMENT,
    candidate_name VARCHAR(255) NOT NULL,
    party_name VARCHAR(255) NOT NULL,
    evm_votes INT NOT NULL,
    postal_votes INT NOT NULL,
    total_votes INT NOT NULL,
    vote_percentage DECIMAL(5,2) NOT NULL,
    constituency_id INT,
    FOREIGN KEY (constituency_id) REFERENCES
constituencywise_results(constituency_id)
);
```

```
CREATE TABLE statewise_results (
    statewise_id INT PRIMARY KEY AUTO_INCREMENT,
    constituency VARCHAR(255) NOT NULL,
    constituency_no INT NOT NULL,
    parliament_constituency VARCHAR(255) NOT NULL,
    leading_candidate VARCHAR(255) NOT NULL,
    trailing_candidate VARCHAR(255) NOT NULL,
    margin INT NOT NULL,
    status VARCHAR(50) NOT NULL,
    state_id INT,
    FOREIGN KEY (state_id) REFERENCES states(state_id)
);
```

Election Data Analysis - Key SQL Queries

a. **Party Performance Analysis** - Find the party with the highest number of seats won.

b. **Winning Margin Analysis** - Identify the candidate with the highest and lowest winning margin.

c. **Total Votes by Party** - Calculate the total votes received by each party across all constituencies.

d. **Closest Contest Analysis** - Find the constituency with the smallest winning margin.

e. **Statewise Voter Turnout** - Determine the state with the highest and lowest voter turnout.

f. **Winning Percentage Calculation** - Calculate the percentage of votes secured by the winning candidate in each constituency.

g. **Majority Constituencies** - Find the number of constituencies where the winning candidate received more than 50% of total votes.

h. **Runner-Up Party Analysis** - Identify the party that secured the most runner-up positions.

i. **Independent Candidates' Performance** - Find the number of independent candidates who won and their constituencies.

j. **Margin of Victory Trends** - Calculate the average margin of victory across all constituencies.