**Practical-1**

**AIM:**

Create a real-time voting system where users can vote on a poll and see the results updated in real-time using only JavaScript, HTML, and CSS.

HTML: A simple poll interface with buttons to vote and display theresults.

CSS: Styles the poll and results.

JavaScript:

Defines a vote function to update the local votes.

Updates the vote counts in the UI.

Simulates real-time voting by randomly incrementing votes.

Notes:

* Votes object: Keeps track of the current vote counts for each language. It initializes each language with 0 votes.
* vote function: This function is called when a button is clicked. It increments the vote count for the selected language and calls updateVotes to refresh the displayed vote counts.
* updateVotes function: Updates the displayed vote counts by setting the text content of the spans in the results section to the current vote counts.
* setInterval: Simulates real-time updates by randomly incrementing the vote counts every 2 seconds. This mimics votes coming in from other users in real time.

**THEORY:**

This project is a real-time voting system built using HTML, CSS, and JavaScript. It allows users to vote for their favorite programming language and view live vote counts. JavaScript handles the logic, updates the UI dynamically, and simulates real-time voting using setInterval(). The project demonstrates basic DOM manipulation and event handling in a browser environment.

**Code:**

**vote.html**

<!DOCTYPE html>

<lang lang="en">

<head>

  <meta charset="UTF-8" />

  <meta name="viewport" content="width=device-width, initial-scale=1.0" />

  <title>Real-Time Voting</title>

  <link rel="stylesheet" href="style.css" />

</head>

<body>

  <script src="vote.js"></script>

  <div class="poll-container">

    <h2>Vote for Your Favorite Language</h2>

    <div class="buttons">

      <button onclick="vote('JavaScript')">JavaScript</button>

      <button onclick="vote('Python')">Python</button>

      <button onclick="vote('Java')">Java</button>

    </div>

    <div class="results">

      <p>JavaScript: <span id="js-votes">0</span></p>

      <p>Python: <span id="python-votes">0</span></p>

      <p>Java: <span id="java-votes">0</span></p>

    </div>

  </div>

</body>

</lang>

</html>

**vote.js**

const votes = {

  JavaScript: 0,

  Python: 0,

  Java: 0

};

function vote(language) {

  votes[language]++;

  updateVotes();

}

function updateVotes() {

  document.getElementById('js-votes').textContent = votes.JavaScript;

  document.getElementById('python-votes').textContent = votes.Python;

  document.getElementById('java-votes').textContent = votes.Java;

}

setInterval(() => {

  const languages = Object.keys(votes);

  const randomLang = languages[Math.floor(Math.random() \* languages.length)];

  votes[randomLang]++;

  updateVotes();

}, 2000);

**OUTPUT:**

****

* The heading asks users to vote for their favorite language.
* Three buttons (JavaScript, Python, Java) let users vote.
* Below the buttons, current vote counts are shown.
* JavaScript updates votes instantly when clicked and also simulates real-time voting every 2 seconds.

**LEARNING OUTCOME:**

* Online Polling in Live Events – Used in webinars, YouTube Live, and conferences to collect audience feedback instantly.
* Social Media Voting (e.g., Instagram Polls, Twitter Polls) – Allows users to vote and see live results.
* Classroom Response Systems – Teachers use real-time polls to check student understanding during lectures.
* TV Shows & Contests – Shows like *Indian Idol* or *Bigg Boss* use real-time voting from the audience.
* Customer Feedback & Surveys – Businesses collect user opinions quickly on websites or apps.

**REFERENCES:**

* W3Schools
* GeeksforGeeks