# 📝 Exercise: Mini Poll with Live Results

**Goal**: Practice JSX, `useState`, **loops** (`array.map`) and **conditional rendering** (`&&`, ternary).

## Task

Build a component called `MiniPoll`.

1. **Initial Data (state)**
    - Keep an array of options in state, e.g. `["React", "Vue", "Svelte"]`.
    - Keep a parallel votes array or an object for counts (your choice).
    - Keep a boolean `showResults` for toggling results.
2. **Render (loops + conditionals)**
    - Loop over the options to render a list.
    - Each option has a **Vote** button that increments its count.
    - Next to each option, conditionally show its current votes **only if** `showResults` is true.
        - Use either `showResults && <span>…</span>` or a ternary.
3. **Controls**
    - A **“Toggle Results”** button that flips `showResults`.
    - An **“Add Option”** mini form (one text input + button) that appends a new option to the list (start its votes at 0).
        - If input is empty, show a small conditional message: “Please enter a name.”
    - A **“Reset Votes”** button that sets all counts to 0.
4. **Highlights**
    - Above the list, conditionally show the **leader**: the option with the highest votes (tie → show “It’s a tie!”).

        - If `showResults` is false, hide the leader section entirely.
    - If there are **no options**, show a placeholder message instead of the list.

5. **Styling (simple)**
    - Title at top.
    - Options as a clean vertical list.
    - Leader text bold and slightly larger.

## Example Behavior

- Start with options and zero votes. Results are hidden.
- Clicking **Toggle Results** shows the vote counts next to each option.
- Clicking **Vote** increases that option’s count; leader updates immediately.
- Adding a new option shows it in the list (0 votes).
- Clicking **Reset Votes** zeros all counts.
- Submitting an empty option name shows a small warning message under the input (conditional).

---

## What this reinforces

- **`useState`** for arrays/objects and booleans
- **Loops** with `map` for rendering lists
- **Conditional rendering** with `&&` and ternaries
- Derived display (leader) computed from current state without extra hooks