

Ramadan Prompting

Nights

Scenario-based prompting competition. 30 nights. 30 real-world coding scenarios.
by Sir Asharib Ali · for GIAIC Students

The screenshot shows a dark-themed coding challenge interface. At the top, there are three colored buttons: 'Day 2' (orange), 'Easy' (green), and 'Weighted scoring' (purple). Below these, the title 'JavaScript Prayer Scheduler' is displayed in a large, bold, white font. Under the title, the word 'Scenario:' is followed by a detailed description in white text: 'Scenario (JavaScript): Write `nextPrayerIndex` that receives current minutes and sorted prayer minutes, returning index of the next prayer, or 0 for next day rollover.' Further down, the 'Example Output:' section shows the code 'nextPrayerIndex(700,[300,600,900,1100]) -> 2' in a light gray box.

PROMPT:

Goal: Write a JavaScript function `nextPrayerIndex` (`currentMinutes`, `prayerMinutes`) that returns the index of the next prayer based on the current time in minutes. If the current time is past all prayer times, the function should return 0 for the next day rollover.

Constraints:

1. `currentMinutes` is an integer between 0 and 1439 (representing minutes from 00:00 to 23:59)
2. `prayerMinutes` is a sorted array of integers in ascending order, each between 0 and 1439.
3. Array length ≥ 1 (at least one prayer time)

Edge Cases:

1. `currentMinutes` is before the first prayer -> return index 0.
2. `currentMinutes` is after the last prayer -> return index 0 (next day rollover).
3. `currentMinutes` matches exactly a prayer time -> return index of that prayer.
4. Only 1 prayer in the array -> always return index 0.

Output Format: Return a single integer representing the index of the next prayer in the `prayerMinutes` array.

Example:

```
nextPrayerIndex(700, [300, 600, 900, 1100]) // returns 2  
nextPrayerIndex(1200, [300, 600, 900, 1100]) // returns 0
```