Optimizing Smart Parking: Data-Driven Operational Insights

Problem Statement

Parking facilities suffer from a critical lack of real-time data, leaving essential questions unanswered: What is the current vehicle count, which zones are utilized, how long do patrons stay, and what is the projected revenue?

Current, unreliable methods—like manual checks, flawed entry/exit logs, and unauditable billing—cause driver frustration and congestion due to unknown occupancy. They also create operational blind spots for management, delaying detection of issues like overcrowding or unauthorized exits, and result in inaccurate, delayed reporting. Furthermore, the lack of data prevents easy performance benchmarking across different periods (weekdays vs. weekends, event days, peak hours).

Proposed Solution

This simulated Smart Parking system uses IoT-style entry/exit events (with timestamps and license plates) to track vehicles. It processes these events into parking sessions to monitor real-time occupancy, duration, and movement. A pricing engine calculates fees based on configurable rules (hourly, free time, peak, daily max). The processed data yields analytics on space utilization, peak hours, average stay, revenue, and anomalies. This fully simulated, scalable model mirrors a real-world smart parking setup.

Dataset Description

This dataset contains simulated IoT data from a smart parking facility, including time-stamped entry and exit events from parking gates. Records detail facility ID, gate ID, event type, license plate, and event time, used to derive parking sessions for calculating duration, occupancy, and charges. Designed to mimic real-world conditions like peak traffic and data inconsistencies, this synthetic data is structured like actual IoT parking data, suitable for testing analytics, dashboards, and billing logic.

List of Tables

Parking Events Table (`PARKING_EVENTS`)

This table stores raw IoT-style data generated at the parking facility's entry and exit gates. Each record represents a vehicle entering or leaving the facility, along with the event timestamp, gate information, and license plate details. Since the data is captured directly from sensors, it may include errors or missing values, reflecting real-world conditions.

Parking Sessions Table (`PARKING_SESSIONS`)

The parking sessions table is derived from the raw events and represents a complete parking activity for each vehicle. It combines entry and exit events to calculate parking duration, charges, and identify abnormal situations such as missing exits or duplicate entries. This table is primarily used for occupancy analysis and billing.

Pricing Rules Table (`PRICING_RULES`)

This table defines the pricing structure applied to parking sessions. It includes hourly rates, free parking limits, peak-hour multipliers, and maximum daily charges. Separating pricing rules from event data allows flexible updates to tariffs without affecting historical records.

Parking Facility Table (`PARKING_FACILITY`)

The facility table stores static information about the parking location, including total capacity and location details. It provides context for utilization and occupancy analytics and supports analysis across multiple facilities.