

DATABASE DESIGN & SQL.

CASE STUDY: FOXCORE RETAIL

Executive Summary

This report outlines the challenges faced by Foxcore Retail, a small business selling novelty items at festivals and tradeshows, in managing sales data. The report proposes a solution through the creation of a custom database to track sales, events, salespeople, venues, products, and booths.

1. Company Background

Foxcore Retail is a small retail business specializing in inexpensive novelty items like bubble guns and cooling towels. As the business grew, they hired salespeople and expanded their product portfolio.

2. Problem Statement

Foxcore Retail faced difficulties managing sales data due to:

- **Manual calculations:** Calculating sales commissions and compiling data became tedious and error-prone.
- **Spreadsheets:** Information scattered across spreadsheets hampered operational efficiency and strategic decision-making.
- **Lack of tracking:** They needed a system to track sales details like what was sold, by whom, when, and where.

3. Constraints

- **Time crunch:** A solution was needed within 30 days before the festival season.
- **Budget constraints:** Recent expansion made cost management challenging.

4. Proposed Solution: Database System

A custom database is the optimal solution for Foxcore Retail to track critical business records. The database will focus on entities like:

- **Events**
- **Sales**
- **Salespeople**
- **Venues**
- **Products**
- **Booths**

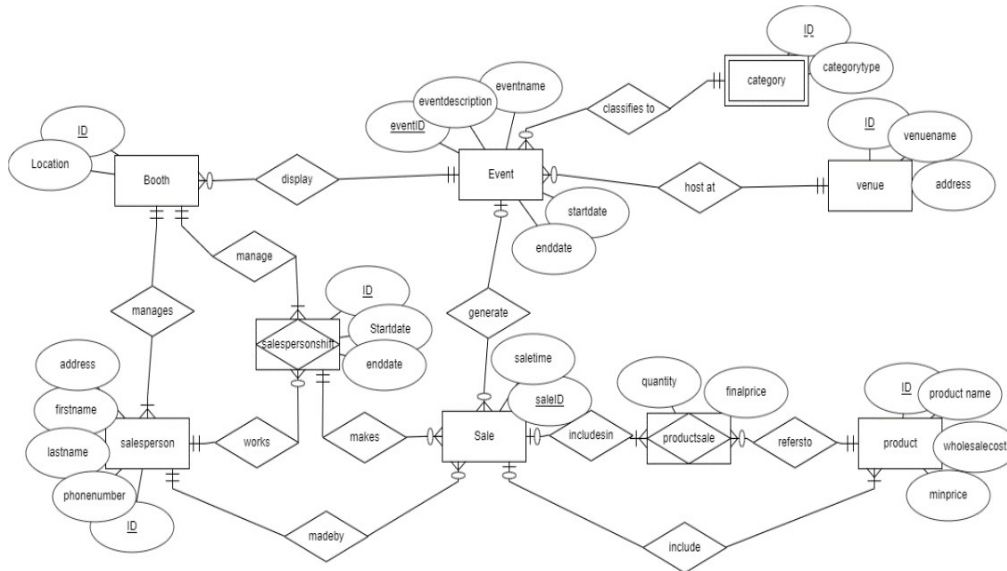
5. Database Design

The database will be designed with the following entities, relationships, and attributes:

Entities	Relationship	Attribute
Product	<ul style="list-style-type: none">• Associates' product sale with the sale.• refer to• Include	ID (primary key), product name, wholesale cost, min price
Sale	<ul style="list-style-type: none">• Includes in,• Associates' product sales with product• Generates• Associates' sale person shifts with salesperson• Made by and makes	Sale ID (primary key), sale time
Salesperson	<ul style="list-style-type: none">• Associates' salesperson shift with sales• Works• Manages• Made by	ID (primary key), address, first name, last name, phone number,
Event	<ul style="list-style-type: none">• Hosts at• Classifies to (category-weak entity)• Generate Display	Event ID (primary key), event description, event name, start date, end date
Booth	<ul style="list-style-type: none">• Manages• Display Associates' salesperson shift with sales person	ID (primary key), location
Venue	<ul style="list-style-type: none">• Host at	ID (primary key), venue name, address
Product sale (Associate entity)	<ul style="list-style-type: none">• Associates' sales and product	Quantity, Final price
Sales person shift (Associate entity)	<ul style="list-style-type: none">• Associates' booth with a salesperson• Associates' sales person with sales	ID (primary key), start date, end date
Category (Weak entity)	<ul style="list-style-type: none">• Classifies event	ID (primary key), venue name, address

6. Creation of ER Diagram

The ER diagram is created by ERD plus tool, aiming to formulate relationships between the entities. Where unique identifiers are selected in each table that is, captured as a *primary key constraint* and to the relationships between entities. The order of creating tables, and the relational constraints to other tables. The following ER diagram shows the relationships between the entities:



7. Database Design (SQL code)