Movie Theater Entity Relational Database

customers

PK	customer_id	SERIAL
	first_name	VARCHAR(100)
	last_name	VARCHAR(100)
	address	VARCHAR(150)
	billing_info	VARCHAR(150)

One cusomer can have zero or many tickets. Similarly one customer can have zero or many concessions.

In this case the movie_id contains all specific information about the movie in question

ticket

PK	ticket_id	SERIAL
FK	movie_id	INTEGER
	movie_time	VARCHAR(10)
	barcode	INTEGER
	ticket_price	NUMERIC(5,2)
	movie_date	VARCHAR(10)
FK	customer_id	VARCHAR(100)

concessions

PK	concessions_id	SERIAL
	order_sub	NUMERIC(5,2)
	order_total	NUMERIC(5,2)
	tax	NUMERIC(5,2)
FK	customer_id	integer
FK	product_id	Integer
	tip	NUMERIC(5,2)

Concessions in this scenario are the food/snacks items the customer will purchase. Each item will have a barcode or UPC which in this case is the product_id. Many products can be associated with one concession purchase.

There could be several tickets sold for the same movie or there could be none for a movie that no one wants to watch. There will be exactly one ticket sold for one showing.

movies

PK	movie_id	SERIAL
	movie_name	VARCHAR(100)
	movie_genre	VARCHAR(100)
	language	VARCHAR(100)
	movie_rating	VARCHAR(6)

product

PK	product_id	SERIAL
	price	NUMERIC(5,2)
	product_name	VARCHAR(100)
FK	inventory_id	integer
	product_type	VARCHAR(100)

one inventory can carry many different products.

inventory

PK	inventory_id	SERIAL
	product_amount	integer
FK	theater_id	integer
FK	warehouse_id	integer

Several different theater inventories can be fulfilled by one warehouse.

For this one imagine the scenario where several different theater locations have the same inventory on one or several items

warehouse

PK	warehouse_id	SERIAL
	location	VARCHAR(100)

Theater

PK	theater_id	SERIAL
	address	VARCHAR(100)
	number_screens	integer

One theater will have many screen rooms.

screen room

PK	room_id	SERIAL
	number_seats	Integer
	number_exits	Integer
FK	theater_id	Integer