

customers		
primary	customer_id	SERIAL
	first_name	VARCHAR(100)
	last_name	VARCHAR(100)
	email	VARCHAR(150)

For the customers table, i chose to use a primary key for the customer id, so that the customer can be connected to everything they buy - the ticket(s) and treats

the first_name, last_name, and email are all for this hypothetical movie theater's desire to flood a customers email with dozens of promotions.

movies		
primary	tickets	INTEGER
	movie_id	SERIAL
	movie_name	VARCHAR(100)
	rating	VARCHAR(20)

the primary key tickets is for connecting the tickets available for each movie to the tickets table

the movie_id is so each movie has a numerical alternative

the movie_name and rating is just for information about the movie

tickets		
foreign	price	NUMERIC(4,2)
foreign	tickets	INTEGER
foreign	customer_id	SERIAL

our customers need to go broke from the absurd prices at the theater, price will determine that

tickets is from movies

customer_id (from the customer table) is to connect the customer with the tickets they choose to bankrupt themselves with

concessions		
	item_id	SERIAL
	item_name	VARCHAR(100)
	price	NUMERIC(4,2)
foreign	customer_id	SERIAL

the item_id, item_name, and price are to assign an id and name to each concession offered and a price associated with that id

the customer_id which is from customers is in the concessions table to connect an order of treats to the customer