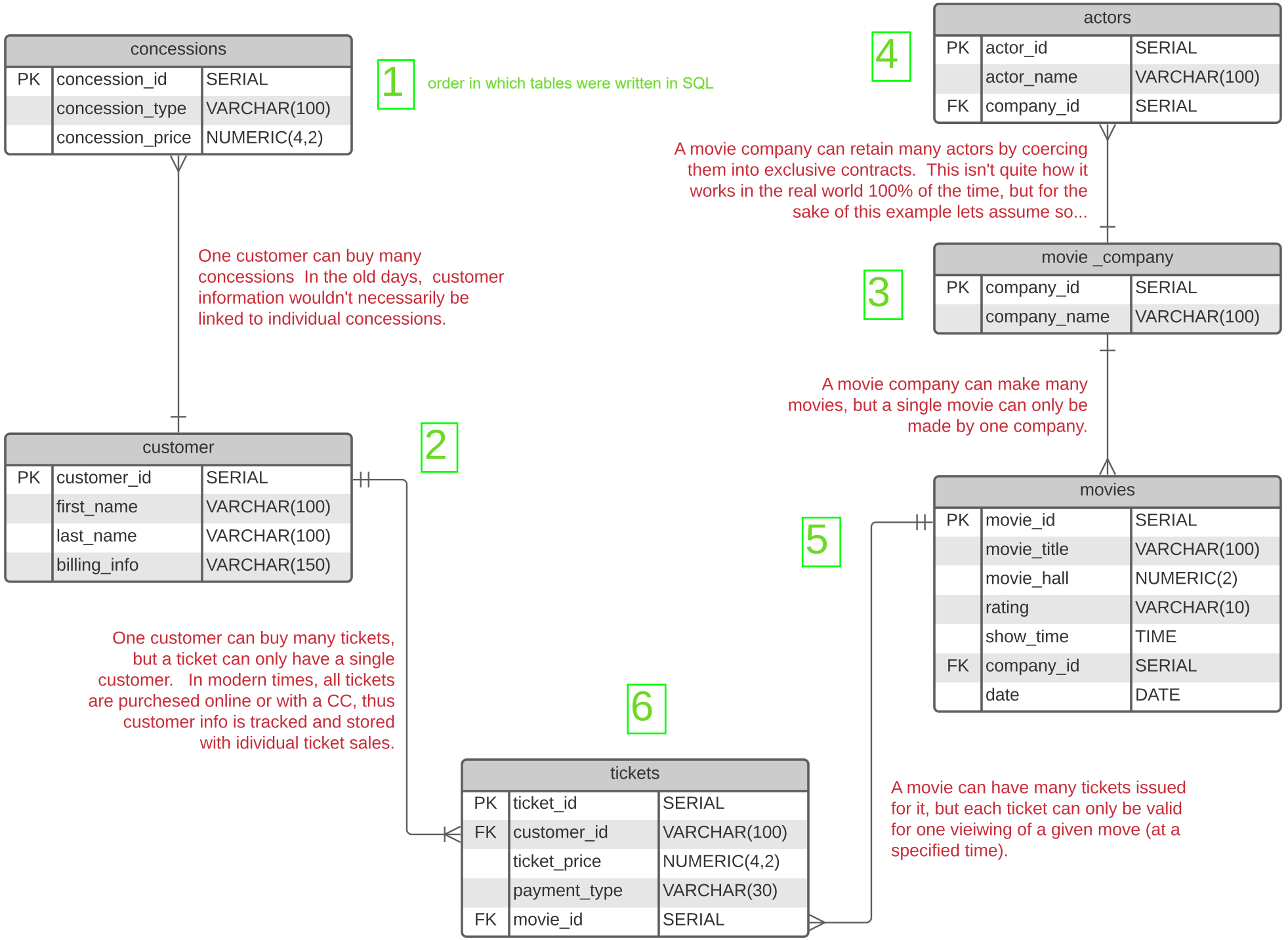
concessions
PK concession_id SERIAL
concession_type VARCHAR(100)
concession_price NUMERIC(4,2)
1
order in which tables were written in SQL
4
actors
PK actor_id SERIAL
actor_name VARCHAR(100)
FK company_id SERIAL
A movie company can retain many actors by coercing them into exclusive contracts. This isn't quite how it works in the real world 100% of the time, but for the sake of this example lets assume so...
One customer can buy many concessions In the old days, customer information wouldn't necessarily be linked to individual concessions.
3
movie _company
PK company_id SERIAL
company_name VARCHAR(100)
A movie company can make many movies, but a single movie can only be made by one company.
2
customer
PK customer_id SERIAL
first_name VARCHAR(100)
last_name VARCHAR(100)
billing_info VARCHAR(150)
5
movies
PK movie_id SERIAL
movie_title VARCHAR(100)
movie_hall NUMERIC(2)
rating VARCHAR(10)
show_time TIME
FK company_id SERIAL
date DATE
One customer can buy many tickets, but a ticket can only have a single customer. In modern times, all tickets are purchesed online or with a CC, thus customer info is tracked and stored with idividual ticket sales.
6
tickets
PK ticket_id SERIAL
FK customer_id VARCHAR(100)
ticket_price NUMERIC(4,2)
payment_type VARCHAR(30)
FK movie_id SERIAL
A movie can have many tickets issued for it, but each ticket can only be valid for one vieiwing of a given move (at a specified time).