Databases 2017 - Assignment 1
Leiden

Leiden Institute of Advanced Computer Science

Introduction

This practical assignment consists of three parts and should be delivered by April 3rd 23:59. You are allowed to work as a pair on these assignments. Include all the results for your assignments in a single zip file and upload it to the Blackboard.

For the ER Modelling part, we recommend that you use software to create an ER diagram for increased usability and legibility; However, if you want to use a scanned-in document please make sure the quality of the handwriting and scan is good. For the SQL parts, you are allowed to use either default Oracle or (if you have issues with Oracle) SQLite to write your queries; However, you must clearly indicate which one you used.

1 ER Modelling - Minions

For this part of the assignment your goal is to design an ER diagram for a database of Minions; The manager of the Minions wants to keep track of the Minions, namely what food they consume, their favourite weapon, who their current leader is, what missions have been assigned to them, and if they have any pets. The following additional information is given by the manager.

- Each Minion has a name (e.g. Dave, Tim, Bob, etc.), an ID, and an age.
- Minions have to register any pet(s) they have because of allergies and house rules. When a Minion flies off to banana heaven it is unfortunately not important anymore to know about that Minion's pets.
- A pet has a name, and the type of organism (e.g. dog, cat, mouse) listed. The pet can only be uniquely identified by its name and the ID of its owner (which the pet has on a nametag around its neck).
- Minions need to eat food. All food items are identified by name and have a health rating (from 1-10), and a size.
- Some of the food that the Minions consume consists of fruit (e.g. Banana, Papple, etc.). Every fruit has a colour and at least one vitamin. Vitamins have a unique name and a known effect. All the rest of the food will be categorised as 'non-fruit'.
- The manager wants to keep track of the amount of calories of each non-fruit food.
- Each Minion has their own personal favourite weapon. A weapon has a unique name, a size, and a weight.
- The Minions are directly ruled by several non-Minion leaders. A leader has a name, an ID, a special skill, and an important goal they would like to achieve.
- Minions are loyal and serve just one leader at a time, but one leader rules over at least one minion.
- The food is bought by leaders for the Minions as payment for their services. The date of this purchase needs to be registered, but only the most recent purchase for each leader purchasing a specific type of food needs to be kept.
- Leaders create missions that need to be taken care of by one or more Minions.
- Each leader supervises the mission they created.
- A mission has a mission ID, a start date, and an end date. Each mission has a head minion (not the leader) and optionally multiple other minions. A record needs to be kept about which minion was the head minion for each undertaken mission.

Try to set up a logical ER diagram for a database that can satisfy as many of these constraints as possible and state any additional assumptions you have made. If certain constraints could not be captured in an ER diagram, explain why.

2 SQL Queries - Cinema

This assignment uses SQL tables and data contained in the file `cinema.sql`, which will run in both Oracle and SQLite. These contain data for a chain of cinemas about movies, customers and locations. The database is represented by the following schema:
Customer(cid, name, age)
Movie(mid, minage, title, director, genre)
Theater(postal code, address, city name, revenue)
SoldTickets(cid, mid, postal code, address)

Write SQL queries for the following statements. Write all of your answers into a single file called `queries.sql` that runs *without errors.*

1. Find the name and age of all customers under the age of 18.
2. Find the name and age of all customers under the age of 18 and all customers over the age of 65.
3. Find the customer cids that have watched a movie of the 'thriller' genre.
4. Find the names of all customers that have not been to watch a movie.
5. The minimum age for movies aren't always strictly upheld; Find the pairs of customer name and movie title that a customer was too young to see, but apparently watched anyway.
6. Assume all movie tickets cost the same, and all profits are made from ticket sales. Find the names of the customers, or customer, which have been the most profitable for the cinema.
7. Find the cid of all customers that have watched two or more movies.
8. Per director, find the average age of all customers that have watched at least one movie from that director.
9. The cinema chain can have multiple theaters in a single city. Find which city has the largest total revenue.
10. Find the customer cid that have only watched movies at a single theater.
11. Find the customers that have watched movies at two or more theater.
12. For each theater, find the title of the movie that has been watched the most.
13. For each theater, find out how many viewers a movie gets on average.
14. Find the theater with the second-lowest revenue.
15. Find the cids of customers that have watched all movies.

3 ER to SQL - Webshop

In Figure 1 an ER diagram is given for a model webshop. Translate this ER diagram to SQL `CREATE TABLE` statements that capture the given constraints as well as possible. Deliver your results in a file called `create.sql` together with a file called `drop.sql` that will delete (`DROP`) your created tables.

In this webshop we have two different types of customers; Customers that are registered get a username and password (stored in hash of course) and registered customers also build up points with every purchase. Customers that do not want to register can shop and checkout as guests. Guests get a session id in the database that will expire after some reasonably large (but finite) amount of time. This IS-A relationship is non-overlapping (customers are registered *or* guest, but not both) and fully covering (any customer is either guest or registered, but not neither). You are not required to enforce these constraints in your database. Multiple approaches exist to translate an IS-A hierarchy, pick the one that seems most sensible to you for this situation.

An order can be ordered by exactly one customer and can be sent to exactly one address. The status of an order (e.g. "processing", "payment pending", "shipped", etc) can be modelled here as a string of characters. When an order is placed by a customer, the order date is set. It is up to you how to store this date into your database and this will also differ per choice of database system (i.e. Oracle or SQLite).

Assume that the order will be sent to an address that belongs to the customer that ordered it.

In general, use your best judgement to figure out which datatypes correspond to which attributes and double-check to see if your work runs without error in your database of choice. And don't forget to write down any assumptions you make.

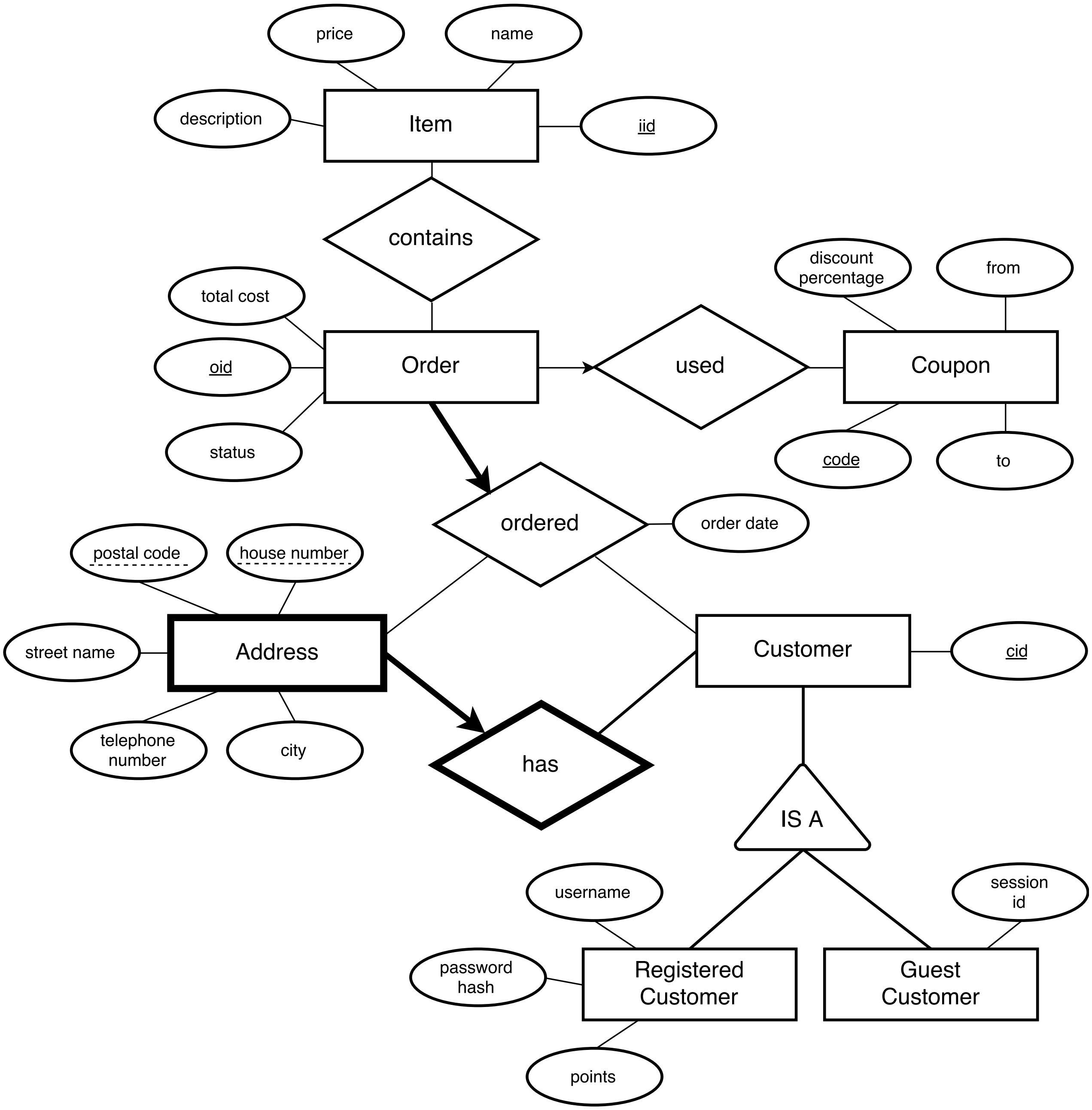

Figure 1: An ER diagram for a webshop