

## HW2: More SQL queries

For the following questions, write down all the queries that answer your questions into one Word file (DOCX), or text file (TXT), or SQL file (SQL). To issue your queries, you need access to our "big IMDB" movie database from HW1 (imdb-cs3200.zip) and have the Chinook database (319), and a small database (395) installed in PostgreSQL. All files are available in our "sql" folder under resources on our website. Create a new database in PostgreSQL for each of the above databases. We recommend using the visual interface PgAdmin for this exercise. Each question need to be answered by one single query. You will get to practice writing nested queries, outer joins, null values, using the with clause, and witnesses for this homework.

### 395

- (1) Create a **Company/Product** database by copying and then executing the commands from the text file 395. We recommend you draw a schema of the database before you try to find the query. The database is small enough for you to verify the correctness of the query results. Then write queries to answer the following questions:

The schema of the database should be familiar to you:

Company(cid, cname, city)

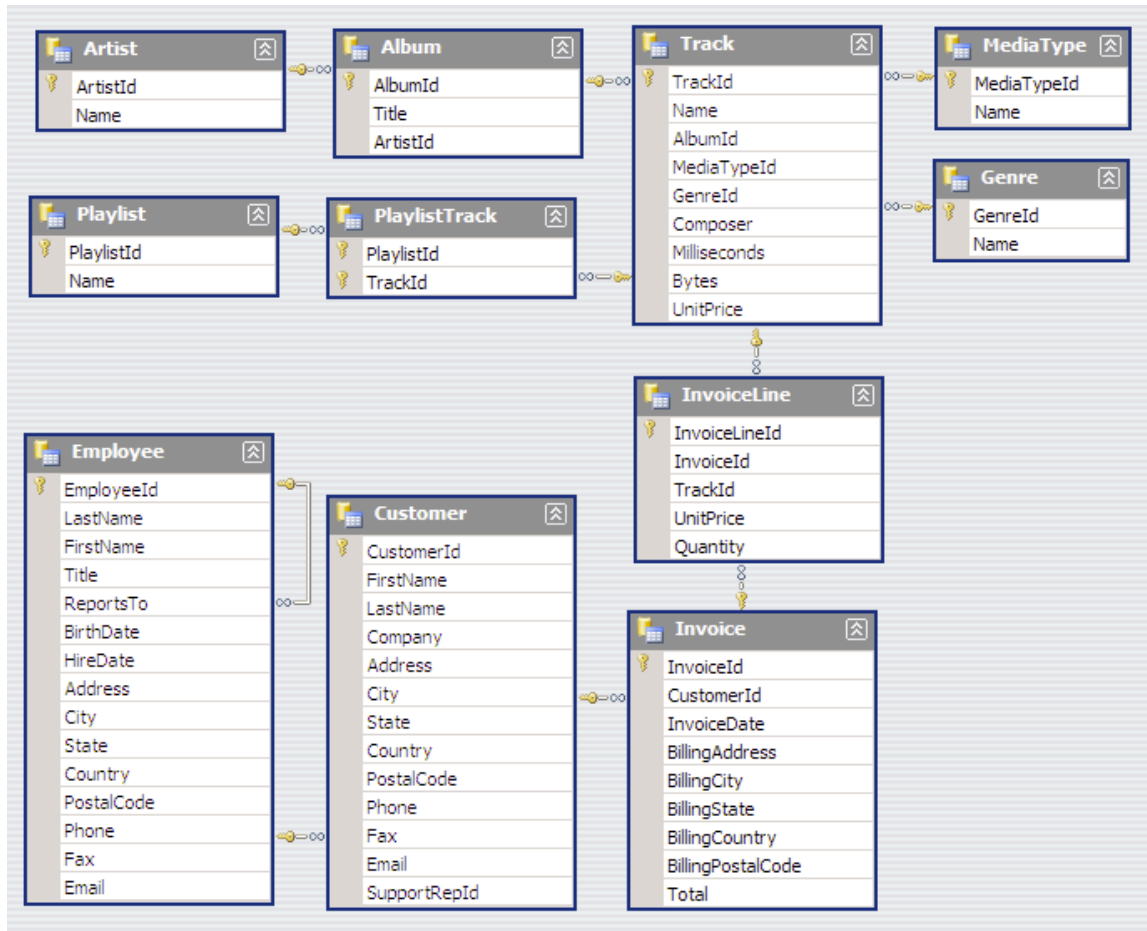
Product(pname, price, cid)

Product.cid is a foreign key to Company

- (a) For each company, find the product(s) with the highest price (return company name, price, and product name) [1pt]:
- (b) For each company, find the number of products produced (return the company name "cname", company "city" and number of products as attribute "num"). Make sure that your query returns a number for *\*all\** companies in the database. Do you notice anything interesting? Explain what you see in maximal 1-3 sentences. [1pt]

## 319

- (2) The next query needs to be written over the Chinook database. This database models a digital media store, including tables for artists, albums, media tracks, invoices and customers. The schema is shown here for you and you find more optional information here: <https://chinookdatabase.codeplex.com/>



We want to find the artists who have the most diverse customers. That means, we like to find the artist who have customers from as many different countries as possible. We write the query in two steps:

- (a) Write a query that lists for each artist the countries of customers who have bought a track from an album produced by them. Your query should return artist.name and customer.country. Make sure to write your query as to show *\*no\** duplicates. [1pt]

- (b) Next write a query that lists for each artist name the number of different countries for which there is a customer listening to their album. Only show those artists who have customers from at least 10 different countries. For your query, assume that there are artists that have the same name, but different ids (in other words, you cannot assume that names are unique). Sort your result in decreasing number of countries. For this query, I like you to use the "WITH" SQL statement. The WITH clause will become handy for part c) [1pt]
- (c) This query is more complicated and combines WITH statements with witnesses. Write a query that returns you the artist (or several artists) with the highest count of different countries with listeners to them. Your query should return artist.name and the count of countries. [1pt]

## IMDB

- (3) The next queries relate to our IMDB movie database. As stated in class, the IMDB dataset is not perfectly clean and some entries in MOVIE\_GENRE.mid refer to non-existing movies (that's why we have no PK-FK relationship). If we had specified that MOVIE\_GENRE.mid is a foreign key to MOVIE.id, how many tuples in MOVIE\_GENRE would have violated this key constraint? In your answer, include both your SQL query and the answer. [1 Pt]
- (4) Find the film(s) with the largest cast. Return the movie title and the size of the cast. By "cast size" we mean the number of distinct actors that played in that movie: if an actor played multiple roles, or if the actor is simply listed more than once in CAST, we still count her/him only once. You may not assume that only one film has the largest cast. [1 pt, tip: **cast smaller or equal than 1300**]
- (5) Actors can sometimes play more than one role in the same movie. Find the actor(s) who has (have) played the most roles in a single movie. Return first name, last name, movie names and number of different roles. [1pt]
- (6) The "Bacon number" of an actor is the length of the shortest path between the actor and Kevin Bacon in the "co-acting" graph. That is, Kevin Bacon has Bacon number 0; all actors who acted in the same film as Kevin Bacon have Bacon number 1; all actors who acted in the same film as some actor with Bacon number 1 (but not with Bacon himself) have Bacon number 2, etc. (You may see that some faculty like to list their Erdoes number on their web page: [https://en.wikipedia.org/wiki/Erd%C5%91s\\_number](https://en.wikipedia.org/wiki/Erd%C5%91s_number)). Write a query that counts how many actors have Bacon number of 1. [1 pt, 1 row. Hint: **5115 expected**]

## DELIVERABLES:

Put together one DOCX or TXT **or SQL** file that contains the answers to all questions above.