
2019 U.S. Information Technology Collegiate Conference

Database Design Competition

Do not put your name(s) or your school's name on anything that you submit.

Doing so will result in disqualification of your team.

The only identifying information you should use is your team number.

Part 1: Database Design

Draw the Entity Relationship Diagram (ERD) for the application described below. Specify the entities, the relationships between the entities, as well as the cardinality and optionality of the relationships. Clearly label all primary and foreign keys. Indicate what type of notation (crow's foot, Chen, or other) you are using to show cardinality (one or many) and optionality (mandatory or optional) at the top of your ERD. You may specify surrogate primary keys as needed to simplify the database design, which should be normalized to 3NF. ***Save your ERD as a pdf document.***

Cirque de la Lune

Cirque de la Lune is a high quality entertainment company and one of the largest theatrical producers in the world. Originally composed of a dozen street performers, it now employs many performers who bring wonder and delight to people through incredible circus performing acts. You have been hired by Cirque to create a database to manage and improve their businesses processes.

The foundation of Cirque are their amazing shows, which are carefully planned to thrill audiences with performances they've never seen before. For the database, a show will have a name, theme, the date it was established, and the date it will expire (or has expired.) Shows travel across the world, and have certain locations where they will be performed. A performance of a show is called an event. A show will have (or has had) many events at many locations. A location consists of the location name, address, phone, and capacity. A location may be the site of many events, but an event has one and only one location. It is also critical that the date of an event be recorded in the database.

The Cirque performers, who have one specialty, are divided into teams. A team has one or more performers, but a performer only belongs to one team. The team has a name and a theme. A specialty has a name, type, and usually a piece of equipment that is required. Specialties may have one or more performers able to perform that specialty. Performers will be performing at several different events of a show, and of course a show has many performers who can do the amazing acts.

Cirque also wishes to keep track of its customers' information, such as their first and last name, phone number, and email. Customers may attend many events, and of course the events have so many customers in attendance, that they are usually sold out for every event.

The owners of Cirque are optimistic that a well-designed database system will substantially improve their business processes. They are depending on you to analyze their business to identify areas that can benefit from such a system, and to design and develop the system itself.

Submit your ERD (saved as pdf) to Part 1, then you may proceed to Part 2.

Once you have access to Part 2, you will be able to download DDL scripts for Oracle, MySQL, and Microsoft SQL Server. Choose the appropriate script for your environment.

Run create_cirque in your SQL environment.

This will create the tables and insert the data into the tables.

Part 2. Queries

Save all queries (and the view) in one SQL script named **TeamNN_All_Queries**. Comment your code so that each of the queries are clearly labeled as **Query1**, **Query2**, etc. When you are done, upload your script as instructed.

For all of the following queries, place the fields in the order that they are requested, and adjust the width of the columns if necessary, so that the column headings as well as the data in the results can be fully viewed. Format all data and column headings as shown and order the records as specified.

Query1. Create a query that lists the team's name and the size of the team. Sort the records by the team size from largest to smallest.

Team	Team Size
Hydra	5
Siren	4
Mermaid	4
Chimera	2

Query2. List the names of all teams and their performers' specialties. Eliminate duplicate rows. Order by the team name, then specialty within the team.

Team	Specialty
Chimera	Trapeze
Hydra	Actor
Hydra	Martial Artist
Hydra	Trapeze
Mermaid	Juggler
Mermaid	Slackline
Mermaid	Trapeze
Siren	Instrumentalist
Siren	Trapeze

Query3. Write a query that shows the shows that have events that occur in the same state as the 'Bayfront Park Amphitheatre'. Format the location address as shown. Eliminate duplicate rows. Order by show then location.

Show	Location	Location Address
Eclipse	Universal Citywalk	6000 Universal Blvd, Orlando, Florida
Symphonia	Bayfront Park Amphitheatre	301 Biscayne Blvd, Miami, Florida
Symphonia	Universal Citywalk	6000 Universal Blvd, Orlando, Florida

Query4. This query will list the event's ID, how many customers are attending that event, the capacity of the location and the remaining capacity. Order by the event ID.

Event	Attendance	Location Capacity	Remaining Capacity
1	5	400	395
2	4	400	396
3	4	350	346
4	5	350	345
5	6	400	394
9	2	400	398

Query5. What customer(s) are not attending any events?

Non-attending Customers
Kimmie Gallaher

Query6. List all the performers who have birthdays for whatever is the current month, and how old they will be turning on their birthday. If there is more than one performer, then order by their age ascending. For example, if today's date is in March, then the result of this query is:

Performer	Birthday	Turning
Starr Berkely	March, 4 1998	22

If today's date is in April, then the result of this query is:

Performer	Birthday	Turning
Issie Norree	April, 30 1990	29
Salvador Agnolo	April, 13 1988	31
Lula Hellings	April, 15 1986	33

VIEW. Create a view, named *upcoming_events* that will show all upcoming events, that is, events that not have occurred yet. Display the name of the show and location, and its date. Order by event date. Include a statement to display all rows of the view, as shown below:

Show	Show Date
Symphonia in Atlantic Station	Friday, June 28, 2019
Eclipse in The Venetian	Tuesday, October 15, 2019
Symphonia in The Magic Castle	Thursday, October 31, 2019
Eclipse in Universal Citywalk	Monday, November 11, 2019

Part 3: Data Definition Language

DDL: Create a new script named **TeamNN_DDL** that implements the following super-class sub-class structure for employees of Cirque:

The superclass EMPLOYEE has 2 subclasses, MANAGEMENT and CREW (*name your tables CIR_EMPLOYEE_DDL, CIR_MANAGEMENT_DDL, and CIR_CREW_DDL*). An employee is either a part of the management or a crew member, i.e. they do not overlap.

All of the employees have attributes employee ID, first and last name, and date of birth. A management employee has the employee's department, along with their job title and annual salary.

For each crew member, we want to store a description of the crew member's technical skill, along with his or her hourly rate. Use your own judgement when determining the datatype of each attribute. That is use a datatype that would be reasonable for the expected data values for that attribute.

What to turn in: a script, named TeamNN_DDL that contains the code to create the 3 tables, along with at least ONE INSERT statement for each table. You should have one **management employee** and one crew. (NOTE: the printed problem statement erroneously says "performer" instead of "management employee" – please make this adjustment on your printed copy!)

Part 4. SQL KNOWLEDGE: This part of the competition tests your knowledge of SQL. Complete the Part 4 Quiz online.

TIE BREAKER – TO BE USED IN CASE OF A TIE.

Write a query to list all shows and whether they have been performed yet or not. Sort by YES then NO.

Show	Starting Date	Ending Date	Show Performed Yet
Symphonia	08-OCT-10	11-OCT-18	YES
Lune	07-OCT-16	10-OCT-19	YES
Eclipse	13-OCT-17	10-OCT-19	YES
Gravity	12-OCT-18	10-OCT-20	YES
Mana	12-OCT-14	12-OCT-18	NO
Liquid	05-MAR-18	02-MAR-23	NO
Mistic	01-MAR-19	04-MAR-21	NO
Savannah	11-OCT-19	07-OCT-21	NO
Peak	06-MAR-20	03-MAR-22	NO
Alien	09-OCT-20	06-OCT-22	NO