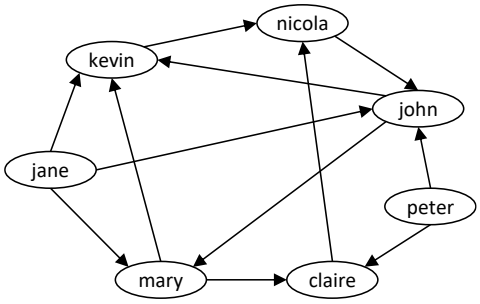
Data Structures
Degree in Computer Science, 2nd year
Exercises

Note: the exercise numbering is aligned with the corresponding exercises in the Spanish version.

1. a. Write in SQL the sentences needed to create a database to store information about an asymmetric social network (of the "follow" kind –as in Twitter) in which users have a (unique) username, name and email.
   b. Set suitable primary keys and constraints for each table.
   c. Insert the data needed to store the social network displayed in the figure (only the usernames are shown in the figure; freely define the rest of the data).



2. Starting from the tables of the previous exercise, write SQL queries to obtain the following information:
   a. Common followers of kevin and mary.
   b. Degree-2 followers of nicola.
   c. Rewrite the previous queries considering the network was symmetric (the "follows" relationship is always mutual –as in Facebook).

3. Write the SQL queries needed to store the information about commercial flights including the following data:
   - Airports with a code (three letters, unique) and city (a city may have several airports).
   - Flights with number (unique), origin and destination (airport codes), departure time, number of seats (assuming it were always the same aircraft model), and airline that operates the flight.
   - Airlines with name and IATA code (unique, e.g. IB is the code for Iberia, BA for British Airways).
   - Passengers with (national or passport) ID and name.
   - Flight tickets booked by passengers on a given date, with its price.

   Illustrate the tables with some data.

4. Write SQL update sentences that apply the following changes in the database of the previous exercise:
   a. Cancel all tickets of flights departing from Madrid.
   b. Reassign to Iberia all British Airways flights departing from Madrid.
   c. Double the price of tickets for all EasyJet flights in August 2021.

5. Write SQL queries to obtain the following information from the database of exercise 3:
   a. Flights departing from Paris.
   b. All flights between Madrid and Paris taking off before 12:00pm.
   c. Name of passengers flying from London to Paris, indicating the date.
   d. Name of passengers flying between London and Paris, in either direction.
   e. Passengers doing a round trip within a single day.

6. Write SQL queries to obtain the following information from the database of exercise 3:
   a. Airlines without flights departing from London.
   b. Sold out flights, indicating the date.
   c. Empty flights (with no sold tickets) on January 1st 2020.
   d. Airlines that only operate flights departing from or arriving in Madrid.

7. Write SQL queries to obtain the following information from the database of exercise 1:
   a. Users who do not follow jane.
   b. Users without any followers.
   c. Users everybody follows.

8. Write SQL queries to obtain the following information from the database of exercise 3:
    a. Airport with most traffic (counting departures and arrivals).
    b. Airlines ordered by the total number of tickets sold in the flights they operate.
    c. Name of the city from which the very first flight of the day departs.
    d. Average total spending by passenger.
    e. Total billing by airline and departure airport.

9. Given the state of the followind database of a company:

PROYECTO

NombreProyecto	NumProyecto	UbicacionProyecto	NumDptoProyecto
ProductoX	1	Valencia	5
ProductoY	2	Sevilla	5
ProductoZ	3	Madrid	5
Computación	10	Gijón	4
Reorganización	20	Madrid	1
Comunicaciones	30	Gijón	4

SUBORDINADO

DniEmpleado	NombSubordinado	Sexo	FechaNac	Relación
333445555	Alicia	M	05-04-1986	Hija
333445555	Teodoro	H	25-10-1983	Hijo
333445555	Luisa	M	03-05-1958	Esposa
987654321	Alfonso	H	28-02-1942	Esposo
123456789	Miguel	H	04-01-1988	Hijo
123456789	Alicia	M	30-12-1988	Hija
123456789	Elisa	M	05-05-1967	Esposa

TRABAJA_EN

DniEmpleado	NumProy	Horas
123456789	1	32,5
123456789	2	7,5
666884444	3	40,0
453453453	1	20,0
453453453	2	20,0
333445555	2	10,0
333445555	3	10,0
333445555	10	10,0
333445555	20	10,0
999887777	30	30,0
999887777	10	10,0
987987987	10	35,0
987987987	30	5,0
987654321	30	20,0
987654321	20	15,0
888665555	20	NULL

EMPLEADO

Nombre	Apellido1	Apellido2	Dni	FechaNac	Dirección	Sexo	Sueldo	SuperDni	Dno
José	Pérez	Pérez	123456789	01-09-1965	Eloy I, 98	H	30000	333445555	5
Alberto	Campos	Sastre	333445555	08-12-1955	Avda. Ríos, 9	H	40000	888665555	5
Alicia	Jiménez	Celaya	999887777	12-05-1968	Gran Vía, 38	M	25000	987654321	4
Juana	Sainz	Oreja	987654321	20-06-1941	Cerquillas, 67	M	43000	888665555	4
Fernando	Ojeda	Ordóñez	666884444	15-09-1962	Portillo, s/n	H	38000	333445555	5
Aurora	Oliva	Avezuela	453453453	31-07-1972	Antón, 6	M	25000	333445555	5
Luis	Pajares	Morera	987987987	29-03-1969	Enebros, 90	H	25000	987654321	4
Eduardo	Ochoa	Paredes	888665555	10-11-1937	Las Peñas, 1	H	55000	NULL	1

DEPARTAMENTO

NombreDpto	NumeroDpto	DniDirector	FechaIngresoDirector
Investigación	5	333445555	22-05-1988
Administración	4	987654321	01-01-1995
Sede Central	1	888665555	19-06-1981

LOCALIZACIONES_DPTO

NumeroDpto	UbicacionDpto
1	Madrid
4	Gijón
5	Valencia
5	Sevilla
5	Madrid

Show the result of the following SQL sentences and, when applicable, the state of the modified tables (assume that the SQL commands always apply on the original DB):

    a. DELETE FROM EMPLEADO WHERE dni='123456789';
    b. DELETE FROM EMPLEADO WHERE Apellido1='Cabrera';
    c. ALTER TABLE EMPLEADO ADD COLUMN Trabajo VARCHAR (12);
    d. ALTER TABLE EMPLEADO DROP COLUMN dirección CASCADE;
    e. INSERT INTO EMPLEADO VALUES ('Ricardo', 'Roca', 'Flores', '653298653', '1962-12-30', 'Los Jarales, 47', 'H', 37000, '653298653', 4);
    f. INSERT INTO EMPLEADO (Nombre, Apellido1, Dno, Dni) VALUES ('Ricardo', 'Roca', 4, '653298653');
    g. UPDATE PROYECTO SET UbicaciónProyecto='Valencia', NumDptoProyecto = 5
       WHERE NumProyecto=10;
    h. SELECT FechaNac, Dirección FROM EMPLEADO
       WHERE Nombre='José' AND Apellido1='Pérez' AND Apellido2='Pérez';
    i. SELECT Nombre, Apellido1, Dirección FROM EMPLEADO, DEPARTAMENTO
       WHERE NombreDpto='Investigación' AND NumeroDpto=Dno;
    j. SELECT NumProyecto, NumDptoProyecto, Apellido1, Dirección, FechaNac

FROM PROYECTO, DEPARTAMENTO, EMPLEADO
WHERE NumDptoProyecto=NumeroDpto AND DniDirector=Dni AND UbicacionProyecto='Gijon';

k. SELECT Nombre, Apellido1, Dirección
FROM (EMPLEADO JOIN DEPARTAMENTO ON Dno=NumeroDpto)
WHERE NombreDpto='Investigación';

10. Design an ER diagram that describes in detail the entities, attributes, relationships and keys to store the following information in a relational database:
    a. Movies with title, nationality, genre, cast, director, writer, release date.
    b. Actors, with name, gender, and movies they have played in (indicating the role).
    c. Directors, with name and directed movies.
    d. Writers, with name and movies the script of which they have written.

    Indicate in detail the cardinality constraints, as well as the keys that the schemas may contain.
    To simplify, consider initially:
    - That two movies cannot have the same title, and two people cannot have the same name.
    - That no one can be an actor and director or writer at the same time, but can only one have of the three jobs.

11. Refine the design of the previous exercise with the following elaborations:
    a. Removing the last constraint, that is, considering that a movie artist can work as an actor, director and/or writer at different times in their career (or even simultaneously).
    b. Adding information about Academy awards, showing who or what earned the award (actor, movie, director, etc.), the category (best picture, screenplay, leading actor, etc.), the year, and any additional information of interest (e.g. the movie and role played by the best actor, the movie directed by the best director, etc.).

12. Convert the models defined in the previous two exercises into tables. Include a few example records in each table.

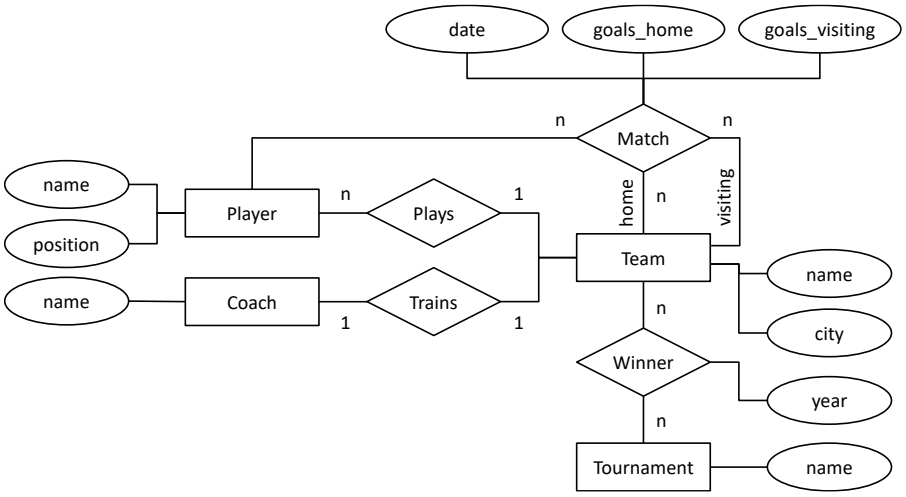
13. Describe the structures of the previous exercise in the relational model.
    a. Describe in detail the schemas and elements of the model in set notation.
    b. Implement the schemas in SQL.

14. A bookstore chain needs an integral business management application supporting the management of the inventory, stock, sales and human resources. Design an ER model for a database supporting such an application, dealing with the following information with the indicated data:
    a. Books for sale: title, author(s), publisher, ISBN and price.
    b. Book publishers: name, VAT number and contact phone numbers.
    c. Book authors: name (we assume no two authors have the same name, or the bookstore does not care).
    d. Bookstore employees: name, VAT number, salary and physical store where the employee is assigned.
    e. Physical stores of the chain: postal address, manager (one of the employees takes this role), and no. of units of each book available in the store.
    f. Sales: book, selling date, store where sold, and employee who carried out the sale.

    Define such details as cardinality, total or partial participation in relationships, primary keys, weak entities if any, and do not include any more data than are strictly indicated above.

15. Convert the structures of the previous exercise to relational model schemas. Then convert to SQL.

16. The following ER model describes database structures to store information about teams, players and results of a football league season. To simplify, we shall assume that players and coaches do not switch between teams during the course of a season; and that two players or two coaches cannot have the same name.



Turn this model into relational schemas, indicating which would be the primary and foreign keys.

17. Write in SQL the structures of the previous exercise.

18. Design an ER model for the structures of exercise 1.

19. Design an ER model for the structures of exercise 3.

20. Design an ER model for the tables of exercise 9.