Homework 2: SQL Exercises

1. Given the following relational schema (primary keys are underlined, optional attributes are denoted as '*'):

   EMPLOYEE (SSN, Name, Surname, BirthDate, Nationality)
   THEME_PARK (TPCode, ParkName, Address, City, Country)
   CAROUSEL (CCode, TPCode, CarouselName, CarouselType)
   EMPLOYEE_WORK (SSN, Date, CCode, TPCode)
   TICKETS (Date, CCode, TPCode, NumTickets)

Write the following queries in SQL language:

a) Find name and surname of each employee born after 1982 who has never worked in theme parks located in France.

b) Find surname and birth date of each Italian employee who has worked in at least 3 carousels of type aquatic in May 2022.

c) For each theme park that has sold more than 10000 tickets for roller coaster-type carousels, find the name and the city of the theme park and the total number of employees who has worked in the park.

2. Given the following relational schema (primary keys are underlined, optional attributes are denoted as '*'):

   TRAINER (SSN, TName, TSurname, TCity)
   GYM (GCode, GName, GCity, Address)
   SPECIALTY (SCode, SName, Description)
   LESSON (SSN, GCode, Date, SCode, ParticipantsNumber)

Write the following queries in SQL language:

a) Show SSN, name and surname of every personal trainer who gave lessons in at least 3 different gyms located in Turin.

b) Show SSN and surname of every personal trainer who gave at least 10 lessons on Karate, but who never gave lessons on Judo

c) For each gym, show the name and the total number of lessons given by personal trainers who gave lessons in at least 3 different gyms located in Turin.

SOLUTIONS

1. Given the following relational schema (primary keys are underlined, optional attributes are denoted as '*'):

EMPLOYEE (SSN, Name, Surname, BirthDate, Nationality)
THEME_PARK (TPCode, ParkName, Address, City, Country)
CAROUSEL (CCode, TPCode, CarouselName, CarouselType)
EMPLOYEE_WORK (SSN, Date, CCode, TPCode)
TICKETS (Date, CCode, TPCode, NumTickets)

Write the following queries in SQL language:

a) Find name and surname of each employee born after 1982 who has never worked in theme parks located in France.

```
SELECT Name, Surname
FROM EMPLOYEE
WHERE BirthDate >= '1983-01-01'
AND SSN NOT IN (SELECT SSN
                FROM EMPLOYEE_WORK EW, THEME_PARK T
                WHERE EW.TPCode=T.TPCode
                AND Country='France')
```

b) Find surname and birth date of each Italian employee who has worked in at least 3 carousels of type aquatic in May 2022.

```
SELECT Surname, BirthDate
FROM EMPLOYEE
WHERE Nationality = 'Italian'
AND SSN IN (SELECT SSN
            FROM EMPLOYEE_WORK EW, CAROUSEL C
            WHERE EW.CCode=C.CCode AND EW.TPCode=C.TPCode
            AND Date>='2022-05-01' AND Date <='2022-05-31'
            AND CarouselType='aquatic'
            GROUP BY SSN
            HAVING COUNT(DISTINCT C.CCode, C.TPCode) >= 3)
```

c) For each theme park that has sold more than 10000 tickets for roller coaster-type carousels, find the name and the city of the theme park and the total number of employees who has worked in the park.

```
SELECT ParkName, City, COUNT(DISTINCT SSN)
FROM THEME_PARK T, EMPLOYEE_WORK EW
WHERE T.TPCode=EW.TPCode
AND TPCode IN (SELECT T.TPCode
        FROM TICKETS T, CAROUSEL C
        WHERE T.CCode=C.CCode AND T.TPCode=C.TPCode
        AND CarouselType='roller coaster'
        GROUP BY T.TPCode
        HAVING SUM(NumTickets) >= 10000)
GROUP BY T.TPCode, ParkName, City
```

2. Given the following relational schema (primary keys are underlined, optional attributes are denoted as '*'):

TRAINER (SSN, TName, TSurname, TCity)
GYM (GCode, GName, GCity, Address)
SPECIALTY (SCode, SName, Description)
LESSON (SSN, GCode, Date, SCode, ParticipantsNumber)

Write the following queries in SQL language:

a) Show SSN, name and surname of every personal trainer who gave lessons in at least 3 different gyms located in Turin.

```
SELECT T.SSN, T.TName, T.TSurname
FROM TRAINER T, GYM G, LESSON L
WHERE G.City = 'Torino' AND G.GCode=L.GCode AND T.SSN = L.SSN
GROUP BY T.SSN, T.TName, T.TSurname
HAVING COUNT (DISTINCT L.GCode) >=3
```

Alternative solution:

```
SELECT T.SSN, T.Name, T.Surname
FROM TRAINER T
WHERE T.SSN IN
        (SELECT L.SSN
         FROM LESSON L, GYM G
         WHERE G.GCity = 'Torino' AND L.GCode = G.GCode
         GROUP BY L.SSN
         HAVING COUNT (DISTINCT L.GCode) >=3)
```

b) Show SSN and surname of every personal trainer who gave at least 10 lessons on Karate, but who never gave lessons on Judo

```
SELECT T.SSN, TSurname
FROM TRAINER T, SPECIALITY S, LESSON L
WHERE S.SName = 'Karate' AND S.SCode = L.SCode AND T.SSN = L.SSN
AND T.SSN NOT IN
                (SELECT L.SSN
                FROM SPECIALITY S, LESSON L
                WHERE S.SName = 'Judo' AND S.SCode=L.SCode)
GROUP BY T.SSN, T.TSurname
HAVING COUNT (*) >=10
```

Alternative solution:

```
SELECT SSN, TSurname
FROM TRAINER
WHERE SSN IN
                (SELECT L.SSN
                FROM SPECIALITY S, LESSON L
                WHERE S.SName = 'Karate' AND S.SCode=L.SCode
                GROUP BY L.SSN
                HAVING COUNT (*) >=10)
AND SSN NOT IN
                (SELECT L.SSN
                FROM SPECIALITY S, LESSON L
                WHERE S.SName = 'Judo' AND S.SCode=L.SCode)
```

c) For each gym, show the name and the total number of lessons given by personal trainers who gave lessons in at least 3 different gyms located in Turin.

```
SELECT GName, COUNT(*)
FROM GYM G, LESSON L
WHERE G.GCode=L.GCode
AND SSN IN (SELECT SSN
                FROM GYM G, LESSON L
                WHERE G.City = 'Torino' AND G.GCode=L.GCode
                GROUP BY SSN
                HAVING COUNT (DISTINCT L.GCode) >=3)
GROUP BY G.GCode, GName
```