

## SW N°2: SQL CORRECTION

**Exercise 1:** Let consider the following relationships:

STATION (StationName, capacity, location, region, price)

ACTIVITY (StationName, label, price)

CUSTOMER (id, lastName, firstName, city, region, balance)

STAY (ClientID, StationName, Start, nbPlaces)

Examples of occurrences:

StationName	Capacity	Location	Region	Price
Venusa	350	Guadeloupe	Antilles	1200
Farniante	200	Seychelles	Océan Indien	1500
Santalba	150	Martinique	Antilles	2000
Passac	400	alpes	Europe	1000

*Station table*

StationName	Label	Price
Vertusa	voile	150
Venusa	Plongée	120
Farniente	Plongée	130
Passac	Ski	200
Passac	Piscine	20
santalba	Kayac	50

*Activity table*

Id	lastName	firstName	city	Region	balance
10	Fogg	Phileas	Londres	Europe	12465
20	Pascal	Blaise	Paris	Europe	6763
30	Kerouac	Jack	New York	Amérique	9812

*Client table*

IdClient	Station	Start	nbPlaces
10	Passac	1998-07-01	2
30	Santalba	1996-08-14	5
20	Santalba	1998-08-03	4
30	Passac	1998-08-15	3
30	Venusa	1998-08-03	3
20	Venusa	1998-08-03	6
30	Farniente	1999-06-24	5
10	Farniente	1998-09-05	3

*Stay table*

Express the following queries in SQL:

1. Extract from the database the name of all the stations located in the Antilles

```
SELECT StationName
FROM Station
WHERE Region = 'Antilles'
```

2. Display the name of customers living in Paris, the resorts where they stayed with the date, and finally the weekly rate for each resort.

```
SELECT lastName, Client, Start, Price
FROM Station, Stay, Client
WHERE city = 'Paris'
AND Client.id = Stay.idClient
AND Station.StationName = Stay.Station
```

3. Give the pairs of stations located in the same region.

```
SELECT s1.StationName, s2.StationName
FROM Station s1, Station s2
WHERE s1.Region = s2.Region
```

4. Give all region names in the table.

```
SELECT Region FROM Station
UNION
SELECT Region FROM Client
```

5. Give the regions where both customers and stations can be found.

```
SELECT region FROM Station
INTERSECT
SELECT region FROM Client
```

6. Which regions have stations but no customers?

```
SELECT region FROM Station
EXCEPT
SELECT region FROM Client
```

7. Names of resorts where Parisian customers have stayed

```
SELECT StationName
FROM Stay, Client
WHERE Client.id = Stay.idClient
AND city = 'Paris'
```

```
SELECT StationName
FROM Stay
WHERE idClient IN ( SELECT id
                    FROM Client
                    WHERE city = 'Paris')
```

8. Where (resort, location) you cannot go skiing?

```
SELECT StationName, Location
FROM Station
WHERE StationName NOT IN ( SELECT StationName
                           FROM Activity
                           WHERE Label = 'ski')
```

9. In which resort do you practice an activity at the same price as in Santalba?

```
SELECT StationName, Label
FROM Activity
WHERE price IN ( SELECT price
                 FROM Activity
                 WHERE StationName='Santalba')
```

10. What are the resorts where diving and sailing are practiced?

```
SELECT a1.StationName
FROM Activity a1, Activity a2
WHERE a1.Label = 'Plongée' AND a2.Label = 'voile'
AND a1.StationName = a2.StationName
```

11. Which stations do not offer sailing?

```
SELECT StationName FROM Station
EXCEPT
SELECT StationName FROM Activity WHERE Label = 'Voile'
```

12. Show the names of guests who have stayed at all resorts in their area.

```
SELECT C.lastName
FROM Station AS S, Stay, Client AS C
WHERE city = 'Paris'
AND C.id = Stay.idClient
AND S.StationName = Stay.Station
GROUP BY C.id
HAVING COUNT(DISTINCT StationName) = ( SELECT COUNT(StationName)
                                       FROM Station
                                       WHERE Region = C.Region)
```

13. Give the names of clients who only stay in the Antilles.

```
SELECT lastName
FROM Client
WHERE id IN (SELECT id FROM Client
            EXCEPT
            SELECT idClient FROM Stay
            WHERE StationName IN ( SELECT StationName
                                   FROM Client
                                   WHERE Region = 'Antilles'))
```

)

14. Give the names of the activities which are only practiced in all stations in Europe.

```
SELECT Label
FROM Activity
GROUP BY Label
HAVING COUNT(DISTINCT StationName) = ( SELECT COUNT(StationName)
                                        FROM Station
                                        WHERE Region = 'Europe')
```

15. Which station charges the highest price?

```
SELECT StationName
FROM Station
where Price = (SELECT MAX(Price) FROM Station)
```

16. Show regions with number of stations.

```
SELECT region, COUNT(StationName)
FROM Station
GROUP BY region
```

17. We would like to consult the number of places reserved, per customer.

```
SELECT lastName, SUM(nbPlaces)
FROM Client, Stay
WHERE id = idClient
GROUP BY id, lastName
```

18. We would like to consult the number of places reserved, per customer, for customers who have reserved more than 10 places.

```
SELECT lastName, SUM(nbPlaces)
FROM Client, Stay
WHERE id = idClient
GROUP BY lastName
HAVING SUM(nbPlaces)>=10
```

## **Exercise 2 :**

We consider the schema of the following database:

FILM (numf, title, genre, year, duration, budget, director, real salary)

DISTRIBUTION (#numf, #numa, role, salary)

PERSON (nump, first name, last name, datenaiss)

ACTOR (numa, agent, specialty, height, weight)

The DIRECTOR attribute of the FILM relationship is the identifier of a PERSON. It is the same for the NUMA and AGENT attributes of the ACTOR relationship.

Give the SQL queries to answer the following questions:

1. Find the list of all films

```
SELECT *  
FROM FILM
```

2. Find the list of films whose length exceeds 180 min.

```
SELECT *  
FROM FILM  
WHERE duration >180
```

3. List all film genres.

```
SELECT DISTINCT genre  
FROM FILM
```

4. Give the number of films by genre.

```
SELECT genre, COUNT(*)  
FROM FILM  
GROUP BY genre
```

5. Find the title(s) and year(s) of the longest film(s).

```
SELECT title, year  
FROM FILM  
WHERE duration = ( SELECT MAX(duration)  
FROM FILM)
```

6. Give the first and last name of the directors who have starred in at least one of their own films

```
SELECT DISTINCT P.first_name, P.last_name  
FROM PERSON P, FILM F, DISTRIBUTION D  
WHERE P.nump = F.director
```

```
AND F.numf = D.numf
AND D.numa = F.director
```

```
SELECT DISTINCT first_name, last_name
FROM PERSON
WHERE P.nump IN (SELECT director
                 FROM FILM
                 WHERE (director, numf) IN ( SELECT numa, numf
                                           FROM DISTRIBUTION)
                )
```

7. What is the total salary of the actors in the film “JOKER”.

```
SELECT SUM(D.salary)
FROM FILM F, DISTRIBUTION D
WHERE F.numf = D.numf
AND F.title = 'JOKER'
```

8. For each “Steven Spielberg” film (title, year), give the total salaries of the actors

```
SELECT F.title, F.year, SUM(D.salary)
FROM FILM F, DISTRIBUTION D, PERSON P
WHERE F.numf = D.numf
AND F.director = P.nump
AND P.last_name = 'Spielberg'
GROUP BY F.title, F.year
```