"BASES DE DATOS Y SISTEMAS DE INFORMACIÓN" EXAM 23/01/2018

UNITS 2 AND 3

Consider the following relational schema about the Milky Way:

Planet(*pla_name*: text, *year*: integer, *gravity*: real, *info*: text)

PK: {pla_name}

NNV: {year}

Astronomer(*as_name*: text, *adate*: date, *university*: text)

PK: {as_name}

NNV: {adate}

Discover(*pla_name*: text, *as_name*: text)

PK: {pla_name, as_name}

FK: {pla_name} → Planet f(pla_name) = pla_name

FK: {as_name} → Astronomer f(as_name) = as_name

Satellite(*sat_code*: text, *name*: text, *namedby*: text, *year*: integer, *planet*: text)

PK: {sat_code}

NNV: {name}

FK: {namedby} → Astronomer(as_name)

FK: {planet} → Planet(pla_name)

NNV: {planet}

Where the meaning of the relations is the following:

- **PLANET:**
  - *pla_name*: Name of the planet
  - *year*: Year in which the planet was discovered
  - *gravity*: Gravity of the planet
  - *info:* Extra information about the planet
- **ASTRONOMER**:
  - *as_name*: Name of the astronomer
  - *adate:* Birthdate of the astronomer
  - *university* where the astronomer is working
- **DISCOVER**:
  - The astronomer named *as_name* discovered the planet called *pla_name*
- **SATELLITE**:
  - *sat_code:* Satellite code
  - *name:* Satellite name
  - *namedby*: Name of the astronomer who named the satellite
  - *year*: Discovery year of the satellite
  - *planet*: The satellite orbits the planet called *planet*

1) Describe briefly the logical and physical independence in the ANSI/SPARC Architecture. (0.5 points)

Logical independence between the logical schema and the external schemas:
The external schemas and the application programs cannot be affected by the modifications in the logical schema of data which are not used by these programs

Physical independence between the internal schema and the logical schema:
The logical schema cannot be affected by changes in the internal schema which refer to the implementation of the data structures, access modes, page size, search path, etc.

2) Write the following queries in SQL:

a) List the code and name of the satellites with no discovery year that are orbiting a planet with a gravity greater than 20. (0,5 points)

```
SELECT sat_code, name
FROM Satellite S
WHERE year IS NULL AND Planet IN (SELECT pla_name FROM Planet
                                  WHERE gravity >20);
```

b) List the code and name of the satellite name by an astronomer working in the 'UPV' university, which has been discovery most recently. (0,75 points)

```
SELECT S.sat_code, S.name
FROM Satellite S
WHERE S.namedby IN (SELECT A.as_name FROM Astronomer A WHERE A.university='UPV')
      AND
      S.year = (SELECT MAX(S1.year)
                FROM Satellite S1
                WHERE S1.namedby IN (SELECT A1.as_name FROM Astronomer A1
                                     WHERE A1.university='UPV')
```

c) List for all the planets in the database, that have been discovered by more than two astronomers, the planet name, its gravity, and amount of satellites orbiting the planet. (0,75 points)

```
SELECT P.pla_name, P.gravity, COUNT(S.sat_code)
FROM Planet P LEFT JOIN Satellite S ON P.pla_name=S.planet
GROUP BY P.pla_name, P.gravity
HAVING P.pla_name IN ( SELECT D.num_pla
                       FROM Discover D
                       GROUP BY D.num_pla
                       HAVING COUNT(*) > 2)
```

d) List the name and university of the astronomers who have only discovered planets with no satellites. (0,75 points)

```
SELECT A.as_name, A.university
FROM Astronomer A
WHERE A.as_name IN (SELECT D.as_name FROM Discover D)
   AND  A.as_name NOT IN (SELECT D1.as_name FROM Discover D1
                                WHERE D1.pla_name IN (SELECT S.Planet
                                                            FROM Satellite S)))
```

e) List the name and university of the astronomer who has named at least two satellites, and has discovered most planets with a gravity lower then 15. (0,75 points)

```
SELECT A.as_name, A.university
FROM Astronomer A, Discover D, Planet P
WHERE A.as_name=D.as_name AND D.pla_name = P.pla_name
            AND P.gravity < 15 AND (SELECT COUNT(*)
                                          FROM Satellite S
                                          WHERE A.as_name=S.namedby) >2
GROUP BY  A.as_name, A.university
HAVING COUNT(*) = (SELECT MAX(COUNT(*)
                           FROM Discover D1,  Planet P1
                           WHERE P1.pla_name = P1.pla_name AND P.gravity < 15
                           GROUP BY D1.as_name);
```