Domanda 1

Risposta non data

Punteggio max.: 3,00

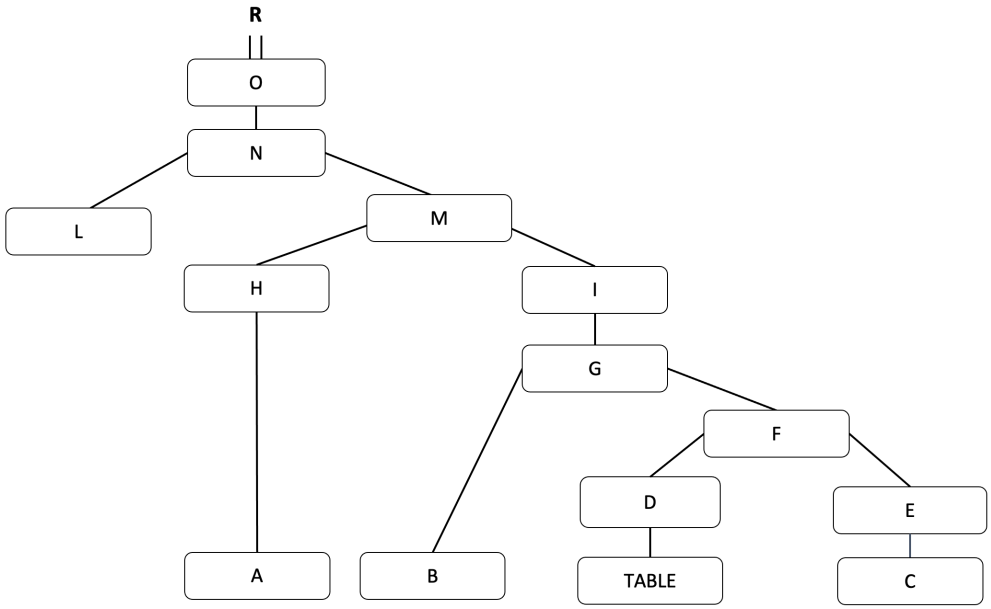
Given the following schema: (primary keys are underlined, optional attributes are indicated by '*'):

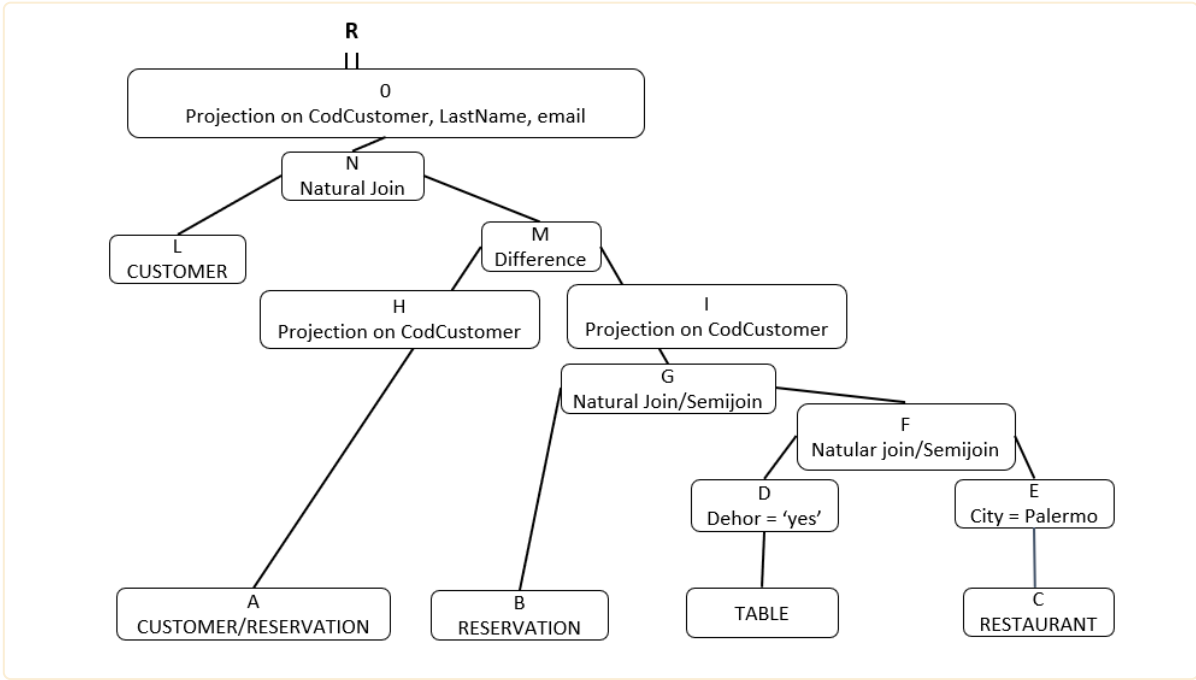
CUSTOMER (CodCustomer, FirstName, LastName, email)
RESTAURANT (CodR, Name, Address, City)
TABLE (CodR, NumTable, #Places, Dehor)
RESERVATION (CodCustomer, Date, CodR, NumTable)

Display code, last name, and email of customers who have never reserved a table in the dehor at a restaurant in Palermo.

Assignment for the exercise:

The following query tree graphically represents the requested algebraic query. You are requested to indicate, for each box in the query tree (i.e., A, B, C, D, E, F, G, H, I, L, M, N, O box), the relational table or the corresponding algebraic operator. Use the text box below to provide your solution. Note: each box in the query tree is associated with only one relational table or one algebraic operator.



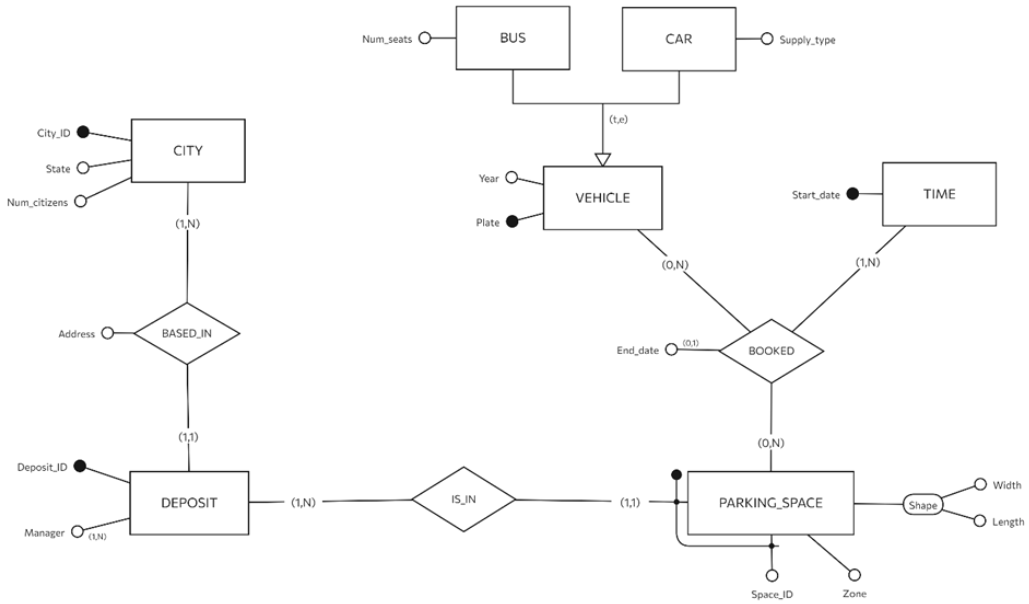

R
0
Projection on CodCustomer, LastName, email
N
Natural Join
L
CUSTOMER
M
Difference
H
Projection on CodCustomer
I
Projection on CodCustomer
G
Natural Join/Semijoin
F
Natular join/Semijoin
D
Dehor = 'yes'
E
City = Palermo
A
CUSTOMER/RESERVATION
B
RESERVATION
TABLE
C
RESTAURANT

Domanda 2

Risposta non data

Punteggio max.: 3,00

Given the following Entity-Relationship diagram



Note: Assume that each manager may be responsible for more than one deposit

You are required to:

1. Construct a normalized relational logical schema for the same database (N.B. It is not necessary to report the restructuring steps of the E-R diagram)
2. Define the referential integrity constraints for 2 relationships to be chosen from those defined in the conceptual scheme.

Indications for the conduct of the exercise

Use the text box below to report the solution.

Normalized relational schema:

CITY (<u>City_ID</u>, State, Num_citizens)
DEPOSIT (<u>Deposit_ID</u>, City_ID, Address)
VEHICLE (<u>Plate</u>, Year, Num_seats*, Supply_type*, Vehicletype)
PARKING_SPACE (<u>Space_ID</u>, <u>Deposit_ID</u>, Zone, Length, Width)
BOOKED (<u>Plate</u>, <u>Start_date</u>, <u>Deposit_ID</u>, <u>Space_ID</u>, End_date*)
MANAGER (<u>Manager</u>, <u>Deposit_ID</u>)
TIME (<u>Start_date</u>) *Note: This can be omitted*

Referential integrity constraints:

PARKING_SPACE (Deposit_ID) REFERENCES DEPOSIT(Deposit_ID)
BOOKED(Plate) REFERENCES VEHICLE(Plate)
BOOKED(Space_ID, Deposit_ID) REFERENCES PARKING_SPACE(Space_ID, Deposit_ID)
BOOKED(Start_date) REFERENCES TIME(Start_date)
Manager(Deposit_ID) REFERENCES DEPOSITO(Deposit_ID)
CITY(City_ID) REFERENCES DEPOSIT(City_ID)

Domanda 3

Risposta non data

Punteggio max.: 3,00

Indications for solving the exercise

Use the text box below to report the ER diagram in textual form. Alternatively, the drawing box can be used to graphically represent the ER diagram.

In case you use the textual form

For each entity indicate the name, the identifier of the entity (internal or external), list of attributes, presence of any composite attributes, the cardinality of the attributes if different from (1,1), and any membership in hierarchies. For each hierarchy indicate the relative classifications.

For each relationship indicate the name, the entities involved with the relative minimum and maximum cardinality, and any attributes of the relationship.

For example, use the following notation:

entity LESSON
internal id: CodLezione
external id: COURSE identifier
attributes: day

entity LOCAL
id: LocalCode
child entities, hierarchy (t, e)
entity CLASSROOM
attributes: capacityMax, floor

entity LABORATORY
attributes: numPC, devices (1, N)

relationship IN
LESSON (1, 1)
LOCAL (0, N)
attributes: start time, end time (0,1)

Describe the Entity-Relationship diagram addressing the following specifications.

- We want to create a database for the consultation of historical books in a museum by some scholars. To consult the books each person must register at the museum and acquire a card. Persons are identified by the identification code of the card; you also know the name, surname, telephone number, e-mail address, and the address of the Facebook page (if available).
- The books available at the museum are characterized by the ISBN unique code, the title, and the list of co-authors of the book. For each book, the list of titles of the chapters of the book is also known. If the book is part of a series, you also know the name of the series and the number that characterizes the book within the series. Each author is identified by a unique code and characterized by the name, surname, and a possible pseudonym.
- Each consultation of a book is characterized by the person who made it, the date on which it was made, the consulted book, and the state of wear of the book when withdrawn by the person for consultation. Consider that the same book can be consulted by at most one person each day. Consider also that the same person can consult several different books on the same day and can consult the same book several times but on different dates.

Entity PERSON
Id: #Card
Attributes: Name, Surname, Phone, Email, Facebook_page (0.1)

Entity BOOK
Id: ISBN
Attributes: Title, Chapter (1,N)

Child Entity, Hierarchy (p,e)
Entity SERIES_BOOK
Attributes: SeriesName, BookNumber

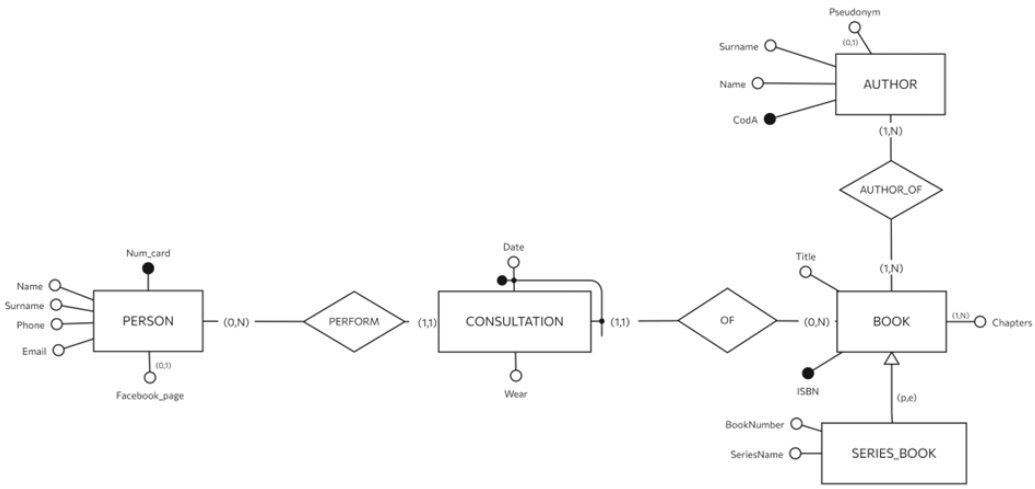
Entity AUTHOR
Id: CodA
Attributes: Name, Surname, Pseudonym (0.1)

Entity CONSULTATION
Internal Id: Date
External Id: from PERSON
Attributes: Wear

Relationship PERFORM
PERSON (1,N)
CONSULTATION (1.1)

Relationship OF
CONSULTATION (1.1)
BOOK (0,N)

Relationship AUTHOR_OF
BOOK (1,N)
AUTHOR (1,N)



Domanda 4

Risposta non data

Punteggio max.: 5,00

Given the following relational schema (primary keys are underlined, optional attributes are indicated with '*'):

WORKSHOP (WID, Name, Address, City)
VEHICLE (LicensePlate#, Model, Maker, Category, FuellingType, RegistrationYear, SSN)
CUSTOMER (SSN, FirstName, LastName, BirthDate, Address, City)
INSPECTION (LicensePlate#, WID, Date, Cost)

Write the following query in the SQL language:

For workshops that have performed inspections on at least 200 different vehicles registered by persons born between 1970 and 1980, display the name and address of the workshop that performed the most inspections (considering all the inspections performed) among workshops located in the same city. Also display the total cost of the inspections performed by the workshop and the number of different models of vehicle inspected.

Assignment for the exercise

Use the text box below to provide your solution.

```
SELECT (W.WID), Name, Address, City, SUM(Cost), COUNT(DISTINCT
 Model)
FROM WORKSHOP W, INSPECTION I, VEHICLE V
WHERE W.WID = I.WID AND I.LicensePlate# = V.LicensePlate#
AND WID IN (SELECT WID
                FROM INSPECTION I, VEHICLE V, CUSTOMER C
                WHERE I.LicensePlate# = V.LicensePlate# AND V.SSN = C.
SSN
                AND BirthDate >= 01/01/1970 AND BirthDate <= 31/12/1980
                GROUP BY WID
                HAVING COUNT(DISTINCT LicensePlate#) >= 200)
GROUP BY W.WID, Name, Address, W.City
HAVING COUNT(*) =
                  (SELECT MAX (Ninsp)
                   FROM (SELECT City, COUNT(*) as Ninsp
                         FROM INSPECTION I3, WORKSHOP W3
                         WHERE W3.WID = I3.WID
                         GROUP BY CitY, WID) AS citY_insp
                  WHERE city_inspe.City = W.City)
```

Domanda 5

Risposta non data

Punteggio max.: 4,00

The following schema is given (primary keys are underlined):

FUNDRAISING (CodF, Name, Description)
FUNDRAISING_DONATION (CodD, DonationDate, CodF, Amount, Comment)
DONATIONS_SUMMARY (CodF, NumDonations, TotalAmount)
DONATIONS_OUTCOME_NOTIFICATION (CodF, DonationDate, AvgAmount)

Write the trigger to manage cash donations for different fundraisers managed through a web portal.

The FUNDRAISING table contains the list of fundraisers for which you can make a donation. The table DONATIONS_SUMMARY contains, for each fundraiser, the *total number* of cash donations received and the *corresponding total amount* of cash. Consider that a fundraiser is in the table DONATIONS_SUMMARY only if at least one donation was made to that fundraiser.

A new donation for a fundraiser is made through the portal (insertion of a record in the table FUNDRISING_DONATION). The table DONATIONS_SUMMARY must be updated taking into account the donation just received. Also, consider the case that this is the first donation received for that fundraiser.

You must then enter a new record in the table DONATIONS_OUTCOME_NOTIFICATION with information about the *average amount* of donations for the fundraiser. This average amount is calculated as the ratio of the total amount received to the number of donations.

Indications for the performance of the exercise:

You are asked to write the trigger to manage cash donations for fundraisers in the manner described above. Use the text box below to report the trigger.

```
CREATE TRIGGER DonationManagement
AFTER INSERT INTO FUNDRISING_DONATION
FOR EACH ROW
DECLARE
BEGIN
---- Check if it is the first Donation

SELECT COUNT(*) into X
FROM DONATIONS_SUMMARY
WHERE CodF = :NEW.CodF;

IF X = 0 THEN
        INSERT INTO DONATIONS_SUMMARY (….)
        VALUES (:NEW.CodF, 1, :NEW.Amount);
ELSE
        UPDATE DONATIONS_SUMMARY
        SET NumDonations = NumDonations +1,
        TotalAmount = TotalAmount + :NEW.Amount
        WHERE CodF = :NEW.CodF;
END IF;

SELECT NumDonations, TotalAmount INTO NumD, TotI
FROM DONATIONS_SUMMARY
WHERE CodF = :NEW.CodF;

INSERT INTO DONATIONS_OUTCOME_NOTIFICATION(….)
VALUES (:NEW.CodF, :NEW.DonationDate, TotI/NumD):
END;
```

Domanda 6

Risposta non data

Punteggio max.: 1,00

The SQL instruction:

```
CREATE TABLE COURSE
(CODC CHAR(20) NOT NULL,
HOLDER INTEGER NOT NULL DEFAULT 1,
NAME CHAR(30) NOT NULL,
PRIMARY KEY (CODC),
FOREIGN KEY (HOLDER) REFERENCES TEACHER(SERIAL),
ON DELETE SET DEFAULT);
```

- (a) creates a COURSE table in which each cancellation operation on COURSE that violates the referential integrity causes the insertion of default values in the corresponding TEACHER attributes
- (b) creates a COURSE table in which each cancellation operation on COURSE or TEACHER that violates the referential integrity causes the insertion of default values in the corresponding attributes of the other table
- (c) none of the answers is correct
- (d) creates a COURSE table in which any deletion operation on TEACHER that violates the referential integrity causes the insertion of default values in the corresponding COURSE attributes

Risposta errata.

La risposta corretta è: creates a COURSE table in which any deletion operation on TEACHER that violates the referential integrity causes the insertion of default values in the corresponding COURSE attributes

Domanda 7

Risposta non data

Punteggio max.: 4,00

Indications for solving the exercise

Use the text box below to report the ER diagram in textual form. Alternatively, the drawing box can be used to graphically represent the ER diagram.

In case you use the textual form

For each entity indicate the name, the identifier of the entity (internal or external), list of attributes, presence of any composite attributes, the cardinality of the attributes if different from (1,1), and any membership in hierarchies. For each hierarchy indicate the relative classifications.

For each relationship indicate the name, the entities involved with the relative minimum and maximum cardinality, and any attributes of the relationship.

For example, use the following notation:

entity LESSON
internal id: CodLezione
external id: COURSE identifier
attributes: day

entity LOCAL
id: LocalCode
child entities, hierarchy (t, e)
entity CLASSROOM
attributes: capacityMax, floor

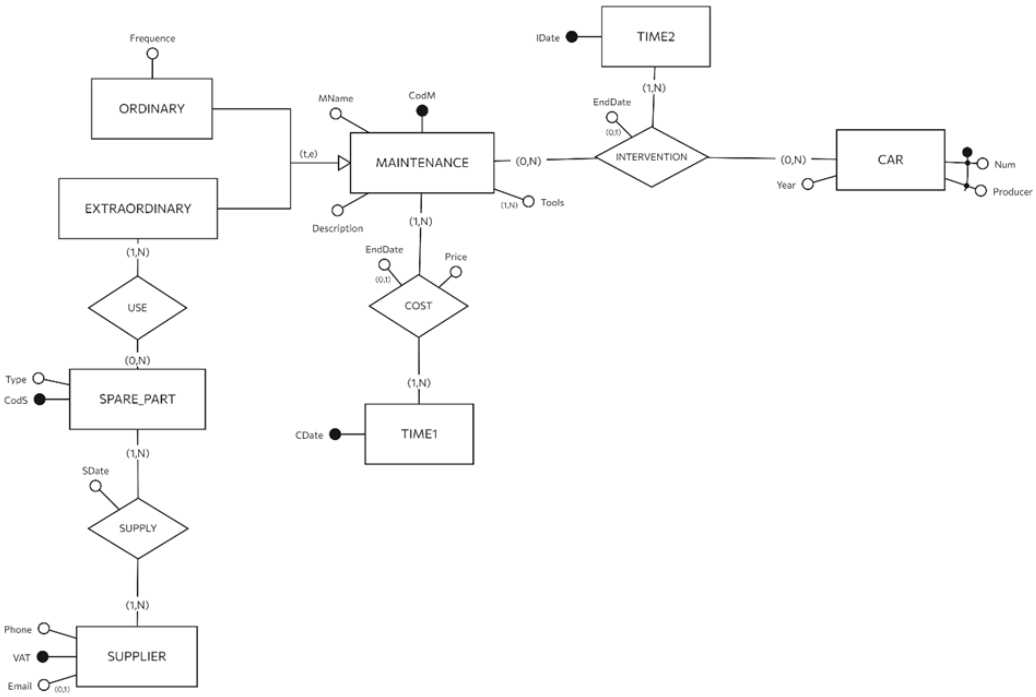
entity LABORATORY
attributes: numPC, devices (1, N)

relationship IN
LESSON (1, 1)
LOCAL (0, N)
attributes: start time, end time (0,1)

Describe the Entity-Relationship diagram addressing the following specifications.

- We want to create a database for the maintenance operations of the cars available in a fleet of cars specialized in vintage cars. Each car is identified by a progressive number and characterized by the name of the manufacturer; for each car, the year of construction is also known.
- Maintenance operations are identified by a unique code and are characterized by a name, a description, and a list of tools to be used to carry out maintenance. Maintenance operations can be ordinary or extraordinary. For ordinary maintenance operations, the frequency with which they must be carried out is known. For extraordinary maintenance operations, the list of spare parts to be used is known.
- Each spare part is identified by a unique code; we also know the type. For each component, the suppliers who supply it and the date of the last relative supply are known. Each supplier is identified by the VAT number and characterized by a telephone number and an email address (if available). Consider that each component can be supplied by multiple suppliers and that each supplier can supply multiple components.

- The cost of maintenance operations varies over time. You want to keep track of the cost assumed over time by each maintenance operation (i.e., in different time intervals characterized by start date and end date).
- You also want to keep track of the maintenance operations carried out on each car over time. Consider that, on the same car, the same maintenance operation can be carried out several times but on different dates, and different maintenance operations can be carried out on the same date. Also, consider that the same maintenance operation can be performed on multiple cars on the same day, but only once a day for the same car.



entity CAR
Id: Num, Producer
Attributes: Year
entity MAINTENANCE
Id: CodM,
Attributes: MName, Tools (multi), Description
Child Entities
entity ORDINARY
Attributes: Frequence
entity EXTRAORDINARY

entity TIME1
Id: CDate

entity TIME2
Id: IDate

entity SPARE_PART
Id: CodS
Attributes: Type

entity SUPPLIER
Id: VAT
Attributes: Phone, Email (optional)

relationship COST
MAINTENANCE: 1,N
TIME1: 1,N
Attributes: Price, EndDate (optional)

relationship INTERVENTION
MAINTENANCE: 0,N,
TIME2: 1,N,
CAR: 0,N
Attributes: EndDate (optional)

relationship USE
EXTRAORDINARY: 1,N,
SPARE_PART: 0,N

relationship SUPPLY
SPARE_PART: 1,N,
SUPPLIER: 1,N
Attributes: SDate

Domanda 8

Risposta non data

Punteggio max.: 1,00

The creation of a view is

- (a) all answers are correct
- (b) a DDL command
- (c) a command at physical level
- (d) a DML command

Risposta errata.

La risposta corretta è: a DDL command

Domanda 9

Risposta non data

Punteggio max.: 1,00

A transaction is "atomic" if

- (a) makes permanent changes made immediately after the end of the transaction
- (b) none of the answers is correct
- (c) it is executed in the system at the same time as other transactions as if it were the only one to be executed
- (d) all the operations that compose it are completed, or are undone, as if they had never been performed

Risposta errata.

La risposta corretta è: all the operations that compose it are completed, or are undone, as if they had never been performed

Domanda 10

Risposta non data

Punteggio max.: 3,00

Given the following relational schema (primary keys are underlined, optional attributes are indicated with '*'):

WORKSHOP (WID, Name, Address, City)
VEHICLE (LicensePlate#, Model, Maker, Category, FuellingType, RegistrationYear, SSN)
CUSTOMER (SSN, FirstName, LastName, BirthDate, Address, City)
INSPECTION (LicensePlate#, WID, Date, Cost)

Write the following query in the SQL language:

Display the name and address of each workshop in the city of Turin that has never performed inspections on LPG-fueled vehicles registered before the year 2015.

Assignment for the exercise

Use the text box below to provide your solution.

```
SELECT Name, Address
FROM WORKSHOP
WHERE City = 'Turin'
AND WID NOT IN
        (SELECT WID
        FROM INSPECTION I, VEHICLE V
        WHERE I.LicensePlate# = V. LicensePlate#
        AND FuellingType='LPG'
        AND RegistrationYear < 2015)
```

Domanda 11

Risposta non data

Punteggio max.: 3,00

Given the following relational schema (primary keys are underlined, optional attributes are indicated with '*'):

WORKSHOP (WID, Name, Address, City)
VEHICLE (LicensePlate#, Model, Maker, Category, FuellingType, RegistrationYear, SSN)
CUSTOMER (SSN, FirstName, LastName, BirthDate, Address, City)
INSPECTION (LicensePlate#, WID, Date, Cost)

Write the following query in the SQL language:

For each customer in the city of Milan who owns at least 3 vehicles in the "motorcar" category, display the customer's last name and, for each of his/her vehicles, display the license plate number, the model, the total number of inspections performed between 2010 and 2020, the number of different workshops at which those inspections were performed, and the total cost of those inspections.

Assignment for the exercise

Use the text box below to provide your solution.

```
SELECT  (C.SSN,) LastName, V.LicensePlate#,
            Model, COUNT (*), COUNT (DISTINCT WID), SUM(Cost)
FROM CUSTOMER C, INSPECTION I, VEHICLE V
WHERE C.SSN=V.SSN
AND I.LicensePlate# = V.LicensePlate#
AND City = 'Milan' AND C.SSN IN
                    (SELECT SSN
                    FROM VEHICLE
                    WHERE Category = "Motorcar"
                    GROUP BY SSN
                    HAVING COUNT(*)>=3)
AND Date >= 1/1/2010 AND Date <= 31/12/2020
GROUP BY C.SSN, LastName, V. LicensePlate#, Model
```