Introduction to databases

Iniziato	domenica, 4 luglio 2021, 15:27
Stato	Completato
Terminato	domenica, 4 luglio 2021, 15:30
Tempo impiegato	3 min. 14 secondi
Valutazione	Non ancora valutato

Domanda 1

Risposta corretta

Punteggio ottenuto 1,00 su 1,00

The primary key of a table

- (a) must be referenced by a foreign key
- (b) may not be composed of a single element
- (c) may not be composite
- (d) must be unique but it might not be minimal
- (e) none of the answers are correct ✓

La risposta corretta è: none of the answers are correct

Domanda 2

Risposta corretta

Punteggio ottenuto 1,00 su 1,00

The PHP instruction:

```
foreach ($var1 as $var2 => $var3) {
    echo "$var3 <br />";
}
```

- (a) none of the other answers are correct
- (b) It is wrong because of
 is not a PHP instruction
- (c) It is correct if $var3 is a scalar array
- (d) It is correct if $var1 is a scalar array
- (e) It is correct if $var1 is an associative array ✓

La risposta corretta è: It is correct if $var1 is an associative array

Domanda 3

Risposta corretta

Punteggio ottenuto 1,00 su 1,00

A transaction has the property of durability if

- (a) it is executed on the system at the same time as other transactions as if it were the only one being executed
- (b) it makes modifications permanent immediately after the transaction has ended ✓
- (c) it takes the system from a valid state to another valid state
- (d) none of the answers are correct
- (e) all of the operations composing it are either completed, or they are undone, as if they had never been executed

La risposta corretta è: it makes modifications permanent immediately after the transaction has ended

Domanda 4

Completo

Punteggio max.: 4,00

Given the following relational tables:

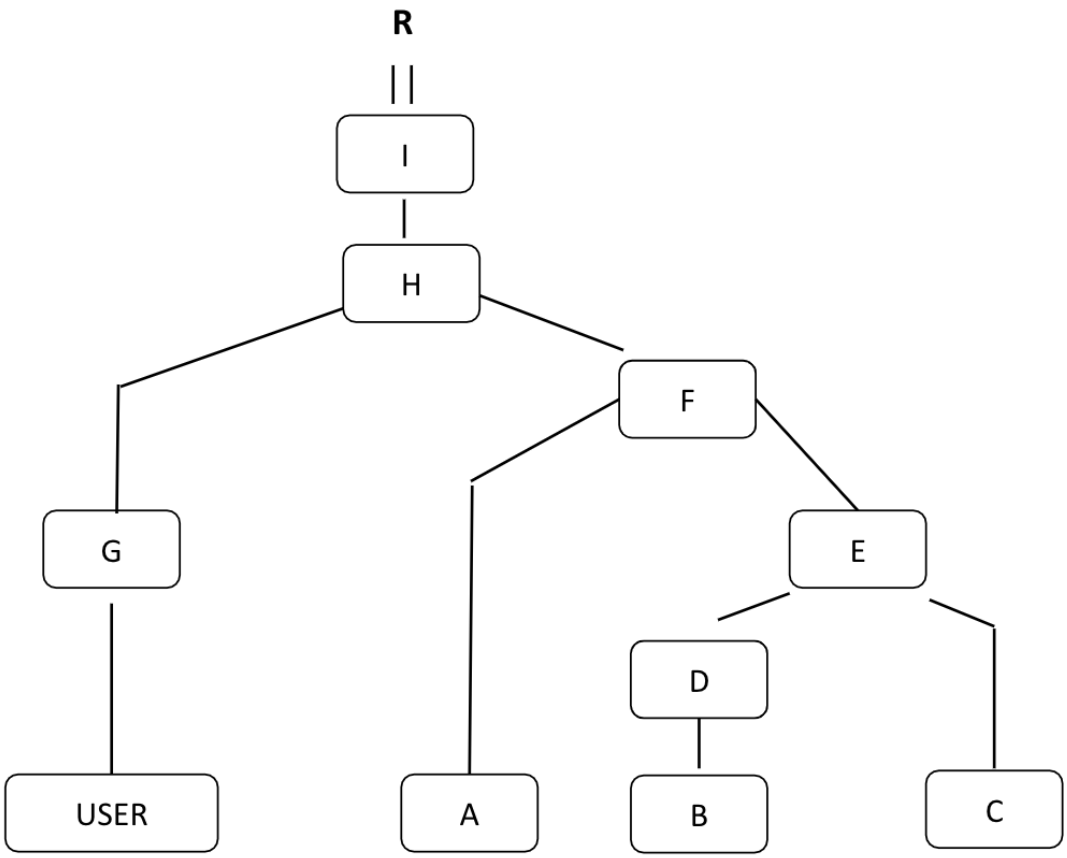
USER (UID, Name, Surname, Birth_date, City)
VIDEO (VID, Title, Category)
VIEW (UID, VID, #Views)

N.B. The #Views attribute has a minimum value of 1.

Select name and surname of users living in Turin who have viewed only videos in the category Entertainment.

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 and I 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.



A. VIEW VW1

B. Video V

C. VIEW VW

D. selection: Category <> Entratainment

E. natural join

F. ANTI-SEMI-JOIN VW1.Uid = V2.Uid

G. selection: City = 'Turin'

H. Natural join

I. projection: name, surname

Draft Solution

A. View1
B. Video
C. View2
D. selection: Category<>'Entertainment'
E. Theta-join: View1.VID=Video.VID or natural join
F. Anti semi-join: View1.UID=View2.UID
G. Selezione: City='Turin'
H. Theta-join: USER.UID=View1.UID or natural join
I. Projection: Name, Surname

Domanda 5

Completo

Punteggio max.: 3,00

Given the following relational tables

WRITER(CodW, Name, Surname, Gender, BirthDate)
BOOK (CodB, Title, Category, CodP, CodW, DateOfPublication)
PUBLISHER (CodP, Name, Headquarter)
WRITER-PRESENT-BOOK (CodW, CodB, PresentationDate, City)

Write the following query in SQL language:

Show name and surname of the writers, born between 1960 and 1970, who have never presented more than one book, belonging to the category “Science fiction”, on the same date.

Assignment for the exercise

Use the text box below to provide your solution.

BLOCK A

SELECT WP.CodW

FROM BOOK B, Write-present-book WP

WHERE B.CodB = WP.CodB AND B.Category = 'Science fiction'

GROUP BY CodW, PresentationDate

HAVING COUNT(*) > 1

MAIN BLOCK

SELECT Name, Surname

FROM WRITER W

WHERE BirthDate>='01/01/1960' AND BirthDate<='01/01/1970' AND CodW NOT IN BLOCK A

Draft solution

```
SSELECT Name, Surname
FROM WRITER S
WHERE S.BirthDate >= “1-1-1960” AND S.BirthDate <= “31-12-1970” AND
S.CodW NOT IN (SELECT SPL.CodW
      FROM WRITER-PRESENT-BOOK SPL, BOOK L
      WHERE SPL.CodP = L.CodP AND
      L.Category = ”Science Finction”
      GROUP BY SPL.CodW, SPL.PresentationDate
      HAVING COUNT(*)>1);
```

Domanda 6

Completo

Punteggio max.: 3,00

Given the following relational tables

WRITER(CodW, Name, Surname, Gender, BirthDate)
BOOK (CodB, Title, Category, CodP, CodW, DateOfPublication)
PUBLISHER (CodP, Name, Headquarter)
WRITER-PRESENT-BOOK (CodW, CodB, PresentationDate, City)

Write the following query in SQL language:

Show the dates on which the number of books presented by female writers is higher than the number of books presented by male writers. Sort the results in decreasing order of date of publication.

Assignment for the exercise

Use the text box below to provide your solution.

SELECT PresentationDate

FROM WPB, Writer W

WHERE W.CodW = WPB.CodW AND W.Gender= 'Female'

GROUP BY PresentationDate

HAVING COUNT(*)> BLOCK B

ORDER BY PresentationDate DESC

BLOCK B

(SELECT COUNT(*)

FROM WP1, Writer W1

WHERE WPB.CodW = W1.CodW AND Gender= 'Male' AND WPB1.PresentationDate = WPB.PresentationDate

GROUP BY W1.CodW, WPB1.PresentationtDate)

Draft solution

```
SELECT PresentationDate
FROM WRITER-PRESENT-BOOK SPL, WRITER S
WHERE S.CodW = SPL.CodW AND Gender = “F”
GROUP BY PresentationDate
HAVING COUNT(*) > (
            SELECT COUNT(*)
            FROM WRITER-PRESENT-BOOK SPL2, WRITER S2
            WHERE S2.CodW = SPL2.CodW AND Gender = “M” AND
SPL2.PresentationDate =             SPL.PresentationDate)
ORDER BY PresentationDate DESC;
```

Domanda 7

Completo

Punteggio max.: 5,00

Given the following relational tables

WRITER(CodW, Name, Surname, Gender, BirthDate)
BOOK (CodB, Title, Category, CodP, CodW, DateOfPublication)
PUBLISHER (CodP, Name, Headquarter)
WRITER-PRESENT-BOOK (CodW, CodB, PresentationDate, City)

Write the following query in SQL language:

Show code, name, and headquarter of the publishers that have published all the books of category “Medicine” presented in at least three different cities during the year 2020.

Assignment for the exercise

Use the text box below to provide your solution.

SELECT CodP, Name, Headquarte

FROM PUBLISHER P

WHERE CodP IN BLOCK A

BLOCK A

SELECT B.CodP

FROM Book B

WHERE Category = 'Medicine' AND B.CodB IN BLOCK B

GROUP BY B.CodP

HAVING COUNT(DISTINCT City) >=3

BLOCK C

SELECT COUNT(*)

FROM BOOK B1

WHERE Category = 'Medicine' AND B.CodB IN BLOCK B

Draft solution

```
SELECT E.CodP, Name, Headquarter
FROM PUBLISHER E, BOOK L
WHERE P.CodB = E.CodB AND Category = “Medicine”
     AND CodB IN (SELECT SPL.CodB
                        FROM WRITER-PRESENT-BOOK SPL
                        WHERE SPL.PresentationDate >= 01-01-2020 AND
                        SPL.PresentationDate <= 31-12-2020
                         GROUP BY SPL.CodB
                          HAVING COUNT(DISTINCT SPL.City) >= 3
GROUP BY E.CodP, Name, Headquarter
HAVING COUNT(*) =  (SELECT COUNT(*)
                           FROM BOOK L2
                           WHERE Category = “Medicine”
                           AND L2.CodB IN (  SELECT L3.CodB
                                                  FROM WRITER-PRESENT-BOOK SPL
```

```
                                    WHERE SPL.PresentationDate >=    01-01-
2020 AND
                                    SPL.PresentationDate <= 31-12- 2020
                                  GROUP BY SPL.CodB
                                 HAVING COUNT(DISTINCT SPL.City) >= 3 )
                    );
```

Domanda 8

Completo

Punteggio max.: 3,00

Describe the Entity-Relationship diagram addressing the following specifications.

You are requested to design the database for the management of staff shifts in different company buildings.

The database must contain a list of employees. Employees are identified by a unique code within the company, they are characterized by their name, email address, and the list of qualifications. Employees can be either factory workers or clerks, for clerks their role is known.

The database must contain the list of the company buildings, which are identified by a unique code and characterized by their address and the list of telephone numbers.

You are requested to keep track of all the work shifts of each employee at the various buildings. A work shift is characterized by a date, a start time and an end time, the building where it takes place, and it can be either ordinary or overtime. A shift is associated with only one employee and one building. The same employee can have work shifts in different buildings, however, an employee can have only one shift on the same day.

Indications for solving the exercise

Use the text box below to report the ER diagram in text form. Alternatively, you can use the drawing box to graphically represent the ER diagram.

ENTITY Worker

PrimaryKey: WCode

Attribute: Name, Surname, telephone (0,1)

ENTITY Product

PrimaryKey: BarCode

Attributes: Categories(1,N), duration, cost

BINARY-RELATIONSHIP ASSIGNED

WORDER(1,N) PRODUCT(1,N)

ENTITY MACHINERY

PrimaryKey: MCode

Attribute: Brand, Model

Generalization(t,e)

ParentEntity: Machinery

ChildrenEntity: ProductionMachinery

Attribute: mxen

ChildrenEntity: QualityControl

Attribute:qualityParameter

ENTITY TIME

PrimaryKey: StartDate, Time

Attribute: endDate, time

TERNARY RELATIONSHIP INSPECTION

WORKER(0,N) TIME(1,N) MACHINERY (0,N)

Draft solution

Entity **EMPLOYEES**
- ID: code
- name, email, qualifications (**1,N**)

GENERALIZATION (**p,e**) children entity **CLERK**
- role

Entity **BUILDINGS**
- ID: code
- address, phones (**1,N**)

Entity **WORK-SHIFTS**
- internal ID: date
- external ID: EMPLOYEE ID (code)
- startTime, endTime, ordinary (T/F)

Relation **WORK:** EMPLOYEE (0,N) – WORK-SHIFT(1,1)

Relation **AT**: WORK-SHIFT (1,1) – BUILDING (0,N)

Domanda 9

Completo

Punteggio max.: 4,00

Describe the Entity-Relationship diagram addressing the following specifications.

We want to design the database for the management of maintenance inspections of workers at different machinery.

The database must contain a list of workers. The workers are identified by a unique code and are characterized by their name, surname, and may have a telephone number.

Each worker is assigned to one or more products. Each product is identified by the bar code and is characterized by the list of categories to which it belongs, the duration in hours of its production process and the cost.

The database must contain a list of machinery. Each machinery is identified by a unique code and is characterized by a brand and a model. The machinery can be either production machinery or quality control machinery. For production machinery, the maximum energy consumption is known. The quality parameter is known for quality control machinery.

You are requested to keep track of all the maintenance inspections that workers have carried out on the various machinery. A worker can carry out multiple inspections on the same machinery at different times. A machinery can receive multiple inspections from the same worker at different times. At the same time, a worker can have multiple inspections on different machinery in progress. You are requested to keep track of the start date and time, and the end date and time of each inspection.

Indications for solving the exercise

Use the text box below to report the ER diagram in text form. Alternatively, you can use the drawing box to graphically represent the ER diagram.

ENTITY EMPLOYEE

PrimaryKey: ECode

Attribute: name, email, listOfQualification(1,N)

ENTITY COMPANY_BUILDING

PrimaryKey: CCode

Attribute: Address, telephones(1,N)

GENERALIZATION (t,e)

ParentEntity: Employee

ChildrenEntity: factory_worker

ChildrenEntity: clerks

Attribte: role

ENTITY Workshift

PrimaryKey: Date

ForeignKey: ECode(EMPLOYEE)

Attribute: startTime, endTime

BINARY-RELATIONSHIP DO:

EMPLOYEE(0,N) WORKSHIFT(1,1)

BINARY-RELATIONSHIP WHERE

WORKSHIFT(1,1) BUILDING(0,N)

Draft solution

Entity **WORKERS**
- ID: code
- name, surname, phone (**0,1**)

Entity **PRODUCTS**
- ID: barCode
- duration, cost, categories (**1,N**)

Entity **MACHINERY**
- ID: code
- brand, model

GENERALIZATION (**t,e**)
children entity **PRODUCTION**, attributes: MaxPowerConsumption
children entity **QUALITY,** attributes: QualityParameter

Entity **INSPECTION**
- interla ID: startDate, StartTime
- external ID: WORKERS (code), MACHINERY (code)
- endDate, endTime

Relations: INSPECTION (1,1) – WORKERS (0,N) e INSPECTION (1,1) – MACHINERY (0,N)
or Relation **INSPECTION**: WORKERS (0,N) – TIME (1,N) – MACHINERY (0,N) + entity TIME(startDate, startTime)

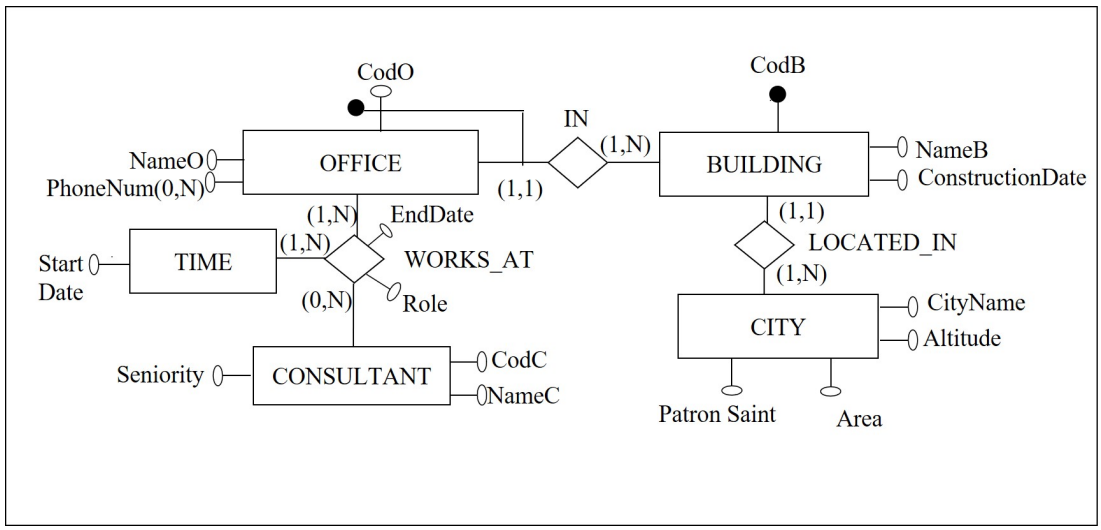
Relation **ASSIGNED:** WORKER (1,N) – PRODUCTS (0,N) or (1,N)

Domanda 10

Completo

Punteggio max.: 3,00

Given the following Entity-Relationship diagram



You are required to:

- Provide a normalized relational logical schema for the same database (N.B. It is not mandatory to report the restructured E-R diagram)
- Define referential integrity constraints for 2 relationships of your choice among those defined in the conceptual schema

Assignment

Use the text box below to provide your solution.

CONSULTALT(CodC_, Name, Seniority)

WORKS_AT(CodC_, StartDate_, CodO_, CodB_, EndDate, role)

OFFICE(CodO_, CodB_, NameO)

HAS_PHONE(PhoneNumb_, CodO, CodB)

BUILDING(CodB_, NameB, ConstructionDate, City, Name)

CITY(CityName_, Altitute, Area, PratonSaint)

INTEGRITY CONSTRAINT

WORKS_AT(CodC) REFERENCES CONSULTANT(CodC)

WORKS_AT(CodO, CodB) REFERENCES CodB (CodO, CodB)

OFFICE(CodB) REFERENCES BUILDING(CodB)

BUILDING(CodB, NameB, ConstructionDate, CityName)
CITY(CityName, Altitude, Patron Saint, Area)
OFFICE(CodO, CodB, NameO)
OFFICE_PHONES(PhoneNum, CodU, CodE)
TIME(StartDate)

CONSULTANT(CodC, NameC, Seniority)
WORK_AT(CodC, CodO, CodB, StartDate, EndDate, Role)

Referential Integrity constraints:

BUILDING(CityName) REFERENCES CITT(CityName)
WORK_AT(CodC) REFERENCES CONSULTANT(CodC)
WORK_AT(CodO, CodB) REFERENCES OFFICE(CodO, CodB)

Domanda 11

Completo

Punteggio max.: 3,00

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

CUSTOMER(CustomerCode, Name, Surname)
CREDIT_CARD (CreditCardNumber, CustomerCode, ActivationDate, ExpirationDate)
EXPENSES_TRANSACTIONS(CreditCardNumber, Date, Hour, Location, ExpendedAmount)
REQUEST_INFO_SUSTAINED_EXPENSES(RequestCode, CreditCardNumber, StartDate , EndDate)
TOTAL_EXPENSES_NOTIFICATION(RequestCode, CreditCardNumber, StartDate, EndDate, CustomerCode, TotalExpendedAmount)

Write a trigger to handle the following activity related to the management of a credit card.

It is requested to calculate the *total amount of expenses* made in a given time period using a credit card (insertion of a new record in the REQUEST_INFO_SUSTAINED_EXPENSES table). The reference time period is defined by the values of the StartDate and EndDate attributes.

The computation of the total amount of expenses must be made by taking into account the information available in the EXPENSES_TRANSACTIONS table. This table contains, for each credit card, the expense transactions made on each date and the corresponding amount.

If there is no expense transaction for the specified credit card within the given time period, the trigger ends with an error. Otherwise, a new record must be inserted into the TOTAL_EXPENSES_NOTIFICATION table with the information on the total amount of expenses and the code of the customer who owns the credit card. The CREDIT_CARD table contains, for each credit card, the code of the customer who is the cardholder.

Indications for carrying out the exercise:

*Given the following incomplete solution of the trigger, you are asked to complete **Part A** in bold by specifying the body of the trigger. Use the text box below to provide your solution.*

Use the raise_application_error (....) function to raise an error. It is not required to specify the parameters passed to the function.

```
create or replace trigger TotalExpenseCalculation
after insert on REQUEST_INFO_SUSTAINED_EXPENSES
for each row
Part A
```

```
DECLARE
N number;
OWNER CHAR(10);
BEGIN
SELECT SUM (ExpendedAmount) INTO N
```

```
FROM EXPENSES_TRANSACTIONS ET
WHERE CreditCardNumber =:NEW.CreditCardNumber AND ET.StartDate>=:
New.StartDate
      AND ET.EndDate<= :NEW.EndDate
IF(N = 0) THEN
      raise_application_error();
END IF;
SELECT CustomerCode AS OWNER
FROM Credit_Card CC
WHERE CreditCardNumber = :New.CreditCardNumber;
INSERT INTO TOTAL_EXPENSES_NOTIFICATION (RequestCode,
CreditCardNumber, StartDate, EndDate, CustomerCode, TotalExpendedAmount)
VALUES (:NEW.RequestCode, :NEW.CreditCard, :NEW.StartDate, :NEW.EndDate,
OWNER, N);
END;
```

```
declare
X number;
Y number;
begin

select SUM(ExpendedAmount) INTO X
FROM EXPENSES_TRANSACTIONS
where  CreditCardNumber = :NEW.CreditCardNumber and Date>= :NEW.StartDate
and Data>=:NEW.EndDate;
IF X IS NULL THEN
    raise_application_error(...);
END IF;

SELECT CustomerCode INTO Y
FROM CREDIT_CARD
WHERE CreditCardNumber = :NEW.CreditCardNumber;
INSERT INTO TOTAL_EXPENSES_NOTIFICATION(RequestCode, CreditCardNumber,
StartDate, EndDate, CustomerCode, TotalExpenses)
values (:NEW.RequestCode, :NEW.CreditCardNumber, :NEW.StartDate,
:NEW.EndDate, Y, X);
END;
```