Introduction to databases

Iniziato	domenica, 4 luglio 2021, 14:05
Stato	Completato
Terminato	domenica, 4 luglio 2021, 14:10
Tempo impiegato	5 min. 7 secondi
Valutazione	Non ancora valutato

Domanda 1

Risposta corretta

Punteggio ottenuto
1,00 su 1,00

A transaction has the property of durability if

- ⦿ (a) it makes modifications permanent immediately after the transaction has ended ✔
- ◯ (b) it takes the system from a valid state to another valid state
- ◯ (c) none of the answers are correct
- ◯ (d) it is executed on the system at the same time as other transactions as if it were the only one being executed
- ◯ (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 2

Risposta corretta

Punteggio ottenuto
1,00 su 1,00

The PHP instruction:

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

○ (a) It is wrong because of
 is not a PHP instruction

○ (b) It is correct if $var1 is a scalar array

○ (c) It is correct if $var3 is a scalar array

○ (d) none of the other answers are correct

◉ (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

The primary key of a table

○ (a) may not be composite

○ (b) must be unique but it might not be minimal

○ (c) may not be composed of a single element

○ (d) must be referenced by a foreign key

◉ (e) none of the answers are correct ✓

La risposta corretta è: none of the answers are correct

Given the following relational tables:

PLAYER (PID, Nickname, BirthDate, Country)
MATCH (PID, VID, Date, #Hours)
VIDEOGAME (VID, Name, Genre)

Select the nickname and the country of players who have played two different videogames belonging to the same genre on the same date.

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.
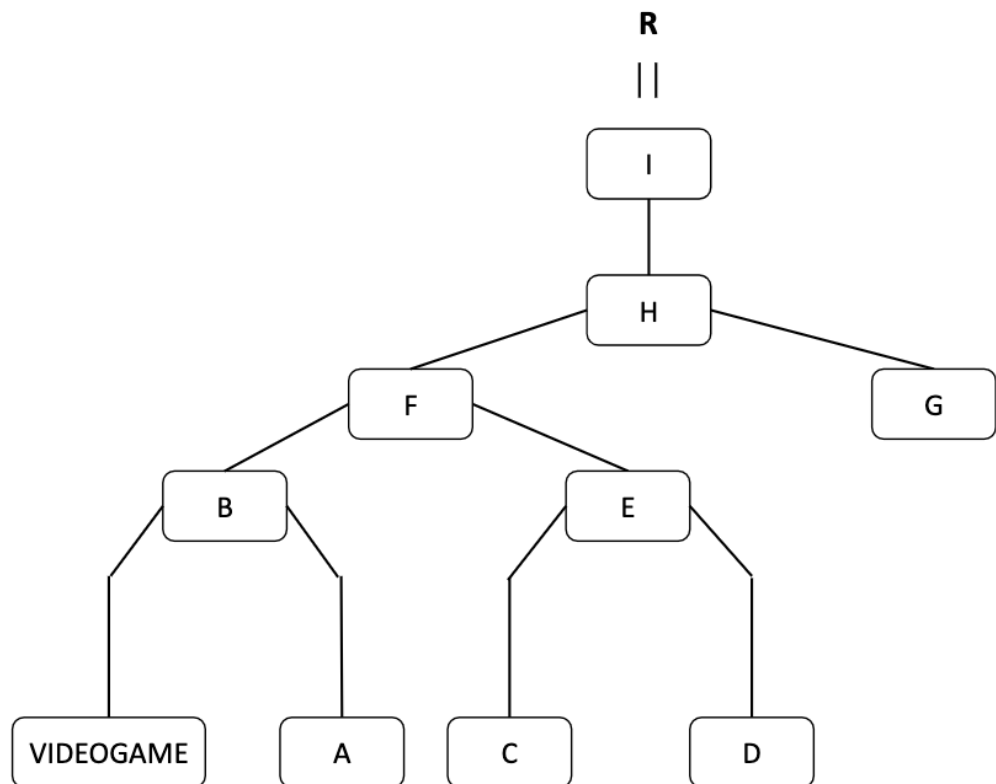
A. Match M

B. Videogame V1

C. Match M1

D. Natural join

E. Natural join

F. Theta-join = V1.Vid <> V.Vid AND V1.Genre= V1.Genre AND V1.Date=V.Date AND M.PID = M1.PID

G. PLAYER P

H. right-semi-join: M.PID = P.PID

I. projection: P.Nickname, P.Country

Draft Solution

    A. MATCH

    B. Natural join

    C. Videogame2

    D. Match2

    E. Natural join

    F. Theta-join: Videogame.VID<>VideoGame2.VID AND Match.PID=Match2.PID AND Match.Date=Match2.Date AND Videogame.genre=Videogame2.genre

    G. Player

    H. Theta-join Player.PID=Match.PID

    I. Projection: Nickname, Country

Given the following relational tables

FOOTBALL-PLAYER(<u>CodFP</u>, Name, Surname, Address, CityOfBirth)

TEAM(<u>CodT</u>, Name, Category, ConstitutionDate)

AWARD(<u>CodA</u>, CodFP, DeliveryDate, DeliveryPlace, Type)

PLAYER-TEAM(<u>CodFP</u>, <u>CodT</u>, <u>StartingDate</u>, EndingDate, NumPlayedMatches, NumGoals)

Write the following query in SQL language:

Find name, surname of the football players who played in at least one team of category "Serie A" and who received at least two awards of type "Top scorer".

Assignment for the exercise

Use the text box below to provide your solution.

<u>BLOCK A</u>

SELECT CodFP

FROM PLAYER-TEAM PT, TEAM T

WHERE PT.CodT = T.CodT AND T.Category = 'Serie A'

<u>BLOCK B</u>

SELECT CodFP

FROM AWARD A1

WHERE Type = 'Top Score'

GROUP BY CodFP

HAVING COUNT(*)>=2

<u>Main block</u>

SELECT Name, Surname

FROM FOOTBALL-PLAYER FP

WHERE CodFP IN (BLOCKA)

    AND CodFP IN (BLOCK B)

Draft solution

```
SELECT DISTINCT Name, Surname
FROM PLAYER-PLAYER PP, AWARD A
WHERE PP.CodFP = A.CodFP AND Type='Top scorer'
AND PP.CodFP IN
    (
    SELECT CodFP FROM TEAM T, PLAYER-TEAM PT
    WHERE PT.CodT = T.CodT
    AND Category = 'Serie A'
    )
GROUP BY PP.CodFP, Name, Surname
HAVING COUNT(*)>=2;
```

Given the following relational tables

FOOTBALL-PLAYER(CodFP, Name,Surname, Address, CityOfBirth)

TEAM(CodT, Name, Category,ConstitutionDate)

AWARD(CodA, CodFP, DeliveryDate, DeliveryPlace, Type)

PLAYER-TEAM(CodFP, CodT, StartingDate,EndingDate, NumPlayedMatches, NumGoals)

Write the following query in SQL language:

For each team in which players living in Turin (attribute "City of birth") have never played, find team name and category and, for each football player, code, surname, and total number of days of militancy in that team (computed as the difference between ending and starting dates).

Assignment for the exercise

Use the text box below to provide your solution.

BLOCK A

SELECT CodT

FROM PLAYER-TEAM PT, FOOTBALL-PLAYER FP

WHERE FP.CodFP= PT.CodFP AND CityOfBirth = 'Turin'

    AND NumPlayedMathces != 0

Main block

SELECT T.Name, T.Category, PT1.CodFP, Surname, FP1.Name, SUM(EndingDate - StartingDate)

FROM TEAM T, PLAYER-TEAM PT1, FOOTBALL-PLAYER FP1

WHERE T.CodT= PT1.CodT AND PT1.CodFP= FP1.CodFP AND T.CoT NOT IN (BLOCK A)

GROUP BY T.CodT, T.Name, T.Category, PT1.CodFP, Surname, FP1.Name

```
Draft solution

SELECT T.Name, T.Category, FP.CodFT, FP.Surname, FP.Name,
SUM(EndingDate - StartingDate)
FROM TEAM T, FOOTBALL-PLAYER FP, PLAYER-TEAM PT
WHERE PT.CodFP = FP.CodFP AND PT.CodT = T.CodT
AND PT.CodT NOT IN
    (
     SELECT PT.CodT
    FROM PLAYER-TEAM PT, FOOTBALL-PLAYER FP
    WHERE PT.CodFP = FP.CodFP and CityOfBirth= "Torino"
     )
GROUP BY T.CodT, T.Name, T.Category, FP.CodFT, FP.Surname,
FP.Name;
```

Given the following relational tables

FOOTBALL-PLAYER(<u>CodFP</u>, Name,Surname, Address, CityOfBirth)

TEAM(<u>CodT</u>, Name, Category,ConstitutionDate)

AWARD(<u>CodA</u>, CodFP, DeliveryDate, DeliveryPlace, Type)

PLAYER-TEAM(<u>CodFP</u>, <u>CodT</u>, <u>StartingDate</u>,EndingDate, NumPlayedMatches, NumGoals)

Write the following query in SQL language:

Find name, category and constitution date of the teams in which have played (in any time) the players who received, in their career as a whole, the highest number of awards of type "Best goalkeeper".

Assignment for the exercise

Use the text box below to provide your solution.

BLOCK A

(SELECT CodFP, COUNT(*) AS NUM

FROM AWARD AW1

WHERE Type = 'BestgoalKeeper'

GROUP BY CodFP) AS AM

BLOCK B

SELECT MAX(NUM)

FROM (BLOCK A)

WHERE AM.CodFP = PT.CodFP

MAIN BLOCK

SELECT DISTINCT T.Name, T.Category, T.ConstitutionDate

FROM TEAM T, PLAYER-TEAM PT

WHERE T.CodT = PT.CodT AND PT.codFP IN (SELECT CodFP

FROM AWARD AW

WHERE A.Type = 'BestgoalKeeper'

GROUP BY CodFP

HAVING COUNT(*) = BLOCK B)

Draft solution

```
SELECT distinct T.Name, T.Category, T.CostituitionDate
FROM TEAM T, PLAYER-TEAM PT
WHERE T.CodFP= PT.CodFP and PT.CodFP IN
     (SELECT CodFP
          FROM AWARD D
          WHERE A.Type = 'Best goalkeeper'
          GROUP BY CodFP
          HAVING COUNT(*) =
               SELECT MAX(AwardTot)
```

```
FROM (SELECT count(*)as AwardTot
           AWARD A
           WHERE A.Type = 'Best goalkeeper'
           GROUP BY CodFP) AS TOTAWARDFP
```

Describe the Entity-Relationship diagram addressing the following specifications.

You are requested to design the database for the management of rental contracts.

The database must contain a list of properties available for rent. The properties are identified by a land code, they are characterized by the address, the area in square meters and the list of rooms. Among the different properties available, for the apartments, the floor number is known.

The database must contain a list of agencies, identified by a code, and characterized by the address and the list of phone numbers.

You are requested to keep track of all the rental contracts of the agencies for each property. Different rental contracts may have been stipulated for the same property, with the same agency or with different agencies. A rental contract is characterized by a start date and an end date, by a monthly price, and can be renewable or non-renewable. A property can only have one rental contract for the same time period. A contract is associated with a single property and a single agency.

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 AGENCY

    PrimaryKey: Code

    Attribute: Address, phoneNumber(1,N)

ENTITY CONTRACT

    PrimaryKey: StartDate

    Attribute: endDate, montlyPrice, renewable*

    ForeingKey: PropertyCode(Property)

BINARY-RELATIONSHIP STIPULATE

    AGENCY(0,N) CONTRACT (1,1)

BINARY-RELATIONSHIP WITH

    CONTRACT(1,1) PROPERTY(0,N)

ENTITY PROPERTY

    PrimaryKey: LandCode

    Attribute: address, Area, rooms(1,N)

GENERALIZATION (p,e)

ParentEntity: Property

ChildrenEntity: Apartment

      Attribute: floornumber

Draft solution

Entity PROPERTY
- ID: codP
- address, area, rooms (1,N)

GENERALIZATION (p,e)
- parent entity: PROPERTY
- child entity: APARTMENT
    - floorNumber

Entità AGENCY
- ID: codA
- address, phoneNumbers (1,N)

Entità CONTRACT
- internal ID: startDate
- external ID: ID of PROPERTY (codP)
- endDate, price, renewable

Relationships: CONTRACT (1,1) – AGENCY (0,N)

                    PROPERTY (0,N) - CONTRACT (1,1)

Describe the Entity-Relationship diagram addressing the following specifications.

You are requested to design the database for the management of rental contracts.

The database must contain a list of properties available for rent. The properties are identified by a land code, they are characterized by the address, the area in square meters and the list of rooms. Among the different properties available, for the apartments, the floor number is known.

The database must contain a list of agencies, identified by a code, and characterized by the address and the list of phone numbers.

You are requested to keep track of all the rental contracts of the agencies for each property. Different rental contracts may have been stipulated for the same property, with the same agency or with different agencies. A rental contract is characterized by a start date and an end date, by a monthly price, and can be renewable or non-renewable. A property can only have one rental contract for the same time period. A contract is associated with a single property and a single agency.

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: AGENCY

PrimaryKey: Code

Attribute: address, phone(1,N)

ENTITY CONTRACT

PrimaryKey: StartDATE

ForeignKey: PropertyCode(Property)

Attribute: endDate, montlyPrice, renewable*

BINARY-RELATIONSHIP STIPULATE

   AGENCY(0,N) CONTRACT(1,1)

BINARY-RELATIONSHIP WITH

   CONTRACT(1,1) PROPERTY(0,N)

Draft solution

Entity PROPERTY
- ID: codP
- address, area, rooms (1,N)

GENERALIZATION (p,e)
- parent entity: PROPERTY
- child entity: APARTMENT
  - floorNumber

Entità AGENCY
- ID: codA
- address, phoneNumbers (1,N)

Entità CONTRACT
- internal ID: startDate
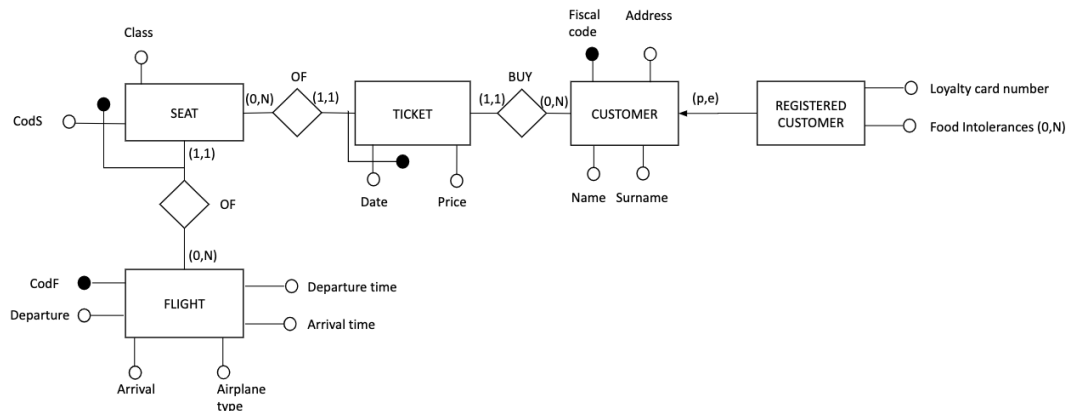- external ID: ID of PROPERTY (codP)
- endDate, price, renewable

Relationships: CONTRACT (1,1) – AGENCY (0,N)

PROPERTY (0,N) - CONTRACT (1,1)

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.

FLIGHT(CodF_, Departure, Arrival, AirplaneType, ArrivalTime, DepartureTIme)

SEAT(CodS_, CodF_, Class)

TICKET(Date_, CodS_, CodF_, Price, FiscalCode)

CUSTOMER(FiscalCode_, Address, Name, Surname)

Registered_Customer(FiscalCode_, LoyaltyCardNumber)

HASINTOLERANCE(FiscalCode_, intolerance_)

SEAT(CodF) REFERENCES Flight(CodF)

TICKET (CodS, CodF) REFERENCES Seat(CodS, CodF)

TICKET(FiscalCode) REFERENCES Customer(FiscalCode)

CUSTOMER(FiscalCode, Name, Surname, Address, LoyaltyCard*, Type)

INTOLERANCES(Intolerances, FiscalCode)

TICKET(CodS, CodF, Date, FisclaCode, Price)

SEAT(CodS, CodF, Class)

FLIGHT(CodF, Departure, Arrival, Departure_date, Arrival_date, Airplane_type)

Integrity constraints:

SEAT(CodF) REFERENCES FLIGHT(CodF)

TICKET(CodS, CodF) REFERENCES SEAT(CodS, CodF)

TICKET(FiscalCode) REFERENCES CUSTOMER(FiscalCode)

Domanda 11

Completo

Punteggio max.: 3,00

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

SENSOR (<u>SCode</u>, AreaCode, Description)
PERFORMED_MEASUREMENTS (<u>SCode</u>, <u>Date</u>, <u>Hour</u>, Value)
MEASUREMENTS_INFO_NOTIFICATION(<u>RequestCode</u>,Date, AreaCode, MinValue, Max Value)
MEASUREMENTS_INFO_REQUEST(<u>RequestCode</u>, Date, AreaCode)

Write the trigger to manage the following activity at a website for monitoring the atmospheric humidity.

You must calculate the minimum and maximum value of the measurements acquired on a specific date in a given geographical area (insertion of a record in the MEASUREMENTS_INFO_REQUEST table).

The calculation of the minimum and maximum value must be performed by considering all the measurements acquired on the specified date, from all the sensors located in the indicated area. The SENSOR table contains, for each sensor, the area where the sensor is located. The PERFORMED_MEASUREMENTS table contains, for each sensor, the list of values acquired at different time instants (given by the Date and Hour attributes).

If no sensor is available in the specified area the trigger ends with an error. Otherwise, once the required maximum and minimum values have been calculated, a new record must be inserted in the MEASUREMENTS_INFO_NOTIFICATION table with the calculated information.

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.

create or replace trigger CalculationOfMeasurementsInfo
after insert on MEASUREMENTS_INFO_REQUEST
for each row
Part A

DECLARE

M Number;

Me Number;

BEGIN

SELECT MIN(Value) INTO mi, MAX(VALUE) INTO me

FROM SENSOR S, PERFORMED_MEASURMENT PM

WHERE S.SCode = PM.SCode AND Date= :New.Date;

IF(MI IS NULL OR ME IS NULL) THEN

    raise_application_error();

END IF;

```
INSERT INTO MEASURMENT_INFO_NOTIFICATION(RequestCode, Date,AreaCode,
MinValue, MaxValue) VALUES (:New.RequestCode, :New.Date, :New.AreaCode,
MI,ME);
END;
```

Part A:

```
declare
NumSensors, myMIN, myMAX number;
begin
select count()* into NumSensors
FROM SENSOR
where AreaCode = :NEW.AreaCode;


if (NumSensors=0) then
 raise_application_error(....);
end if;


select MIN(Value), MAX(Value) INTO myMIN, myMAX
from SENSOR S, PERFORMED_MEASUREMENTS M
where S.SCode = M.SCode AND AreaCode = NEW.AreaCode AND Date = :NEW.Date


INSERT INTO MEASUREMENTS_INFO_NOTIFICATION(RequestCode, Date,
AreaCode, MinValue, MaxValue) values (:NEW.RequestCode, :NEW.date,
:New.AreaCode, MyMin, MyMax);


END;
```