Introduction to databases

Iniziato	sabato, 3 luglio 2021, 19:06
Stato	Completato
Terminato	sabato, 3 luglio 2021, 19:31
Tempo impiegato	25 min. 1 secondo
Valutazione	Non ancora valutato

Domanda 1

Risposta corretta

Punteggio ottenuto 1,00 su 1,00

A transaction is atomic if

- (a) none of the other answers are correct
- (b) takes the system from a valid state to another valid state
- (c) makes permanent the changes made immediately after the end of the transaction itself
- (d) is performed in the system concurrently with other transactions, as if it were the only one to be executed
- (e) all the operations that compose the transaction itself are completed, or are undone as if they had ever been performed ✓

La risposta corretta è: all the operations that compose the transaction itself are completed, or are undone as if they had ever been performed

Domanda 2

Risposta corretta

Punteggio ottenuto 1,00 su 1,00

The SQL statement

```
CREATE TABLE T1
(A1 CHAR (5) NOT NULL,
A2 INTEGER,
A3 CHAR (5) NOT NULL,
PRIMARY KEY (A1, A3),
FOREIGN KEY (A2) REFERENCES T2,
ON UPDATE NO ACTION);
```

- (a) creates a table T1 where any UPDATE operation is prevented
- (b) creates a T1 table in which, following an UPDATE operation on T2 that violates the referential integrity, no actions are executed
- (c) creates a table T1 in which, as a result of an UPDATE operation, no action is executed
- (d) creates a T1 table in which it is required to prevent the execution of any UPDATE operation on T2 violating the referential integrity ✓
- (e) none of the other answers are correct

La risposta corretta è: creates a T1 table in which it is required to prevent the execution of any UPDATE operation on T2 violating the referential integrity

Domanda 3

Risposta corretta

Punteggio ottenuto 1,00 su 1,00

The PHP function mysqli_query()

- (a) is used for displaying the result of a SQL query
- (b) none of the other answers are correct
- (c) is used for the prepared execution of SQL statements
- (d) is used for the immediate execution of SQL statements ✓
- (e) is used for both the immediate execution and the prepared execution of SQL statements (subject to parameter binding)

La risposta corretta è: is used for the immediate execution of SQL statements

Domanda 4

Completo

Punteggio max.: 4,00

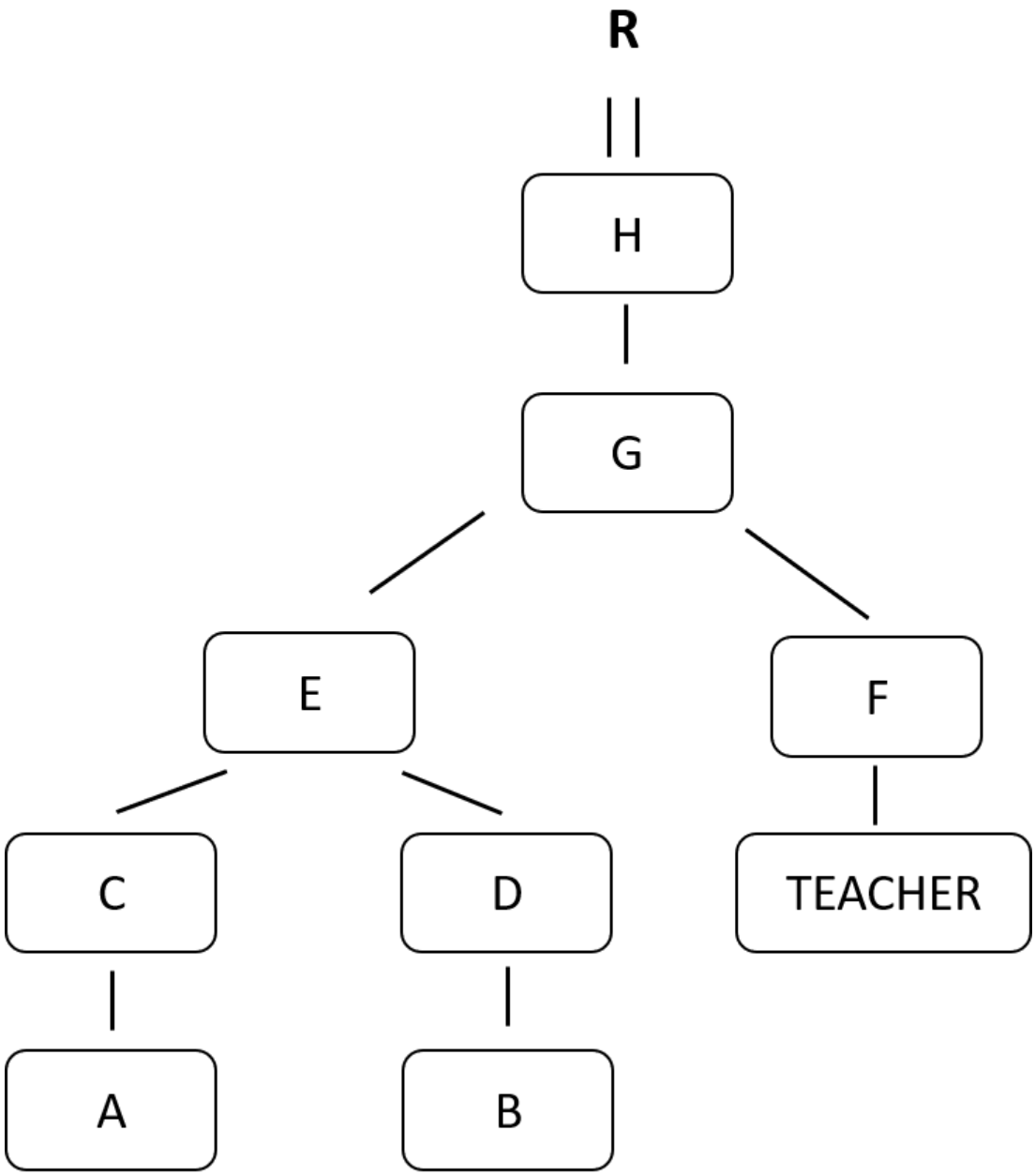
Given the following relational tables:

TEACHER (TeacherID, FirstName, LastName, Department)
COURSE (CourseID, CourseName, Year, Semester, TeacherID, Language)

Select the first and last names of the teachers of the Electronics department who teach at least two English-language courses in the first semester of the second year.

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, and H 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. COURSE AS C1

B. COURSE AS C2

C. selection: Year= 2 AND Semester= First AND Language = English

D. selection: Year= 2 AND Semester= First AND Language = English

E. theta-join: C1.CourseID <> C2.CourseID AND C1.TeacherID = C2.TeacherID

F. selection: Department = 'Electronic'

G. theta-join: TeacherID = Teacher.TeacherID

H. projection: FirstName, LastName

Domanda 5

Completo

Punteggio max.: 3,00

Given the following relational tables

TEACHER (TeacherID, FirstName, LastName, Department)
COURSE (CourseID, CourseName, Year, Semester, MainTeacherID, Language)
STUDENT (StudentID, FirstName, LastName, BirthDate, Nationality)
STUDENT-COURSE-ENROLLMENT (CourseID, StudentID, EnrollmentYear)
STREAMING-VIDEOLECTURE (VideoLectureID, CourseID, Topic, Date, DurationInMinutes, TeacherID)
STREAMING-VIDEOLECTURE-ATTENDANCE (StudentID, VideoLectureID, CourseID, AttendanceInMinutes)

Write the following query in SQL language:

Select the student ID, first name, last name, and nationality of each student who has never attended a streaming video-lecture longer than 10% of the video-lecture duration.

Assignment for the exercise

Use the text box below to provide your solution.

```
SELECT StudentID, FirstName, LastName, Nationality
FROM Student S
WHERE StudentID NOT IN (
                        SELECT StudentID
                          FROM SV, SVA
                           WHERE SV.VideoLectureID = SVA.VideoLectureID AND
                                 AttendanceinMinutes >= 0.1*DurationInMinutes
                )
```

Domanda 6

Completo

Punteggio max.: 3,00

Given the following relational tables

TEACHER (TeacherID, FirstName, LastName, Department)
COURSE (CourseID, CourseName, Year, Semester, TeacherID, Language)
STUDENT (StudentID, FirstName, LastName, BirthDate, Nationality)
STUDENT-COURSE-ENROLLMENT (CourseID, StudentID, EnrollmentYear)
STREAMING-VIDEOLECTURE (VideoLectureID, CourseID, Topic, Date, DurationInMinutes, TeacherID)
STREAMING-VIDEOLECTURE-ATTENDANCE (StudentID, VideoLectureID, CourseID, AttendanceInMinutes)

Write the following query in SQL language:

For each teacher of the Department of "Control and Computer Engineering" who teaches at least three courses, select the teacher's first and last name, and the total number of the teacher's streaming video-lectures.

Assignment for the exercise

Use the text box below to provide your solution.

MAIN QUERY

SELECT FirstName, LastName, COUNT(*)

FROM T, SA

WHERE T.TeacherID = SA.TeacherID AND T.TeacherID IN (QUERY A) AND Department = 'CCE'

GROUP BY T.TeacherID

QUERY A

SELECT TeacherID

FROM COURSE C

GROUP BY TeacherID

HAVING COUNT(*)>=3

Domanda 7

Completo

Punteggio max.: 5,00

Given the following relational tables

TEACHER (TeacherID, FirstName, LastName, Department)
COURSE (CourseID, CourseName, Year, Semester, TeacherID, Language)
STUDENT (StudentID, FirstName, LastName, BirthDate, Nationality)
STUDENT-COURSE-ENROLLMENT (CourseID, StudentID, EnrollmentYear)
STREAMING-VIDEOLECTURE (VideoLectureID, CourseID, Topic, Date, DurationInMinutes, TeacherID)
STREAMING-VIDEOLECTURE-ATTENDANCE (StudentID, VideoLectureID, CourseID, AttendanceInMinutes)

Write the following query in SQL language:

For each student enrolled in at least 3 courses in the 2019-2020 enrollment year, select the student ID and last name, and the ID of each course for which the student attended all streaming video-lectures of that course.

Assignment for the exercise

Use the text box below to provide your solution.

QUERY A.

```
SELECT StudentID
FROM SCE
WHERE EnrollmentYear = '2019-2020'
GROUP BY StudentID
HAVING COUNT (*)>= 3
```

QUERY B

```
  ( SELECT StudentID, CourseID
   FROM SVA
   GROUP BY StudentID, CourseID
   HAVING COUNT(*) = (SELECT COUNT(*)
                      FROM SV
                       WHERE SVA. CourseID = SV.CourseID) ) AS TMP
```

MAIN QUERY

```
SELECT S.StuentID, LastaName, TMP.CourseID
FROM Student S, QUERY B
WHERE S.StudentID IN (QUERY A)
          AND TMP. StudentID = S.StudentID
```

Domanda 8

Completo

Punteggio max.: 2,00

Describe the Entity-Relationship diagram corresponding to the following specifications.

You are requested to design the database for managing the users of a COVID-19 infection-tracking mobile app.

Users of the app are characterized by their SSN (Social Security Number), first name, last name, date of birth, and place of residence. Users are divided into occasional users and registered users.

For registered users, their Local Health Agency and their registration date are known, whereas for occasional users, the brand and type of their mobile device are known. The Local Health Agency is identified by its reference city and a code (e.g. Turin TO1) and is characterized by an email address, a venue, and a list of phone numbers.

Directions for the exercise to run

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

ENTITY USER

Primarykey: SSN

Attribute: FirstName, LastName, DateofBirth, Residence

GENERALIZATION(t,e)

Parent-entity: USER

Children entity: Occasional

Attribute: brand, typeMobileDevice

Children-entity: registeredU

Foreign Key: Code(LHA), RCity(LHA)

BINARY-RELATIONSHIP REGISTRERED

registeredU (1,1) LHA(0,N)

Attribute: registrationDate

BINARY-RELATIONSHIP PARTECIPA

Ospiti(1,N) SCAMBIO(1,1)

ENTITY LHA

Primarykey: Code, RCity

Attribute: email, value, numbers (1,N)

Domanda 9

Completo

Punteggio max.: 2,00

Describe the Entity-Relationship diagram for the following specifications.

You want to design the database for handling alert messages for an app that is useful for tracking COVID-19 infections.

App users are identified by a code and are characterized by their first and last name. If the app incurs an issue or the Bluetooth protocol is turned off, the app notifies an alert to the user. Each alert notification is identified by a unique code and is characterized by a predefined message text (e.g. “The Bluetooth connection is turned off. Restore it as soon as possible.”).

For each user, record the date and time of receipt of the alert notifications. Assume that the same alert notification can be sent multiple times to the same user (i.e., several times on the same date with different times or on different dates) or to different users (on the same date and time, or on different dates and/or times).

Directions for the exercise to run

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

ENTITY USER

Primarykey: UCode

Attribute: FirstName, LastName

ENTITY Notification

Primarykey: NCode

Attribute: predefinedmessage

ENTITY Time

Primarykey: Data, time

TERNARY-RELATIONSHIP GOT

USER(0,N) Notification(0,N), Time(1,N)

Domanda 10

Completo

Punteggio max.: 3,00

Describe the Entity-Relationship diagram for the following specifications.

You want to design the data base for managing notifications of an app useful for co-tracking COVID-19 infections and doctor visits booked by users

The users of the app are identified by a code and are characterized by their first and last name. Users receive notifications of potential exposure to infection. Each notification is identified by a progressive number, unique for each user, and is characterized by a description and a specific risk level.

Users can book medical visits to doctors in the area. Each doctor, identified by the VAT registration number, has a first name, last name, and number of years of experience. For each booking of a medical examination, you are requested to record the date, the requesting user, the doctor in charge of performing the examination, and the place where the examination will be held. Assume that a user can book multiple visits on different dates with the same doctor or with different doctors, but on the same date a user can book at most one examination. Assume also that a doctor can have multiple examinations on the same date.

Directions for the exercise

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

ENTITY USER

Primarykey: UCode

Attribute: FirstName, LastName

ENTITY Notification

Primarykey: NCode

ForeignKey: UCode (USER)

Attribute: description, riskLevel

BINARY-RELATIONSHIP GOT

USER(1,N) Notification(1,1)

ENTITY Doctor

Primarykey: VAT

Attribute: First, last, numberofyearexp

ENTITY Booked

Primarykey: data

Externalkey: UCode

Attribute: place

BINARY-RELATIONSHIP HAS

USER(0,N) Booked(1,1)

BINARY-RELATIONSHIP BY

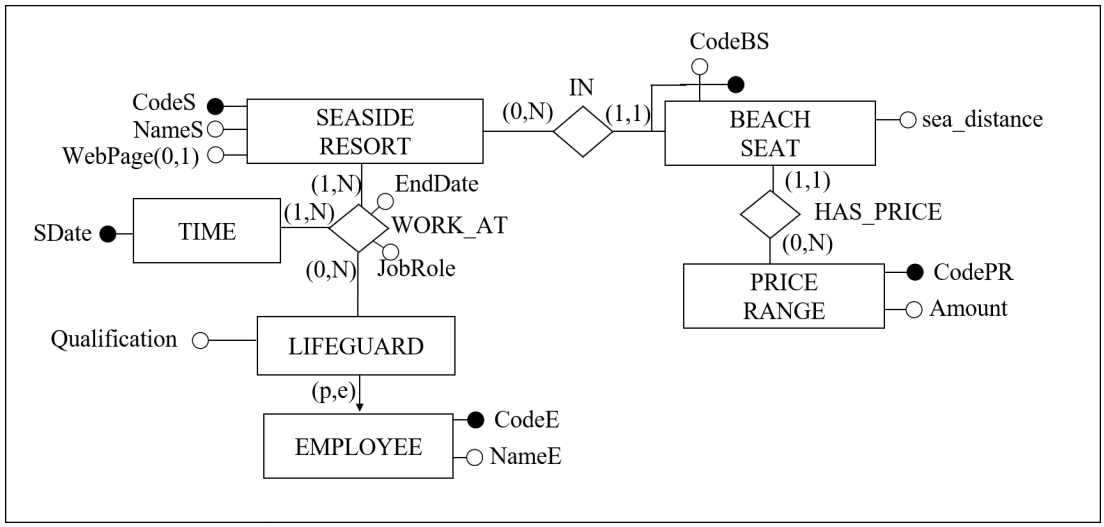
DOCTOR(0,N) Booked(1,1)

Domanda 11

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.

EMPLOYEE(CodeE, NomeE)

LIFEGUARD(CodE, Qualification)

WORKAT(CodE,CodeS, SDate, JobRole, EndDate)

SEASIDERESORT(CodS, NameS, WebPage*)

BEACHSEAT(CodeBS, CodeS, Codepr, sea_distance)

PRINCERANGE(CodePR, Amount)

REFERENTIAL_INTEGRITY_CONSTRAINT

WORKAT(CodeS) REFERENCES(Seaside_Resort)

WORKAT(CodeE) REFERENCES(LIFEGUARD)

IN(CodeS) REFERENCES(Seaside_Resort)

Domanda 12

Completo

Punteggio max.: 3,00

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

STUDENT (StudentID, Name, Surname, MasterDegree)
COURSE (CourseID, Name, NumOfCredits)
EXAM (StudentID, CourseID, Date, Mark)
STUDENT_CAREER_INFO_NOTIFICATION (StudentID, RequestDate, NumOfPassedExams)
STUDENT_CAREER_REQUEST (RequestID, StudentID, RequestDate)

A student requests her/his career information (i.e., insertion of a record in the table STUDENT_CAREER_REQUEST). Write the trigger to address the following activities.

The number of exams passed by the student must be calculated. The EXAM table contains the list of exams taken by students. Consider that an exam is passed if the mark is greater than or equal to 18.

If the student has not passed any exam, the trigger ends with an error. Otherwise, a new record must be inserted in the STUDENT_CAREER_INFO_NOTIFICATION table with the computed 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.

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 StudentCareerInfo
after insert on STUDENT_CAREER_REQUEST
for each row
Part A

```
DECLARE
NUMBER N;

BEGIN
    SELECT COUNT(*) INTO N
    FROM EXAM E
    WHERE E.StudentID = :NEW.StudentID AND Mark >= 18

    IF(N = 0) THEN
        raise_application_error();
     ELSE
        INSERT INTO S_C_I_N()
        VALUES(:NEW.StudentID, :NEW.RequestDate, N)
```