CSCI585 Fall '18 Midterm Exam

October 19th, 2018

CLOSED book and notes. No electronic devices. DO YOUR OWN WORK. Duration: 1 hour. If you are discovered to have cheated in any manner, you will get a 0 and be reported to SJACS. If you continue working on the exam after time is up you will get a 0. This document contains 12 pages including this one.

Signature: _____

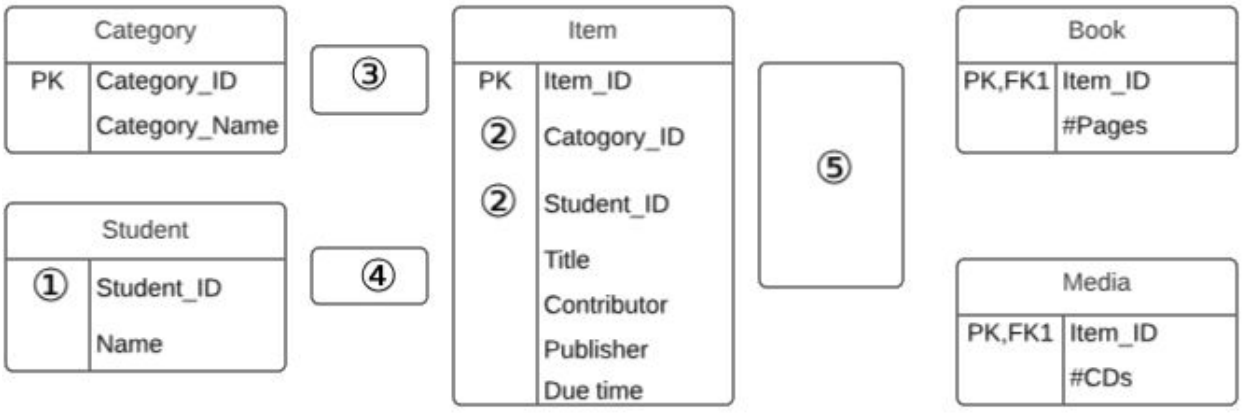
Problem Set	Number of Points
Q1	5
Q2	5
Q3	6
Q4	7
Q5	7
Q6	4
Q7	1
Total	**35**

Q1. (5 points total) ER MODELING

You are required to fill in the five blanks in the ER Diagram of a library database so it meets the following requirements. For blanks 1 and 2, please write the key type. For blanks 3, 4 and 5, please draw an edge to represent the relationship between its entities. Feel free to draw edges on the diagram, but please copy them on the blanks as well (to be graded).

The library has two types of items to check out, books and media. For each item, the database needs to record its unique Item_ID, title, contributor and publisher. Each item is also assigned one category like Science, Art, History and so on and each category is assigned to one or more items. Each item can be checked out by at most one student and the database should record who borrowed the item and due date for return. For a book, the database should record its number of pages. For a media item, the database should record number of CDs contained in it. All items should be in the database regardless whether they are available or have been already checked out.

Students can borrow zero, one or more items from the library. Each student has a unique Student_ID. The database should record all students' Student_IDs and names.



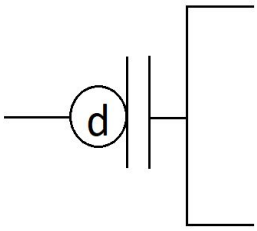
Solution

①PK

② FK1, FK2 (or just FK)

③

④



⑤

Rubrics: -1 for each wrong answer

Q2. (5 points total) SQL

A. (2 points) Write a brief description of what the following query does. The semantics should be straightforward, but you can make any reasonable assumptions (ie: Viterbi is a school within USC, etc.)

B. (3 points) Sketch the basic ER diagram/schema, show entities, attributes, and connections between them (relationships). Table names are: uscstudent, course, coursedescription, uscschool, semester, and studentsemesterenrollment.

```
SELECT stu.student_id, stu_fname, stu_lname, stu_email, totalunits
FROM   uscstudent stu
JOIN (
  SELECT uscstudent.student_id, Sum(course.course_numofunits) AS totalunits
  FROM (
    SELECT *
    FROM   studentsemesterenrollment sse
    JOIN uscstudent scs ON ( sse.student_id = scs.student_id )
    JOIN semester sem ON ( sse.semester_id = sem.course_id )
    ) sem
  JOIN course c ON sem.semester_code = c.semester_code
  JOIN coursedescription cd ON c.course_id = cd.course_id
  JOIN uscschool sch ON sch.school_id = cd.school_id
  WHERE uscschool.school_name = 'VITERBI'
  AND semester.semester_date BETWEEN '01-JAN-18' AND '31-DEC-18'
  GROUP BY uscstudent.student_id
  ) tommy ON stu.student_id = tommy.student_id
WHERE totalunits = (
  SELECT Max(totalunits)
  FROM (
    SELECT uscstudent.student_id,Sum(course.course_numofunits) AS totalunits
    FROM (
      SELECT *
      FROM   studentsemesterenrollment sse
      JOIN uscstudent scs ON ( sse.student_id = scs.student_id )
      JOIN semester sem ON ( sse.semester_id = sem.course_id )
      ) sem
    JOIN course c ON sem.semester_code = c.semester_code
```

```
    JOIN coursedescription cd ON c.course_id = cd.course_id
    JOIN uscschool sch ON sch.school_id = cd.school_id
    WHERE uscschool.school_name = 'VITERBI'
    AND semester.semester_date BETWEEN '01-JAN-18' AND '31-DEC-18'
    GROUP BY uscstudent.student_id
    )
);
```

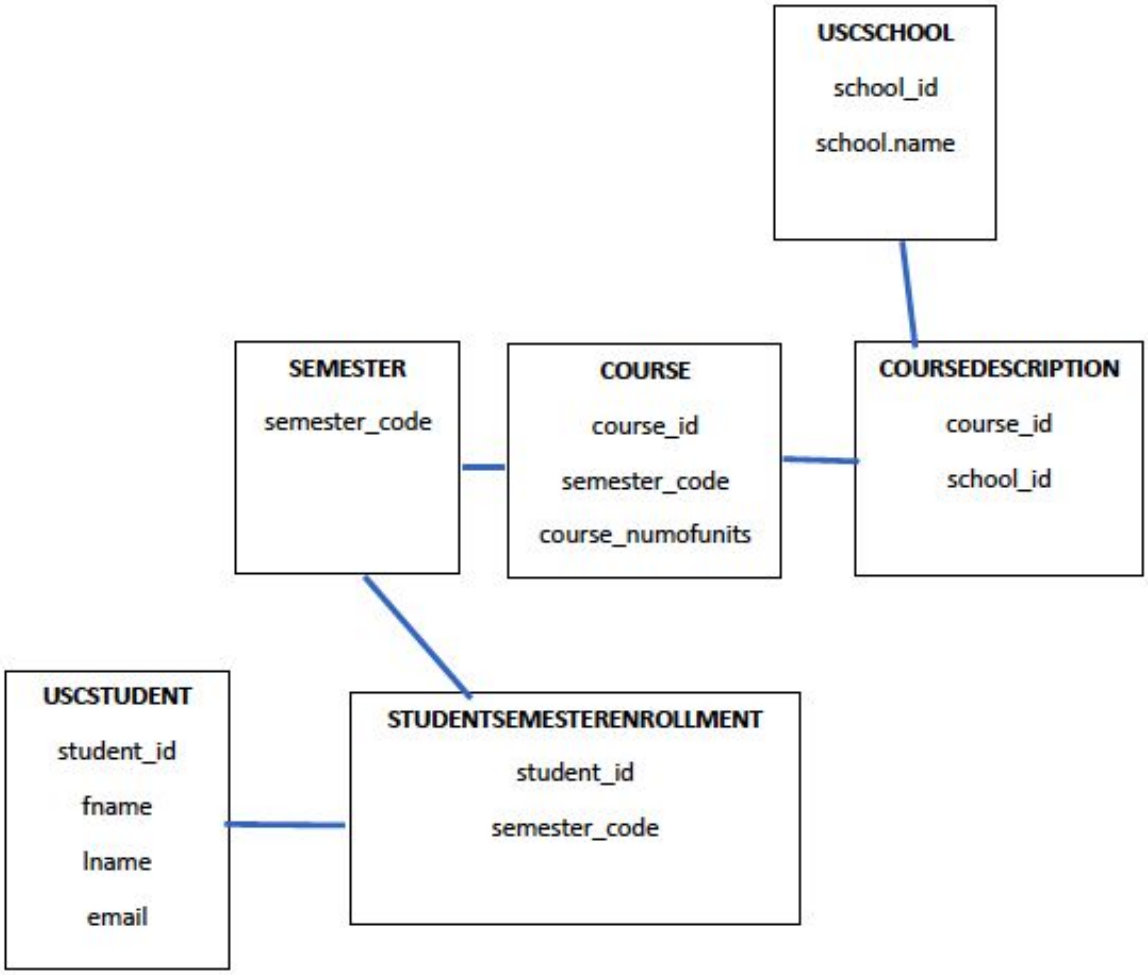
Q2. Solution

This is a query to display the student id, student first name, student last name, e-mail, and total course units taken for the student who took the most Viterbi school classes between January 1, 2018, and December 31, 2018.

The following sub query:
FROM (SELECT * FROM studentsemesterenrollment sse JOIN uscstudent scs ON (sse.student_id = scs.student_id) JOIN semester sem ON (sse.semester_id = sem.course_id)) sem
Is a bridge table that links the uscstudent and semester tables with M:N relationship. The rest should be clear with the following diagram:

Rubrics

The query displays **student information** for the student who took the **most Viterbi school classes (1 point)** between **January 1, 2018, and December 31, 2018 (1 point).**



Rubrics :

+2 All 6 entities are present. (-1 if half or more of the entities are missing)

+1 Correctly linking most of the entities

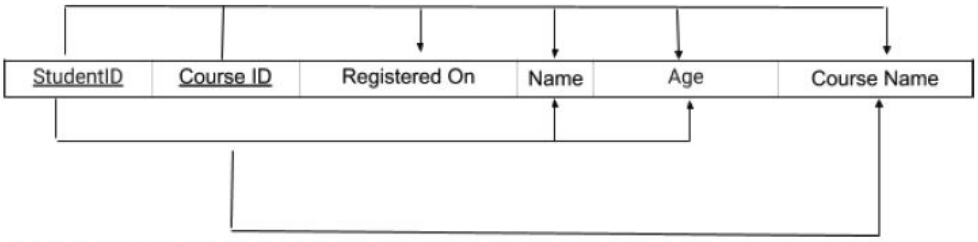
Q3. (6 points total) NORMALIZATION

Show dependency diagram and normalize the following table in 3 NF.

StudentID	Name	Age	Course ID	Course Name	Registered On
12	Alex	19	CSCI 511	C++	08/11/2018
			CSCI 510	Java	08/12/2018
123	Bin	20	CSCI 511	C++	08/05/2018
			CSCI 670	Algorithms	08/05/2018
32	Young	18	CSCI 550	Data Structures	08/15/2018
			CSCI 511	C++	08/11/2018
			CSCI 585	Database Systems	08/11/2018
133	Tracy	20	CSCI 520	Math	08/09/2018
			CSCI 510	Java	08/09/2018

Solution

Dependency Diagram



(StudentID, Course ID) → (Registered On, Name, Age, Course Name)
StudentID → (Name, Age)
CourseID → Course Name

3NF transformation

StudentID	Name	Age
12	Alex	19
123	Bin	20

32	Young	18
133	Tracy	20

Course ID	Course Name
CSCI 511	C++
CSCI 510	Java
CSCI 670	Algorithms
CSCI 585	Database Systems
CSCI 520	Math

StudentID	Course ID	Registered On
12	CSCI 511	08/11/2018
12	CSCI 510	08/12/2018
123	CSCI 511	08/05/2018
123	CSCI 670	08/05/2018
32	CSCI 550	08/15/2018
32	CSCI 511	08/11/2018
32	CSCI 585	08/11/2018
133	CSCI 520	08/09/2018
133	CSCI 510	08/09/2018

Rubrics

3 points for correct dependency diagram indicating full and two partial dependencies (-1 for each of the three dependencies missing).
3 points for listing attributes and primary keys for the three tables in 3NF (-1 for each incorrect table).
The question clearly asked for dependency diagram and tables in 3NF, so no need to bother with 1NF and 2 NF

Q4. (7 points) TRANSACTION MANAGEMENT

You are given the example tables that represent information of a factory, a retailer, and a customer. Each table has information of products and their counts. Also provided is a transaction log (on the next page), which contains 2 transactions: one represents production of 100 products from factory to retailer, the other represents a purchase of 150 products by a customer.

(1) Consider the case that locking is not properly implemented in the DBMS. Discuss whether the results of the two transactions are deterministic. (No need to consider other external transaction, but failure or roll back can happen).
(2) Consider that the DBMS in use is implementing a locking mechanism. Is two-phase locking required to ensure correctness of the two transactions? State your reasons. (No need to consider other external transactions, but failure or roll back can happen).
(3) Consider that pessimistic locking is implemented with two-phase locking protocol. Create a chronological list of locking, unlocking, and data manipulation activity that would occur during the completion of the two given transactions. (No step fails and no rollback happens).

Example tables:

FACTORY

PRODUCT_ID	PRODUCT_COUNT
42	1000

RETAILER

PRODUCT_ID	PRODUCT_COUNT
42	58

CUSTOMER

CUSTOMER_ID	PRODUCT_ID	PRODUCT_COUNT
1007	42	3

Q4. (Continued)
Transaction log

TRL_ID	TRX_NUM	PREV PTR	NEXT PTR	OPERATION DESCRIPTION
214	101	Null		****Start Transaction
216	101	214	225	Update "RETAILER" table on the row with PRODUCT_ID = 42 and add 100 to PRODUCT_COUNT
225	101	216	233	Update "FACTORY" table on the row with PRODUCT_ID = 42 and subtract 100 from PRODUCT_COUNT
233	101	225	Null	****End of Transaction
220	105	Null		****Start Transaction
227	105	220	239	Check that PRODUCT_ID = 42 in RETAILER table has PRODUCT_COUNT > 150 and wait until the condition is met.
239	105	227	243	Update "RETAILER" table on the row with PRODUCT_ID = 42 and subtract 150 to PRODUCT_COUNT
243	105	239	252	Update "CUSTOMER" table on the row with PRODUCT_ID = 42 and CUSTOMER_ID = 1007 and then add 100 to PRODUCT_COUNT
252	105	243	Null	****End of Transaction

Solution and Rubrics

(1) ***2 points. 1 point for identifying it to be non-deterministic. 1 point for providing a brief discussion.***

The result will be non-deterministic if no locking is implemented. Even if transaction 105 checks that PRODUCT_ID 42 should have at least 150 items before proceeding which seems to suggest that transaction 105 will not proceed before transaction 101 is done, it is still possible that one of the transaction is aborted that may cause inconsistencies, for example, consider the following events:

* TRX 101 starts
* TRX 101 updates RETAILER table, now RETAILER.PRODUCT_COUNT = 158 for

RETAILER.PRODUCT_ID = 42
* TRX 105 starts
* TRX 105 checks RETAILER table, find the RETAILER.PRODUCT_COUNT > 150 for RETAILER.PRODUCT_ID = 42, and proceed to the next step
* TRX 105 updates RETAILER table by subtracting 150 for RETAILER.PRODUCT_ID = 42, now RETAILER.PRODUCT_COUNT = 8
* TRX 101 failed to update FACTORY table in its next step, and the whole TRX 101 is reverted, now RETAILER.PRODUCT_COUNT = 58 again.
* TRX 105 updates CUSTOMER table, now that CUSTOMER.PRODUCT_COUNT = 153 for CUSTOMER.PRODUCT_ID = 42 and CUSTOMER.CUSTOMER_ID = 1007
The result of this example shows the customer successfully bought 150 items while the other tables are not updated properly. Hence, a proper locking mechanism is required.

(2) ***2 points. 1 point for identifying that 2-phase locking is required. 1 point for giving a valid reason. 0 if invalid reason and not mentioning that two phase locking is required.***
Two-phase locking protocol is required, because in TRX 101, updating of RETAILER table happens before updating FACTORY table, which has the possibility that the latter may fail and rollback the transaction (like the example given in the above answer). Without two-phase locking protocol, only locking one table may not ensure correctness once errors occur.

(3) ***2 points. 1 point for identifying that 2-phase locking is required. 1 point for giving a valid reason. 0 if invalid reason and not mentioning that two phase locking is required.***
Example of chronological events

Time	TRX_NUM	Event
1	101	Lock table RETAILER
2	101	Lock table FACTORY
3	101	Update table RETAILER by adding 100 to PRODUCT_COUNT of PRODUCT_ID = 42
4	101	Update table FACTORY by subtracting 100 to PRODUCT_COUNT of PRODUCT_ID = 42
5	101	Unlock table FACTORY
6	101	Unlock table RETAILER

7	105	Lock table RETAILER
8	105	Lock table CUSTOMER
9	105	Check table RETAILER of PRODUCT_ID = 42 that PRODUCT_COUNT > 150
10	105	Update table RETAILER by subtracting 150 to PRODUCT_COUNT of PRODUCT_ID = 42
11	105	Update table CUSTOMER by adding 150 to PRODUCT_COUNT with PRODUCT_ID = 42 and CUSTOMER_ID = 1007
12	105	Unlock table CUSTOMER
13	105	Unlock table RETAILER

Q5. (7 points) OPTIMIZATION

Consider the three following tables for an airport database and all attributes are neither indexed nor sorted.

- AIRPLANES (aid, brand, size), aid is the primary key.
- PILOTS (pid, name, age), pid is the primary key.
- LastFlight (aid, pid, date), aid and pid are a composite primary key.

And we want to execute the following SQL query:

```
SELECT P.name
FROM AIRPLANES A, PILOTS P, LastFlight L
WHERE A.aid = L.aid AND P.pid = L.pid
AND P.age < 35 AND A.brand = 'Boeing 737';
```

Assuming:

- There are 1,000 rows in AIRPLANES, 1,000 rows in PILOTS and 1,000,000 rows in LastFlight.
- PILOTS.age ranges from [30 to 49] (both inclusive) equally distributed in PILOTS.
- AIRPLANES.brand has 100 distinct values equally distributed in AIRPLANES.
- LastFlight has every combination of aid and pid.

Suppose the cost of running a SELECT operation is the number of rows in the source table and the cost of running a JOIN operation (Cartesian product) is the total rows of the two source tables. If we execute the query with following access plan, the cost will be 1,001,001,002,000.

Step	Operation	Cost	Estimated result rows
A1	Cartesian product (A, L)	1,001,000	1,000,000,000
A2	Cartesian product (A1, P)	1,000,001,000	1,000,000,000,000
A3	Select rows in A2 with all conditions	1,000,000,000,000	2,500*

* Here is how the number of resulting rows were estimated:
- The possibility of A.aid = L.aid is 1/1,000 for there are 1,000 different aid.
- The possibility of P.pid = L.pid is 1/1000 for there are 1,000 different pid.
- The possibility of an airplane brand = 'Boeing 737' is 1/100 for there are 100 different brands.
- The possibility of P.age < 35 is 5/20.
- Since all conditions are independent, the number of resulting rows in A3 is about:

1,000,000,000,000 * (1/1,000) * (1/1,000) * (1/100) * (5/20) = 2,500.

Do you have a better access plan to execute the query with a lower total cost?
Please fill the following form about your access plan.

- You don't have to fill all rows depending on how many steps are in your access plan.
- There should be enough room in each cell for you to answer and make corrections.

Q5. Solution

Best answer:

Step	Operation	Cost	Estimated result rows
B1	Select rows in P with ages < 35	1,000	250
B2	Select rows in A with brand = 'Boeing 737'	1,000	10
B3	Cartesian product (L, B2)	1,000,010	10,000,000
B4	select rows in B3 with A.aid = L.aid	10,000,000	10,000
B5	Cartesian product (B1, B4)	10,250	2,500,000
B6	select rows in B5 with P.pid = L.pid	2,500,000	2,500

Total cost: 13,512,260 (not required to answer)

-1 Point

Step	Operation	Cost	Estimated result rows
B1	Select rows in P with ages < 35	1,000	250
B2	Select rows in A with brand = 'Boeing 737'	1,000	10
B3	Cartesian product (L, B1)	1,000,250	250,000,000
B4	select rows in B3 with P.pid = L.pid	250,000,000	250,000
B5	Cartesian product (B2, B4)	250,010	2,500,000
B6	select rows in B5 with A.aid = L.aid	2,500,000	2,500

Total cost: 253,752,260 (not required to answer)

-2 Points

Step	Operation	Cost	Estimated result rows
B1	Select rows in P with ages < 35	1,000	250
B2	Select rows in A with brand = 'Boeing 737'	1,000	10
B3	Cartesian product (L, B2)	1,000,010	10,000,000
B4	Cartesian product (B1, B3)	10,000,250	2,500,000,000
B5	select rows in B4 with P.pid = L.pid and A.aid = L.aid	2,500,000,000	2,500

Total cost: 2,511,002,260 (not required to answer)

Rubrics

Plus Points

1. **According to the answer key**
2. **In case the overall solution is wrong, +1 for each correct line**

Negative Points

1. **If estimated result rows value is incorrect, -1. However, if estimated rows value is incorrect and calculations based on that are correct, we do not deduct points for each row of wrong estimates, we deduct it only for the first incorrect value.**
2. **If final row of answer is incorrect, -1**
3. **No points for rows having answer as 'Create index'**
4. **If calculations are missing, but the order of operations is correct (-2)**

Q6 (4 points) DISTRIBUTED DATABASES

List and explain characteristics of distributed databases (provide clear explanation and/or examples).

Solution

This question was designed to test student's understanding of distributed database systems.

One potential answer is to list and explain several DDBMS functions. For sample answer, please refer to chapter 12-4 on page 559 of class textbook.

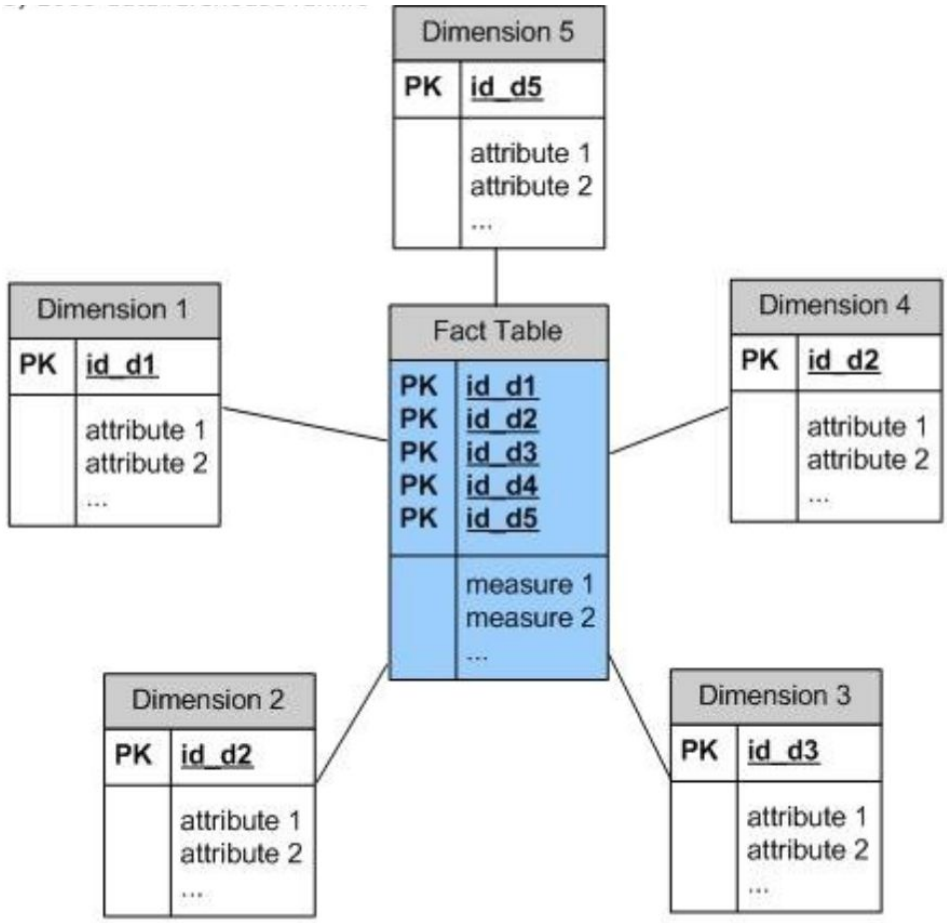
The alternate answer was to list and explain the distributed database transparency features: distribution, transaction, failure, performance, and heterogeneity transparencies. For sample answers, please refer to chapter 12-7 on page 564 of class textbook.

Rubrics

1. No fractional points
2. list and explain 4 or more DDBMS functions (-1 for each missing point)
3. OR list and explain 4 or more distributed database transparency features (-1 for each missing point)
4. -2 for listing all function/ feature name correctly but no rational explanation
5. ACID cannot get points

Q7. (1 point) BUSINESS INTELLIGENCE

What kind of schema does the ER diagram demonstrate?



Solution

Star schema

Rubrics: -1 for wrong answer

BONUS!!! (1 point)

What was your favorite part of Science documentary shown in class? If you have seen the entire movie, feel free to reference the part not displayed in class.

Solution
Your mileage may vary ☺

Rubrics: -1 for answers not related to the documentary. Also it should be a meaningful example discussed in the documentary.