

**Adama Science and Technology University
School of Electrical Engineering and Computing
Computer Science and Engineering Program**

Date: April 26, 2018

Time allowed: 2:15

Database Systems (CSE 3207) - Mid Exam (25%)

Name _____

Id. No. _____ Section/Group _____

General instruction:

- Read the questions carefully!!!
- Make sure the exam contains **3 parts and 6 pages** including the cover page and excluding blank page attached
- **Use only the space provided and for part II use the attached paper for part I and part II**
- Unreadable answer may worth ***no mark!***
- Using a ***mobile phone***, during exam time, is ***totally forbidden*** and anyone using it will be disqualified from the exam!!!
- **Writing unrelated answer will result a deduction from your mark**

Part I	Part II	Part III+Bonus	Total

Part I: Choose the best answer and write it in a block latter in the answer sheet provided [0.5 pts. each]

1. Which are the two ways in which entities can participate in a relationship?

[b]

- (a) Passive and active
- (b) Total and partial
- (c) Simple and Complex
- (d) All of the above

2. Which of the following operation is used if we are interested in only certain columns of a table?

- (a) PROJECTION
- (b) SELECTION [a]
- (c) UNION
- (d) JOIN

3. Disadvantage of a file based system to store data is

[a]

- (a) Difficulty in accessing data
- (b) Data redundancy and inconsistency
- (c) Data isolation
- (d) All
- (e) None

4. For each attribute of a relation, there is a set of permitted values called the _____ of that attribute?

- (a) Relation
- (b) Set
- (c) Schema
- (d) Domain [d]
- (e) None

5. Consider a relation R(A,B,C,D,E) with the following functional dependencies:

ABC → DE and D → AB

what is the number of superkeys of R?

[c]

- (a) 2
- (b) 7
- (c) 10
- (d) 12

ABC → DE

D → AB

Candidate keys of these FD's are ABC and CD

Number of super keys with ABC are ABC __ = 4

Number of superkeys with CD are __ CD = 8

Total number of superkeys = $n(ABC \cup CD) = n(ABC) + n(CD) - n(ABC \cap CD)$

since ABCD ,ABCDE are common in both keys so $n(ABC \cap CD) = 2$

$$= 4+8-2 = 10$$

Therefore total number of superkeys are 10

6. Process of analyzing relation schemas to achieve minimal redundancy and insertion or update anomalies is classified as

- (e) Functional dependency
- (f) Database modeling
- (g) Normalization [c]
- (h) Decomposition

6. Property of normalization of relations which guarantees that functional dependencies are represented in separate relations after decomposition is classified as

- (a) Non additive join property
- (b) independency reservation property
- (c) dependency preservation property
- (d) additive join property

7. Joining property which guarantees that spurious tuple generation problem is not created after decomposition is called

- (a) lossless join property
- (b) nonadditive join property
- (c) additive join property
- (d) both a and b

8. Indexes which specifies address of records on disk with a physical pointer are

- (a) structural index
- (b) hashing index
- (c) physical index [c]
- (d) logical index

9. Technique which is used to retrieve data from disk in form of continuous blocks of stream and eliminates seek time is classified as

- (a) concurrent buffering
 - (b) parallel buffering
 - (c) single buffering
 - (d) double buffering
- [d]

Part II: short answer questions: use onlu the space provided

10. Define Integrity Constraints. Why we need integrity constraints? Support your answer with example? **[3pts]**

Constraints are useful because they allow a designer to specify the semantics of data in the database. *Constraints* are the rules that force DBMSs to check that data satisfies the semantics.

Examples of referential integrity constraint in the Customer/Order database of the Company:

- Customer(**CustID**, CustName)
- Order(**OrderID**, CustID, OrderDate)

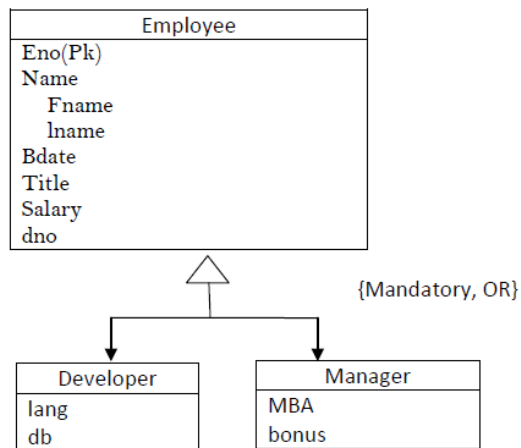
To ensure that there are no orphan records, we need to enforce referential integrity. An *orphan record* is one whose foreign key FK value is not found in the corresponding entity - the entity where the PK is located. Recall that a typical join is between a PK and FK.

requires that every table have a primary key; neither the primary key, nor any part of it, can contain null values

logical statements that state what data values are or are not allowed and which format is suitable for an attribute

requires that a foreign key must have a matching primary key or it must be null

11. Given the following enhanced Entity Relationship Diagram , indicate the schema(s) of the model.**[2pts]**



12. Consider the following tables: [5 pts.]

Employee (Emp_no, Name, Emp_city)

Company (Emp_no, Company_name, Salary)

(a) Write both SQL query and relational algebra to display Employee name and company name. [2pts]

Ans:- Select e.empno,name,company_name from Employee e, Company c
where e.Emp_no=c.Empno

Relational Algebra $\rightarrow \pi_{\text{name,company_name}} (\text{Employee} \Join \text{Company})$

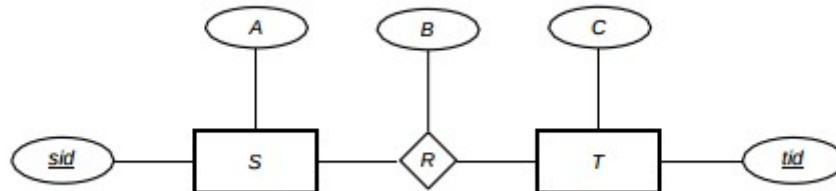
(b) Write a SQL query to display employee name, employee city, company name and salary of all the employees whose salary > the average number of salary [1pts]

select ename, city,name,sal from emp where empno in(select sal from emp)
and sal > (select avg(sal) from emp);

(c) Write a both the SQL query and relational algebra to display all companies and employees working in one of the companies [5pts]

Select empno,company from employee e, company c where e.empno in (select * from company)

13. Consider the following Entity-Relationship (ER) diagram. [5 pts]



Suppose we wish to implement this diagram in a relational database using three

tables, S(sid, A), T(tid, C), and R(· · ·). Describe the **schema** you would use for

R depending on the cardinality of the relationship.

(a) When R is a many-to-many relationship between S and T. [1 marks]

(b) When R is a one-to-many relationship between S and T. [1 marks]

(c) When R is a many-to-one relationship between S and T. [1 pt.]

(d) When R is a one-to-one relationship between S and T. [1 pt.]

(e) Write the DDL command to create table S and T if the two relations are in one-to-one relationship taking the referential in to consideration[2 pt.]

14. What are NULL values in SQL, and with what problems are they associated?[1 pts.]

Ans. Why NULL Values Should Not Be Used in a Database Unless Required

Null (or NULL) is a special marker used in [Structured Query Language](#) to indicate that a data value does not exist in the database. Introduced by the creator of the [relational](#) database model, [E. F. Codd](#), [SQL](#) Null serves to fulfil the requirement that all *true relational database management systems (RDBMS)* support a representation of "missing information and inapplicable information". Codd also introduced the use of the lowercase [Greek omega](#) (ω) symbol to represent Null in [database theory](#). In SQL, `NULL` is a [reserved word](#) used to identify this marker.

This should not be confused with a value of 0. A null value indicates a lack of a value — a lack of a value is not the same thing as a value of zero in the same way that a lack of an answer is not the same thing as an answer of "no". For example, consider the question "How many books does Adam own?" The answer may be "zero" (we *know* that he owns *none*) or "null" (we *do not know* how many he owns). In a database table, the column reporting this answer would start out with no value (marked by Null), and it would not be updated with the value "zero" until we have ascertained that Adam owns no books.

SQL null is a state, not a value. This usage is quite different from most programming languages, where [null value](#) of a reference means it is not pointing to any object.

Then you set up a database (at least in MS SQL Server) you can flag a field as allowing NULL values and which default values to take. If you look through people's DB structures, you'll see that a lot of people allow NULL values in their database. This is a very bad idea. I would recommend never allowing NULL values unless the field can logically have a NULL value (and even this I find this only really happens in DATE/TIME fields).

NULL values cause several problems. For starters, NULL values are not the same as data values. A NULL value is basically an undefined values. On the ColdFusion end, this is not terrible as NULL values come across as empty strings (for the most part). But in SQL, NULL and empty string are very different and act very differently. Take the following data table for example:

id	name
1	Ben
2	Jim
3	Simon
4	<NULL>
5	<NULL>
6	Ye
7	

8
9 Dave
10

This table has some empty strings (id: 7, 8, 10) and some NULL values (id: 4, 5). To see how these behave differently, look at the following query where we are trying to find the number of fields that do not have values:

Part III: case based question: write your answer in the blank sheet attached sequentially and label your answer with the question number.

ASTU student union is planning to organize half life football competition between 3rd year students in each program in the next semester. For this event the union wants to develop a database system to keep track of all events during the match period and you are selected to design the database.

Requirements

A player can only play in one team but may participate in all match at most. Where as, each match, which will be held at a specific time and date, will involve two teams. On the other hand, for each match all the events that can happen such as fouls, red cards, yellow cards, goal scorer... must be must be recorded in relation with the players and team.

The score boards of all the matches must be recorded so as to generate the fixture of the competition

Bellow are list of entities and and a partial list of attributes

- ➔ match(MID, time, date, Team1_ID, Team2-ID)
- ➔ players(PID, Pname, position, squadNumber)
- ➔ teams(TID, ProgramName)
- ➔ events(EID, MatchID, PlayerID, event)
- ➔ fixture (FID, Team1_ID, Team2_ID, MatchDate, Team1_Score, Team1_Score, Rank)

Consider the above scenario and answer the following questions

1. which entities are strong and which are weak?**[1pts]**

2. Identify the primary key of fixture table? Explain why you choose it? **[2pts]**
3. Draw the complete ER-diagram? **[3pts]**
4. Modify the above schema by adding a foreign key attribute for the tables having relationship? **[1pts]**
5. Check each schema and make sure each of them are normalized up to 3rd normal form. If not normalize it? **[2pts]**

Bonus [2pts]

What is the objective of this course?

Q4.Ans

ABC->DE

D->AB

Candidate keys of these FD's are ABC and CD

Number of super keys with ABC are ABC _ _ =4

Number of superkeys with CD are _ _ CD_ =8

Total number of superkeys = $n(ABC \cup CD) = n(ABC) + n(CD) - n(ABC \cap CD)$

since ABCD ,ABCDE are common in both keys so $n(ABC \cap CD) = 2$

$= 4 + 8 - 2 = 10$

Therefore total number of superkeys are 10

5. For each attribute of a relation, there is a set of permitted values, called the _____ of that attribute.

- a) Domain
- b) Relation
- c) Set
- d) Schema

View Answer

Answer: a

Explanation: The values of the attribute should be present in the domain. Domain is a set of values permitted .