International Institute of Information Technology, Hyderabad
(Deemed to be University)

CS4.301: Data and Applications
IIIT Hyderabad Question cum Answer Booklet
End Semester Examination

Max. Time: 2 Hrs   **Max. Marks:45**

Roll No: _____   **Programme: _____**   **Date of Exam: _____**

Room no: _____ Seat No: _____   **Invigilator's Signature: _____**

Instructions about the exam

1. Answer all questions. There are seven questions.
2. Make appropriate assumptions.

Additional sheet for rough work will not be provided

Marks Table (To be filled by the Examiner)

Question No / Marks							Name of the Examiner who marked
1							
2							
3							
4							
5							
6							
7							

General Instructions to the students

1. Place your Permanent / Temporary Student ID card on the desk during the examination for verification by the Invigilator.
2. Reading material such as books (unless open book exam) are not allowed inside the examination hall.
3. Borrowing writing material or calculators from other students in the examination hall is prohibited.
4. If any student is found indulging in malpractice or copying in the examination hall, the student will be given 'F' grade for the course and may be debarred from writing other examinations.

Best of Luck
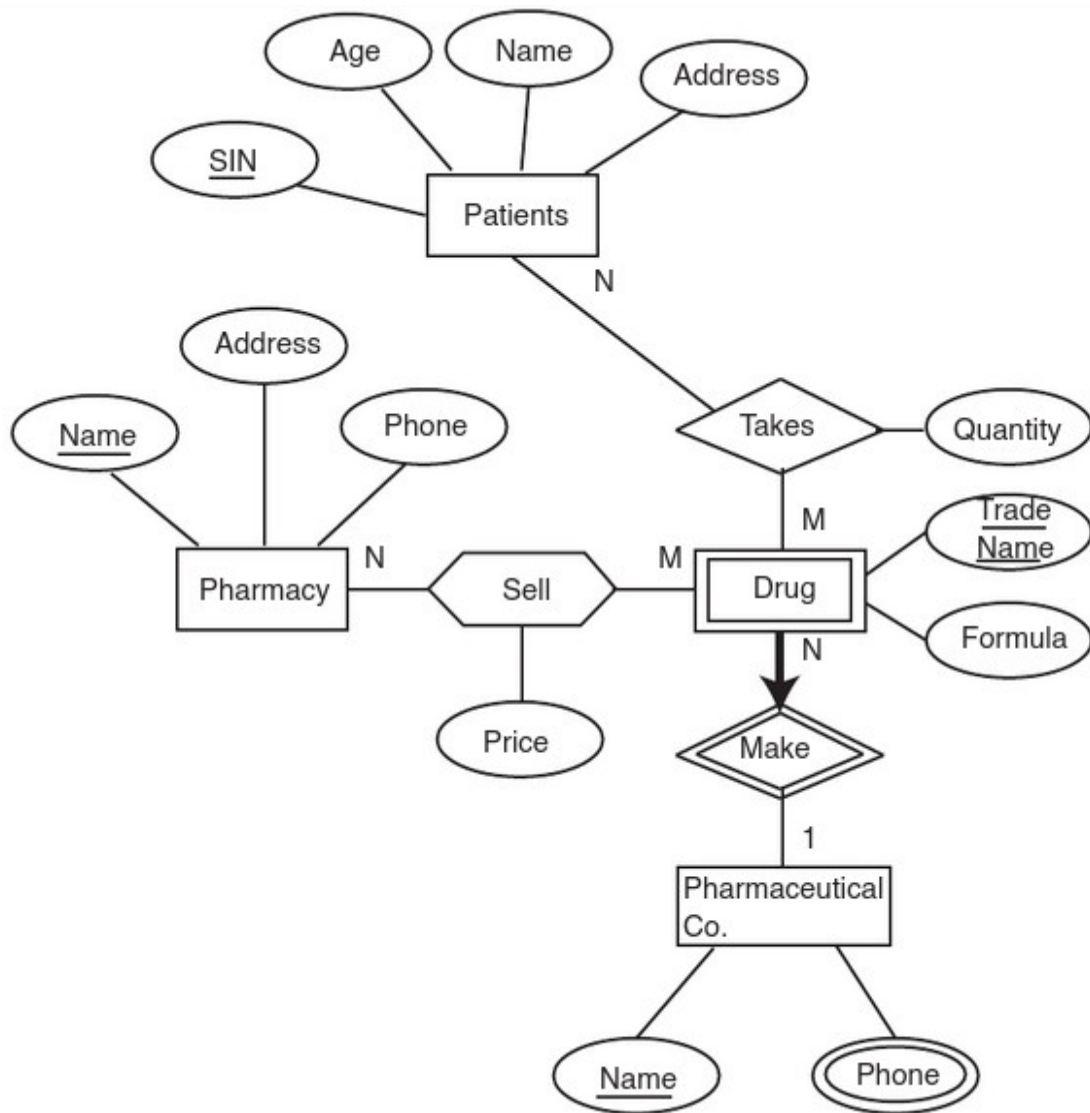
Q1.   Modify the model (by adding to the Figure, **not** by drawing another figure) so that you can represent the following:

- Each patient has to have one and only one primary physician. Each physician has at least one patient. We want to know at least the specialty and the date of entry into the profession of each physician.

- Instead of modelling only the fact that a patient takes certain drugs, model the fact that a patient takes certain drugs that are prescribed by a physician and the prescription date.

- Pharmaceutical companies have long-term contracts with pharmacies. A pharmaceutical company can contract with several pharmacies, and a pharmacy can contract with several pharmaceutical companies. For each contract we want to store a start date, an end date.

Q2.   **The holidays are coming and that means that Santa Claus is preparing to distribute toys to boys and girls of all ages around the world. Santa is joining the modern age and wants to use databases to keep track of his operation but needs some help.**

The following questions will involve the given database schema, used to keep track of toys, clients, and requests.

> **Toys (id, name, color, min_age, weight, number_available)**
> **Client (id, name, age, address, city)**
> **Request (client_id, toy_id)**

The underlined attributes are keys for each relation. The tables contain the following information:

- *Toys* records information about all of the toys in Santa's workshops. Each Toy has a unique integer id. Attributes *name* and *color* are strings that describe the toy; *min_age*, *weight*, and *number_available* are integers giving the minimum age a child should be to use the toy, the weight of the toy in pounds, and how many are currently available in Santa's workshops.

- *Client* stores information about each of the boys and girls in Santa's database. Each client has a unique integer *id*, strings with the client's *name*, *address*, and *city*, and an integer *age*. (To simplify things for the exam, we will ignore issues about different cities that might have the same name and assume that all cities have a unique name.)

- *Request* has a pair of integers indicating that a particular client has requested a particular toy.

You should assume that all entries in all of the tables are not null.

Santa has some old programs that retrieve information from an earlier version of the database. In that earlier database there was a table that contained information about clients and toy requests with this schema:

> **ToyList (client_id, client_name, client_age, toy_id)**

We want to define a SQL view to be used by the old programs to read data from the current tables.

a) **Complete the following SQL view definition to define the ToyList view as an appropriate query on the existing Toys, Client, and Request tables:**

```
CREATE VIEW ToyList AS
```

b) Now suppose we have the following query that was written to use the old table, and now will be executed using the view defined in part (a):

> SELECT client_name, client_age
> FROM ToyList
> WHERE client_age > 2
> ORDER BY client_name

Show how this query would be expanded and processed by a SQL database engine using the view defined in the first part (i.e. write the expanded query that references the underlying database tables).

Try to simplify this expanded query to get one that does not involve nested queries, if that is possible. Be sure to explain well enough (in 3-4 lines) so we can follow how you arrived at your final answer.

Q3.   There is a SQL Table (named *IncomeData*) with Two Columns *EmployeeID* and *Income*. You are required to find the second highest Salary earned by any Employee. Which of the following SQL Queries will be able to do this:

Note: All Income Values are Unique

a. SELECT Income FROM FinancialData GROUP BY Income HAVING Income < MAX(Income) ORDER BY Income DESC LIMIT 1;

b. SELECT MAX(Income) FROM IncomeData WHERE Income < (SELECT MAX(Income) FROM IncomeData);

c. SELECT Income FROM (SELECT DISTINCT Income FROM FinancialData ORDER BY Income DESC LIMIT 2) AS TopTwo ORDER BY Income ASC LIMIT 1;

d. SELECT MAX(Income) FROM FinancialData WHERE Income != MAX(Income);

Q4.   Given below are two tables:

Books

book_id	book_name	author
01_17	Jane Eyre	Charlotte Bronte
01_18	Pride and Prejudice	Jane Austen
02_10	The BFG	Roald Dahl
03_25	The Da Vinci Code	Dan Brown

Readers

reader_id	reader_name	book_id
2809	Juliet	01_17
2809	Juliet	01_18
2513	Blair	03_25
2290	Henry	01_18

(a)   What JOIN can you use to extract all useful rows with reader name, book name, and author, and avoid NULL values (without the use of any WHERE clause)?

     1. Inner, Right, Natural
     2. Left, Right, Natural
     3. Inner, Left, Full
     4. Full, Cross, Natural

(b)   Write the proper, syntactically correct SQL query using one of the selected JOIN to get the above information

(c)   Write the output you expect to obtain from the above query (**show populated table**)

Q5. Given the database schema R(A, B, C), and a relation r on the schema R, write an SQL query to test whether the functional dependency B → C holds on relation r.

Q6. Consider the following relation $R\,(\,A\,,\,B\,,\,C\,,\,D\,,\,E\,,\,F\,)$ with no multi valued attribute.
Consider the following FDs to hold: B -> EBF; E -> D, D -> C, BD -> E.

a) What is/are the candidate keys of R?   What normal form(s) does R satisfy?

b) Which of the following FDs hold for the table, truncate the RHS to make the FD valid in case it is not already valid(RHS can be empty, truncate as less as possible). Show derivation:

   a. BC -> EDA
   b. AB -> FCA
   c. AE -> C
   d. D -> E

Q7. **Show that every schema consisting of exactly two attributes must be in BCNF regardless of the given set *F* of functional dependencies.**

(Please give a logical, concrete and general proof. Providing or discussing examples does **not** constitute a proof, neither does providing only an intuitive explanation. No partial marks will be awarded for these approaches)

Rough Work