QUESTION STYLES

Faculty of Applied Science
Department of: Information Technology
Course: Database Systems I

Q. Fill in the blanks with correct words.

A. Cardinality of the relationship between Student and Course tables is ……**many-to-many**…….

B. ……**end users**… are people that the database is designed to serve the information needs of them.

C. ……**structured data**…… is type of data that fits neatly into data tables.

Q. Answer the following questions.

A. Explain the difference between structured data and unstructured data.

- **Structured data is data that fits neatly into data tables and includes discrete data types such as numbers, short text, and dates.**
- **Unstructured data doesn't fit neatly into a data table because of its size or nature: for example, audio and video files and large text documents.**

B. Name three type of cardinality of relationships in database systems. Take example of each type.

One-to-One → Example: One **Student** can have only One **Contact information**.

One-to-Many → Example: One **Department** can have Many **Students**.

Many-to-Many → Example: Many **Students** can have Many **Courses**.

Q. Choose the correct answer.

1. Cardinality of the relationship between Student and Course tables is …………………. .

A. one-to-one **B. many-to-many**

C. many-to-one D. one-to-many

2. The database is designed to serve the information needs of ………………… .

A. program developers B. database administrator

C. end-users D. none of them

Q. Write required SQL query/command for each part, according to the following tables.

deptName	building	NumberOfStudents
English	4	250
Dentistry	4	300
IT	1	220

Department Table

stuID	stuName	deptName	credits
20	Saeed	IT	55
21	Kawa	IT	36
22	Lana	English	60

Student Table

A. Write an SQL command to enter a new record in student table with these values →(23, Karez, Dentistry, 70).

INSERT INTO Student
VALUES (23, 'Karez', 'Dentistry', 70);

B. Write an SQL command to delete information of students in IT department.

DELETE FROM student
WHERE deptName = 'IT';

C. Write an SQL query to find the number of departments in each building.

SELECT building, count(*)
FROM department
GROUP BY building;

D. Write an SQL query to show ID, name and creditsStatus of students according to the following conditions:

High credits : credits > 50

Low credits : credits <= 50

SELECT stuID, stuName, **iif**(credits>50, 'High credits' , 'Low credits') **AS** creditsStatus
FROM student ;

Q. Write the required SQL query/command for each part, according to the following tables.

deptCode	deptName	NumberOfStudents
E55	English	250
D19	Dentistry	300
I78	IT	220

Department Table

stuID	stuName	deptCode	stage	credits
20	Saeed	I78	second	55
21	Kawa	I78	second	36
22	Lana	E55	third	60

Student Table

A. Write an SQL command to find total number of students in all departments.

```
SELECT sum(NumberOfStudents)
FROM department;
```

B. Write an SQL command to add 10 credits to the credits of second stage students.

```
UPDATE   student
SET      credits = credits + 10
WHERE    stage = 'second';
```

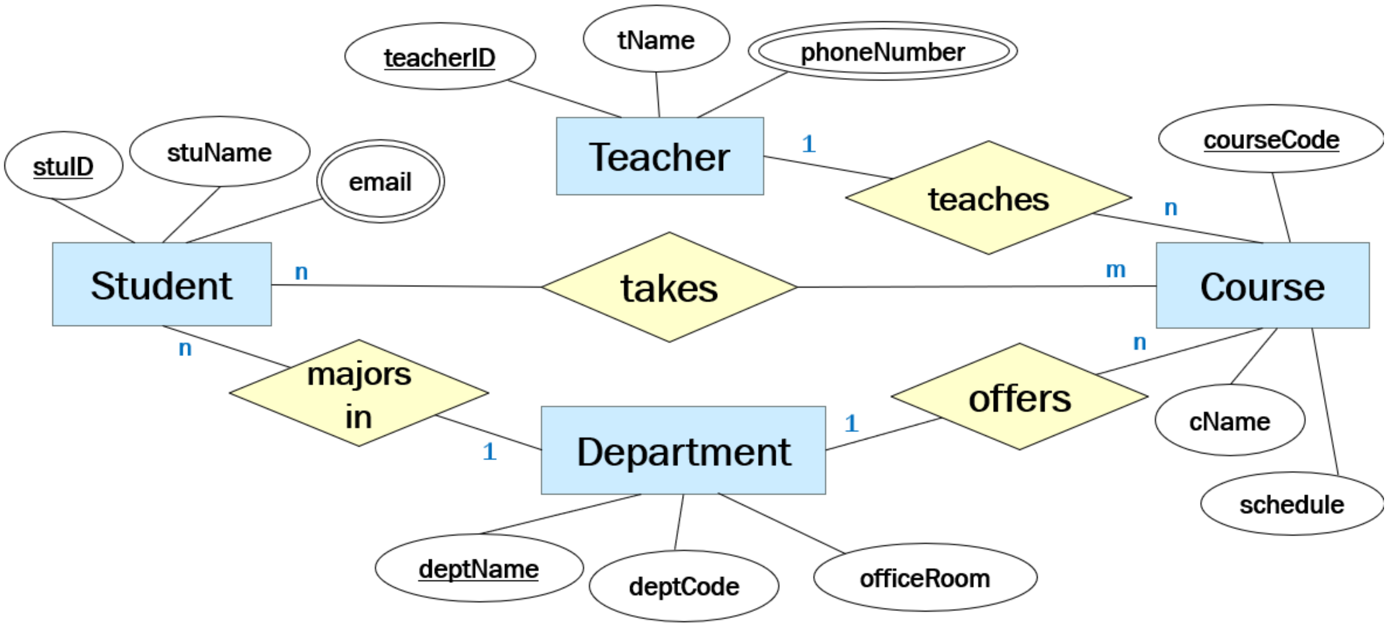
C. Write an SQL query to retrieve average credits of students in IT department.

```
SELECT avg(credits)
FROM student, department
WHERE student.deptCode = department.deptCode  AND  deptName = 'IT';
```

Q. Draw an Entity-Relationship Diagram for **Bank** database, considering the following requirements:

- Tables:
  Student (**<u>stuID</u>**, stuName, email)
  Teacher (**<u>teacherID</u>**, tName, phoneNumber)
  Department (**<u>deptName</u>**, deptCode, officeRoom)
  Course (**<u>courseCode</u>**, cName, schedule)

- At least four relationship and type of each relationship (cardinality).



Q. For the given table, specify all its possible super keys, candidate keys, alternate keys and primary key.

bookID	ISBN	bookName	bookPrice
12	5610888110	Basic C#	20
23	1986622017	Adv. Java	50
24	2677709914	SQL	45

Super Keys:

1. {bookID}
2. {ISBN}
3. {bookID, ISBN}
4. {bookID, bookName}
5. {bookID, bookPrice}
6. {ISBN, bookName}
7. {ISBN, bookPrice}
8. {bookID, ISBN, bookName}
9. {bookID, ISBN, bookPrice}
10. {ISBN, bookName, bookPrice}
11. {bookID, ISBN, bookName, bookPrice}

Candidate Keys:

1. {bookID}
2. {ISBN}

Primary Key:

{bookID}

Alternate Key:

{ISBN}

Q. Normalize the given table by applying Third Normal Form.

Teacher ID	courseName	course Type	teacher Name	Phone Number	numberOf StudentIn Course	student Feedback Score	feedback Status
20	Web Design	Core	John	312-555-1212, 312-561-1901	40	3	Neutral
21	Python	Core	Kate	310-677-1145	47	4	Good
21	Algorithms	Basic	Kate	310-677-1145	35	5	Very Good
20	Database	Core	John	312-555-1212, 312-561-1901	50	4	Good
24	English	Basic	William	122-331-0876	55	2	Bad

Answer:

The result of the third normal form is the following four tables.

TeacherID	courseName	courseType	teacherName	phoneNumber	numberOfStudent InCourse	student FeedbackScore	feedback Status
20	Web Design	Core	John	312-555-1212, 312-561-1901	40	3	Neutral
21	Python	Core	Kate	310-677-1145	47	4	Good
...	...	...	...	....	...	...	...

1NF

TeacherID	courseName	courseType	teacherName	phoneNumber	numberOfStudent InCourse	student FeedbackScore	feedback Status
20	Web Design	Core	John	312-555-1212	40	3	Neutral
20	Web Design	Core	John	312-561-1901	40	3	Neutral
...	...	...	...	....	...	...	...

2NF

Teacher ID	teacher Name	phone Number

Table 1

Course Name	Course Type	numberOfStudent InCourse

Table 2

Teacher ID	course Name	student FeedbackScore	feedback Status

3NF

Teacher ID	course Name	student FeedbackScore

Table 3

student FeedbackScore	feedback Status

Table 4

As a result of the third normal form, the original table is converted into four tables: Table1, Table2, Table3, and Table4.

Q. Normalize the given table by applying Second Normal Form.

NewClass

classNo	stuId	lastName	facId	schedule	room	grade
ART103A	S1001	Smith	F101	MWF9	H221	A
ART103A	S1010	Burns	F101	MWF9	H221	
ART103A	S1006	Lee	F101	MWF9	H221	B
CSC201A	S1003	Jones	F105	TuThF10	M110	A
CSC201A	S1006	Lee	F105	TuThF10	M110	C
HST205A	S1001	Smith	F202	MWF11	H221	

Answer:

It is already in 1NF because all attributes are single-valued.

The result of the second normal form is the following three tables.

Register

classNo	stuId	grade
ART103A	S1001	A
ART103A	S1010	
ART103A	S1006	B
CSC201A	S1003	A
CSC201A	S1006	C

Stu

stuId	lastName
S1001	Smith
S1010	Burns
S1006	Lee
S1003	Jones

Class2

classNo	facId	schedule	room
ART103A	F101	MWF9	H221
CSC201A	F105	TuThF10	M110
HST205A	F202	MWF11	H221

Q. By having the Employee table, write an SQL query to find the minimum amount of salary of employees hired after 2015 in each position, if the minimum salary at that position is greater than $2000. Arrange the final result in alphabetical order of position.

EmpID	HiringDate	Position	Salary
011	2014	Engineer	4000
012	2016	Technician	1500
013	2020	Technician	1600
014	2017	Manager	7000
015	2020	Engineer	2500
016	2013	Engineer	3000
017	2011	Manager	4500
018	2018	Manager	5000
019	2018	Engineer	3000

```
SELECT   Position , min(Salary)  AS  minimum_salary
FROM     Employee
WHERE    HiringDate > 2015
GROUP BY Position
HAVING   min(Salary) > 2000
ORDER BY Position;
```