QUESTION 1

a) Contrast define and construct in database management systems functionality.

Ans:- Contrast define and construct in database management systems functionality:

i. Define: In the context of database management systems (DBMS), "define" refers to the process of specifying the structure and organization of a database. This involves defining the tables, their attributes (columns), data types, keys, and relationships between tables. It's essentially creating the blueprint or schema for the database.

ii. Construct: "Construct" in DBMS functionality involves populating the defined database with actual data. This includes inserting, updating, or deleting records in the database tables. In essence, it's the process of filling the database with real-world data that conforms to the predefined structure.

b) Explain the following advantages of database.

i. Enforcing integrity constraints on the database

ii. Restricting unauthorized access to data

iii. Representing complex relationships among data

Ans:- Advantages of a database:

i. Enforcing integrity constraints on the database: A database management system allows you to define integrity constraints such as primary keys, foreign keys, unique constraints, and check constraints. These constraints ensure the accuracy and consistency of data in the database. For example, a primary key constraint ensures that each record in a table has a unique identifier, preventing duplicates.

ii. Restricting unauthorized access to data: DBMS provides security features to restrict unauthorized access to sensitive data. User authentication and authorization mechanisms control who can access, modify, or delete data in the database. This enhances data security and privacy, reducing the risk of data breaches.

iii. Representing complex relationships among data: Databases are excellent at representing complex relationships among data through features like foreign keys and join operations. This allows for efficient querying and reporting on interconnected data, making it easier to derive valuable insights and analyze relationships.

c) Clean Carpet is opening the 3' branch, and has come to you for consultation. Upon the investigation, you found that Clean Carpet Is keeping track of their sales and service records manually by multiple staffs in each branch, which potentially lead to errors in their records. Would you recommend Clean Carpet to use database management system? Discuss THREE (3) reasons.

Ans:- Recommending the use of a database management system (DBMS) for Clean Carpet:

1. Data Accuracy: A DBMS ensures data accuracy by enforcing data integrity constraints. With manual record-keeping, errors are more likely to occur due to human mistakes. Using a DBMS would reduce data errors, ensuring Clean Carpet's sales and service records are reliable.

2. Data Consistency: A DBMS helps maintain consistent data across multiple branches. Inconsistent data between branches can lead to confusion and inefficiencies. A centralized database would ensure data consistency and facilitate data sharing between branches.

3. Improved Efficiency: Managing data manually by multiple staff members is time-consuming and prone to duplication of effort. A DBMS streamlines data entry, retrieval, and reporting processes, saving time and reducing the workload on staff.

d) Contrast between database schema and database state, with a diagram for each.

Ans:- Contrast between database schema and database state with diagrams:

Database Schema:

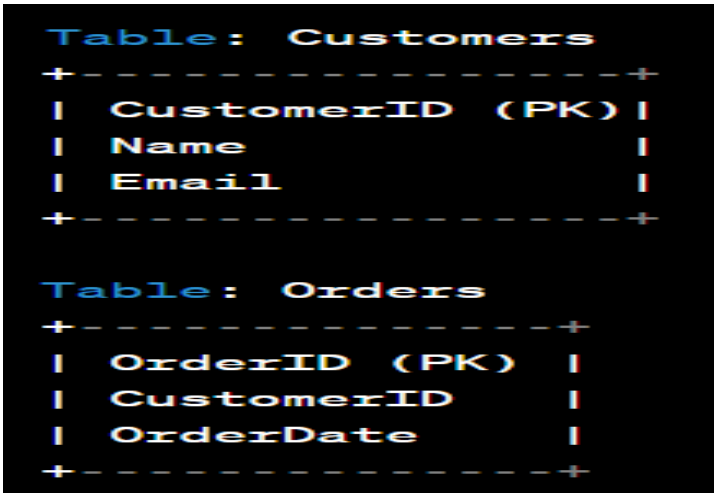
- A database schema defines the logical structure of a database.

- It includes the structure of tables, attributes (columns), data types, keys, and relationships.

- A schema is static and doesn't change unless intentionally modified.

- It serves as a blueprint for organizing data.

- Here's a simplified diagram of a database schema:



Database State:

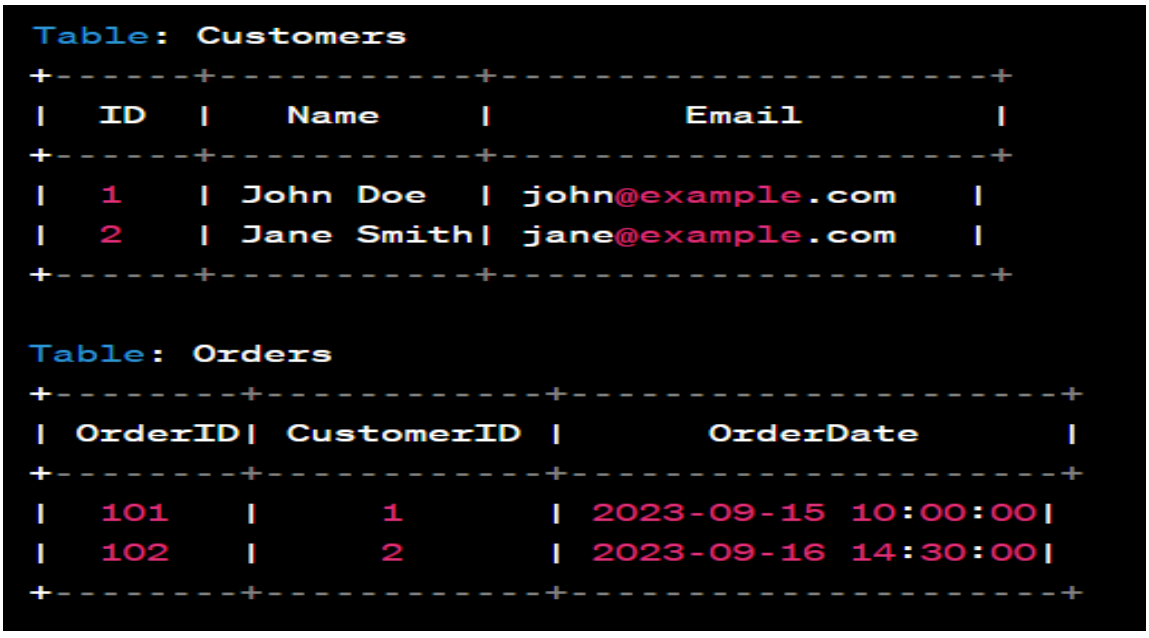
- A database state represents the actual data stored in the database at a specific point in time.

- It changes as data is inserted, updated, or deleted.

- A state is dynamic and reflects the current content of the database.

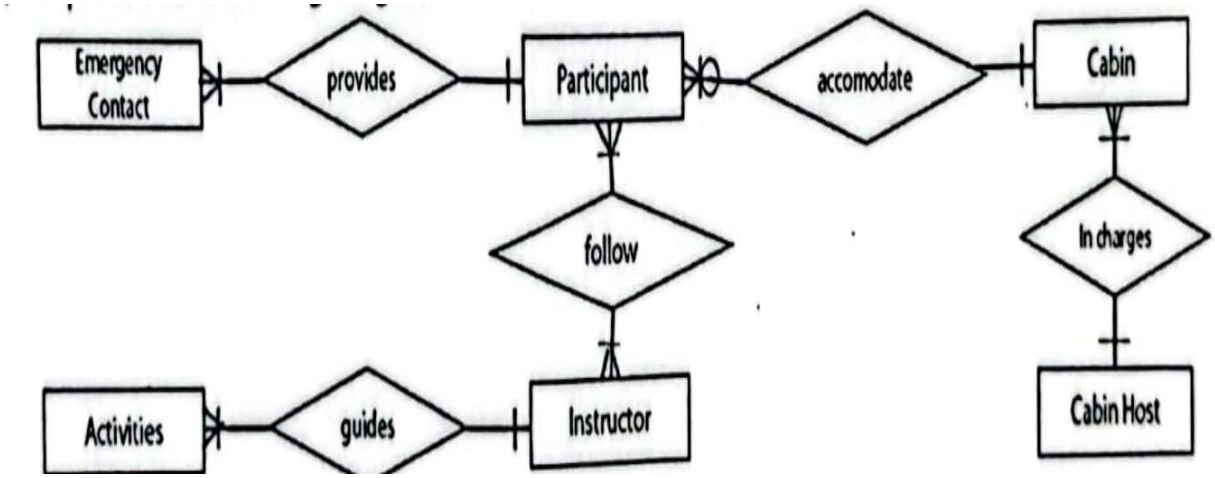
- It can be queried to retrieve real data.

- Here's a simplified diagram of a database state:



QUESTION 2

a) Explain the following diagram.



Ans:-

1. Emergency contact (One to Many) -- Provides -- (One) Participant:

- An emergency contact can be provided to one or more participants.
- Each participant must have exactly one emergency contact.

2. Participant (Zero or One or Many) – accomodate – (One) Cabin

- Participant can be accomodated in zero, one or many cabins.
- Each cabin must accomodated at least one participant.

3. Cabin (One or Many) – in charge – (One) Cabin Host

- Each cabin have one or many cabin host.
- Each cabin host must be in charged in one cabin.

4. Participant (One or Many) – follow – (One to Many) Instructor

- A participant can follow one or many instructors.
- An instructor can guide one or many participants.

5. Instructor (One) – guides – (One or Many) Activities

- An instructor can guide one or many activities.
- Each activity must have at least one instructor.

b) Draw an entity relationship diagram for the following statement, complete with the entities, key attributes with the primary keys underlined, relationships, and cardinalities.

In order to make a dinner reservation at the Aurore's (a well-known restaurant), a customer Is required to at least provide his/her name. The customer may reserve one or more tables, depending on the number of party. The customer may order one or more of the same or different meals.

Ans:- Creating an entity-relationship diagram (ERD) for the given statement involves identifying entities, their attributes, primary keys, relationships, and cardinalities.

Here's the ERD for the scenario described:

Entities:

1. Customer

2. Table

3. Reservation

4. Meal

Attributes and Primary Keys:

- Customer (CustomerID [Primary Key], Name)

- Table (TableID [Primary Key], Number)

- Reservation (ReservationID [Primary Key], CustomerID [Foreign Key], TableID [Foreign Key], PartySize, ReservationDate)

- Meal (MealID [Primary Key], Name, Price)

Relationships and Cardinalities:

1. Customer - Reservation (One-to-Many):

- A customer can make multiple reservations (One).

- A reservation is made by one customer (Many).

2. Reservation - Table (Many-to-One):

- A reservation can include one table (One).

- A table can be associated with multiple reservations (Many).

3. Customer - Meal (Many-to-Many):

- A customer can order multiple meals (Many).
- A meal can be ordered by multiple customers (Many).

QUESTION 3

The Tablet below is provided as a reference for the following questions.

Table 1: Unnormalized Form (UNF)

CustomerName	Address	Contact	Item	Price	Seller	Extension
Britney House	15 Lilly Dr.	012 555 5555	La Liberta Luggage	$36	Brandon	0121
Justin Dear	191 Madison St.	016 333 3333	Case Valker Luggage	$58	Molly	0122
Nick Hammer	201 Florida Ave.	019 777 7777	Case Valker Luggage	$58	Molly	0122
Nick Hammer	201 Florida Ave.	019 777 7777	La Liberta Luggage	$36	Brandon	0121

a) Apply first normalization form (I NF) to Table 1.

Ans:- 1NF requires that each column in a table must contain atomic (indivisible) values, and each row must be unique. In this case, there are some repeating groups, so we will create two separate tables: one for customers and one for orders.

Table 2: Customers

CustomerName	Address	Contact
Britney House	15 Lilly Dr.	012 555 5555
Justin Dear	191 Madison St.	016 333 3333
Nick Hammer	201 Florida Ave.	019 777 7777

Table 3: Orders

CustomerName	Item	Price	Seller	Extension
Britney House	La Liberta Luggage	$36	Brandon	0121
Justin Dear	Case Valker Luggage	$58	Molly	0122
Nick Hammer	Case Valker Luggage	$58	Molly	0122
Nick Hammer	La Liberta Luggage	$36	Brandon	0121

b) Apply the second normalization form (2NF) to your answer for Question 3a. Treat this question as an independent separate question, and provide the full answer. Construct all/full table(s), regardless whether there is no change to the table(s) from your answer for Question 3a.

Ans:- 2NF requires that each non-key attribute (columns other than the primary key) be fully functionally dependent on the entire primary key. In this case, we need to create a composite primary key in the "Orders" table and remove any partial dependencies.

Table 2: Customers

CustomerName	Address	Contact
Britney House	15 Lilly Dr.	012 555 5555
Justin Dear	191 Madison St.	016 333 3333
Nick Hammer	201 Florida Ave.	019 777 7777

Table 3: Orders

CustomerName	Item	Price	Seller	Extension
Britney House	La Liberta Luggage	$36	Brandon	0121
Justin Dear	Case Valker Luggage	$58	Molly	0122
Nick Hammer	Case Valker Luggage	$58	Molly	0122
Nick Hammer	La Liberta Luggage	$36	Brandon	0121

c) Apply the third normalization form (3NF) to your answer for Question 3b. Treat this question as an Independent separate question, and provide the full answer. Construct all/full table(s), regardless whether there is no change to the table(s) from your answer for Question 3b.

Ans:- 3NF requires that there should be no transitive dependencies between non-key attributes. In this case, we need to remove the transitive dependency between "Seller" and "Extension" in the "Orders" table.

Table 2: Customers

CustomerName	Address	Contact
Britney House	15 Lilly Dr.	012 555 5555
Justin Dear	191 Madison St.	016 333 3333
Nick Hammer	201 Florida Ave.	019 777 7777

Table 3: Orders

Item	Price	Seller
La Liberta Luggage	$36	Brandon
Case Valker Luggage	$58	Molly
Case Valker Luggage	$58	Molly
La Liberta Luggage	$36	Brandon

Table 4: OrderExtensions

Item	Extension
La Liberta Luggage	0121
Case Valker Luggage	0122
Case Valker Luggage	0122
La Liberta Luggage	0121

d) Apply the fourth normalization form (4NF) to your answer for Question 3c. Treat this question as an independent separate question, and provide the full answer. Construct all/full table(s), regardless whether there Is no change to the table(s) from your answer for Question 3c.

Ans:- 4NF is used to eliminate multi-valued dependencies. In this case, there are no multi-valued dependencies, so no further normalization is needed.

Table 2: Customers

CustomerName	Address	Contact
Britney House	15 Lilly Dr.	012 555 5555
Justin Dear	191 Madison St.	016 333 3333
Nick Hammer	201 Florida Ave.	019 777 7777

Table 3: Orders

Item	Price	Seller
La Liberta Luggage	$36	Brandon
Case Valker Luggage	$58	Molly
Case Valker Luggage	$58	Molly
La Liberta Luggage	$36	Brandon

Table 4: OrderExtensions

Item	Extension
La Liberta Luggage	0121
Case Valker Luggage	0122
Case Valker Luggage	0122
La Liberta Luggage	0121

The data is now in 4NF, and we have successfully normalized the original unnormalized table into separate tables while maintaining data integrity and eliminating redundancy.

QUESTION 4

The Table 2 below is provided as a reference for the following questions. Answers are expected to be in Structured Query Language (SQL).

Table 2: Student Table Definition

Field Name	Data Type	Primary Key
Patron ID	Number	Yes
Patron Name	Text- 255 characters max	No
Patron Contact Number	Text- 10 cgaracters max	No

a) Construct a patron table using SQL with the appropriate field names and properties.

Ans:- To construct a patron table in SQL, you can use the following SQL statement:

```
CREATE TABLE Patron (
  PatronID NUMBER PRIMARY KEY,
  PatronName TEXT(255),
  PatronContactNumber TEXT(10)
);
```

b) Insert the following data into the table using SQL

Patron ID	Patron Name	Patron Contact Number
45660	Damien Tan	0123456789

Ans:- To insert the given data into the table, you can use the following SQL statement:

```
INSERT INTO Patron (PatronID, PatronName, PatronContactNumber)
VALUES (45660, 'Damien Tan', '0123456789');
```

c) Assume the participant table has the following data.

Patron ID	Patron Name	Patron Contact Number
45660	Damien Tan	0123456789
21089	Larry Smith	0124463840
19876	Sarah Tan	0127896789
75832	Sandra Chow	0128787878

i)Create an SQL query to display only the Patron Name and the Patron Contact Number.

Ans:- To display only the Patron Name and Patron Contact Number:

```
SELECT PatronName, PatronContactNumber

FROM Patron;
```

ii) Create an SQL query to display the Patron Name and the Patron Contact Number where the Patron ID is 45660.

Ans:- To display the Patron Name and Patron Contact Number where the Patron ID is 45660:

```
SELECT PatronName, PatronContactNumber

FROM Patron

WHERE PatronID = 45660;
```

iii) Create an SQL query to display the Patron ID, Patron Name, and Patron Contact Number for all participants with the last name Tan.

iii) To display the Patron ID, Patron Name, and Patron Contact Number for all participants with the last name "Tan," you can use the following query:

```
SELECT PatronID, PatronName, PatronContactNumber

FROM Patron

WHERE PatronName LIKE '% Tan';
```

iv) Create an SQL query to display the Patron Name, where the Patron ID is either 19876 or 21089.

Ans:-To display the Patron Name where the Patron ID is either 19876 or 21089, you can use the following query:

```
SELECT PatronName

FROM Patron

WHERE PatronID IN (19876, 21089);
```