Q.No.1. LMG Sdn. Bhd is a train production company that assemble train couch for the local transportation company operating in Peninsular Malaysia. The management of this company decided to build a database to manage its operations. Currently, the business operations are managed manually and there is no database to integrate all the data which cause a delay in tracking of information and also in management decision making task.

a. Using this introductory description of LMG Sdn. Bhd, make a list of SEVEN (7) entities required to create a database for this company to maintain their data.

Ans:- List of SEVEN entities required to create a database for LMG Sdn. Bhd:

1. Train Models:- This entity would store information about the different train models produced by LMG, including specifications, features, and production details.

2. Parts Inventory:- This entity would track the inventory of train components and parts, including quantities, suppliers, and reorder levels.

3. Production Orders:- This entity would manage production orders for assembling train coaches, including order details, timelines, and status tracking.

4. Employees:- Store information about the company's employees, including their roles, contact information, and work schedules.

5. Customers:- Maintain a database of customer information, such as transportation companies, including contact details, contracts, and order history.

6. Maintenance Records:- Record maintenance and repair data for the train coaches, including schedules, maintenance history, and associated costs.

7. Financial Records:- Keep track of financial transactions, including revenue, expenses, payroll, and accounts payable/receivable.

b. List THREE (3) reports that LMG Sdn. Bhd could generate by using the entities suggested in 1(a) to improve its operations.

Ans:- THREE reports LMG Sdn. Bhd could generate using the entities suggested in 1(a):

1. Production Status Report:- This report would provide an overview of the current status of production orders, including the progress of each train coach assembly, any delays, and estimated completion dates. It would help in managing production timelines efficiently.

2. Inventory Replenishment Report:- This report would automatically generate reorder notifications for parts and components when their quantities reach predefined reorder levels. It ensures that LMG never runs out of essential parts, preventing production delays.

3. Financial Performance Analysis:- This report would provide insights into the company's financial health, including revenue, expenses, and profit margins. It can help in making informed financial decisions and identifying areas for cost reduction or revenue improvement.

c. Describe THREE (3) advantages of applying database systems for LMG Sdn. Bhd.

Ans:- THREE advantages of applying database systems for LMG Sdn. Bhd:

1. Data Centralization:- With a database system, all relevant data is centralized, making it easier to access and update. This eliminates the need for manual data management, reducing errors and redundancy.

2. Improved Data Accuracy:- Databases enforce data integrity constraints, ensuring that data is accurate and consistent. This reduces errors in operations, such as order processing and inventory management.

3. Enhanced Decision Making:- Access to real-time data and the ability to generate reports and analytics allows for better decision-making. LMG can make informed choices about production scheduling, resource allocation, and financial planning.

d. Describe TWO (2) integrity constraint that could be applied in the entities suggested in 1(a) with an example.

Ans:- TWO integrity constraints that could be applied in the entities suggested in 1(a) with an example:

1.Unique Constraint:

- Example: In the "Employees" entity, the "Employee ID" field should have a unique constraint. This ensures that each employee has a unique identifier, preventing multiple employees from having the same ID.

2. Referential Integrity Constraint:

- Example: In the "Production Orders" entity, there is a foreign key referencing the "Train Models" entity to specify which train model is being produced for each order. The referential integrity constraint ensures that a production order can only be associated with an existing train model in the "Train Models" entity, preventing orphaned records.

Q.NO.2. Read the following description carefully.

The Nepal Government Rail (NGC) is the first railway in Nepal which operates passengers train service. This 39-kilometre-long (24 mi) line allowed people from different areas of the country to reach Amlekhganj, and helped move heavy vehicles to Bhimphedi. It was then possible to reach Kathmandu from Bhimphedi on foot. The railway possessed seven steam locomotives, 12 coaches and 82 wagons.

It owns a substantial number of trains. Each train is allocated to a particular route although some routes may have several trains. Each train has a unique train number. It is important to store information about the seating capacity and the make/type of all the train.

Each route, distinguished by a route number, passes through a number of towns. Several routes may serve the same town. Information is available on the average number of passengers carried per day for each route.

Due to long travelling time, one or more drivers are assigned to each stage of a route, which corresponds to a journey through a town on a route. Drivers have an employee number, name, address and sometimes a telephone number.

a. **State appropriate assumptions and devise the corresponding Entity Relationship Model (ERD) for the above scenario.**

Ans:- a. Assumptions and Entity Relationship Model (ERD):

Assumptions:

1. Each train operates on a specific route.

2. Each route passes through multiple towns.

3. Multiple routes may serve the same town.

4. Each train has a unique train number.

5. Information on seating capacity and make/type is stored for each train.

6. Information is available on the average number of passengers carried per day for each route.

7. Each stage of a route has one or more drivers assigned.

8. Drivers have employee numbers, names, addresses, and optionally, telephone numbers.

Entity Relationship Model (ERD):

Entities:

1. Train

- Attributes: TrainNumber (Primary Key), SeatingCapacity, Make/Type

2. Route

- Attributes: RouteNumber (Primary Key)

3. Town

- Attributes: TownName (Primary Key)

4. RouteTown (Associative Entity)

- Attributes: No attributes (This entity exists to establish the many-to-many relationship between Route and Town)

5. AveragePassengers

- Attributes: RouteNumber (Foreign Key), AveragePassengersPerDay

6. Stage

- Attributes: StageID (Primary Key)

7. Driver

- Attributes: EmployeeNumber (Primary Key), Name, Address, TelephoneNumber

Relationships:

- Train to Route: Each train operates on a specific route. (One-to-Many)

  - Train (1) ---< Route (Many)

- Route to Town: Each route passes through multiple towns. (Many-to-Many)

  - Route (Many) ---< RouteTown >--- (Many) Town

- Route to AveragePassengers: Information on average passengers per day for each route. (One-to-Many)

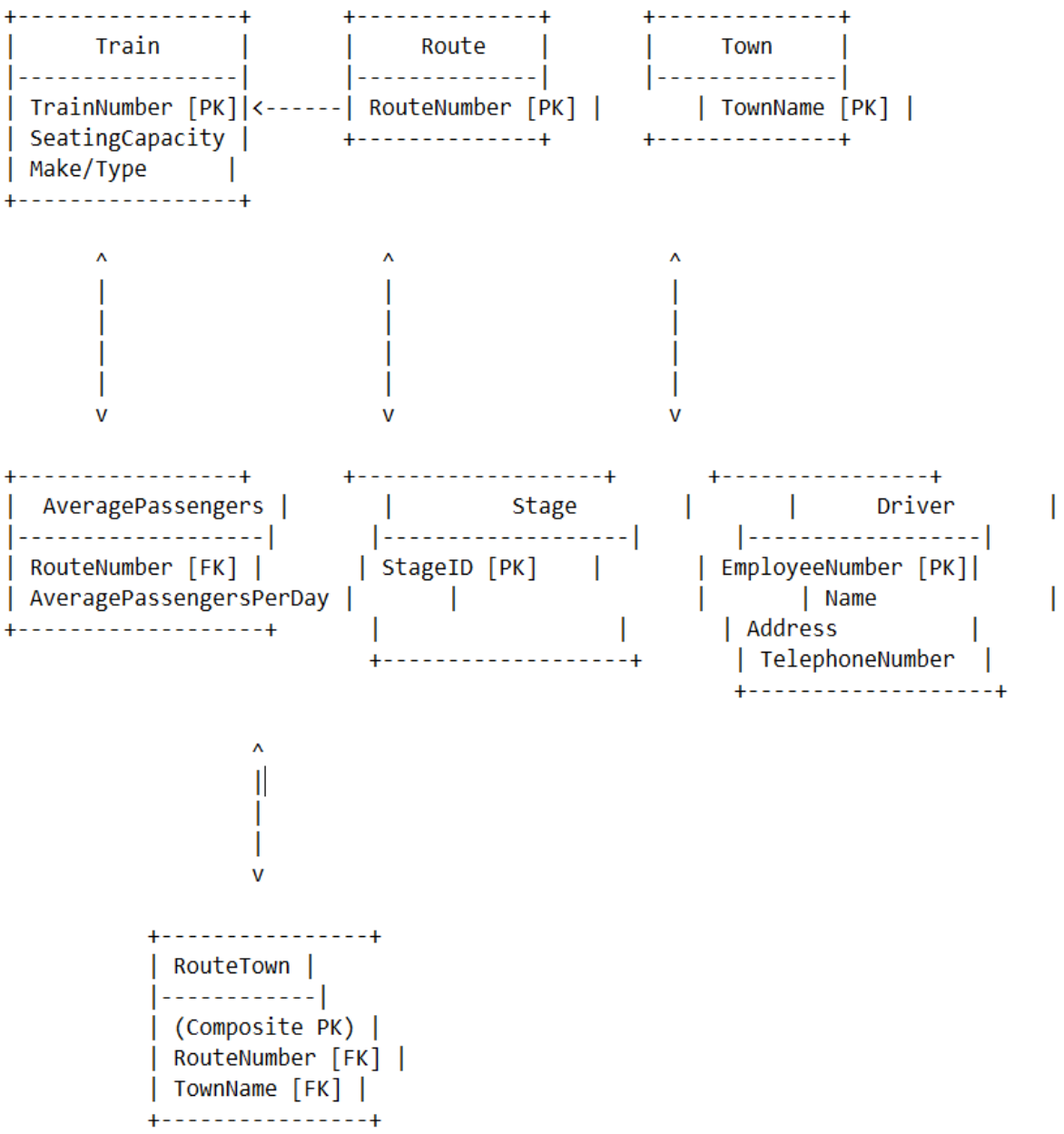
  - Route (1) ---< AveragePassengers (Many)

- Stage to Route: Each stage corresponds to a journey through a town on a route. (Many-to-One)

  - Stage (Many) ---< Route (1)

- Driver to Stage: One or more drivers are assigned to each stage of a route. (Many-to-Many)

- Driver (Many) ---< StageDriver >--- (Many) Stage



b. Devise a relational database schema to demonstrate how you will map your ERD model in 2(a) onto tables in a relational database. Identify the primary keys / foreign keys and attributes in each entity.

Ans:- Relational Database Schema:

Tables:

1. Train
   - TrainNumber (Primary Key)
   - SeatingCapacity
   - MakeType
2. Route
   - RouteNumber (Primary Key)
3. Town
   - TownName (Primary Key)
4. RouteTown (Associative Table)
   - RouteNumber (Foreign Key)
   - TownName (Foreign Key)
5. AveragePassengers
   - RouteNumber (Foreign Key)
   - AveragePassengersPerDay
6. Stage
   - StageID (Primary Key)
   - RouteNumber (Foreign Key)
7. Driver
   - EmployeeNumber (Primary Key)
   - Name
   - Address
   - TelephoneNumber
8. StageDriver (Associative Table)
   - StageID (Foreign Key)
   - EmployeeNumber (Foreign Key)

Primary Keys:

- TrainNumber (Train)
- RouteNumber (Route)
- TownName (Town)
- StageID (Stage)
- EmployeeNumber (Driver)

Foreign Keys:

- RouteNumber (AveragePassengers, Stage)
- RouteNumber, TownName (RouteTown)
- StageID, EmployeeNumber (StageDriver)

This schema maps the ERD model into a relational database with appropriate primary keys, foreign keys, and attributes for each entity. It captures the relationships between trains, routes, towns, average passenger data, stages, and drivers as described in the scenario.

3. South East Asia Ships (SEAS) is an agency that hires ship charter companies that rent ships. SEAS have no ships. Instead, it leases ships on behalf of shipowners who want to earn income from their ships when they are not using the ships themselves and SEAS charge the owners for this service. SEAS specialize in renting ships that can be used for multiday or weekly rentals. The smallest ship available is 30 feet long, and the largest is 70 feet long. Table 1 below present the data that SEAS captured in spreadsheet.

Table 1- Ship_Data Spreadsheet

BoatName	Make	Model	Length	OwnerLastName	Ownerfirst Name	OwnerPhone	Address	City	State	ZIP
Far Horizon	Catain a	Morgan	38	Princeton	Darryl	206-543-6677	2345 15th NE	Seattle	WA	98115
Ebb Tide	Hunter	38	38	Tulsa	Bill	503-486-8786	1324 24th NE	Portland	OR	97215
Foreign Shores	Hans Christian	38 MK II	38	Berkely	George	425-765-4455	4567 35th W	Bellevue	WA	98040
Seafarer V	Endeavotr	37	37	Tulsa	Bill	503-486-8786	1324 24th NE	Portland	OR	97215
Midnight on the Water	Sabre	32	32	Oxford	Kelly	503-578-7574	2435 36th SE	Astoria	OR	97103

a. Identify THREE (3) anomalies problems that are likely to occur if SEAS attempt to maintain its data in the spreadsheet as presented in Table 1. Provide suitable example for each anomalies case.

Ans:- Maintaining data in a spreadsheet can lead to various anomalies and problems. In the case of SEAS, here are three potential anomalies along with suitable examples:

1. Data Redundancy

-Anomaly:- Data redundancy occurs when the same information is duplicated in the spreadsheet, which can lead to inconsistencies and errors if not updated consistently.

- Example:- If multiple ships owned by the same owner share the same contact information, such as phone number or address, and this information is repeated for each ship they own, any changes to the owner's contact details would need to be updated for each ship individually. For instance, if Bill Tulsa's phone number changes, SEAS would have to update it for every ship he owns, which is prone to errors.

2. Inconsistent Data Entry

- Anomaly:- Inconsistent data entry involves variations in data formatting or data types, making it challenging to analyze or query the data accurately.

- Example:- In the spreadsheet, the length of the ship is listed both under the "Model" and "Length" columns. This inconsistency can lead to confusion when trying to extract information about ship lengths. For instance, if a query is performed to find all ships longer than 35 feet, it might miss ships with the length listed under the "Model" column.

3. Missing or Incomplete Data

- Anomaly:- Missing or incomplete data points can result in gaps in information, making it challenging to make informed decisions or perform comprehensive analysis.

- Example:- The "State" and "ZIP" columns in the spreadsheet contain missing or incomplete data for some entries. For example, "Seafarer V" has no state or ZIP code listed. This missing data can be problematic when trying to generate location-based reports or analyses.

b. Describe THREE (3) benefits that SEAS will gain from Normalizing table 1.

Ans:-Three benefits that SEAS will gain from normalizing Table 1:

1. Reduced Data Redundancy:- Normalization helps eliminate data redundancy by organizing data into separate tables and linking them using relationships. This reduces the risk of inconsistencies and saves storage space. For example, owner information would be stored in a separate table, reducing the need to repeat it for each boat entry.

2. Improved Data Integrity:- Normalization enforces data integrity by reducing the chances of anomalies like insertion and deletion anomalies. Each piece of data is stored in one place, making it easier to maintain and update without risking data inconsistencies.

3. Efficient Querying:- Normalized tables are typically more efficient for querying and reporting because they allow for smaller, well-structured tables. This can improve the speed and efficiency of data retrieval operations, which is crucial for a business like SEAS that needs to handle a large volume of ship data.

c. Normalize Table 1 to Third Normal Form to eliminate the anomalies problems mentioned in 3(a).

Ans:- To normalize Table 1 to the Third Normal Form (3NF), we need to do the following:

1. Create a separate table for boat information (Boat_ID, BoatName, Make, Model, Length), where Boat_ID is the primary key.

2. Create a separate table for owner information (Owner_ID, OwnerLastName, OwnerFirstName, OwnerPhone, Address, City, State, ZIP), where Owner_ID is the primary key.

3. Create a third table (Boat_Owner) to establish a many-to-many relationship between boats and owners. This table should include Boat_ID and Owner_ID as foreign keys.

Boat Table:

Boat_ID	BoatName	Make	Model	Length
1	Far Horizon	Cataina	Morgan	38
2	Ebb Tide	Hunter	38	38
3	Foreign Shores	Hans Christian	38 MK II	38
4	Seafarer V	Endeavotr	37	37
5	Midnight on ...	Sabre	32	32

Owner Table:

Owner_ID	OwnerLastName	OwnerFirstName	OwnerPhone	Address	City	Stat
1	Princeton	Darryl	206-543-6677	2345 15th NE	Seattle	WA
2	Tulsa	Bill	503-486-8786	1324 24th NE	Portland	OR
3	Berkely	George	425-765-4455	4567 35th W	Bellevue	WA
4	Oxford	Kelly	503-578-7574	2435 36th SE	Astoria	OR

Boat_Owner Table:

Boat_ID	Owner_ID
1	1
2	2
3	3
4	2
5	4

This normalization eliminates the anomalies and improves data integrity while allowing efficient querying of boat and owner information.

4. Consider the following relational database for Nepal Airlines.

Nepal Airlines has to keep track of its flight and airplane history. A flight is uniquely identified by the combination of a flight number and a date. Every passenger who has flown on Nepal Airlines has a unique passenger number. For a particular passenger who has taken a particular flight, the company wants to keep track of the fare that she paid for it and the date that she made the reservation for it. Clearly, a passenger may have taken many flights (he must have taken at least one to be in the database) and every flight has had many passengers on it.

A pilot is identified by a unique pilot (or employee) number. A flight on a particular date has exactly one pilot. Each pilot has typically flown many flights but a pilot may be new to the company, is in training, and has not flown any flights, yet. Each airplane has a unique serial number. A flight on a particular date used one airplane. Each airplane has flown on many flights and dates, but a new airplane may not have been used at all, yet.

The following describes the attributes for each entity.

PILOT

PILOTNUM	PILOTNAME	BIRTHDATE	HIREDATE

FLIGHT

FLIGHTNUM	DATE	DEPTIME	ARRTIME	PILOTNUM	PLANENUM

PASSENGER

PASSNUM	PASSNAME	ADDRESS	PHONE

RESERVATION

FLIGHTNUM	DATE	PASSNUM	FARE	RESVDATE

AIRPLANE

PLANENUM	MODEL	CAPACITY	YEARBUILT	MANUF

Referring to the above schema, write SQL code for the questions listed below:

a. Find the records for the airplanes manufactured by Boeing.

b. List the flights on May 1, 2017 that are scheduled to depart between 10 and 11AM.

c. How many of each Boeing aircraft model does Nepal Airlines have?

d. List the names, addresses, and telephone numbers of the passengers who have reservations on Flight 562 on January 15, 2017.

e. List the smallest (in terms of passenger capacity) Boeing 737.

```
a. Find the records for the airplanes manufactured by Boeing:
SELECT *
FROM AIRPLANE
WHERE MANUF = 'Boeing';

b. List the flights on May 1, 2017 that are scheduled to depart between 10 and 11 AM:
SELECT *
FROM FLIGHT
WHERE DATE = '2017-05-01'
  AND DEPTIME >= '10:00:00'
  AND DEPTIME < '11:00:00';

c. How many of each Boeing aircraft model does Nepal Airlines have?
SELECT MODEL, COUNT(*) AS COUNT
FROM AIRPLANE
WHERE MANUF = 'Boeing'
GROUP BY MODEL;

d. List the names, addresses, and telephone numbers of the passengers who have
 reservations on Flight 562 on January 15, 2017:
SELECT P.PASSNAME, P.ADDRESS, P.PHONE
FROM PASSENGER P
INNER JOIN RESERVATION R ON P.PASSNUM = R.PASSNUM
WHERE R.FLIGHTNUM = 562 AND R.DATE = '2017-01-15';

e. List the smallest (in terms of passenger capacity) Boeing 737:
SELECT *
FROM AIRPLANE
WHERE MANUF = 'Boeing' AND MODEL = '737'
ORDER BY CAPACITY
LIMIT 1;
```

5. a. Differentiate dirty read, nonrepeatable read, and phantom read with suitable example.

b. Based on Nepal Airlines schema shown in question 4, discuss which schema suitable for exclusive lock and shared lock. Justify your answer.

Ans:- In the context of a relational database like the one described for Nepal Airlines, exclusive locks and shared locks are used for controlling concurrent access to data. Here's how the schema would be suitable for these types of locks:

1. Exclusive Locks:- Exclusive locks are used when a transaction needs to perform write operations on data. These locks prevent other transactions from acquiring shared or exclusive locks on the same data simultaneously, ensuring data consistency during updates.

In the Nepal Airlines schema, the following tables and scenarios are suitable for exclusive locks:

- PASSENGER:- Exclusive locks would be used when adding, updating, or deleting passenger records. This prevents multiple transactions from modifying passenger data at the same time, ensuring data integrity.

- RESERVATION:- Exclusive locks would be needed when making changes to reservations, such as updating fares or reservation dates. This prevents conflicting updates to reservation records.

- PILOT:- If there are operations to add, update, or remove pilot records (such as changing pilot names or hire dates), exclusive locks would be required here to maintain consistency.

2. Shared Locks:- Shared locks are used when a transaction needs to read data without intending to modify it. Multiple transactions can hold shared locks on the same data simultaneously, allowing for concurrent reads without conflicts.

In the Nepal Airlines schema, the following tables and scenarios are suitable for shared locks:

- FLIGHT:- Shared locks can be used when retrieving flight information, such as departure times and arrival times. Multiple transactions can read this data simultaneously without affecting each other.

- **PASSENGER**:- Shared locks can be employed when reading passenger information, such as names and addresses, as long as no modifications are intended.

- **AIRPLANE:-** Shared locks are appropriate when querying airplane details, such as model, capacity, and year built. Multiple transactions can concurrently access this information without issues.

c. Ms. Shreya owns a cookies stall. Shreya remain competitive in the market and distinguish her business by providing homely taste cookies. As the business expanding, she wants to keep track of each of her customers and their orders. Eventually, she wants to notify them that their orders are ready via email. As her business growing, Shreya has hired you as a database consultant to develop an operational database having the following four tablesas shown in Table 2:

Table 2 - Cake & Cookies Database

CUSTOMER (CustomerlD, FirstName, LastName, Phone, EmailAddress)
ORDER (Ordeal) CustomerlD, Dateln, DateOut, Subtotal, Tax, TotalAmt)
ORDER ITEM (InvoireNtimher)temNumber OrderlD, Quantity, UnitPrice, Extended_Price)
ITEM (ItemID Item description, UnitPrice)

You as a database administrator, required to assign privilege for the database that you have draft. Assume that Shreya has a sales clerk. Tabulate the privilege granted for the users as shown in Table 3. Justify your answer.

Table 3: Home Cake & Cookies Database Privilege

	Customer	**Order**	**Order_Item**	**Item**
Owners				
Sales clerks				
System Admin				

Ans:- To assign privileges to the database tables for the different roles (Owners, Sales clerks, and System Admin), you need to consider the specific actions each role should be able to perform on each table. Here's a table showing the privileges for each role:

Owners

	Customer	Order	Order_Item	Item
Owners	SELECT, INSERT, UPDATE, DELETE	SELECT, INSERT, UPDATE, DELETE	SELECT, INSERT, UPDATE, DELETE	SELECT, INSERT, UPDATE, DELETE

Owners should have full access to all tables in the database. They can select, insert, update, and delete records in the Customer, Order, Order_Item, and Item tables because they have full control over the business.

Sales clerks

	Customer	Order	Order_Item	Item
Sales clerks	SELECT	SELECT, INSERT	SELECT, INSERT, UPDATE	SELECT

Sales clerks should have limited access to the database. They can only select records from the Customer and Item tables. In the Order table, they can select and insert records, indicating they can create new orders for customers. In the Order_Item table, they can select, insert, and update records, allowing them to add and modify order items. However, they cannot delete records in any table to maintain data integrity.

System Admin

	Customer	**Order**	**Order_Item**	**Item**
System Admin	SELECT, INSERT, UPDATE, DELETE	SELECT, INSERT, UPDATE, DELETE	SELECT, INSERT, UPDATE, DELETE	SELECT, INSERT, UPDATE, DELETE

The System Admin should have full access to all tables in the database, similar to the Owners. They can select, insert, update, and delete records in the Customer, Order, Order_Item, and Item tables to perform maintenance and administrative tasks.

Justification

1. Owners:- Owners have complete control over the business and the database, so they need full privileges to perform any action on the tables.

2. Sales clerks:- Sales clerks are responsible for creating and managing orders, so they have limited access to specific tables. They can select customer information, insert new orders, update order details, and select item information, but they can't delete any data to avoid accidental data loss.

3. System Admin:- System Admins require full privileges to manage and maintain the entire database, similar to the Owners. They should be able to perform any action on the tables for administrative purposes.