

## Q,NO.1



### Sales Files

**PropertyForRent (propertyNo, street, city, postcode, type, rooms, rent, ownerNo)**

**PrivateOwner (ownerNo, (Name, IName, address, telNo)**

**Client (clientNo, fName, (Name, address, telNo, prefType, maxRent)**

### Contract Files

**Lease (leaseNo, propertyNo, clientNo, rent, paymentMethod, deposit, paid, rentStart, rentFinish, duration)**

**PropertyForRent (propertyNo, street, city, postcode, rent)**

**Client (clientNo, (Name, (Name, address, telNo)**

**a. Describe any FOUR (4) limitations of a file-based processing as shown in Figure 1.**

Ans:- Four limitations of file-based processing:

**1. Data Redundancy:** In a file-based system, data is often duplicated across multiple files. For example, in the provided Sales and Contracts files, client information appears in both Sales and Contracts files. This redundancy can lead to inconsistencies and increased storage requirements.

**2. Data Inconsistency:** Because data is duplicated in multiple files, there is a risk of data inconsistencies. If data is updated or modified in one file and not in others, it can result in inaccurate and unreliable information.

**3. Data Isolation:** File-based systems tend to isolate data, making it difficult to access and share data across different parts of an organization. For example, if the Sales department and Contracts department need to access the same client information, they may need separate copies of the data, leading to synchronization challenges.

**4. Limited Data Security:** File-based systems often lack robust security features. It can be challenging to control access to specific data elements and ensure data privacy and integrity.

**b. Explain THREE (3) main functions of a database management system (DBMS).**

Ans:- Three main functions of a Database Management System (DBMS):

**1. Data Storage and Management:** A DBMS stores and manages data efficiently. It provides mechanisms for creating, updating, deleting, and retrieving data in a structured manner. This includes defining tables, relationships, and data constraints.

**2. Data Retrieval and Querying:** DBMS enables users to retrieve and query data using a standardized query language like SQL (Structured Query Language). Users can specify complex criteria to filter and retrieve the required information from the database.

**3. Data Security and Access Control:** DBMS offers robust security features to control access to data. It allows administrators to define user roles and permissions, ensuring that only authorized users can perform specific operations on the data. This helps protect data from unauthorized access and manipulation.

**c. Provide THREE (3) examples of a real-world database (e.g., the library contains a database of books).**

Ans:-Three examples of real-world databases:

**1. Library Database:** A library maintains a database of books, including information such as titles, authors, ISBN numbers, publication dates, and availability status. This database helps library staff manage the collection, track loans, and assist patrons in finding books.

**2. Healthcare Electronic Health Record (EHR) System:** Healthcare facilities use EHR systems to store patient information, including medical histories, diagnoses, prescriptions, and treatment plans. EHR databases help healthcare providers make informed decisions and ensure patient data is securely managed.

**3. Online Retail Store Database:** Online retailers maintain databases of products, customer profiles, orders, and inventory. These databases enable e-commerce platforms to display product catalogs, process orders, manage customer accounts, and track stock levels in real-time.

**Q.NO.2.**

**Table 1 — Customer Bank Account**

<b>AccountNo</b>	<b>Balance</b>	<b>Cust_ID</b>	<b>BranchNo</b>	<b>Address</b>
A-101	500	1313131	01	Nilai
A-102	400	2121234	02	Melaka
A-113	600	9876543	03	Mantin
A-201	900	2323412	01	Nilai
A-215	700	5434323	04	Seremban
A-222	700	1231231	04	Seremban
A-305	350	4543453	02	Melaka

- a. Using correct terminology, identify and describe all the components in Table 1. Illustrate each of the components with an example by referring to Table 1.**

Ans:- Components in Table 1:

**1. Table Name:** The name of the table is "Customer Bank Account."

**2. Columns/Attributes**

**AccountNo:** This is a column that stores the account number.

Example: A-101.

**Balance:** This column stores the balance of the bank account.

Example: 500.

**Cust\_ID:** It stores the customer's unique identification number.

Example: 1313131.

**BranchNo:** This column contains the branch number associated with the account.

Example: 01.

**Address:** It stores the address of the customer associated with the account.

Example: Nilai.

**3. Rows/Records:** Each row in the table represents a specific bank account with its associated information. For example, the first row represents an account with AccountNo A-101, Balance 500, Cust\_ID 1313131, BranchNo 01, and Address Nilai.

**b. Based on Table 1, describe problems that will arise for insert, update and delete. Illustrate your answers with an example.**

Ans:- Problems with Insert, Update, and Delete:

### **Insert**

Problem: Inserting duplicate values in the AccountNo column may lead to data integrity issues as each account should have a unique account number.

Example: If you try to insert a new record with AccountNo A-101, it will conflict with the existing record for A-101.

### **Update**

Problem: Incorrect or unauthorized updates to the Balance column can result in incorrect account information.

Example: If an unauthorized user updates the Balance of A-102 to 1000 without proper authorization, it will lead to incorrect data.

### **Delete**

Problem: Deleting a record without considering referential integrity constraints may result in orphaned records in related tables.

Example: If you delete the record with Cust\_ID 9876543 without updating or removing related records in other tables that reference this Cust\_ID, you will have orphaned data.

**c. Differentiate Primary key and Foreign key.**

Ans:-

**Primary Key**

A primary key is a column or set of columns in a table that uniquely identifies each row in the table.

It enforces data integrity by ensuring that each value in the primary key is unique and not null.

In Table 1, "AccountNo" can be a primary key as it uniquely identifies each bank account.

**Foreign Key**

A foreign key is a column or set of columns in one table that is used to establish a link between the data in two tables.

It creates a relationship between tables by referencing the primary key of another table.

In Table 1, "BranchNo" could be a foreign key if it references the primary key of a "Branch" table, linking bank accounts to specific branches.

### Q.NO.3

a. Create an ER model for each of the following descriptions:

- i. A large organisation has several parking lots, which are used by staff.
- ii. Each parking lot has a unique name, location, capacity, and number of floors (where appropriate).
- iii. Each parking lot has parking spaces, which are uniquely identified using a space number.
- iv. Members of staff can request the sole use of a single parking space. Each member of staff has a unique number, name, telephone extension number, and vehicle license number.
- v. Represent all the ER models described in parts (i), (ii), (iii), and (iv) as a single ER model. Provide any assumptions necessary to support your model.

Ans:- Entity-Relationship (ER) Model for the Descriptions:

#### **Assumptions:**

- Each parking space is uniquely identified within the organization.
- Staff members can only request one parking space.
- Visitors are not staff members but can use specific parking spaces for a day visit.

#### **Entities and their attributes:**

##### **1. Organization**

- Attributes: Name

##### **2. Parking Lot**

- Attributes: Name (unique), Location, Capacity, Number of Floors

##### **3. Parking Space**

- Attributes: Space Number (unique), Type (undercover or not)

#### **4. Staff Member**

- Attributes: Staff ID (unique), Name, Telephone Extension Number, Vehicle License Number

#### **5. Visitor**

- Attributes: Visitor ID (unique), Vehicle License Number

#### **6. Booking**

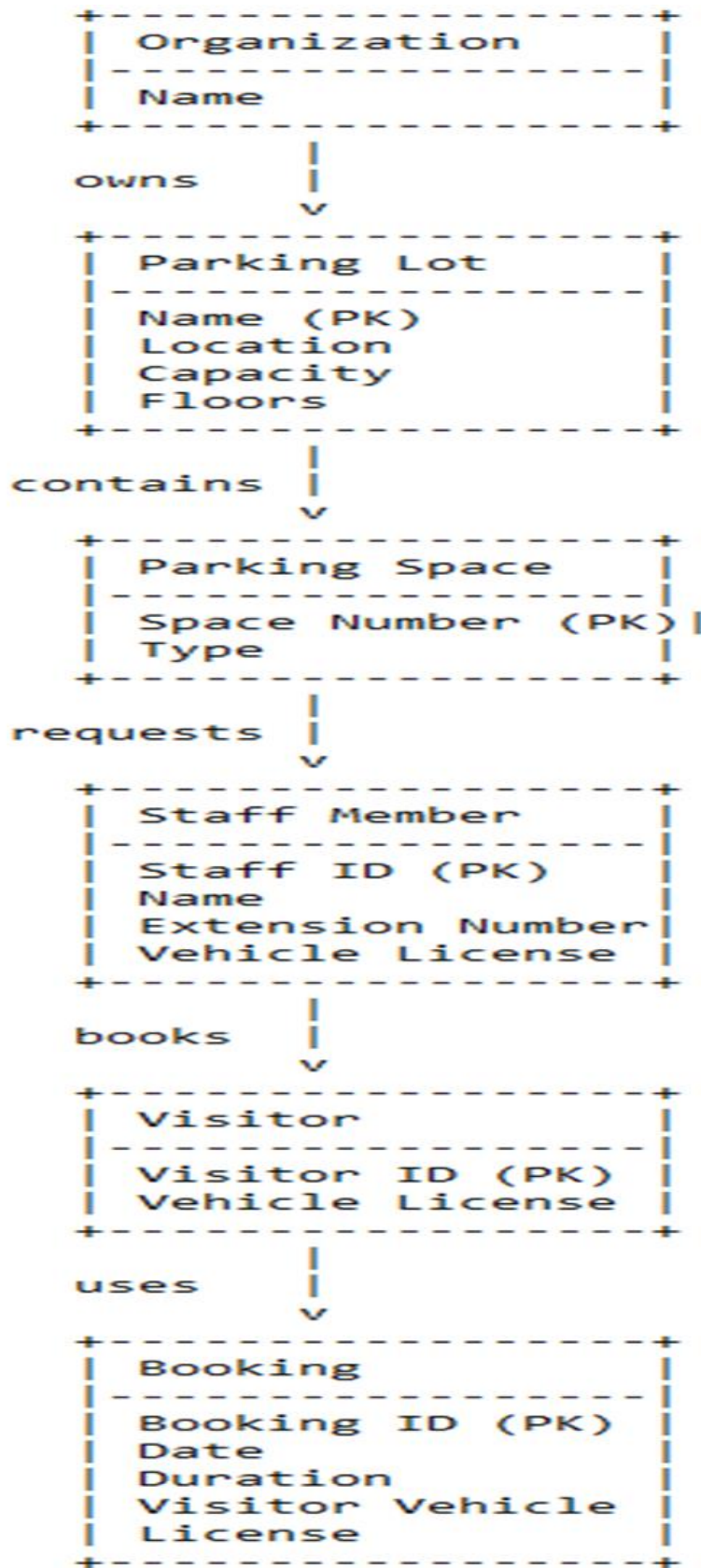
- Attributes: Booking ID (unique), Date, Duration, Visitor Vehicle License Number

#### **Relationships:**

- Organization owns Parking Lot (1 to many)
- Parking Lot contains Parking Space (1 to many)
- Staff Member requests Parking Space (1 to 1)
- Visitor books Parking Space (1 to 1)

Here is a simplified ER diagram for the above descriptions:





**b. Illustrate specialization and generalisation concepts into Entity Relational Model created in 3a(v) for the following added conditions:**

**i. The majority of parking spaces are under cover and each can be allocated for use by a member of staff for a monthly rate.**

**ii. Parking spaces that are not under cover are free to use and each can be allocated for use by a member of staff.**

**iii. Up to twenty covered parking spaces are available for use by visitors to the company. However, only members of staff are able to book out a space for the day of the visit. There is no charge for this type of booking, but the member of staff must provide the visitor's vehicle license number.**

Ans:- To represent specialization and generalization in the ER model, we can introduce a superclass "Parking Space" and then create two subclasses "Covered Parking Space" and "Open Parking Space" with specific attributes.

**Entities and their attributes:**

1. Parking Space (Superclass)

- Attributes: Space Number (unique)

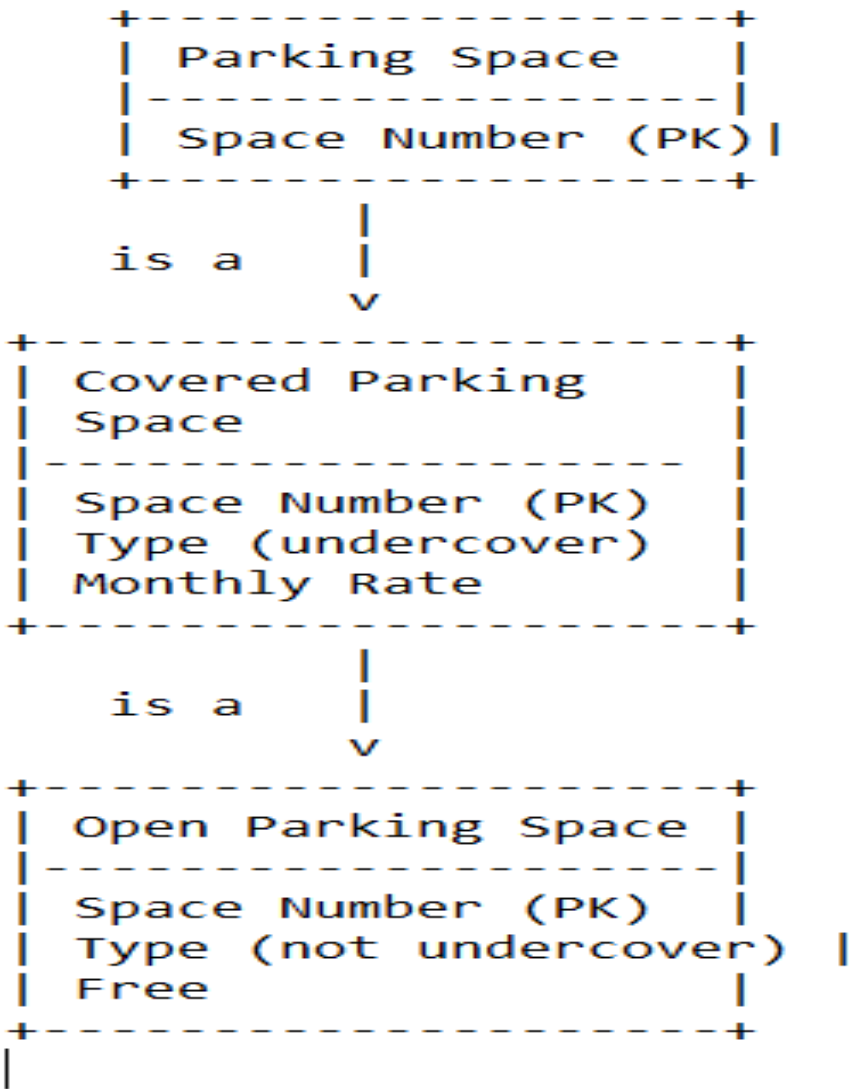
2. Covered Parking Space (Subclass of Parking Space)

- Attributes: Type (undercover), Monthly Rate

3. Open Parking Space (Subclass of Parking Space)

- Attributes: Type (not undercover), Free

Now, the ER diagram will include a specialization/generalization relationship:



**c. Describe the following terms:**

**i. Participation Constraint**

**ii. Disjoint Constraints**

**iii. Aggregation**

Ans:-

i. **Participation Constraint:-** Participation constraint in an ER model specifies whether an entity's participation in a relationship is mandatory (total participation) or optional (partial participation). It indicates whether every instance of the entity must be associated with the relationship (mandatory) or if it can exist without being part of the relationship (optional).

**ii. Disjoint Constraints:-** Disjoint constraints are used in specialization/generalization hierarchies to specify whether or not subtypes are mutually exclusive (disjoint) or can overlap (non-disjoint). Disjoint means that an instance can belong to only one subtype, while non-disjoint allows instances to belong to multiple subtypes within the hierarchy.

**iii. Aggregation:-** Aggregation in an ER model represents a relationship between a whole (the aggregate) and its parts (the components). It is used to model complex objects as a composition of simpler objects. Aggregation is typically depicted by a diamond shape in the ER diagram and is used to show that one entity is composed of or contains other entities.

**Q.NO.4.**

**Table 2 - Patient Appointment Data**

<b>StaffNo</b>	<b>Dent_Name</b>	<b>patNo</b>	<b>patName</b>	<b>Appointment date time</b>	<b>surgeryNo</b>
S1011	Tiaga	P100	Armin	12-Sep-19 10.00	S15
S1011	Tiaga	P105	Aines	12-Sep-19 12.00	S15
S1024	Helen	P108	Brian	12-Sep-19 10.00	S10
S1024	Helen	P108	Brian	14-Sep-19 14.00	S10
S1032	Robin	P105	Aines	14-Sep-19 16.00	S15
S1032	Robin	P110	Manu	15-Sep-19 18.00	S13

**a. Identify the functional dependencies represented by the attributes shown in Table 2.**

Ans:-The functional dependencies in Table 2:

- StaffNo → Dent\_Name

(StaffNo uniquely determines Dent\_Name)

- patNo → patName

(patNo uniquely determines patName)

- Appointment date time → surgeryNo

(Appointment date time uniquely determines surgeryNo)

- StaffNo, Appointment date time → patNo, patName

(The combination of StaffNo and Appointment date time uniquely determines patNo and patName)

- StaffNo, Appointment date time → surgeryNo

(The combination of StaffNo and Appointment date time uniquely determines surgeryNo)

**b. Illustrate the process of normalizing the above table to 3NF relations. Identify the primary, alternate, and foreign keys in your 3NF relations.**

Ans:- To normalize the given table into 3NF (Third Normal Form) relations, we'll first identify the functional dependencies and then create separate tables to eliminate any partial or transitive dependencies.

**1. Staff Table (StaffNo, Dent\_Name)**

- Primary Key: StaffNo
- Alternate Key: Dent\_Name (since Dent\_Name can uniquely identify a staff member)
- This table stores information about the dental staff.

StaffNo	Dent_Name
S1011	Tiaga
S1024	Helen
S1032	Robin

**2. Patient Table (patNo, patName)**

- Primary Key: patNo
- This table stores information about patients.

patNo	patName
P100	Armin
P105	Aines
P108	Brian
P110	Manu

**3. Appointment Table (StaffNo, patNo, AppointmentDateTime, surgeryNo)**

- Primary Key: (StaffNo, patNo, AppointmentDateTime)
- Foreign Keys: StaffNo (references Staff Table), patNo (references Patient Table)
- This table stores information about appointments.

StaffNo	patNo	Appointment Date Time	surgeryNo
S1011	P100	12-Sep-19 10.00	S15
S1011	P105	12-Sep-19 12.00	S15
S1024	P108	12-Sep-19 10.00	S10
S1024	P108	14-Sep-19 14.00	S10
S1032	P105	14-Sep-19 16.00	S15
S1032	P110	15-Sep-19 18.00	S13

**c. Describe the concept of transitive dependency. Provide an example to illustrate your answer.**

Ans:-Transitive Dependency: A transitive dependency is a type of functional dependency where an attribute depends on another attribute through a third attribute. In other words, if  $A \rightarrow B$  and  $B \rightarrow C$ , then  $A \rightarrow C$  is a transitive dependency.

Example: Let's consider a table called "StudentInfo" with the following attributes:

- StudentID (Primary Key)
- StudentName
- Department

In this table, if we observe the functional dependencies:

- StudentID  $\rightarrow$  StudentName
- Department  $\rightarrow$  StudentID

### Q.NO.5

a. EMP (emp\_no, emp\_name, salary, address, dept\_id) DEPT (deptid, dept\_name)

From the schema above, write a SQL code to retrieve emp\_no, emp\_name, dept\_id and dept name by using:

- i. Join with "Using" clause
- ii. Join with "ON" clause

Ans:- SQL code to retrieve emp\_no, emp\_name, dept\_id, and dept\_name using both "Using" and "ON" clauses:

Using "Using" clause:

```
SELECT emp.emp_no, emp.emp_name, emp.dept_id, dept.dept_name
FROM EMP emp
JOIN DEPT dept
USING (dept_id);
```

Using "ON" clause:

```
SELECT emp.emp_no, emp.emp_name, emp.dept_id, dept.dept_name
FROM EMP emp
JOIN DEPT dept
ON emp.dept_id = dept.deptid;
```



**b. Describe TWO (2) restrictions imposed on using "Group By" clause.**

Ans:- Two restrictions imposed on using the "GROUP BY" clause:

**1. Aggregations:** When you use the "GROUP BY" clause in SQL, you must include at least one aggregation function (e.g., SUM, COUNT, AVG) in your SELECT statement for the columns that are not part of the grouping. This means that you can only select aggregated values for non-grouped columns. For example, if you are grouping by department and want to retrieve the total salary for each department, you can use SUM(salary) in the SELECT clause, but you cannot simply select the individual employee's salary without an aggregation function.

**2. Column Selection:** When using "GROUP BY," you can only include columns in the SELECT clause that are either part of the grouping or are used with an aggregation function. Columns that are neither part of the grouping nor aggregated cannot be selected in the result set. This restriction ensures that the query result is meaningful and consistent because each row in the result represents an aggregated group, and selecting non-aggregated, non-grouped columns could result in ambiguous or incorrect data.