**Name :** Harsh Tankariya

**E-mail :** [tankariyaharsh7@gmail.com](mailto:tankariyaharsh7@gmail.com)

**Mobile no :** 7990748717

**Subject :** Task 1

**Task 1** :

Database Setup and Schema Design.

Objective: Learn to create databases, tables, and define relationships.

Tools:MySQL Workbench / pgAdmin / SQLiteStudio. Deliverables: SQL script to create schema and ER diagram

1. domain is **Hospital Management system**

2.**Identify entities and relationships**

| **Relationship** | **Type** | **Table Used** |
| --- | --- | --- |
| Patient–MedicalRecord | 1-to-Many | Medical\_Record FK of PatID |
| Patient–Hospital | Many-to-One | Patient.HosID |
| Hospital–Doctor | 1-to-Many | Doctor.HosID |
| Doctor–Patient | Many-to-Many | Doctor\_Patient |
| ISA (Indoor/Outdoor) | 1-to-1 | PatID as FK and UNIQUE in both subclasses |

3.**Create tables using CREATE TABLE**

-- Hospital Table

CREATE TABLE Hospital (

HosID INT PRIMARY KEY,

Name VARCHAR(100)

);

-- Doctor Table

CREATE TABLE Doctor (

DrID INT PRIMARY KEY,

Dr\_Name VARCHAR(100),

HosID INT,

FOREIGN KEY (HosID) REFERENCES Hospital(HosID)

);

-- Patient Table

CREATE TABLE Patient (

PatID INT PRIMARY KEY,

Name VARCHAR(100),

HosID INT,

FOREIGN KEY (HosID) REFERENCES Hospital(HosID)

);

-- Medical Record Table

CREATE TABLE Medical\_Record (

MRID INT PRIMARY KEY,

Report\_Name VARCHAR(100),

PatID INT,

FOREIGN KEY (PatID) REFERENCES Patient(PatID));

-- Indoor Patient (subclass of Patient)

CREATE TABLE Indoor (

IPDID INT PRIMARY KEY,

RoomNo VARCHAR(20),

PatID INT UNIQUE,

FOREIGN KEY (PatID) REFERENCES Patient(PatID));

-- Outdoor Patient (subclass of Patient)

CREATE TABLE Outdoor (

OPDID INT PRIMARY KEY,

Charge DECIMAL(10,2),

PatID INT UNIQUE,

FOREIGN KEY (PatID) REFERENCES Patient(PatID)

);

-- Doctor-Patient Relationship (Many-to-Many)

CREATE TABLE Doctor\_Patient (

DrID INT,

PatID INT,

PRIMARY KEY (DrID, PatID),

FOREIGN KEY (DrID) REFERENCES Doctor(DrID),

FOREIGN KEY (PatID) REFERENCES Patient(PatID)

);

4**.Define primary and foreign keys**

**Entities & Attributes:**

1. **Patient**
   * PatID (Primary Key)
   * Name
2. **Medical\_Record**
   * MRID (Primary Key)
   * Report\_Name
   * Related to **Patient** (1-to-1 or 1-to-many)
3. **Hospital**
   * HosID (Primary Key)
   * Name
4. **Doctor**
   * DrID (Primary Key)
   * Dr\_Name
   * Related to **Hospital**
5. **Indoor** (Subtype of Patient)
   * IPDID (Primary Key)
   * RoomNo
   * PatID (Foreign Key)
6. **Outdoor** (Subtype of Patient)
   * OPDID (Primary Key)
   * Charge
   * PatID (Foreign Key)

**Interview Questions:**

**1. What is normalization?**

Normalization is the process of organizing data in a database to reduce data redundancy and improve data integrity by dividing large tables into smaller related ones.

**2. Explain primary vs foreign key.**

* **Primary Key**: Uniquely identifies each record in a table. Cannot be NULL.
* **Foreign Key**: Refers to the **primary key** in another table to create a relationship

**3. What are constraints?**

* **Constraints** are rules applied to table columns to control the data.

Examples: NOT NULL, UNIQUE, PRIMARY KEY, FOREIGN KEY, CHECK, DEFAULT.

**4. What is a surrogate key?**

A **surrogate key** is an artificial or auto-generated key (like an ID number) used to uniquely identify a row, instead of using real data.

**5. How do you avoid data redundancy?**

* By using **normalization**
* Creating **relationships** with foreign keys
* Avoiding storing the same data in multiple places

**6. What is ER diagram?**

* An **ER (Entity-Relationship) diagram** shows the **entities**, **attributes**, and **relationships** in a database system. It's like a visual plan of the database.

**7. What are the types of relationships in DBMS?**

* **One-to-One (1:1)**
* **One-to-Many (1:N)**
* **Many-to-One (N:1)**
* **Many-to-Many (M:N)**

**8. Explain the purpose of AUTO\_INCREMENT.**

* AUTO\_INCREMENT automatically increases the value of a column (usually the **primary key**) when a new row is added.

**9. What is the default storage engine in MySQL?**

* The default storage engine is **InnoDB**.  
  It supports transactions, foreign keys, and better performance.

**10. What is a composite key?**

* A **composite key** is a **combination of two or more columns** used together as a **primary key** to uniquely identify a row.

**Key Concepts: DDL, Normalization, ER Diagrams**

**1. DDL (Data Definition Language):**

**Definition:**  
DDL is a set of SQL commands used to **define and manage database structure**.

**Common DDL commands:**

* CREATE – to create tables, databases, etc.
* ALTER – to modify existing structures
* DROP – to delete tables/databases
* TRUNCATE – to remove all data from a table (but keep the structure)

**2. Normalization:**

**Definition:  
Normalization is the process of organizing data in a database to:**

* Reduce redundancy (no duplicate data)
* Improve data integrity

**Types (Forms):**

* 1NF: Remove repeating groups
* 2NF: Remove partial dependency
* 3NF: Remove transitive dependency

**Goal:** Make the database efficient and clean.

**3. ER Diagrams (Entity-Relationship Diagrams):**

**Definition:**An ER Diagram is a visual representation of a database showing:

* Entities (tables)
* Attributes (columns)
* Relationships (connections between tables)

**Use:** It helps in planning the structure of a database before actual implementation.

**Example:  
Entity:** Student **Attributes:** StudentID, Name, Age **Relationship:** Enrolls in → Course