Database Management System

5/7/25

1] Domain / Scenario chosen :-

Hospital Management System

2] Data Description :-

- Patient :- Patient_ID , Name , Age , Gender , Contact , Address , Disease , Admit_date , discharge_date

- Doctor :- Doctor_ID , Name , Specialization , Contact , department , Shift .

- Appointment :- Appointment_ID , Patient_ID , Doctor_ID , Date , time , Status .

- Treatment / Prescription :- Treatment_ID , Patient_ID , Doctor_ID , Medicine , Dosage , Duration

- Billing : Bill_ID , Patient_ID , Total_Amount , Payment_Status , Payment_Date .

3] Types of users

1] Admin Staff :- Manage patient records, billing and reports

2] Doctors :- Views patients, update treatment and prescriptions

3] Nurses / Ward Staff :- Track admitted patients, update status

4] Patients / Visitors :- Book appointments, check reports

4] Challenges faced in the traditional implementation of the system

-> • Data redundancy and inconsistency
- Difficult to search and retrieve patient history quickly
- No centralized access - doctors, nurses and admin work in isolation.
- Manual report generation is slow and error-prone
- Security and privacy issues for patient data.

5] How DBMS Helps Overcome Challenges
- Centralized database ensure data consistency
- Easy retrieval of patient and doctor records using queries.
- Data security with user authentication and authorization (DCL)
- Reduced redundancy via normalization
- Quick reporting with SQL queries and views
- Concurrent access for doctors, nurses and admin staff

6] List of tables in the system.

1] Patients

2] Doctors

3] Appointments

4] Treatments / Prescriptions

5] Billing.

7] Schema for Each Table / Relation

1] Patients Table

```
Create Table Patients (
Patient_ID INT PRIMARY KEY,
Name VARCHAR (100) NOT NULL,
Age INT,
GENDER CHAR (1);
Contact VARCHAR (15),
Address VARCHAR (255);
Disease Varchar (100),
Admit_Date DATE,
Discharge_Date DATE );
```

2. Doctors Table

```
CREATE TABLE Doctors (
    Doctor_ID INT PRIMARY KEY,
    Name VARCHAR (100) NOT NULL,
    Specialization VARCHAR (100),
    Contact VARCHAR (15),
    Department VARCHAR (50),
    Shift VARCHAR (20)
);
```

3. Appointments Table

```
Create TABLE Appointments (
    Appointment_ID INT Primary KEY,
    Patient_ID INT,
    DOCTOR-ID INT,
    Date DATE,
    Time TIME,
    Status VARCHAR (20),
```

4. Treatments Table

```
Create Table Treatments (
Treatment_ID INT PRIMARY KEY,
Patient_ID INT,
Doctor_ID INT,
Medicine VARCHAR (100),
Dosage VARCHAR (50),
Duration VARCHAR (50),
);
```

5. Billing Table

```
CREATE TABLE Billing (
Bill_ID INT BPRIMARY KEY,
Patient_ID INT,
Total_Amount DECIMAL (10,2)
Payment_Status VARCHAR (20),
Payment_Date DATE,
);
```