

# HOSPITAL\_MANAGEMENT\_SYSTEM(H\_M\_S)

**1. Introduction to MySQL:** MySQL is an open-source relational database management system (RDBMS) that uses structured query language (SQL) to manage and manipulate data in a database. It is widely used for various applications, from small web applications to large enterprise systems. MySQL's key features include:

- Scalability: Capable of handling large amounts of data and concurrent connections.
- Flexibility: Supports various data types and storage engines.
- Performance: Optimized for speed and efficiency.
- Reliability: Known for its stability and robustness.

**2. Installation of MySQL:** MySQL can be installed on various operating systems, including Windows, macOS, and Linux. Here are the general steps to install MySQL:

## **Windows:**

- Download the MySQL installer from the official website.  
<https://dev.mysql.com/downloads/installer/>
- Run the installer and follow the on-screen instructions.
- Choose the installation type (Typical, Complete, or Custom). Recommended Custom.
- Set a root password for the MySQL server.

## **3. E-R Diagram (ERD)**

An Entity-Relationship Diagram (ERD) is a visual representation of the data model that shows the entities, attributes, relationships between entities, and cardinality. ERDs are commonly used in database design to help developers and stakeholders understand the structure and relationships within a database.

### **Identify Entities**

- Start by identifying the main entities in your system. These are the objects or concepts about which you want to store data.
- Each entity should correspond to a table in your database.

### **Define Attributes**

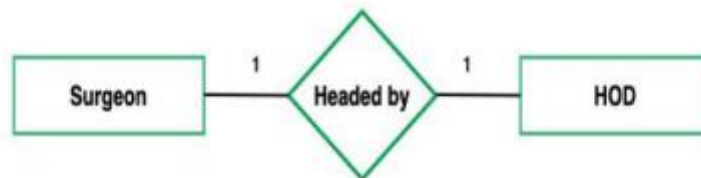
- For each entity, list the attributes (properties or fields) that describe it.
- These attributes will become columns in the corresponding database table.

## Identify Relationships

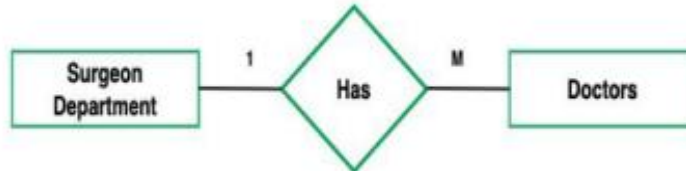
- Determine how entities are related to each other. There are three types of relationships: one-to-one (1:1), one-to-many (1:N), and many-to-many (N:M).
- Represent these relationships using lines connecting the entities.

Let's see a few examples of relationships:

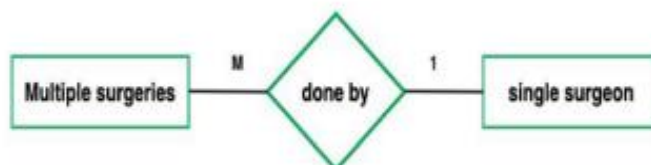
### One to One



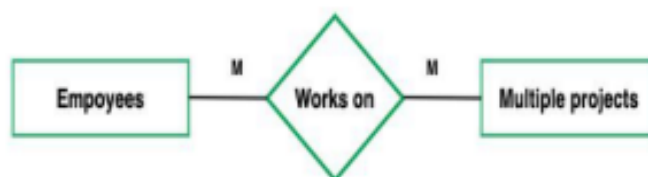
### One to Many



### Many to One



### Many to Many



## Cardinality Notation

Cardinality represents the number of times an entity of an entity set participates in a relationship set. Or we can say that the cardinality of a relationship is the number of tuples (rows) in a relationship.

- Use notation (such as Crow's Foot Notation or Chen Notation) to indicate the cardinality of each relationship.
- Cardinality describes how many instances of one entity are related to how many instances of another entity.
- Common notations include:
  - ★ One (1)
  - ★ Zero or one (0..1)
  - ★ Many (N)
  - ★ Zero or many (0..N)

## Optional:

### Add Attributes and Constraints

- Include additional information in your ERD, such as primary keys, foreign keys, and constraints (e.g., unique constraints).

### Create the diagram

- Use specialized diagramming software or tools (e.g., Lucidchart, draw.io, or even pen and paper) to create your ERD.

### Refine and Review:

- Review your ERD with stakeholders and team members to ensure it accurately represents the data model and relationships. Make any necessary refinements.

## Let's identify the entities of the Hospital management system

- 1.admissions
- 2.appointments
- 3.billing\_invoices
- 4.departments
- 5.doctors
- 6.insurance

- 7.inventory
- 8.lab\_tests
- 9.medical\_records
- 10.patients
- 11.staff
- 12.suppliers
- 13.wards\_rooms

Now let's identify the attributes and relationships of each entity for the Hospital Management System.

## DATABASES

### →HOSPITAL

#### TABLES

##### →admissions

```
mysql> desc admissions;
```

Field	Type	Null	Key	Default	Extra
admission_id	int	NO	PRI	NULL	auto_increment
patient_id	int	YES	MUL	NULL	
ward_room_id	int	YES	MUL	NULL	
admission_date	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
discharge_date	timestamp	YES		NULL	
admission_status	enum('Admitted','Discharged')	YES		NULL	

6 rows in set (0.00 sec)

#### CODE:

```
CREATE TABLE admissions (
    admission_id INT AUTO_INCREMENT PRIMARY KEY,
    patient_id INT NOT NULL,
    ward_room_id INT NOT NULL,
    admission_date TIMESTAMP NOT NULL,
    discharge_date TIMESTAMP,
    admission_status VARCHAR(50) NOT NULL
);
```

##### →appointments

```
mysql> desc appointments;
```

Field	Type	Null	Key	Default	Extra
appointment_id	int	NO	PRI	NULL	auto_increment
patient_id	int	YES	MUL	NULL	
doctor_id	int	YES	MUL	NULL	
appointment_date	date	YES		NULL	
appointment_time	time	YES		NULL	
appointment_status	enum('Scheduled','Cancelled','Completed')	YES		NULL	
appointment_notes	text	YES		NULL	

7 rows in set (0.00 sec)

### CODE:

CREATE TABLE appointments (

    appointment\_id INT AUTO\_INCREMENT PRIMARY KEY,

    patient\_id INT NOT NULL,

    doctor\_id INT NOT NULL,

    appointment\_date DATETIME NOT NULL,

    appointment\_status VARCHAR(50) NOT NULL,

    appointment\_notes TEXT,

    CONSTRAINT fk\_patient FOREIGN KEY (patient\_id) REFERENCES patients(patient\_id),

    CONSTRAINT fk\_doctor FOREIGN KEY (doctor\_id) REFERENCES doctors(doctor\_id)

);

### →billing\_invoices

```
mysql> desc billing_invoices;
```

Field	Type	Null	Key	Default	Extra
invoice_id	int	NO	PRI	NULL	auto_increment
patient_id	int	YES	MUL	NULL	
total_amount	decimal(10,2)	YES		NULL	
payment_status	enum('Paid','Pending')	YES		NULL	
invoice_date	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED

5 rows in set (0.01 sec)

### CODE:

CREATE TABLE invoices (

    invoice\_id INT AUTO\_INCREMENT PRIMARY KEY,

    patient\_id INT,

    total\_amount DECIMAL(10,2),

    payment\_status ENUM('Paid','Pending'),

    invoice\_date TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

    CONSTRAINT fk\_patient FOREIGN KEY (patient\_id) REFERENCES patients(patient\_id)

);

## →departments

```
mysql> desc departments;
```

Field	Type	Null	Key	Default	Extra
department_id	int	NO	PRI	NULL	auto_increment
department_name	varchar(100)	NO		NULL	
head_of_department	varchar(100)	YES		NULL	
contact_number	varchar(15)	YES		NULL	
email	varchar(100)	YES		NULL	

5 rows in set (0.01 sec)

### CODE:

```
CREATE TABLE departments (  
    department_id INT AUTO_INCREMENT PRIMARY KEY,  
    department_name VARCHAR(100) NOT NULL,  
    head_of_department VARCHAR(100),  
    contact_number VARCHAR(15),  
    email VARCHAR(100)  
);
```

## →doctors

```
mysql> desc doctors ;
```

Field	Type	Null	Key	Default	Extra
doctor_id	int	NO	PRI	NULL	auto_increment
first_name	varchar(50)	NO		NULL	
last_name	varchar(50)	NO		NULL	
gender	enum('Male','Female','Other')	YES		NULL	
date_of_birth	date	YES		NULL	
contact_number	varchar(15)	YES		NULL	
email	varchar(100)	YES		NULL	
specialization	varchar(100)	YES		NULL	
qualification	varchar(100)	YES		NULL	
years_of_experience	int	YES		NULL	
registration_date	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED

11 rows in set (0.01 sec)

### CODE:

```
CREATE TABLE doctors (  
    doctor_id INT AUTO_INCREMENT PRIMARY KEY,  
    first_name VARCHAR(50) NOT NULL,  
    last_name VARCHAR(50) NOT NULL,  
    gender ENUM('Male','Female','Other'),  
    date_of_birth DATE,  
    contact_number VARCHAR(15),  
    email VARCHAR(100),  
    specialization VARCHAR(100),  
    qualification VARCHAR(100),
```

```

years_of_experience INT,

registration_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP

);

```

#### →insurance

```
mysql> desc insurance ;
```

Field	Type	Null	Key	Default	Extra
insurance_id	int	NO	PRI	NULL	auto_increment
patient_id	int	YES	MUL	NULL	
insurance_company	varchar(100)	NO		NULL	
policy_number	varchar(50)	YES		NULL	
policy_details	text	YES		NULL	
expiration_date	date	YES		NULL	

6 rows in set (0.00 sec)

#### CODE:

```

CREATE TABLE insurance (

insurance_id INT AUTO_INCREMENT PRIMARY KEY,

patient_id INT,

insurance_company VARCHAR(100) NOT NULL,

policy_number VARCHAR(50),

policy_details TEXT,

expiration_date DATE

);

```

#### →inventory

```
mysql> desc inventory ;
```

Field	Type	Null	Key	Default	Extra
item_id	int	NO	PRI	NULL	auto_increment
item_name	varchar(100)	NO		NULL	
item_description	text	YES		NULL	
quantity	int	YES		NULL	
unit_price	decimal(10,2)	YES		NULL	
supplier_id	int	YES	MUL	NULL	
last_restock_date	timestamp	YES		NULL	

7 rows in set (0.00 sec)

#### CODE:

```

CREATE TABLE items (

item_id INT AUTO_INCREMENT PRIMARY KEY,

item_name VARCHAR(100) NOT NULL,

item_description TEXT,

quantity INT,

```

```

unit_price DECIMAL(10,2),

supplier_id INT,

last_restock_date TIMESTAMP

);

```

### →lab\_tests

```
mysql> desc lab_tests;
```

Field	Type	Null	Key	Default	Extra
test_id	int	NO	PRI	NULL	auto_increment
patient_id	int	YES	MUL	NULL	
test_name	varchar(100)	NO		NULL	
test_date	date	YES		NULL	
test_results	text	YES		NULL	
test_status	enum('Pending','Completed')	YES		NULL	
test_notes	text	YES		NULL	

7 rows in set (0.00 sec)

### CODE:

```

CREATE TABLE lab_tests (

test_id INT AUTO_INCREMENT PRIMARY KEY,

patient_id INT,

test_name VARCHAR(100) NOT NULL,

test_date DATE,

test_results TEXT,

test_status ENUM('Pending','Completed'),

test_notes TEXT

);

```

### →medical\_records

```
mysql> desc medical_records;
```

Field	Type	Null	Key	Default	Extra
record_id	int	NO	PRI	NULL	auto_increment
patient_id	int	YES	MUL	NULL	
doctor_id	int	YES	MUL	NULL	
diagnosis	text	YES		NULL	
treatment	text	YES		NULL	
prescription	text	YES		NULL	
date_recorded	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED

7 rows in set (0.00 sec)

### CODE:

```

CREATE TABLE medical_records (

record_id INT AUTO_INCREMENT PRIMARY KEY,

patient_id INT,

```



```

doctor_id INT,

diagnosis TEXT,

treatment TEXT,

prescription TEXT,

date_recorded TIMESTAMP DEFAULT CURRENT_TIMESTAMP

```

```
);
```

### →patients

```
mysql> desc patients;
```

Field	Type	Null	Key	Default	Extra
patient_id	int	NO	PRI	NULL	auto_increment
first_name	varchar(50)	NO		NULL	
last_name	varchar(50)	NO		NULL	
date_of_birth	varchar(20)	YES		NULL	
gender	varchar(20)	YES		NULL	
marital_status	varchar(20)	YES		NULL	
blood_type	varchar(20)	YES		NULL	
contact_number	varchar(15)	YES		NULL	
emergency_contact_name	varchar(100)	YES		NULL	
emergency_contact_number	varchar(15)	YES		NULL	
address	varchar(255)	YES		NULL	
city	varchar(100)	YES		NULL	
state	varchar(100)	YES		NULL	
zip_code	varchar(20)	YES		NULL	
country	varchar(100)	YES		NULL	
email	varchar(100)	YES		NULL	
occupation	varchar(100)	YES		NULL	
registration_date	varchar(20)	YES		NULL	

18 rows in set (0.01 sec)

### CODE:

```

CREATE TABLE patients (

    patient_id INT AUTO_INCREMENT PRIMARY KEY,

    first_name VARCHAR(50) NOT NULL,

    last_name VARCHAR(50) NOT NULL,

    date_of_birth VARCHAR(20),

    gender VARCHAR(20),

    marital_status VARCHAR(20),

    blood_type VARCHAR(20),

    contact_number VARCHAR(15),

    emergency_contact_name VARCHAR(100),

    emergency_contact_number VARCHAR(15),

    address VARCHAR(255),

    city VARCHAR(100),

    state VARCHAR(100),

    zip_code VARCHAR(20),

```

```

country VARCHAR(100),

email VARCHAR(100),

occupation VARCHAR(100),

registration_date VARCHAR(20)

);

```

### →staff

```
mysql> desc staff;
```

Field	Type	Null	Key	Default	Extra
staff_id	int	NO	PRI	NULL	auto_increment
first_name	varchar(50)	NO		NULL	
last_name	varchar(50)	NO		NULL	
gender	enum('Male','Female','Other')	YES		NULL	
date_of_birth	date	YES		NULL	
contact_number	varchar(15)	YES		NULL	
email	varchar(100)	YES		NULL	
position	varchar(100)	YES		NULL	
department_id	int	YES	MUL	NULL	

9 rows in set (0.00 sec)

### CODE:

```

CREATE TABLE staff (

    staff_id INT AUTO_INCREMENT PRIMARY KEY,

    first_name VARCHAR(50) NOT NULL,

    last_name VARCHAR(50) NOT NULL,

    gender ENUM('Male','Female','Other'),

    date_of_birth DATE,

    contact_number VARCHAR(15),

    email VARCHAR(100),

    position VARCHAR(100),

    department_id INT,

    FOREIGN KEY (department_id) REFERENCES departments(department_id)

);

```

### →suppliers

```
mysql> desc suppliers ;
```

Field	Type	Null	Key	Default	Extra
supplier_id	int	NO	PRI	NULL	auto_increment
supplier_name	varchar(100)	NO		NULL	
contact_number	varchar(15)	YES		NULL	
email	varchar(100)	YES		NULL	
address	varchar(255)	YES		NULL	

5 rows in set (0.00 sec)

### CODE:

```
CREATE TABLE suppliers (  
    supplier_id INT AUTO_INCREMENT PRIMARY KEY,  
    supplier_name VARCHAR(100) NOT NULL,  
    contact_number VARCHAR(15),  
    email VARCHAR(100),  
    address VARCHAR(255)  
);
```

### →wards\_rooms

```
mysql> desc wards_rooms;
```

Field	Type	Null	Key	Default	Extra
ward_room_id	int	NO	PRI	NULL	auto_increment
ward_room_number	varchar(20)	NO		NULL	
ward_room_type	enum('ICU','General','Surgical','Maternity','Pediatric','Isolation','Other')	NO		NULL	
capacity	int	YES		NULL	
availability	enum('Available','Occupied')	YES		Available	

5 rows in set (0.00 sec)

### CODE:

```
CREATE TABLE wards_rooms (  
    ward_room_id INT AUTO_INCREMENT PRIMARY KEY,  
    ward_room_number VARCHAR(20) NOT NULL,  
    ward_room_type ENUM('ICU','General','Surgical','Maternity','Pediatric','Isolation','Other') NOT NULL,  
    capacity INT,  
    availability ENUM('Available','Occupied') DEFAULT 'Available'  
);
```

## 4. Creating a Database

Using MySQL server, create a new database for your hospital management system. You can do this with SQL commands or through the graphical interface.

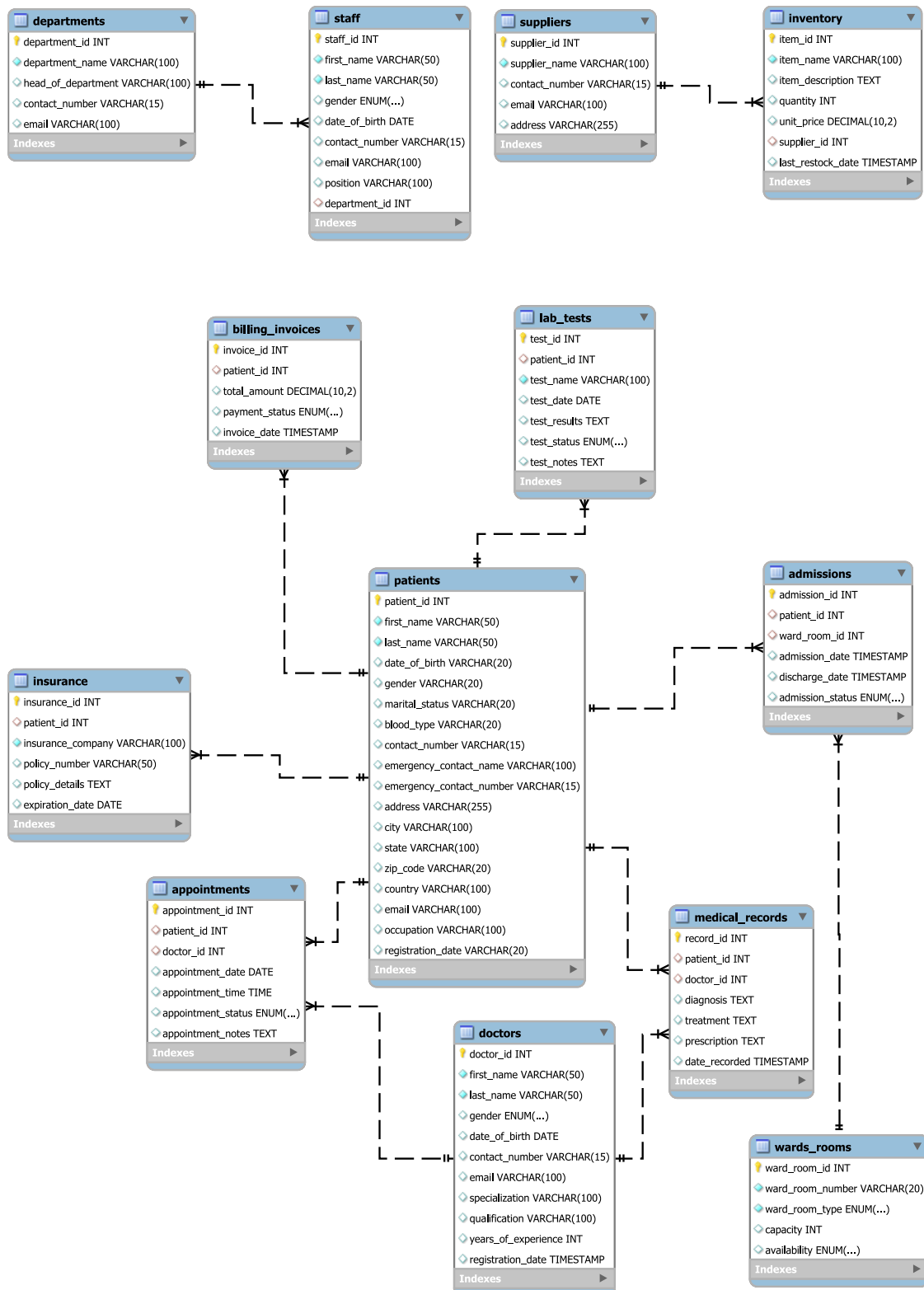
**CREATE DATABASE hospital;**

## 5. Using a Database

Before performing any operations on a database, you need to select it using the USE statement:

**USE hospital**

# ERD Diagram



PN: Ideally no data should be deleted from any tables. You can use an additional column to set the status of that record to 'Active/Inactive', etc. Or you can use an Archive table to move the unnecessary records out of the main table.

THANK YOU.....