**Entity Relationship Diagram - Document**

**Name of Database: Hospital Management Database**

**DATABASE DESCRIPTION:**

**This database contains 18 tables.**

**TABLE DESCRIPTIONS ARE AS FOLLOWS:**

1. Patients

Stores patient details.

* Patient\_ID (INT, PRIMARY KEY, NOT NULL)
* First\_Name (VARCHAR(50), NOT NULL)
* Last\_Name (VARCHAR(50), NOT NULL)
* DOB (DATE, NOT NULL)
* Gender (CHAR(1), NOT NULL)
* Contact\_Number (VARCHAR(15), NOT NULL)
* Email (VARCHAR(100), NULL)
* Blood\_Type (VARCHAR(3), NULL)
* Insurance\_Provider (VARCHAR(100), NULL)
* Insurance\_Number (VARCHAR(50), NULL)

2. Patient\_Addresses

Stores addresses of patients.

* Address\_ID (INT, PRIMARY KEY, NOT NULL)
* Patient\_ID (INT, FOREIGN KEY, NOT NULL)
* Address\_Line1 (VARCHAR(255), NOT NULL)
* Address\_Line2 (VARCHAR(255), NULL)
* City (VARCHAR(100), NOT NULL)
* State (VARCHAR(100), NOT NULL)
* ZIP\_Code (VARCHAR(10), NOT NULL)

3. Doctors

Stores doctor details.

* Doctor\_ID (INT, PRIMARY KEY, NOT NULL)
* First\_Name (VARCHAR(50), NOT NULL)
* Last\_Name (VARCHAR(50), NOT NULL)
* Specialization (VARCHAR(100), NOT NULL)
* Contact\_Number (VARCHAR(15), NOT NULL)
* Email (VARCHAR(100), NOT NULL)
* Experience\_Years (INT, NOT NULL)
* Department\_ID (INT, NULL)

4. Appointments

Stores appointment details between patients and doctors.

* Appointment\_ID (INT, PRIMARY KEY, NOT NULL)
* Patient\_ID (INT, FOREIGN KEY, NOT NULL)
* Doctor\_ID (INT, FOREIGN KEY, NOT NULL)
* Appointment\_Date (DATE, NOT NULL)
* Appointment\_Time (TIME, NOT NULL)
* Status (VARCHAR(50), NOT NULL)
* Reason (VARCHAR(255), NULL)

5. Medical\_Records

Stores patient medical records.

* Record\_ID (INT, PRIMARY KEY, NOT NULL)
* Patient\_ID (INT, FOREIGN KEY, NOT NULL)
* Doctor\_ID (INT, FOREIGN KEY, NOT NULL)
* Record\_Date (DATE, NOT NULL)

6. Diagnosis

Stores diagnoses and prescriptions related to medical records.

* Diagnosis\_ID (INT, PRIMARY KEY, NOT NULL)
* Record\_ID (INT, FOREIGN KEY, NOT NULL)
* Diagnosis\_Details (TEXT, NOT NULL)
* Prescription (TEXT, NULL)

7. Billing

Stores patient billing details.

* Bill\_ID (INT, PRIMARY KEY, NOT NULL)
* Patient\_ID (INT, FOREIGN KEY, NOT NULL)
* Total\_Amount (DECIMAL(10,2), NOT NULL)
* Bill\_Date (DATE, NOT NULL)
* Payment\_Status (VARCHAR(50), NOT NULL)

8. Payment\_Details

Stores payment transactions.

* Payment\_ID (INT, PRIMARY KEY, NOT NULL)
* Bill\_ID (INT, FOREIGN KEY, NOT NULL)
* Amount\_Paid (DECIMAL(10,2), NOT NULL)
* Payment\_Date (DATE, NOT NULL)
* Payment\_Mode (VARCHAR(50), NOT NULL)

9. Staff

Stores hospital staff details.

* Staff\_ID (INT, PRIMARY KEY, NOT NULL)
* First\_Name (VARCHAR(50), NOT NULL)
* Last\_Name (VARCHAR(50), NOT NULL)
* Role (VARCHAR(100), NOT NULL)
* Contact\_Number (VARCHAR(15), NOT NULL)
* Email (VARCHAR(100), NULL)
* Department\_ID (INT, NULL)
* Salary (DECIMAL(10,2), NOT NULL)
* Shift\_Timings (VARCHAR(50), NOT NULL)

10. Rooms

Stores room information.

* Room\_ID (INT, PRIMARY KEY, NOT NULL)
* Room\_Type (VARCHAR(50), NOT NULL)
* Capacity (INT, NOT NULL)
* Occupied\_Status (VARCHAR(50), NOT NULL)

11. Admissions

Stores patient admission records.

* Admission\_ID (INT, PRIMARY KEY, NOT NULL)
* Patient\_ID (INT, FOREIGN KEY, NOT NULL)
* Room\_ID (INT, FOREIGN KEY, NOT NULL)
* Admission\_Date (DATE, NOT NULL)
* Discharge\_Date (DATE, NULL)
* Diagnosis (TEXT, NOT NULL)

12. Inventory

Stores inventory details.

* Item\_ID (INT, PRIMARY KEY, NOT NULL)
* Item\_Name (VARCHAR(100), NOT NULL)
* Category (VARCHAR(100), NOT NULL)
* Quantity (INT, NOT NULL)
* Supplier\_ID (INT, FOREIGN KEY, NOT NULL)
* Purchase\_Date (DATE, NOT NULL)

13. Suppliers

Stores supplier details.

* Supplier\_ID (INT, PRIMARY KEY, NOT NULL)
* Supplier\_Name (VARCHAR(100), NOT NULL)
* Contact\_Number (VARCHAR(15), NULL)
* Email (VARCHAR(100), NULL)
* Address (TEXT, NULL)
* Category\_Provided (VARCHAR(100), NOT NULL)

14. Tests

Stores diagnostic test details.

* Test\_ID (INT, PRIMARY KEY, NOT NULL)
* Test\_Name (VARCHAR(100), NOT NULL)
* Description (TEXT, NULL)
* Price (DECIMAL(10,2), NOT NULL)

15. Lab\_Reports

Stores lab reports for patients.

* Report\_ID (INT, PRIMARY KEY, NOT NULL)
* Patient\_ID (INT, FOREIGN KEY, NOT NULL)
* Doctor\_ID (INT, FOREIGN KEY, NOT NULL)
* Report\_Date (DATE, NOT NULL)

16. Test\_Results

Stores results of medical tests.

* Result\_ID (INT, PRIMARY KEY, NOT NULL)
* Report\_ID (INT, FOREIGN KEY, NOT NULL)
* Test\_ID (INT, FOREIGN KEY, NOT NULL)
* Results (TEXT, NOT NULL)

17. Pharmacy

Stores medicine information.

* Medicine\_ID (INT, PRIMARY KEY, NOT NULL)
* Medicine\_Name (VARCHAR(100), NOT NULL)
* Category (VARCHAR(100), NOT NULL)
* Price (DECIMAL(10,2), NOT NULL)

18. Stock

Stores stock details of medicines.

* Stock\_ID (INT, PRIMARY KEY, NOT NULL)
* Medicine\_ID (INT, FOREIGN KEY, NOT NULL)
* Quantity\_Available (INT, NOT NULL)
* Expiry\_Date (DATE, NOT NULL)

**THE RELATIONSHIP DESCRIPTIONS ARE AS FOLLOWS:**

1. Patients and Patient\_Addresses

Type: One-to-Many  
Cardinality:

* Patients: 1 (Each patient must have at least one address)
* Patient\_Addresses: 1..\* (A patient can have multiple addresses)

2. Patients and Appointments

Type: One-to-Many  
Cardinality:

* Patients: 1 (A patient must have at least one appointment)
* Appointments: 0..\* (A patient can have multiple appointments or none)

3. Doctors and Appointments

Type: One-to-Many  
Cardinality:

* Doctors: 1 (A doctor must have at least one appointment)
* Appointments: 0..\* (A doctor can have multiple appointments or none)

4. Patients and Medical\_Records

Type: One-to-Many  
Cardinality:

* Patients: 1 (A patient must have at least one medical record)
* Medical\_Records: 0..\* (A patient can have multiple medical records or none)

5. Doctors and Medical\_Records

Type: One-to-Many  
Cardinality:

* Doctors: 1 (A doctor must have at least one medical record)
* Medical\_Records: 0..\* (A doctor can create multiple medical records or none)

6. Medical\_Records and Diagnosis

Type: One-to-One  
Cardinality:

* Medical\_Records: 1 (Each medical record must have one diagnosis)
* Diagnosis: 1 (Each diagnosis must belong to one medical record)

7. Patients and Billing

Type: One-to-Many  
Cardinality:

* Patients: 1 (Each patient must have at least one bill)
* Billing: 0..\* (A patient can have multiple bills or none)

8. Billing and Payment\_Details

Type: One-to-Many  
Cardinality:

* Billing: 1 (Each bill must have at least one payment)
* Payment\_Details: 0..\* (A bill can have multiple payments or none)

9. Patients and Admissions

Type: One-to-Many  
Cardinality:

* Patients: 1 (Each patient can be admitted at least once)
* Admissions: 0..\* (A patient can have multiple admissions or none)

10. Rooms and Admissions

Type: One-to-Many  
Cardinality:

* Rooms: 1 (Each room can have at most one admission at a time)
* Admissions: 0..\* (A room can be used for multiple admissions over time)

11. Inventory and Suppliers

Type: Many-to-One  
Cardinality:

* Inventory: 1..\* (Each inventory item must have one supplier)
* Suppliers: 1 (A supplier can supply multiple inventory items)

12. Patients and Lab\_Reports

Type: One-to-Many  
Cardinality:

* Patients: 1 (Each patient must have at least one lab report)
* Lab\_Reports: 0..\* (A patient can have multiple lab reports or none)

13. Doctors and Lab\_Reports

Type: One-to-Many  
Cardinality:

* Doctors: 1 (Each doctor must be associated with at least one lab report)
* Lab\_Reports: 0..\* (A doctor can be associated with multiple lab reports or none)

14. Lab\_Reports and Test\_Results

Type: One-to-Many  
Cardinality:

* Lab\_Reports: 1 (Each lab report must have at least one test result)
* Test\_Results: 0..\* (A lab report can have multiple test results or none)

15. Tests and Test\_Results

Type: One-to-Many  
Cardinality:

* Tests: 1 (Each test must be performed at least once)
* Test\_Results: 0..\* (A test can be performed multiple times or none)

16. Pharmacy and Stock

Type: One-to-Many  
Cardinality:

* Pharmacy: 1 (Each medicine must have at least one stock entry)
* Stock: 0..\* (A medicine can have multiple stock entries or none)