HOSPITAL DATABASE MANAGEMENT SYSTEM

UCS310 -DBMS Project Report End Semester Lab Evaluation Submitted By :

Ishrat Bhullar(102203144) Sparsh Khanna(102203081) Kushagra Nagpal(102213018) Aditya Bora(102203467)

Submitted To:

Mrs.Jhilik Bhattacharya



Contents

1. **Basic Introduction**
   1. **Database Management System**
   2. **Relational Database Management System**
   3. **ER diagram**
   4. **Terminologies and Symbols of ER diagram**
2. **Hospital Database Management System**
   1. **Introduction**
   2. **Objectives of the system**
   3. **ER diagram: Hospital Database**
   4. **Physical Schema**
   5. **SQL commands**
   6. **Normalisation**
   7. **PL/SQL Queries**

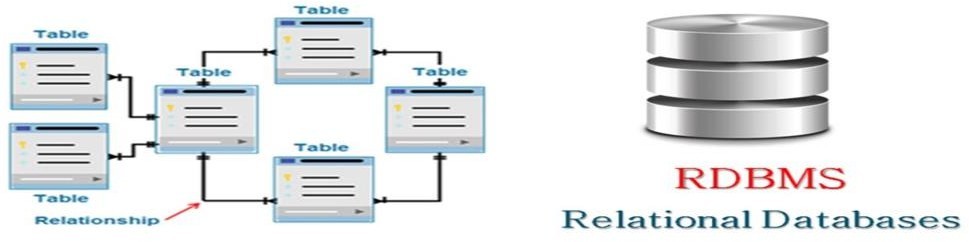


**BASIC INTRODUCTION**

# DATABASE MANAGEMENT SYSTEM: -

A Database Management System (DBMS) encompasses a compilation of interconnected data along with a suite of programs designed for accessing this data. Functioning as a repository of correlated information with inherent significance, it embodies the concept of a database. The fundamental objective of a DBMS revolves around furnishing a mechanism for storing and retrieving database information in a manner that is both expedient and effective.

# RELATIONAL DATABASE MANAGEMENT SYSTEM: -



***Fig. 2.2 : RDBMS***

An RDBMS, or Relational Database Management System, encompasses a suite of programs and functionalities that facilitate user interaction with a relational database. Distinguished by its row-based table structure, an RDBMS serves as a specialized type of DBMS tailored to manage relational databases.

# ER DIAGRAM: -

The Entity-Relationship (ER) model is a high-level data model utilized to delineate data elements and relationships within a particular system. Serving as the foundation for conceptual database design, it offers a simplified and user-friendly depiction of data. In ER modeling, the database structure is represented graphically through an entity-relationship diagram, providing a visual representation of entities, their attributes, and the relationships between them.



# TERMINOLOGIES AND SYMBOLS OF ER DIAGRAM: -

|  |  |
| --- | --- |
| **SYMBOL** | **DESCRIPTIO N** |
|  | This is a basic entity that is represented by a rectangle with its name inside. |
|  | This is an entity that cannot solely be identified with its attributes (due to the absence of a primary key). It inherits the identifier of its parent entityand often integrates it with a partial key. |
|  | A strong relationship is depicted by a single rhombus with its name inside.In this, an entity is independent, that is, its primary key for any child doesn't contain the primary key of the linked entity. |
|  | A weak relationship is depicted by a double rhombus with the name inside. In this, the child is. dependent on the parent entity as its primarykey would contain a component of the parent's primary key. |
|  | A basic attribute is represented by a single oval with its name writteninside. |
|  | This is a special attribute that is used to uniquely identify an entity. It isrepresented by an oval with its name underlined. |
|  | These are the attributes that can have multiple values (like the Name attribute can have First and Last name) and are represented by a doubleoval. |
|  | A derived attribute might not be physically present in the database andcould be logically derived from any other attribute (Represented by a dotted oval). |
|  | Composite attributes are those attributes which are composed of manyother simple attributes. |
|  | This depicts that not all the entities in the set are a part of the relationshipand is depicted by a single line. |
|  | This means that all the entities in the set are in a relationship and aredepicted by a double line. |

***Fig. 3.4 : Terminologies and Symbols of ER Diagram***

**HOSPITAL MANAGEMENT SYSTEM**

## Introduction: -

The surge in population and the emergence of new diseases, coupled with recent pandemics, have necessitated the establishment of new hospitals across various cities to ensure the delivery of quality healthcare services to patients.

Hospitals play a vital role in our lives, offering top-notch medical facilities to individuals grappling with a myriad of ailments stemming from factors such as climatic changes, increased workloads, emotional stress, and more. Given the pivotal role hospitals play, it is imperative for them to maintain meticulous records of their day-to-day operations, including patient information, staff details, and various activities critical to the smooth functioning of the facility.

However, relying on manual paper-based record-keeping systems proves to be both cumbersome and error-prone. With the continuous influx of patients and the expanding population, maintaining accurate records on paper becomes increasingly unreliable, inefficient, and time-consuming. Recognizing these challenges, we have developed an automated solution named the "Hospital Management System."

The primary objective of our project is to transition hospitals to a paperless environment, aiming for up to 90% reduction in paper usage. Additionally, our system focuses on delivering cost-effective and reliable automation of existing manual processes. By implementing our solution, hospitals can benefit from enhanced data security across all user-system interactions, along with robust storage and backup capabilities to safeguard critical information.



## Objectives of the system: -

The "Hospital Management System" project is geared towards streamlining the day-to-day operations of hospitals, encompassing tasks such as patient admission and discharge, doctor management, and report generation. To this end, the project aims to achieve the following objectives:

a) Digitizing all patient and hospital-related information for improved accessibility and efficiency.

b) Implementing an appointment scheduling system to facilitate seamless interactions between patients and doctors.

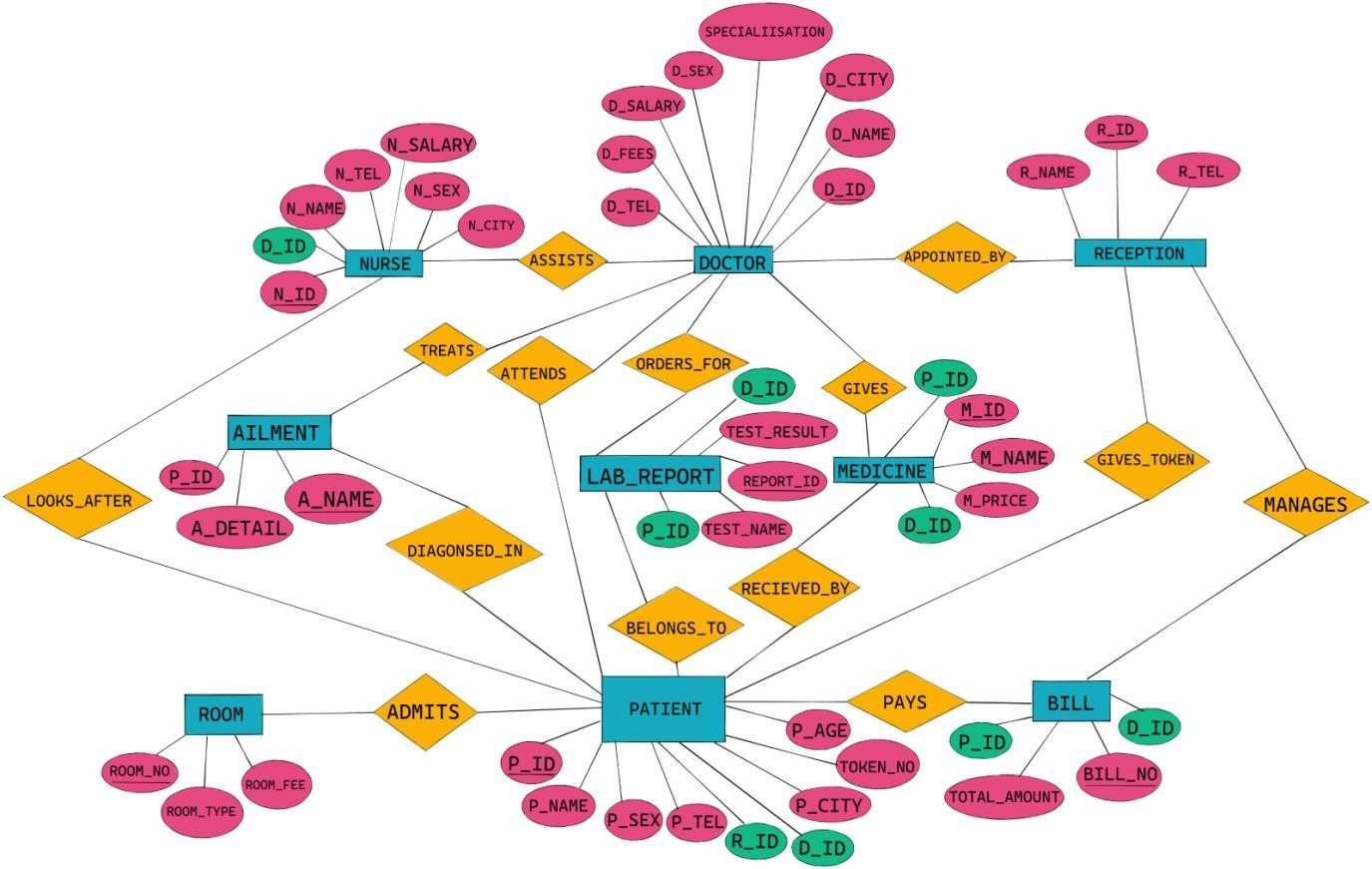
c) Optimizing the scheduling of specialized medical services and emergencies to maximize the utilization of hospital resources.

d) Ensuring seamless integration between the hospital's medical store and patient treatment records to facilitate accurate inventory management.

e) Facilitating the management of patient test reports conducted in the hospital's pathology lab.

f) Automating inventory updates to reflect real-time transactions and maintain accurate stock levels.

g) Maintaining up-to-date patient records within the system for historical reference and continuity of care.



## ER Diagram: Hospital Database: -

***Fig. 2.3 : ER Diagram***



**a) The patient's ID ('P\_ID')** serves as the key attribute, uniquely identifying patients and including their name, gender, address, contact details, date of birth, and admission details.

**b) The doctor's ID ('D\_ID')** acts as the key attribute, uniquely identifying doctors and including their name, contact details, service fee, salary, gender, and specialization.

**c)** To mitigate redundancy, medicine information is stored separately from prescriptions, with the medicine ID ('M\_ID') serving as the key attribute, along with the medicine name ('M\_NAME') and price ('M\_PRICE').

**d)** The bill for each patient is identified by the 'BILL\_NO.' (partial key) and 'TOTAL\_AMOUNT', capturing the total billed amount.

e) Nurse information includes the nurse ID ('N\_ID') as the key attribute, along with their name, contact details, and gender.

f) Room information for patients includes the room number ('ROOM\_NO') as the key attribute, along with room type and fee.

g) Ailment information for patients includes the ailment name ('A\_NAME') as the key attribute, along with ailment details.

h) Reception information encompasses the reception ID ('R\_ID') as the key attribute, along with reception name and contact details.

i) Lab report information includes the report ID ('REPORT\_ID'), test name ('TEST\_NAME'), and test result ('TEST\_RESULT').

**Relationships:**

**- APPOINTED\_BY:** Between Doctor and Reception, with a cardinality ratio of 1:n from Reception to doctor, both with total participation.

**- LOOKS\_AFTER:** Between Nurse and Patient, with a cardinality ratio of m:n for Nurse to Patient, featuring partial participation from Nurse and total from Patient.

**- ASSISTS**: Between Doctor and Nurse, with a cardinality ratio of 1:n from doctor to nurse, both with total participation.

**- TREATS:** Between Doctor and Ailment, with a cardinality ratio of m:n, featuring total participation from doctor and partial from ailment.

**- GIVES:** Between Medicine and Doctor, with a cardinality ratio of n:1 from Doctor to Medicine, both with total participation.

**- ATTENDS:** Between Doctor and Patient, with a cardinality ratio of 1:n from Doctor to Patient, both with total participation.

**- GIVES\_TOKEN**: Between Reception and Patient, with a cardinality ratio of 1:n from Reception to Patient, both with total participation.

**- MANAGES:** Between Reception and Bill, with a cardinality ratio of 1:n from Reception to Bill, both with total participation.

**- ADMITS:** Between Patient and Room, with a cardinality ratio of m:n, featuring partial participation from Patient and total from Room.

**- ORDERS\_FOR:** Between Doctor and Lab Report, with a cardinality ratio of 1:n from doctor to Lab Report.

**- BELONGS\_TO:** Between Lab Report and Patient, with a cardinality ratio of 1:n from Patient to Lab Report.

**- PAYS:** Between Bill and Patient, with a cardinality ratio of 1:n from Patient to Bill.

**- RECEIVED\_BY:** Between Medicine and Patient, with a cardinality ratio of 1:n from Medicine to Patient.

**- DIAGNOSED\_IN:** Between Ailment and Patient, with a cardinality ratio of 1:n from Patient to Ailment.

**- PAYS:** Between Bill and Patient, with a cardinality ratio of 1:n from Patient to Bill.



## Physical Schema: -

**Table Name: NURSE**

|  |  |  |
| --- | --- | --- |
| **FIELDS** | **DATA TYPE** | **RELATIONSHIPS** |
| N\_ID | NUMBER (5) | PRIMARY KEY |
| D\_ID | NUMBER (5) | FOREIGN KEY |
| N\_NAME | VARCHAR (20) | NOT NULL |
| N\_SEX | VARCHAR (2) | NOT NULL |
| N\_TEL | NUMBER (10) | NOT NULL |
| N\_CITY | VARCHAR (10) | NOT NULL |
| N\_SALARY | NUMBER (10) | NOT NULL |

**Table Name: DOCTOR**

|  |  |  |
| --- | --- | --- |
| **FIELDS** | **DATA TYPE** | **RELATIONSHIPS** |
| D\_ID | NUMBER (5) | PRIMARY KEY |
| D\_NAME | VARCHAR (20) | NOT NULL |
| D\_SEX | VARCHAR (2) | NOT NULL |
| D\_TEL | NUMBER (10) | NOT NULL |
| D\_CITY | VARCHAR (10) | NOT NULL |
| D\_SALARY | NUMBER (10) | NOT NULL |
| D\_FEES | NUMBER (10) | NOT NULL |
| SPECIALISATION | VARCHAR (50) | NOT NULL |

**Table Name: RECEPTION**

|  |  |  |
| --- | --- | --- |
| **FIELDS** | **DATA TYPE** | **RELATIONSHIPS** |
| R\_ID | NUMBER (5) | PRIMARY KEY |
| R\_NAME | VARCHAR (20) | NOT NULL |
| R\_TEL | NUMBER (10) | NOT NULL |

**Table Name: PATIENT**

|  |  |  |
| --- | --- | --- |
| **FIELDS** | **DATA TYPE** | **RELATIONSHIPS** |
| P\_ID | NUMBER (5) | PRIMARY KEY |
| P\_NAME | VARCHAR (20) | NOT NULL |
| P\_AGE | NUMBER (10) | NOT NULL |
| P\_SEX | VARCHAR (2) | NOT NULL |
| P\_TEL | NUMBER (10) | NOT NULL |
| P\_CITY | VARCHAR (10) | NOT NULL |
| D\_ID | NUMBER (5) | FOREIGN KEY |
| R\_ID | NUMBER (5) | FOREIGN KEY |
| TOKEN\_NO | NUMBER (5) | NOT NULL |

**Table Name: BILL**

|  |  |  |
| --- | --- | --- |
| **FIELDS** | **DATA TYPE** | **RELATIONSHIPS** |
| BILL\_NO | NUMBER (10) | PRIMARY KEY |
| P\_ID | NUMBER (5) | PRIMARY KEY |
| D\_ID | NUMBER (5) | PRIMARY KEY |
| TOTAL\_AMOUNT | NUMBER (10) | NOT NULL |

**Table Name: ROOM**

|  |  |  |
| --- | --- | --- |
| **FIELDS** | **DATA TYPE** | **RELATIONSHIPS** |
| ROOM\_NO. | NUMBER (5) | PRIMARY KEY |
| ROOM\_FEE | NUMBER (5) | NOT NULL |
| ROOM\_TYPE | VARCHAR (20) | NOT NULL |

**Table Name: AILMENT**

|  |  |  |
| --- | --- | --- |
| **FIELDS** | **DATA TYPE** | **RELATIONSHIPS** |
| A\_NAME | VARCHAR (50) | PRIMARY KEY |
| P\_ID | NUMBER (5) | PRIMARY KEY |
| A\_DETAIL | VARCHAR (50) | NOT NULL |

**Table Name: LAB\_REPORT**

|  |  |  |
| --- | --- | --- |
| **FIELDS** | **DATA TYPE** | **RELATIONSHIPS** |
| REPORT\_ID | NUMBER (5) | PRIMARY KEY |
| TEST\_NAME | VARCHAR (20) | NOT\_NULL |
| TEST\_RESULT | VARCHAR (10) | NOT NULL |
| D\_ID | NUMBER (5) | FOREIGN KEY |
| P\_ID | NUMBER (5) | FOREIGN KEY |

**Table Name: MEDICINE**

|  |  |  |
| --- | --- | --- |
| **FIELDS** | **DATA TYPE** | **RELATIONSHIPS** |
| M\_ID | NUMBER (5) | PRIMARY KEY |
| M\_NAME | VARCHAR (20) | NOT NULL |
| M\_PRICE | NUMBER (5) | NOT NULL |
| D\_ID | NUMBER (5) | FOREIGN KEY |
| P\_ID | NUMBER (5) | FOREIGN KEY |



## SQL Commands: -

**CREATE TABLE DOCTOR (D\_ID NUMBER (5),**

**D\_NAME VARCHAR (20),**

**D\_SEX VARCHAR (2),**

**D\_TEL NUMBER (10),**

**D\_CITY VARCHAR (10),**

**SPECIALISATION VARCHAR (50),**

**D\_FEES NUMBER (10),**

**D\_SALARY NUMBER (10),**

**CONSTRAINT PK\_DOCTOR PRIMARY KEY(D\_ID));**



-

**CREATE TABLE NURSE (N\_ID NUMBER (5),**

**N\_NAME VARCHAR (20),**

**N\_SEX VARCHAR (2),**

**N\_TEL NUMBER (10),**

**N\_CITY VARCHAR (10),**

**N\_SALARY NUMBER (10),**

**D\_ID NUMBER (5),**

**CONSTRAINT PK\_NURSE PRIMARY KEY (N\_ID), CONSTRAINT FK\_NURSE FOREIGN KEY(D\_ID)REFERENCES DOCTOR);**

****

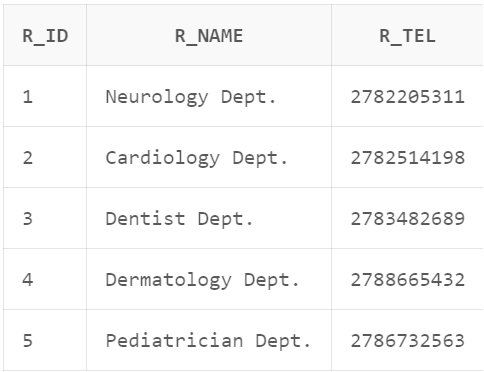


**CREATE TABLE RECEPTION (R\_ID NUMBER (5),**

**R\_NAME VARCHAR (20),**

**R\_TEL NUMBER (10),**

**CONSTRAINT PK\_RECEPTION PRIMARY KEY(R\_ID));**

****

**CREATE TABLE PATIENT (P\_ID NUMBER (5),**

**P\_NAME VARCHAR (20),**

**P\_AGE NUMBER (10),**

**P\_SEX VARCHAR (2),**

**P\_TEL NUMBER (10),**

**P\_CITY VARCHAR (10),**

**D\_ID NUMBER (5),**

**R\_ID NUMBER (5),**

**TOKEN\_NO NUMBER (5),**

**CONSTRAINT PK\_PATIENT PRIMARY KEY(P\_ID),**

**CONSTRAINT FK\_P1 FOREIGN KEY(D\_ID) REFERENCES DOCTOR(D\_ID), CONSTRAINT FK\_P2 FOREIGN KEY(R\_ID) REFERENCES RECEPTION(R\_ID));**

****



**CREATE TABLE MEDICINE (M\_ID NUMBER (5),**

**M\_NAME VARCHAR (20),**

**M\_PRICE NUMBER (5),**

**D\_ID NUMBER (5),**

**P\_ID NUMBER (5),**

**CONSTRAINT PK\_MEDICINE PRIMARY KEY(M\_ID),**

**CONSTRAINT FK\_MEDICINE FOREIGN KEY (D\_ID) REFERENCES DOCTOR, CONSTRAINT FK\_MEDICINE\_PATIENT FOREIGN KEY (P\_ID) REFERENCES PATIENT);**

****

**CREATE TABLE ROOM (ROOM\_NO NUMBER (5),**

**ROOM\_FEE NUMBER (5),**

**ROOM\_TYPE VARCHAR (20),**

**CONSTRAINT PK\_ROOM PRIMARY KEY(ROOM\_NO));**

****



**CREATE TABLE LAB\_REPORT (REPORT\_ID NUMBER (5),**

**TEST\_NAME VARCHAR (20),**

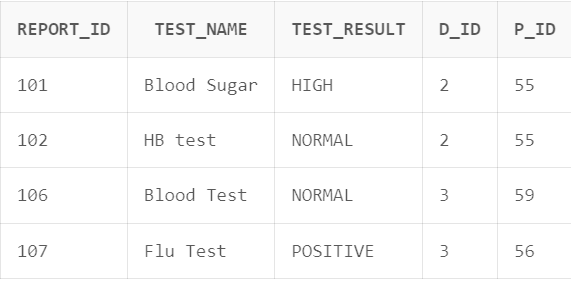
**TEST\_RESULT VARCHAR (10),**

**D\_ID NUMBER (5),**

**P\_ID NUMBER (5),**

**CONSTRAINT PK\_REPORT PRIMARY KEY(REPORT\_ID),**

**CONSTRAINT FK\_REPORT FOREIGN KEY (D\_ID) REFERENCES DOCTOR, CONSTRAINT FK\_REPORT\_PATIENT FOREIGN KEY (P\_ID) REFERENCES PATIENT);**

****

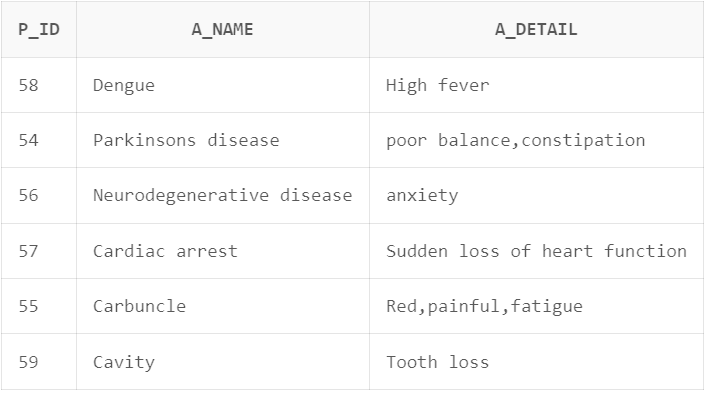
**CREATE TABLE AILMENT (P\_ID NUMBER (5),**

**A\_NAME VARCHAR (50),**

**A\_DETAIL VARCHAR (50),**

**CONSTRAINT PK\_AILMENT PRIMARY KEY (P\_ID, A\_NAME),**

**CONSTRAINT FK\_AILMENT FOREIGN KEY (P\_ID) REFERENCES PATIENT);**

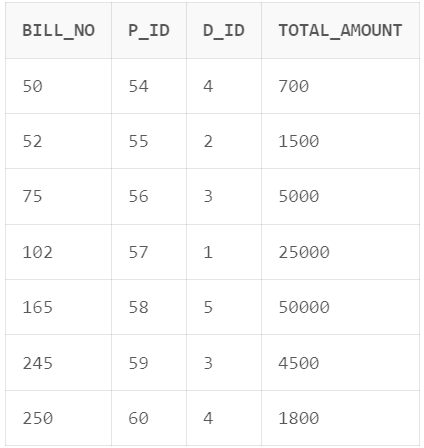
****



**CREATE TABLE BILL**

**( BILL\_NO NUMBER (10), P\_ID REFERENCES PATIENT, D\_ID REFERENCES DOCTOR, TOTAL\_AMOUNT NUMBER (10),**

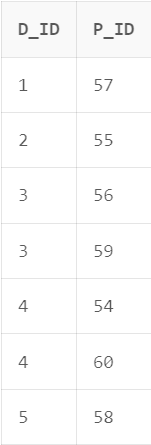
**CONSTRAINT PK\_BILL PRIMARY KEY (P\_ID, BILL\_NO, D\_ID));**

****

**CREATE TABLE TREATS (D\_ID NUMBER (5),**

**P\_ID NUMBER (5),**

**CONSTRAINT FK1\_TREATS FOREIGN KEY(D\_ID) REFERENCES DOCTOR(D\_ID), CONSTRAINT FK2\_TREATS FOREIGN KEY(P\_ID) REFERENCES PATIENT(P\_ID),CONSTRAINT PK\_TREATS PRIMARY KEY (D\_ID, P\_ID));**

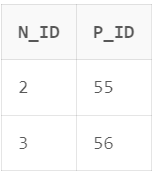
****



**CREATE TABLE LOOKS\_AFTER (N\_ID NUMBER (5),**

**P\_ID NUMBER (5),**

**CONSTRAINT FK\_NP1 FOREIGN KEY(P\_ID) REFERENCES PATIENT(P\_ID),CONSTRAINT FK\_NP2 FOREIGN KEY(N\_ID) REFERENCES NURSE(N\_ID), CONSTRAINT PK\_LOOKS\_AFTER PRIMARY KEY (N\_ID, P\_ID));**

****

**CREATE TABLE MANAGES (R\_ID NUMBER (5),**

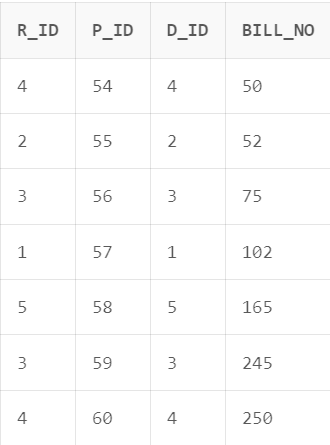
**P\_ID NUMBER (5),**

**D\_ID NUMBER (5),**

**BILL\_NO NUMBER (10),**

**CONSTRAINT FK\_MANAGES1 FOREIGN KEY(R\_ID) REFERENCES RECEPTION(R\_ID), CONSTRAINT FK\_MANAGESBILL FOREIGN KEY (BILL\_NO, P\_ID, D\_ID) REFERENCES BILL(BILL\_NO, P\_ID, D\_ID),**

**CONSTRAINT PK\_MANAGES PRIMARY KEY (BILL\_NO, P\_ID, R\_ID, D\_ID));**

****

**CREATE TABLE ADMITS (ROOM\_NO NUMBER (5),**

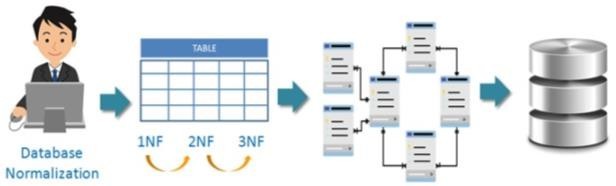
**P\_ID NUMBER (5), DATE\_OF\_ADMISSION DATE,**

**CONSTRAINT FK\_PRO1 FOREIGN KEY(P\_ID) REFERENCES PATIENT(P\_ID), CONSTRAINT FK\_PRO2 FOREIGN KEY(ROOM\_NO) REFERENCES ROOM(ROOM\_NO),CONSTRAINT PK\_ADMITS PRIMARY KEY (ROOM\_NO, P\_ID));**

****

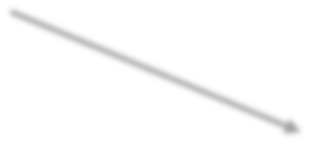
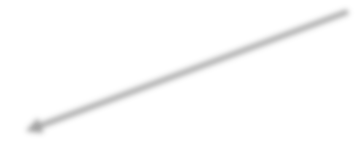


## Normalization: -

**PATIENT**

|  |  |
| --- | --- |
| **Database Type** | **Tables** |
| Normalized (1F) | DOCTOR, NURSE, RECEPTION, PATIENT, MEDICINE, ROOM, LAB\_REPORT, AILMENT, BILL, TREATS, LOOKS\_AFTER, MANAGES, ADMITS |
| Normalized (2F) | DOCTOR, NURSE, RECEPTION, PATIENT, MEDICINE, ROOM, LAB\_REPORT, AILMENT, BILL, TREATS, LOOKS\_AFTER, MANAGES, ADMITS |
| Normalized (3F) | DOCTOR, NURSE, RECEPTION, BILL, TREATS, LOOKS\_AFTER, MANAGES, ADMITS |

****

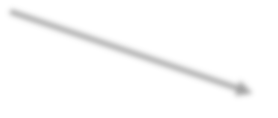
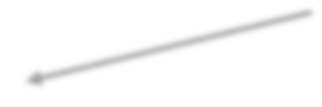


**APPOINTMENT PATIENT DETAILS**

**** ****

**MEDICINE**

****



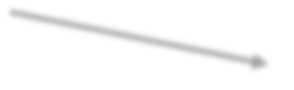
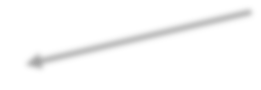
**PRESCRIBES**

**MEDICINE DETAILS**

** **

**ROOM**

****

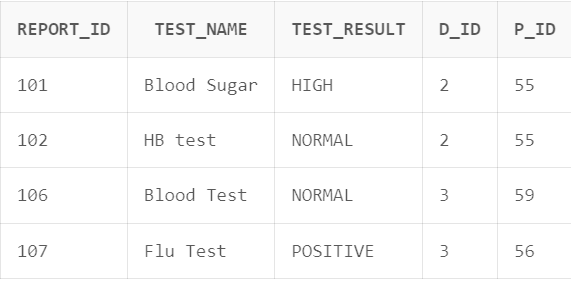


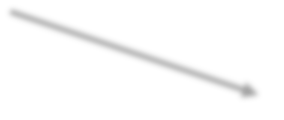
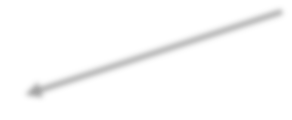
**ROOM DETAIL**

**ROOM CHARGES**

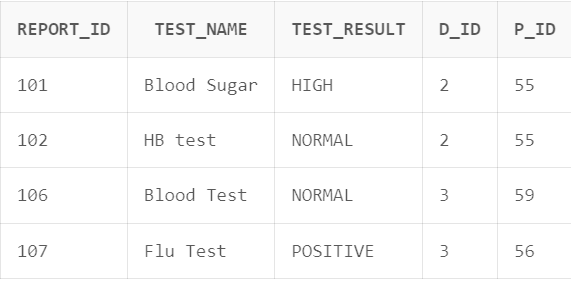
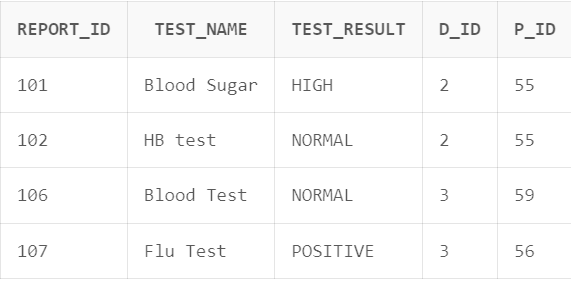
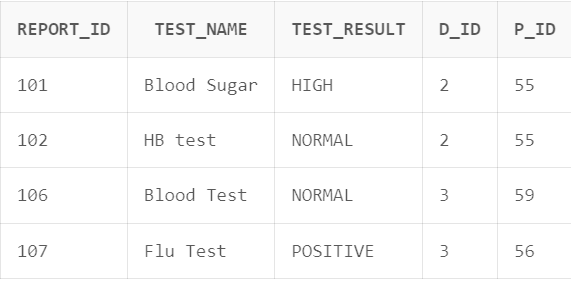
** **

**LAB REPORT**

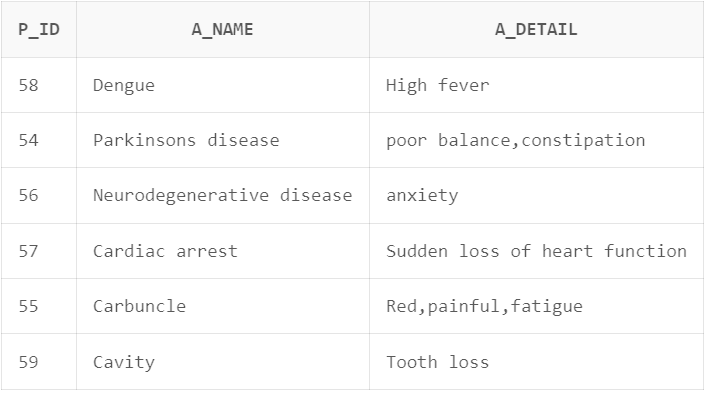
****

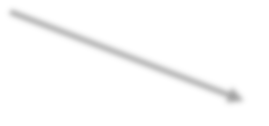
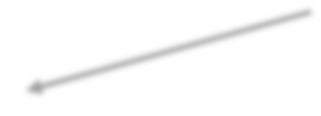


**LAB TESTS TEST REPORT**

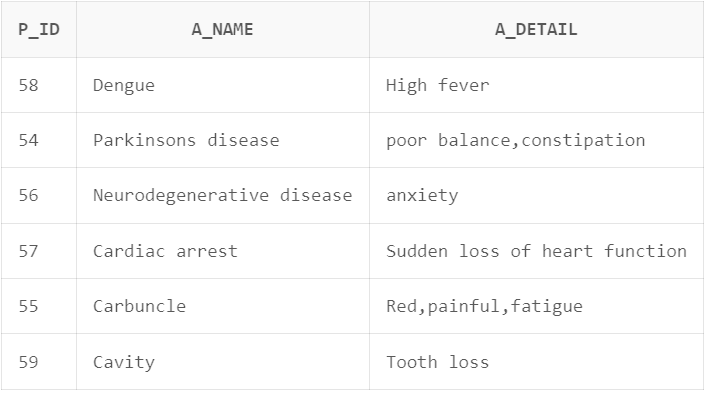
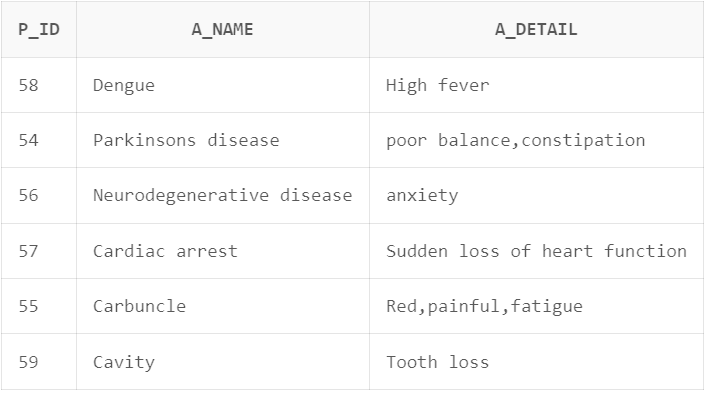
**** ****

**AILMENT**

****



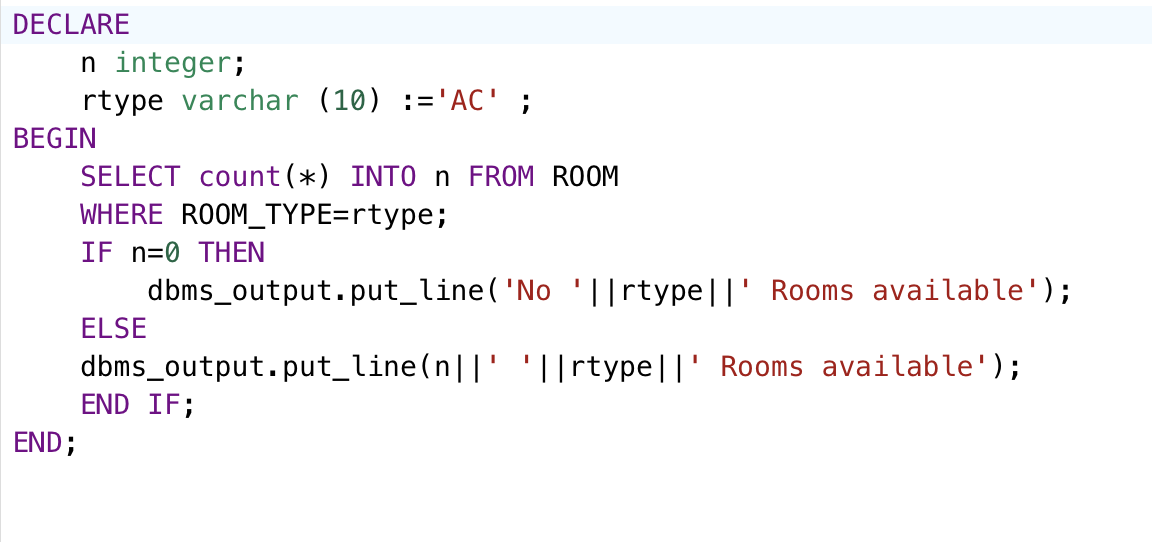
**DIAGNOSIS AILMENT DETAILS**

**** ****



## PL/SQL Queries: -

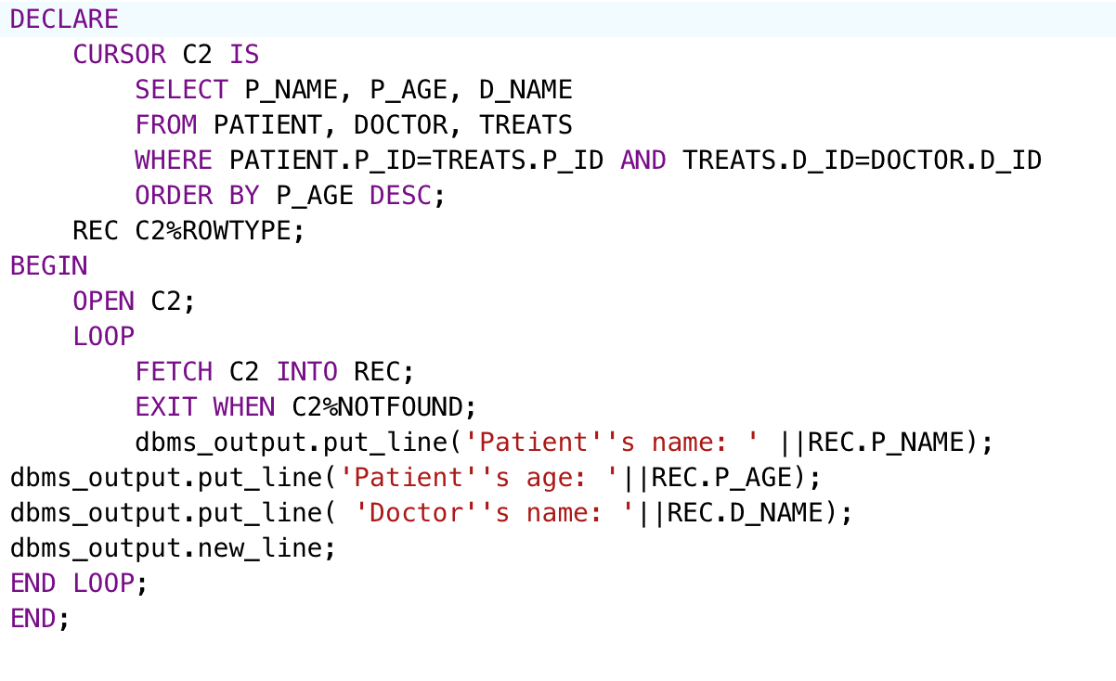
1. **Write a SQL query to count the number of “AC rooms” available. Return the count as “Number of AC Rooms”.**

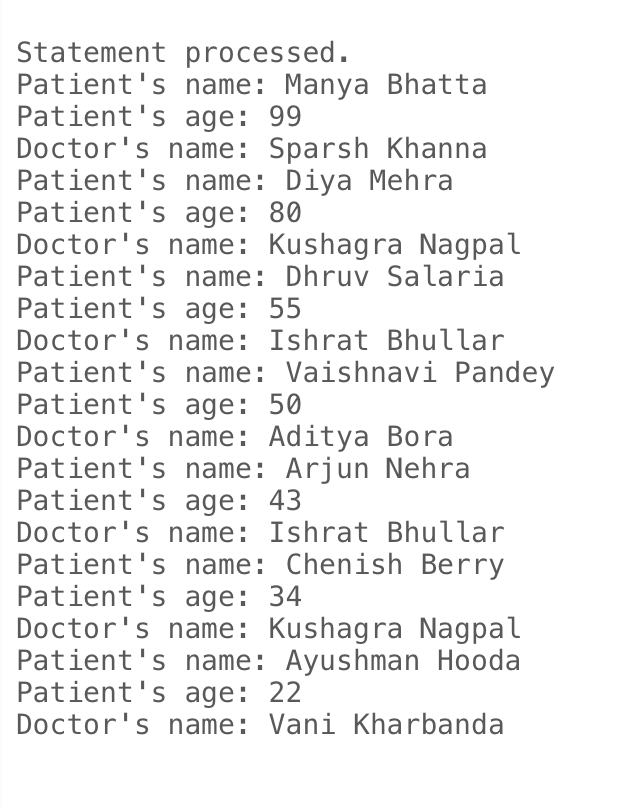
****

**A black text on a white background

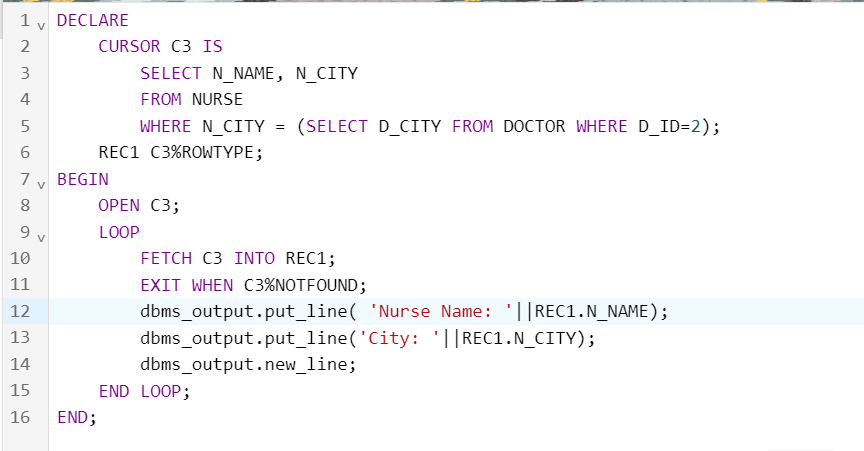
Description automatically generated**

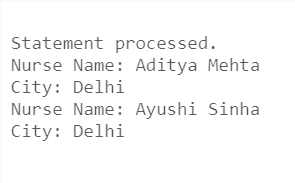
1. **List the name of the patient descending order of their age with their doctor's name.**

****

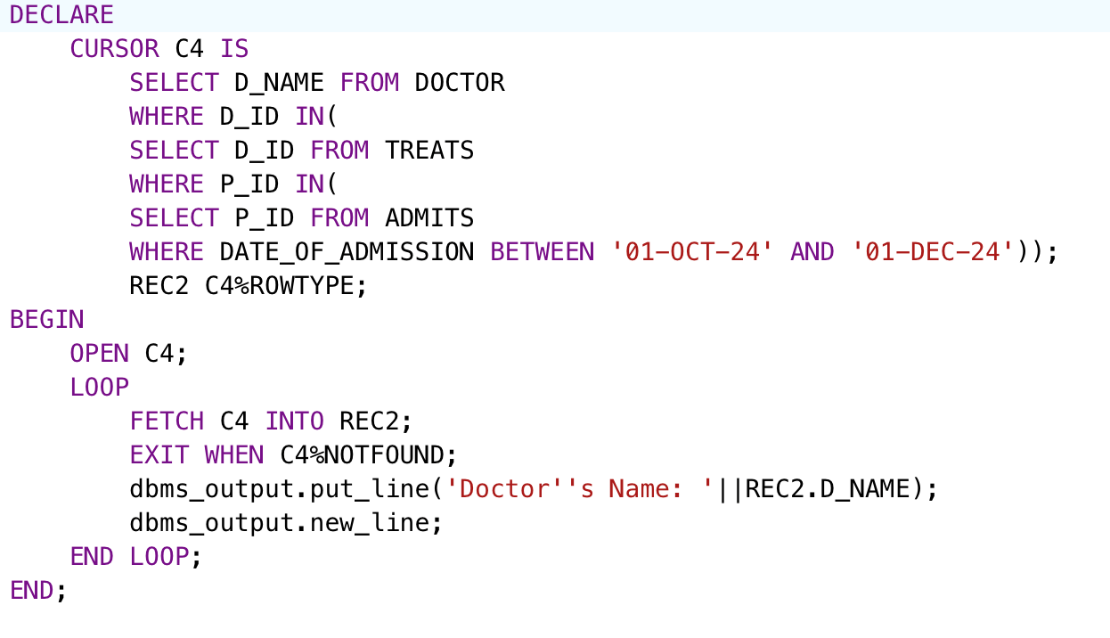


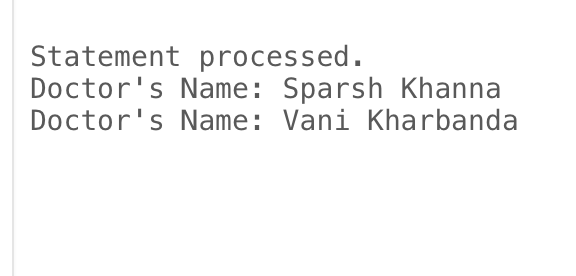
1. **List the name of the nurses who have the same city as the doctor id‘2’**

****

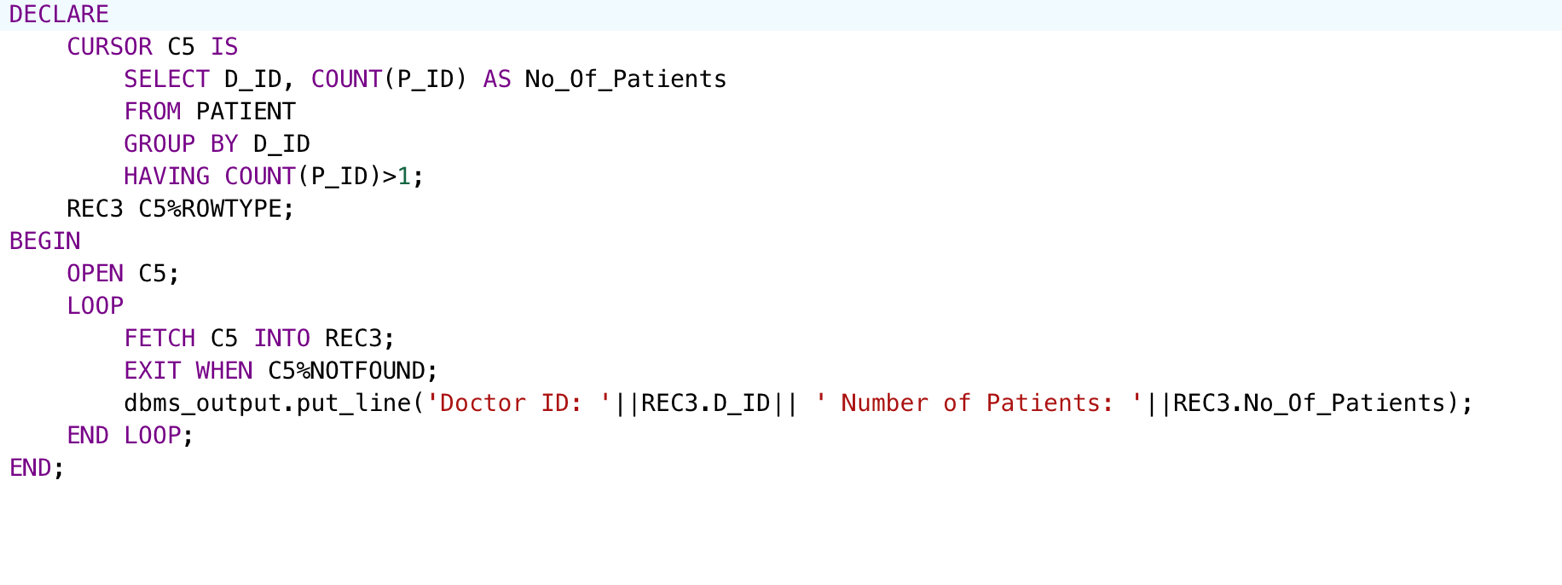
****

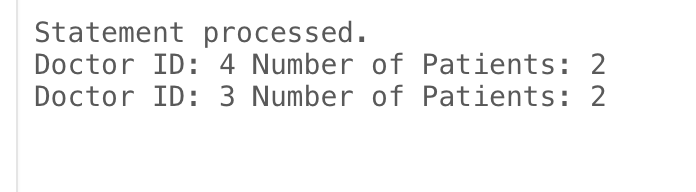
1. **List the name of the doctor who treat the patients with the date of admission between 01-Oct-2021 and 01-Dec-2021**

****

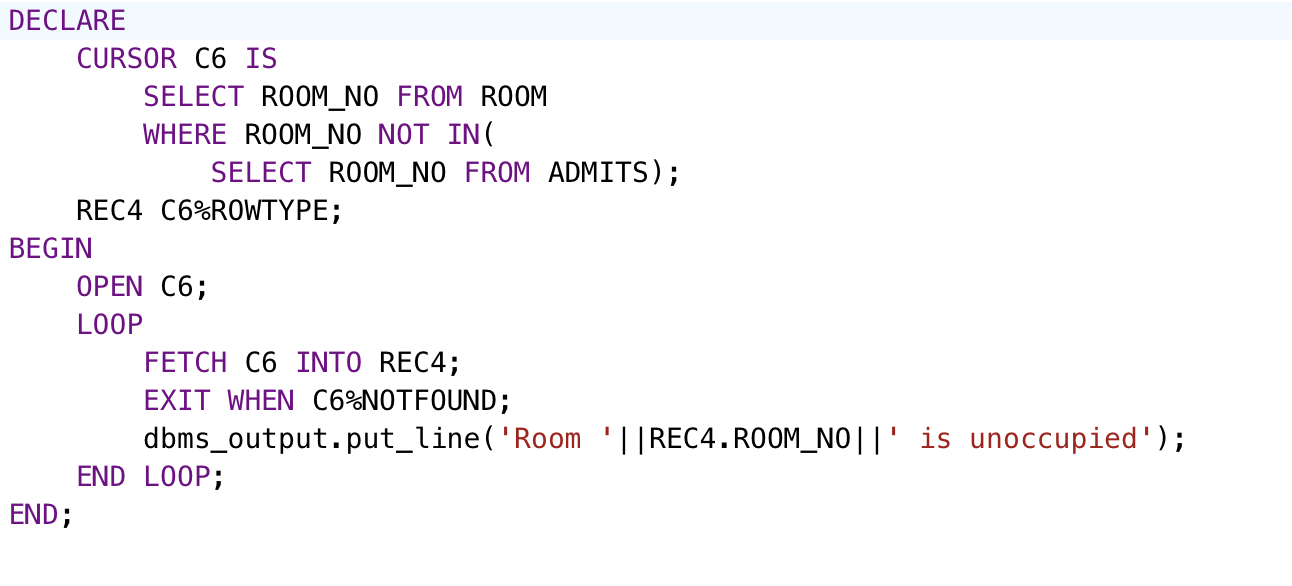
****

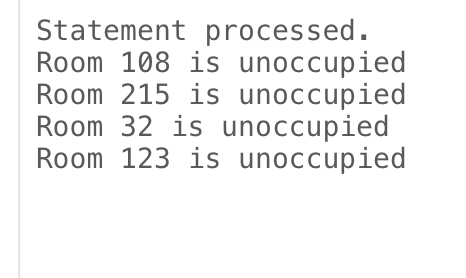
1. **List the name of doctors treating more than 1 patient.**

****



1. **List the room numbers that are unoccupied.**



****