

**HOSPITAL MANAGEMENT SYSTEM**

**SUBMITTED FOR THE COURSE,  
DATABASE MANAGEMENT SYSTEM**

**ON**

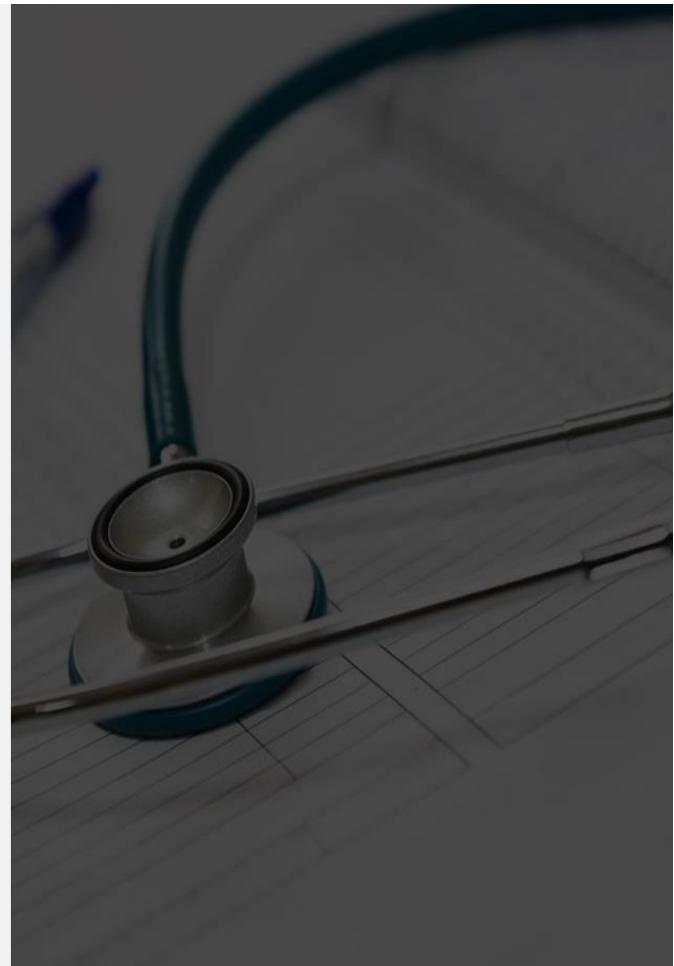
**MAY 31, 2021**

**BY**

**2019103523 | HARIPRIYA K**

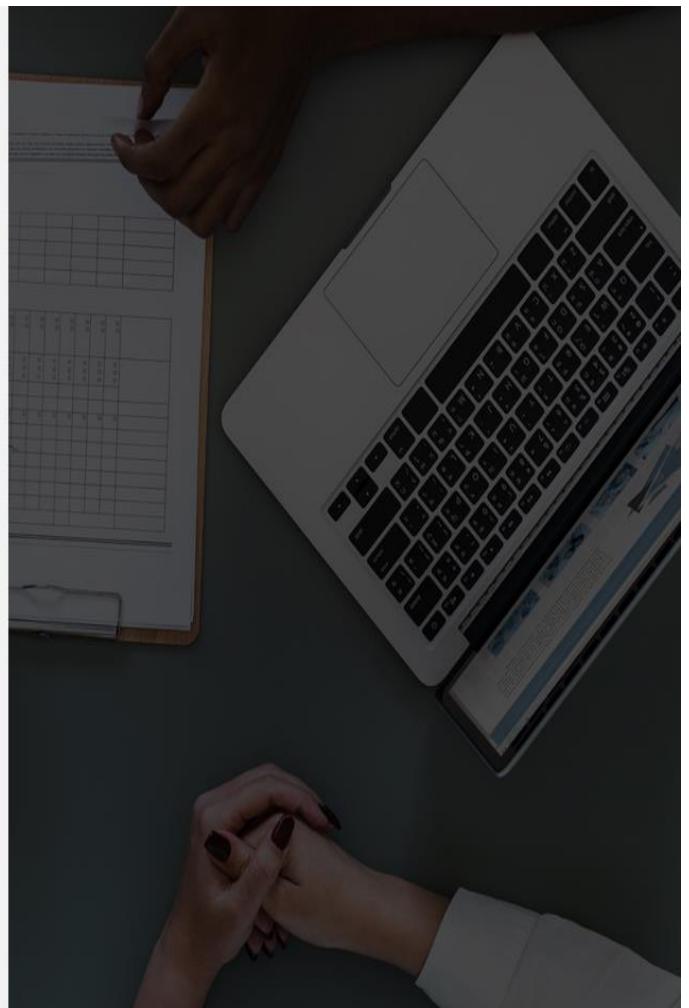
**2019103526 | HEMAVARSHINI C**

**2019103553 | POOJASRI S**



# **CONTENT**

- INTRODUCTION - 3**
- FUNCTIONAL REQUIREMENTS - 4**
- NON-FUNCTIONAL REQUIREMENTS - 8**
- ER DIAGRAM - 9**
- SCHEMA DESIGN - 10**
- USER-INTERFACE DESIGN - 14**
- IMPLEMENTATION - 15**
- RESULTS SNAPSHOTS - 16**
- TABLES - 37**
- SCHEMAS - 41**
- QUERIES AND SUBQUERIES - 44**
- CONCLUSION AND REFERENCES - 56**



## **ABSTRACT:**

- ✓ A database is a collection of data in an organized manner so that its content can easily be accessed, retrieved, managed and updated as per need.
- ✓ It is a think of electronic filing system. DBMS (Database Management System) is a keeper of database facilitating the creation and maintenance of database.
- ✓ It provides fast secure access to the data in the database.
- ✓ A database contains objects which are: tables, views, indexes, sequences, types, packages, procedures, functions, triggers, database links, materialized view and synonyms.

## **INTRODUCTION:**

- ✓ Hospital are the essential part of our lives, providing best medical facilities to people suffering from various ailments, which may be due to change in climatic conditions, increased work-load, emotional trauma stress etc.
- ✓ It is necessary for the hospitals to keep track of its day-to-day activities & records of its patients, doctors, inpatients, outpatients and other staff personals that keep the hospital running smoothly & successfully. But keeping track of all the activities and their records on paper is very cumbersome and error prone.
- ✓ It is also very inefficient and a time-consuming process Observing the continuous increase in population and number of people visiting the hospital recording and maintaining all these records is highly unreliable, inefficient and error- prone.
- ✓ It is also not economically & technically feasible to maintain these records on paper. Thus, keeping the working of the manual system as the basis of our project.
- ✓ We have developed an automated version of the manual system, named as "Hospital Management System".
- ✓ The main aim of our project is to provide a paper-less hospital up to 90%. It also aims at providing low-cost reliable automation of the existing systems.
- ✓ The system also provides excellent security of data at every level of user-system interaction and also provides robust & reliable storage.

## **FUNCTIONAL REQUIREMENTS:**

### **DOCTOR:**

#### **DESCRIPTION:**

Ideally, medical software should have a section with information about physicians that provide their service at the establishment. The section should contain a list of the available doctors with their respective unique id's and expertise of the medical professionals.

#### **EVENTS:**

##### a) ADD DOCTOR:

It facilities the admin to add a new doctor by filling the form with a unique id and their field of expertise and their name when a new doctor joins in service.

##### b) DELETE DOCTOR:

It facilities the admin to delete the doctor who resigns for one or more reason. It can be easily deleted by entering the doctor id in the given text box.

### **STAFF:**

#### **DESCRIPTION:**

It is also mandatory to keep account of the staffs who are working in the hospitals. This section should contain the list of the staffs available with their name, unique id and their salary. The staffs can belong to the category like nurse, cleaner, sweeper, ward boy, receptionist, lab technicians.

#### **EVENTS:**

##### a) ADD STAFF:

It facilities the admin to add a new staff by filling the form with necessary details like name, unique id and their salary whenever a new staff joins in.

##### b) DELETE STAFF:

It provides the admin to delete a particular staff when a staff resigns for one or more reason. The admin can delete a staff by simply giving the staff id who is no more working in the hospital.

## INPATIENT:

### DESCRIPTION:

Inpatient case management has the potential to improve both processes and outcomes of hospital care. This section lists all the inpatient's details with their relevant information like name, gender, address, patient's unique id, treating doctor id, room no, date of admit, date of discharge.

### EVENTS:

#### a) ADD INPATIENT:

Inpatient care starts with the admission to the hospital for medical treatment. Whenever a new patient visits as an inpatient the admin must register the patient details by filling the form.

## OUTPATIENT:

### DESCRIPTION:

First and foremost, management software should have hospital registration software to provide the possibility to register outpatients into the hospital's system and file in all relevant information. Outpatient care involves any sort of care provided without admission into the hospital. This section list all the outpatient's details which includes the patient's unique id, name, date, their treating doctor id.

### EVENTS:

#### a) ADD OUTPATIENT:

Procedure within an outpatient clinic include filling the form with the necessary details like the name, patient's unique id, date and their treating doctor id which is done by the admin.

## ROOMS:

### DESCRIPTION:

A good hospital management software must list the room details with their availability and non-availability status. This section lists the room no, room type, status, patient-id. The patient id will be included only when it is occupied by a patient else it would be left unfilled.

### EVENTS:

#### a) UPDATE ROOM:

It is very essential to update the room whenever the patient gets admitted and when the patient gets discharged. It is a mandatory for the admin to update the room with the status 'NON- AVAILABLE' and fill it with the patient's id who accommodates the room. The similar updating must be done when a patient gets discharged where the admin must update the room no with the status 'AVAILABLE'.

## LAB:

### DESCRIPTION:

Laboratory management is quite an essential part of every hospital management system. This section lists the lab no, date, patient id who takes test and the doctor id who have made reference to take test.

### EVENTS:

#### ADD LAB DETAILS:

This allows the admin to add the details like lab no, date, doctor id along with patient id who takes the test.

## BILL:

### **DESCRIPTION:**

A hospital management system should have a hospital billing system that provides the possibility to manage and monitor finances. This section lists the details like the bill no which is unique, patient-id, medical fee, room fee and no of days. The room fee and no of days will be zero in case of outpatients and will be filled in case of the inpatients.

### **EVENTS:**

#### a) ADD BILL DETAILS:

The admin must add the bill details of the patients who have taken treatment by filling the form with necessary details like bill no, patient-id, medical fee, room fee and no of days.

#### b) PAYMENT:

This function enables the admin to compute the total bill of a particular patient-id by simply giving the patient id whose bill amount is to be computed.

## **NON- FUNCTIONAL REQUIREMENTS:**

### **i) PERFORMANCE:**

#### **USER INTERFACE:**

The user-interface shall respond within 5 seconds.

### **ii) SECURITY:**

#### **LOGIN ID:**

Any user who uses the system shall have a Login id, password and username.

#### **MODIFICATION:**

Any modification (insert, delete, update) for the database shall be synchronized only by the administrator.

#### **ADMINISTRATOR-RIGHTS:**

Administrator shall be able to view and modify all information.

### **iii) AVAILABILITY:**

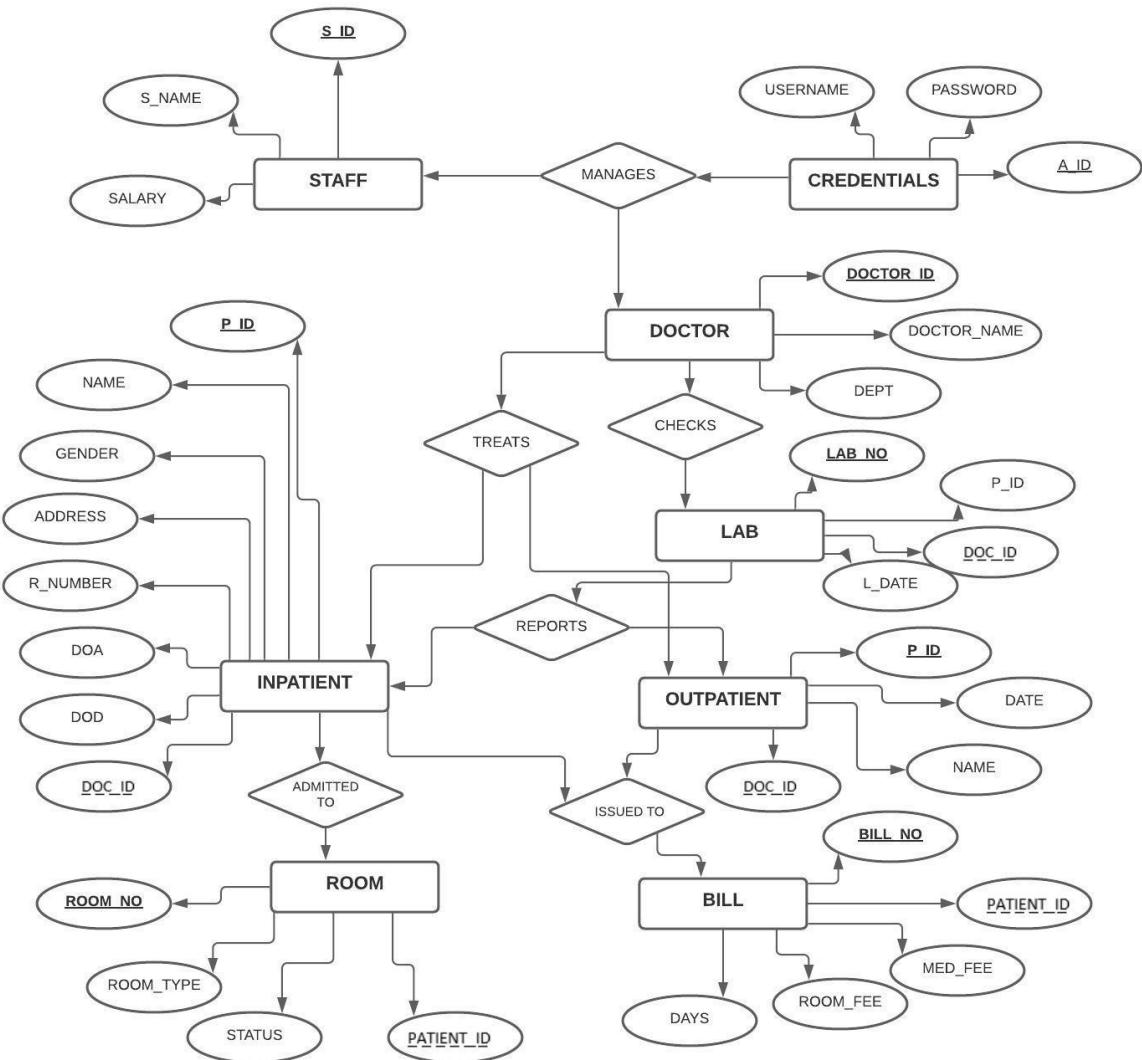
The system shall be available at the working time of the hospital.

### **iv) SOFTWARE QUALITY ATTRIBUTES:**

The quality of the system is maintained in such a way so that it can be very user-friendly. The software quality attributes are assumed as under:

- Accurate and hence reliable
- Secured
- Fast Speed
- Compatibility

## ER DIAGRAM:



## SCHEMA DESIGN:

### 1.CREDENTIALS:

<u>FIELDS</u>	<u>DATA TYPE</u>	<u>RELATIONSHIPS</u>
Username	Varchar (100)	Allow Null
Password	Varchar (100)	Primary key
a_id	Varchar (100)	Allow Null

### 2.STAFF:

<u>FIELDS</u>	<u>DATA TYPE</u>	<u>RELATIONSHIPS</u>
S_id	Varchar (50)	Primary key
S_name	Varchar (100)	Allow null
Salary	Int (11)	Allow Null

### 3.DOCTOR:

<u>FIELDS</u>	<u>DATA TYPE</u>	<u>RELATIONSHIPS</u>
Doctor_id	Varchar (50)	Primary key
Doctor_name	Varchar (100)	Allow null
dept	Varchar (100)	Allow null

### 4.IN-PATIENT:

<u>FIELDS</u>	<u>DATA TYPE</u>	<u>RELATIONSHIPS</u>
P_id	Varchar (20)	Primary key
Name	Varchar (150)	Allow null
Gender	Varchar (10)	Allow null
Address	Varchar (500)	Allow null
R_number	Int (11)d	Allow null
DOA	Date	Allow null
DOD	Date	Allow null
Doc_id	Varchar (10)	Foreign key

## **5.OUT-PATIENT:**

<b><u>FIELDS</u></b>	<b><u>DATA TYPE</u></b>	<b><u>RELATIONSHIPS</u></b>
P_id	Varchar (20)	Primary key
DATE	Date	Allow null
Name	Varchar (100)	Allow null
Doc_id	Varchar (10)	Allow null

## **6.ROOM:**

<b><u>FIELDS</u></b>	<b><u>DATA TYPE</u></b>	<b><u>RELATIONSHIPS</u></b>
room_no	Int (11)	Primary key
room_type	Varchar (50)	Allow null
status	Varchar (50)	Allow null
Patient_id	Varchar (20)	Allow null

## 7.LAB:

<u>FIELDS</u>	<u>DATA TYPE</u>	<u>RELATIONSHIPS</u>
Lab_no	Int (11)	Allow null
P_id	Varchar (20)	Allow null
Doc_id	Varchar (20)d	Foreign key
L_date	Date	Allow null

## 8.BILL:

<u>FIELDS</u>	<u>DATA TYPE</u>	<u>RELATIONSHIPS</u>
Bill_no	Int (11)	Primary key
P_id	Varchar (50)	Allow null
Med_fee	Int (11)	Allow null
room_fee	Int (11)	Allow null
Days	Int (11)	Allow null

## **USER-INTERFACE DESIGN:**

- ❖ Hospital management system is the system which is developed to minimize the pen and paper work at the hospitals.
- ❖ This system includes the registration of inpatients and outpatients and storing their details in the system.
- ❖ Alongside it maintains the records of the doctors, staffs, room and lab details along with the bill details.
- ❖ It allows the admin to add new doctor, new staff, new lab and new bill details.
- ❖ In addition, it also allows the admin to delete a specific doctor and staff information when they resign or leave for one or more reason.
- ❖ This system gives access to the admin to update the room details whenever a patient gets admitted and when the patient gets discharged.
- ❖ The overall labs present in the hospital can be known by clicking a button “lab details” in the lab page which is directed after clicking the lab button.
- ❖ It provides the facility for the admin to calculate the total bill amount by clicking the button “payment” in the bill page which is directed after clicking the bill button.
- ❖ The whole process is conducted by the Administrator.

## **IMPLEMENTATION:**

The various system tools that have been used in developing both the front end, back end of the project is being discussed in this section:

### **1.FRONT END:**

HTML, CSS are utilized to implement the frontend.

#### HTML (Hyper Text Mark-up Language):

HTML is a syntax used to format a text document on the web.

#### CSS (Cascading Style Sheet):

CSS is a style sheet language used for describing the look and formatting of a document written in a mark-up language.

### **2.BACKEND:**

The back end is implemented using MYSQL which is used to design the databases.

#### MYSQL:

MYSQL is the world's second most widely used open source relational database management system (RDMS). The SQL phrase stands for structured query.

#### PHP:

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. PHP code is interpreted by a web server with a PHP processor module, which generates the resulting web page: PHP commands can be embedded directly into an HTML document and also implemented by calling an external file to process data.

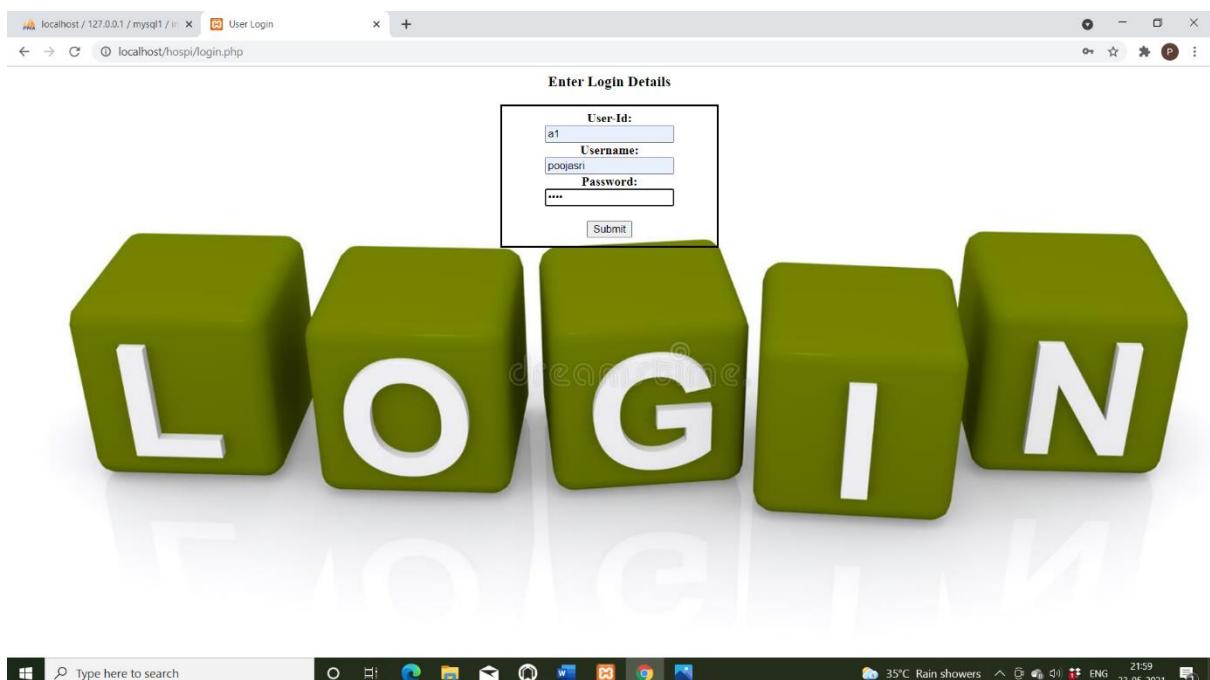
### **IMPLEMENTATION CONSTRAINTS:**

- System is wirelessly networked with an encryption.
- System is only accessible within the hospital premises only.
- Database is password protected.
- Should use less RAM and processing power.
- Only administrator can access the whole system.

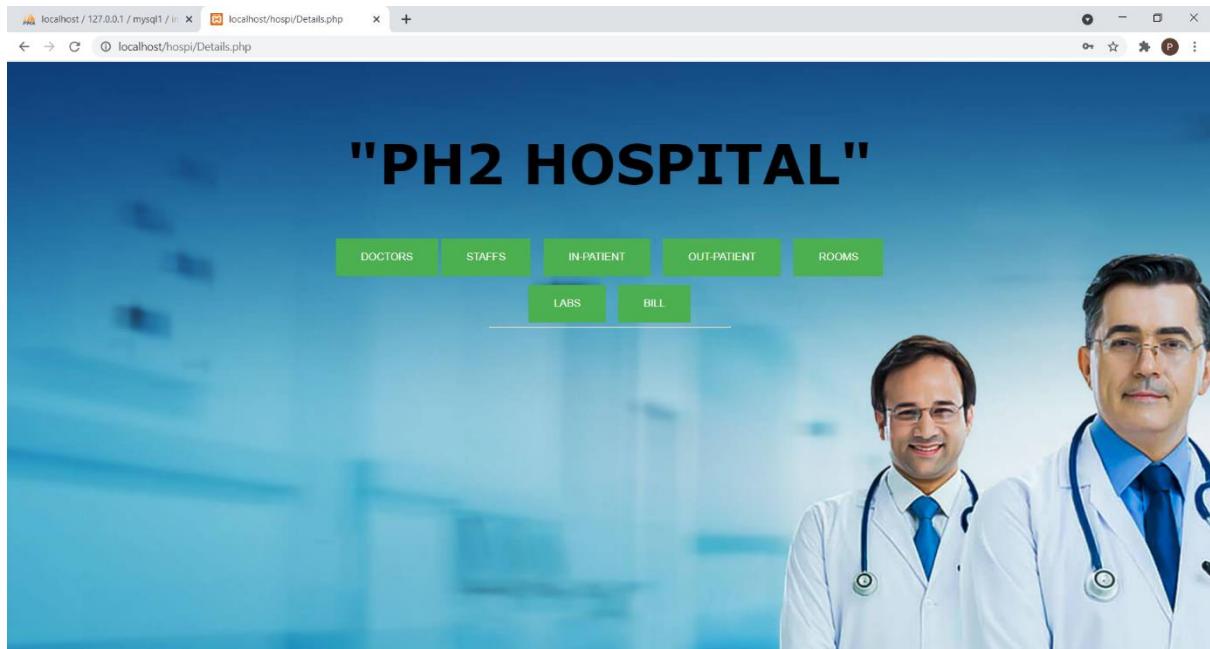
## RESULTS SNAPSHTOS:



After clicking the “admin login” button:



After clicking submit button it is directed to the below page:



## 1) DOCTORS:

On clicking the “DOCTORS” button it directs to the below page where it displays the list of doctors:

LIST OF DOCTORS		
<a href="#">Insert New doctor</a> <a href="#">Delete doctor</a> <a href="#">Go Back</a>		
DOCTOR_ID	NAME	DEPARTMENT
D3	DHANU	DENTIST
D4	DIVYA	GYNECOLOGIST
D8	HEMA	ENT
D1	KARAN	CARDIOLOGIST
D7	POOJA	DENTIST
D10	PRIYA	SURGEON
D11	RAMIYA S	CARDIOLOGIST
D2	SANDIYA	DERMATOLOGIST
D6	SHANTHANU	PHYSICIAN
D5	SUBHI	ENT
D9	VARSHINI	CARDIOLOGIST

On clicking the “Insert New doctor” it takes to the following page where we must enter the details of the doctor to be added:

localhost / 127.0.0.1 / mysql1 / doctor / doctor.php

Enter Doctor Details

DOCTOR-ID	D12
DOCTOR-NAME	SOUNDARYA S
DEPARTMENT	GYNACAELOGIST
<input type="button" value="submit"/>	

On clicking submit button the newly added record gets added to the doctor table and gets displayed along with the List of doctors:

localhost / 127.0.0.1 / mysql1 / doctor / doctor.php

LIST OF DOCTORS

DOCTOR_ID	NAME	DEPARTMENT
D3	DHANU	DENTIST
D4	DIVYA	GYNECOLOGIST
D8	HEMA	ENT
D1	KARAN	CARDIOLOGIST
D7	POOJA	DENTIST
D10	PRIYA	SURGEON
D11	RAMYA S	CARDIOLOGIST
D2	SANDIYA	DERMATALOGIST
D6	SHANTHANU	PHYSICIAN
D12	SOUNDARYA S	GYNACAELOGIST
D5	SUBHI	ENT
D9	VARSHINI	CARDIOLOGIST

On clicking the “Delete doctor” button it takes to the below page where we must give the doctor id to delete:

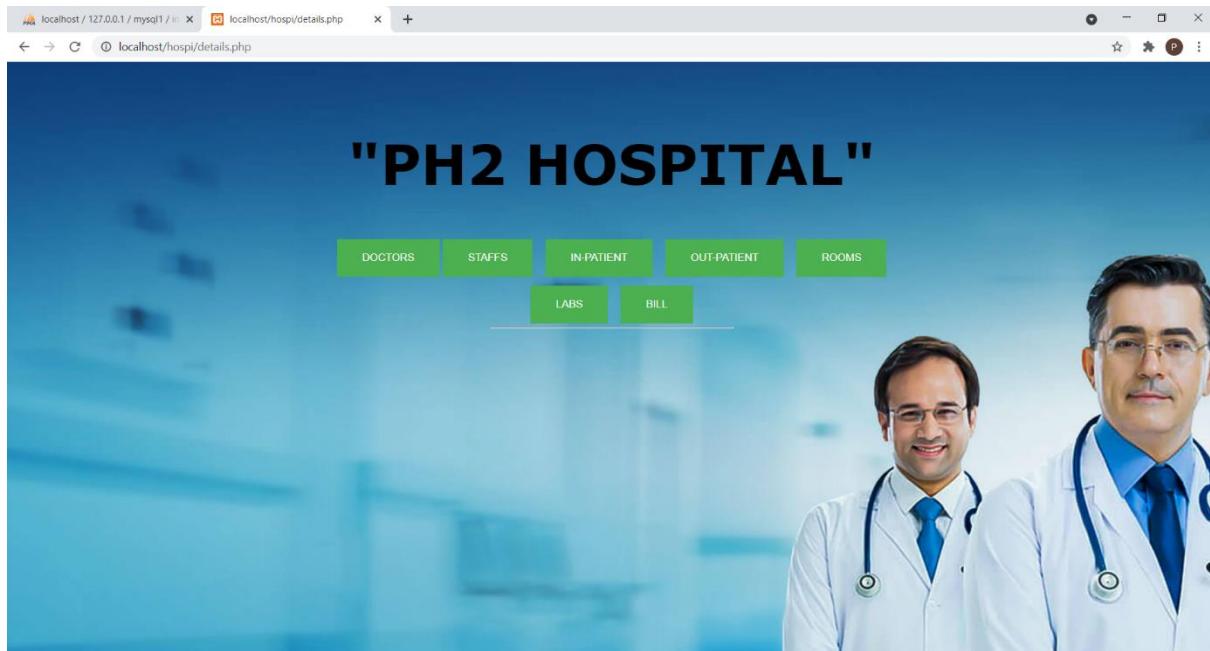
A screenshot of a web browser window. The address bar shows "localhost / 127.0.0.1 / mysql1 / In Patients". The title bar says "In Patients". The main content area has a heading "Enter Doctor ID TO Delete". Below it is a form with a text input field labeled "DOCTOR\_ID" containing "D12" and a submit button.

On clicking the “submit” button the particular doctor gets deleted from the doctor's table:

A screenshot of a web browser window. The address bar shows "localhost / 127.0.0.1 / mysql1 / DOCTOR". The title bar says "DOCTOR". The main content area has a heading "LIST OF DOCTORS" and three buttons: "Insert New doctor", "Delete doctor", and "Go Back". Below is a table with columns "DOCTOR\_ID", "NAME", and "DEPARTMENT". The table contains 12 rows of data.

DOCTOR_ID	NAME	DEPARTMENT
D3	DHANU	DENTIST
D4	DIVYA	GYNECOLOGIST
D8	HEMA	ENT
D1	KARAN	CARDIOLOGIST
D7	POOJA	DENTIST
D10	PRIYA	SURGEON
D11	RAMYA S	CARDIOLOGIST
D2	SANDIYA	DERMATALOGIST
D6	SHANTHANU	PHYSICIAN
D5	SUBHI	ENT
D9	VARSHINI	CARDIOLOGIST

On clicking the “Go Back” button it takes to the home page:

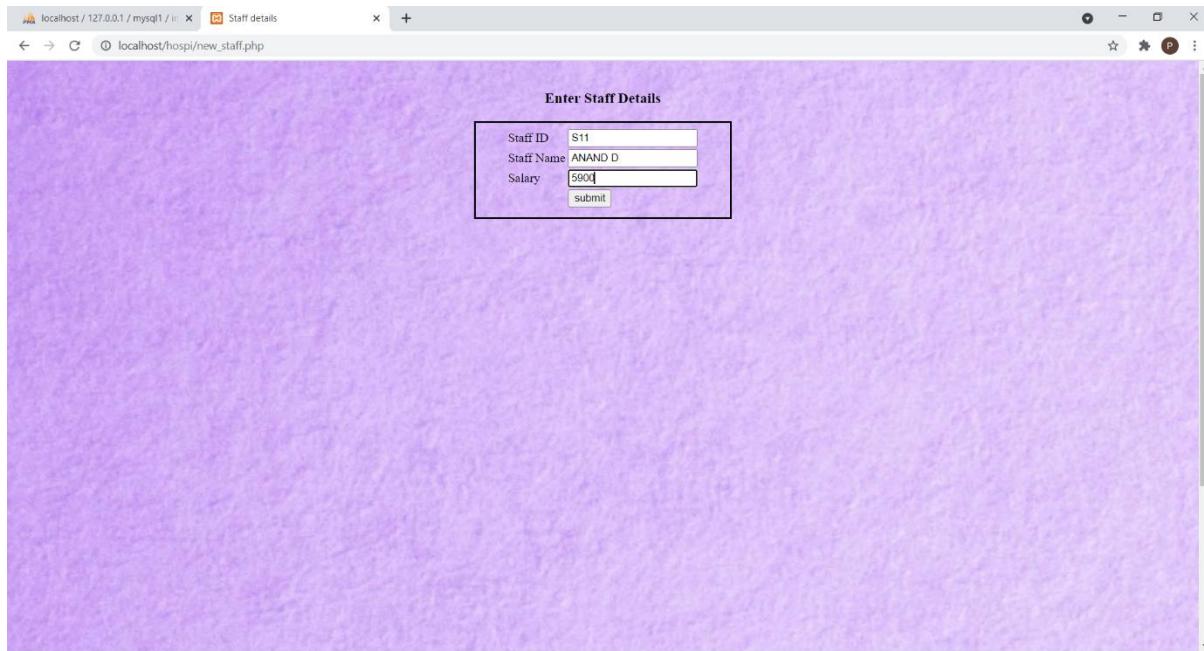


## 2)STAFFS:

On clicking the “STAFFS” button it takes to the below page where it displays the details of the staffs:

STAFF DETAILS		
<a href="#">Insert New staff</a>	<a href="#">Delete staff</a>	<a href="#">Go Back</a>
STAFF-ID	STAFF-NAME	SALARY
S5	GAYATHRI	17000
S2	HARSHA	10000
S8	KAVIN	16000
S10	KRITHI	20000
S9	MANISHA	24000
S7	NIVI	11000
S1	SATHIYA	23000
S3	SUGANA	18000
S6	VENUGOPAL	22000
S4	YAAZHINI	27000

On clicking the “Insert New staff” it takes to the following page where we must enter the details of the staff to be added:



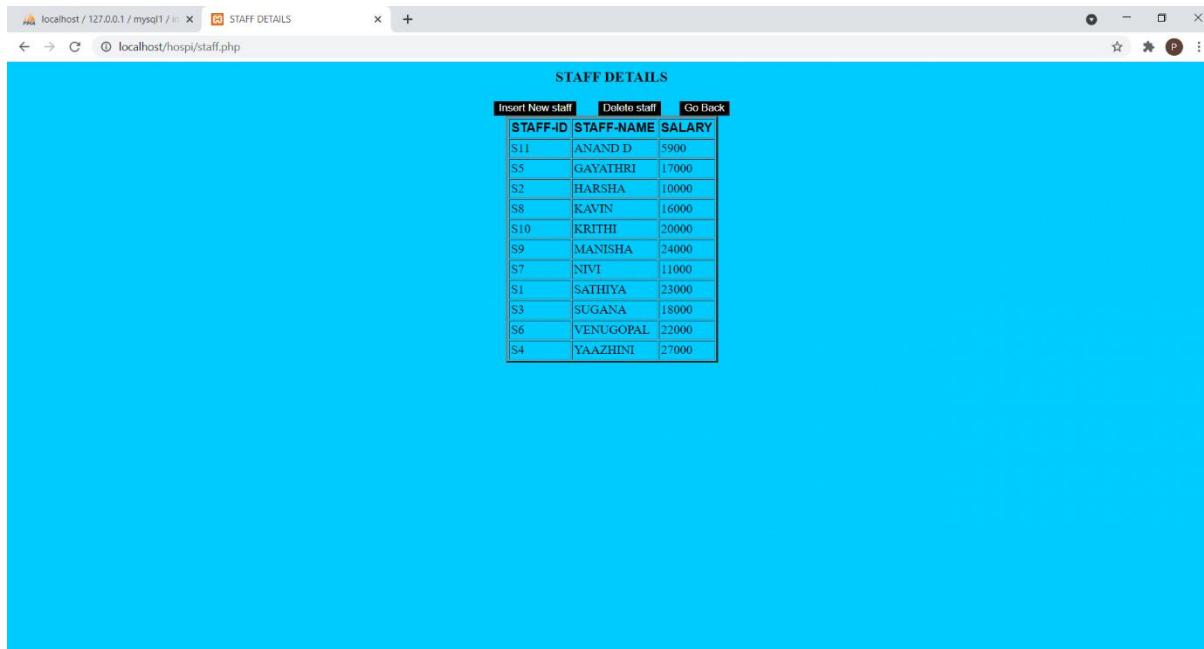
localhost / 127.0.0.1 / mysql1 / Staff details

localhost/hospi/new\_staff.php

Enter Staff Details

Staff ID	S11
Staff Name	ANAND D
Salary	5900
<input type="button" value="submit"/>	

On clicking submit button the newly added record gets added to the staffs table and gets displayed along with the staff details :



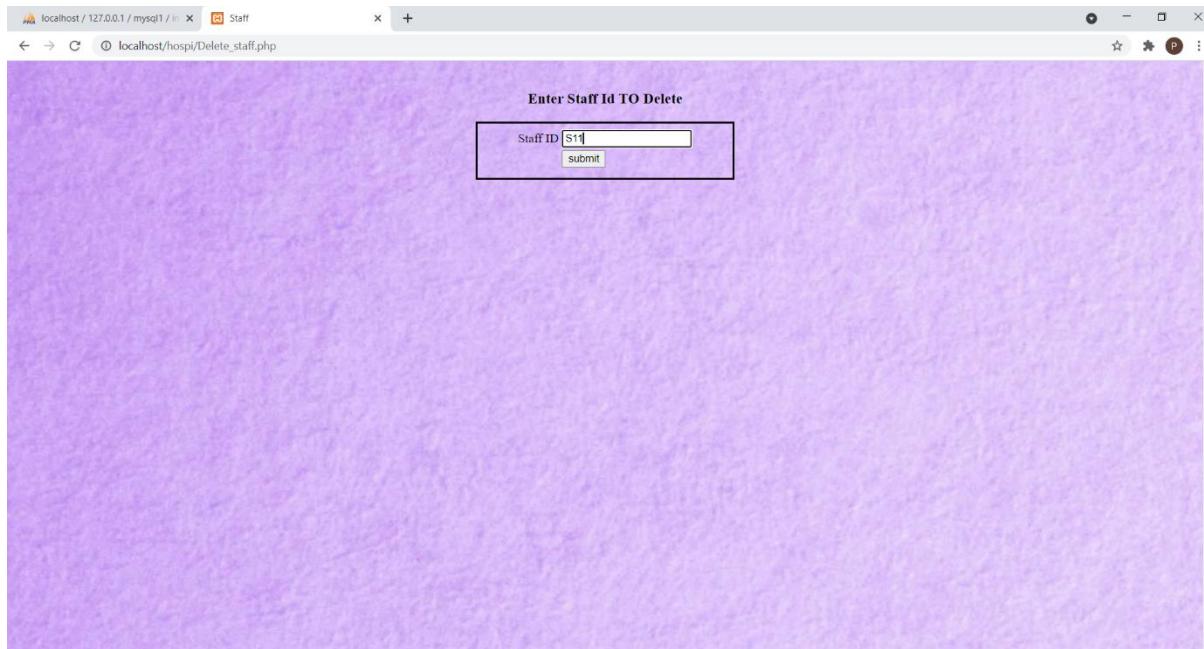
localhost / 127.0.0.1 / mysql1 / STAFF DETAILS

localhost/hospi/staff.php

STAFF DETAILS

<input type="button" value="Insert New staff"/>	<input type="button" value="Delete staff"/>	<input type="button" value="Go Back"/>
STAFF-ID	STAFF-NAME	SALARY
S11	ANAND D	5900
S5	GAYATHRI	17000
S2	HARSHA	10000
S8	KAVIN	16000
S10	KRITHI	20000
S9	MANISHA	24000
S7	NIVI	11000
S1	SATHIYA	23000
S3	SUGANA	18000
S6	VENUGOPAL	22000
S4	YAAZHINI	27000

On clicking the “Delete staff” button it takes to the below page where we must give the staff id to delete:



A screenshot of a web browser window. The address bar shows "localhost / 127.0.0.1 / mysql1 / Staff". The title bar says "Staff". The main content area has a heading "Enter Staff Id TO Delete". Below it is a form with a text input field containing "S11" and a submit button.

Staff ID	Staff Name	Salary
S1	SATHIYA	23000
S2	HARSHA	10000
S3	SUGANA	18000
S4	YAAZHINI	27000
S5	GAYATHRI	17000
S6	VENUGOPAL	22000
S7	NIVI	11000
S8	KAVIN	16000
S9	MANISHA	24000
S10	KRITHI	20000

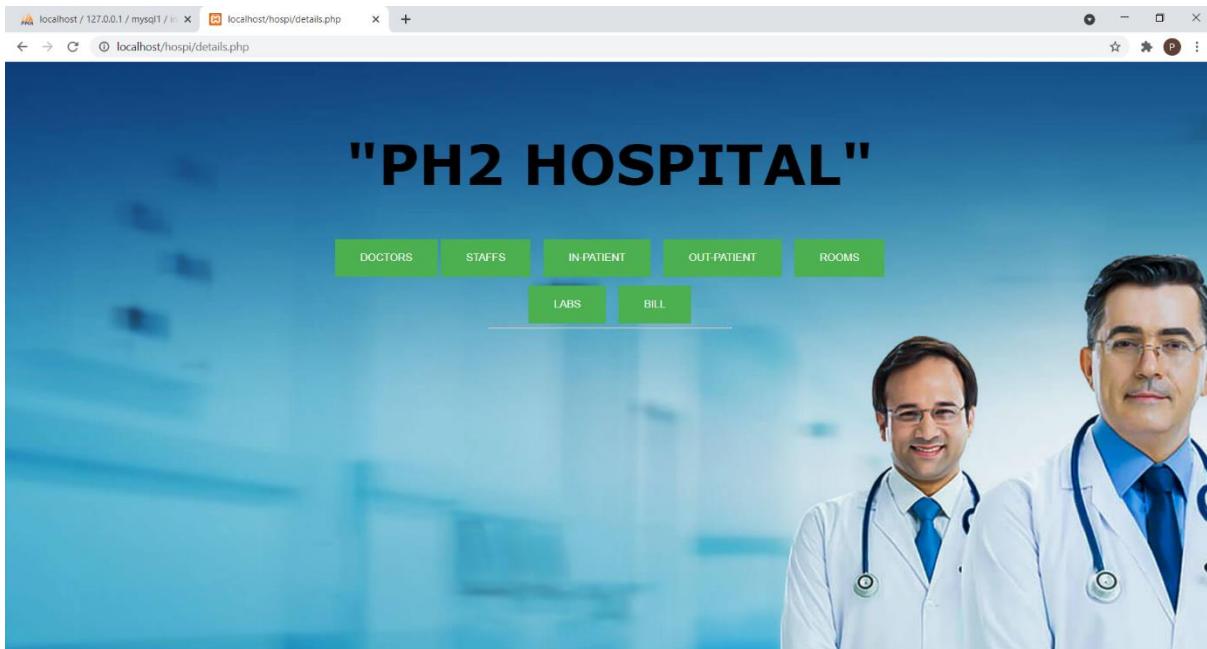
On clicking the “submit” button the particular staff gets deleted from the staff’s table:



A screenshot of a web browser window. The address bar shows "localhost / 127.0.0.1 / mysql1 / STAFF DETAILS". The title bar says "STAFF DETAILS". The main content area has a table titled "STAFF DETAILS" with columns "STAFF-ID", "STAFF-NAME", and "SALARY". The table contains 10 rows of data. At the top of the table, there are three buttons: "Insert New staff", "Delete staff", and "Go Back".

STAFF-ID	STAFF-NAME	SALARY
S1	SATHIYA	23000
S2	HARSHA	10000
S3	SUGANA	18000
S4	YAAZHINI	27000
S5	GAYATHRI	17000
S6	VENUGOPAL	22000
S7	NIVI	11000
S8	KAVIN	16000
S9	MANISHA	24000
S10	KRITHI	20000

On clicking the “Go Back” button it takes to the home page:



### 3) IN-PATIENT:

On clicking the “IN-PATIENT” button it takes to the below page where it displays the details of the inpatients:

localhost / 127.0.0.1 / mysql1 / In Patients

localhost/hospIP.php

## LIST OF INPATIENTS

[Insert New Patient](#) | [Go Back](#)

PATIENT-ID	NAME	GENDER	ADDRESS	ROOM_Number	DATE OF ADMISSION	DATE OF DISCHARGE	DOCTOR-ID
P11	HARI R	MALE	ANNA-NAGAR	10	2021-05-01	2021-05-05	D1
P12	KANAGA H	FEMALE	OMR-ROAD	20	2021-04-13	2021-04-16	D2
P13	VINCY K	FEMALE	BESANT-NAGAR	30	2021-04-28	2021-04-30	D3
P14	KRISHNA M	MALE	MADHAVARAM	40	2021-04-20	2021-04-22	D4
P15	AASIQ M	MALE	VELACHERY	50	2021-04-15	2021-04-17	D4
P16	ABIRAMI P	FEMALE	ADVAR	60	2021-05-04	2021-05-06	D6
P17	SUPRAJA K	FEMALE	MYLAPOOR	70	2021-04-26	2021-05-01	D7
P18	HARSHAT T	MALE	CHINDAMBARAM	10	2021-05-19	2021-05-22	D1

On clicking “Insert New Patient” it takes to the following page where we must enter the details of the patient to be added:

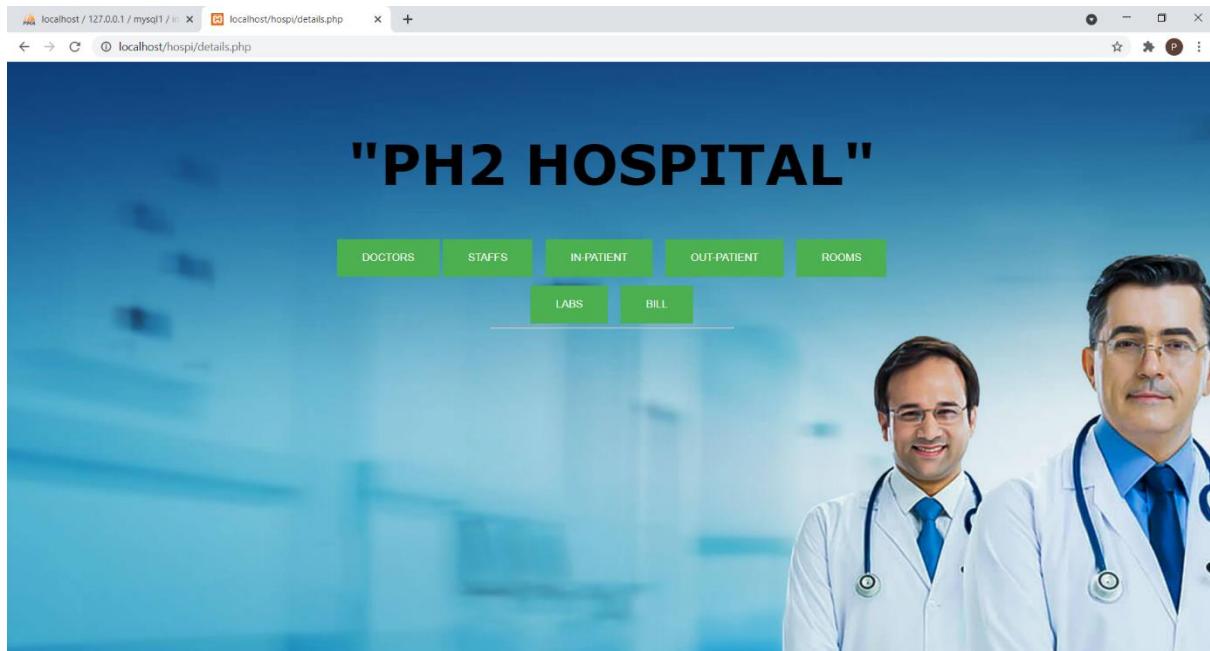
PATIENT-ID	P19
Name	HARISHRAJ G
Gender	MALE
Address	BAIPASS-ROAD 2
Room No	1
Date Of Admit	23-05-2021
Date Of Discharge	24-05-2021
Doctor ID	D2
<input type="button" value="submit"/>	

On clicking submit button the newly added record gets added to the inpatient table and gets displayed along with the LIST OF INPATIENTS:

**LIST OF INPATIENTS**

Insert New Patient		Go Back					
PATIENT-ID	NAME	GENDER	ADDRESS	ROOM_Number	DATE_OF_ADMISSION	DATE_OF_DISCHARGE	DOCTOR-ID
P11	HARI R	MALE	ANNA-NAGAR	10	2021-05-01	2021-05-05	D1
P12	KANAGA H	FEMALE	OMR-ROAD	20	2021-04-13	2021-04-16	D2
P13	VINCY K	FEMALE	BESANT-NAGAR	30	2021-04-28	2021-04-30	D3
P14	KRISHNA M	MALE	MADHAVARAM	40	2021-04-20	2021-04-22	D4
P15	AASIQ M	MALE	VELACHERRY	50	2021-04-15	2021-04-17	D4
P16	ABIRAMI P	FEMALE	ADYAR	60	2021-05-04	2021-05-06	D6
P17	SUPRAJA K	FEMALE	MYLAPOOR	70	2021-04-26	2021-05-01	D7
P18	HARSHAT T	MALE	CHINDAMBARAM	10	2021-05-19	2021-05-22	D1
P19	HARISHRAJ G	MALE	BAIPASS-ROAD 2	1	2021-05-23	2021-05-24	D2

On clicking the “Go Back” button it takes to the home page:



#### 4)OUT-PATIENT:

On clicking the “OUT-PATIENT” button it takes to the below page where it displays the details of the outpatients:

List of Out Patients			
		<a href="#">Insert New Patient</a>	<a href="#">Go Back</a>
PATIENT-ID	DATE	NAME	DOCTOR-ID
P1000	2021-04-25	RAJU S	D1
P1001	2021-05-17	AISHWARYA K	D4
P1002	2021-05-24	MALIGA H	D4
P111	2021-05-03	VAANATHI G	D4
P1200	2021-05-19	ADITYA D	D2
P222	2021-05-05	MITHUN K	D2
P333	2021-04-21	BHARATHI A	D9
P444	2021-04-19	ARUN T	D9
P555	2021-05-09	KISHORE J	D1
P666	2021-04-14	VICKY J	D3
P777	2021-04-28	ARJUN P	D3
P888	2021-05-11	ROJA A	D4
P999	2021-05-07	PREETHI G	D4

On clicking “Insert New Patient” it takes to the following page where we must enter the details of the patient to be added:

localhost / 127.0.0.1 / mysql1 / out Patients

localhost/hospi/New\_OP.php

Enter Patient Details

p_ID	P1003
DATE	23-05-2021
NAME	KARTHICK D
DOCTOR-ID	D1
<input type="button" value="Submit"/>	

On clicking the submit button the newly added record gets added to the outpatient table and gets displayed along with the List of out-patients:

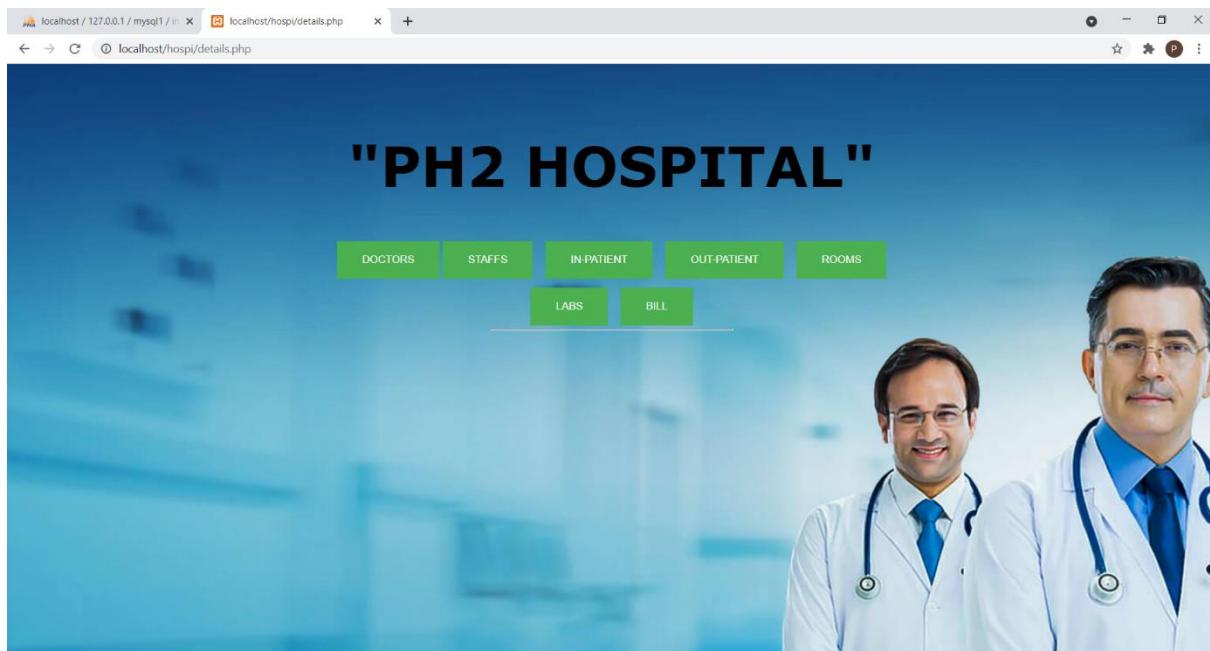
localhost / 127.0.0.1 / mysql1 / out Patients

localhost/hospi/OP.php

List of Out Patients

Insert New Patient				Go Back
PATIENT-ID	DATE	NAME	DOCTOR-ID	
P1000	2021-04-25	RAJU S	D1	
P1001	2021-05-17	AISHWARYA K	D4	
P1002	2021-05-24	MALIGA H	D4	
P1003	2021-05-23	KARTHICK D	D1	
P111	2021-05-03	VAANATHI G	D4	
P1200	2021-05-19	ADITYA D	D2	
P222	2021-05-05	MITHUN K	D2	
P333	2021-04-21	BHARATHIA A	D9	
P444	2021-04-19	ARUN T	D9	
P555	2021-05-09	KISHORE J	D1	
P666	2021-04-14	VICKY J	D3	
P777	2021-04-28	ARJUN P	D3	
P888	2021-05-11	ROJA A	D4	
P999	2021-05-07	PREETHI G	D4	

On clicking the “Go Back” button it takes to the home page:



## 5)ROOMS:

On clicking the “ROOMS” button it takes to the below page where it displays the details of the rooms:

Room Details			
ROOM-NO	ROOM-TYPE	STATUS	PATIENT-ID
1	NON-AC	AVAILABLE	
2	AC	AVAILABLE	
3	NON-AC	AVAILABLE	
4	NON-AC	AVAILABLE	
5	AC	AVAILABLE	
6	NON-AC	AVAILABLE	
7	AC	AVAILABLE	
8	AC	AVAILABLE	
9	NON-AC	AVAILABLE	
10	AC	NON-AVAILABLE	P18
20	AC	NON-AVAILABLE	P12
30	NON-AC	NON-AVAILABLE	P13
40	AC	NON-AVAILABLE	P14
50	NON-AC	NON-AVAILABLE	P15
60	NON-AC	NON-AVAILABLE	P16
70	AC	AVAILABLE	
80	NON-AC	AVAILABLE	
90	NON-AC	AVAILABLE	
100	AC	AVAILABLE	

On clicking the “update room” it takes to the following page where we must enter the details to be updated:

localhost / 127.0.0.1 / mysql1 / □ ROOM

localhost/hospi/roomupdate.php

Enter room no and status TO Update

ROOM-NO	1
STATUS	NON-AVAILABLE
PATIENT-ID	P19
<input type="button" value="submit"/>	

On clicking the “submit” button the given room no gets updated in the room table and gets displayed in the room details:

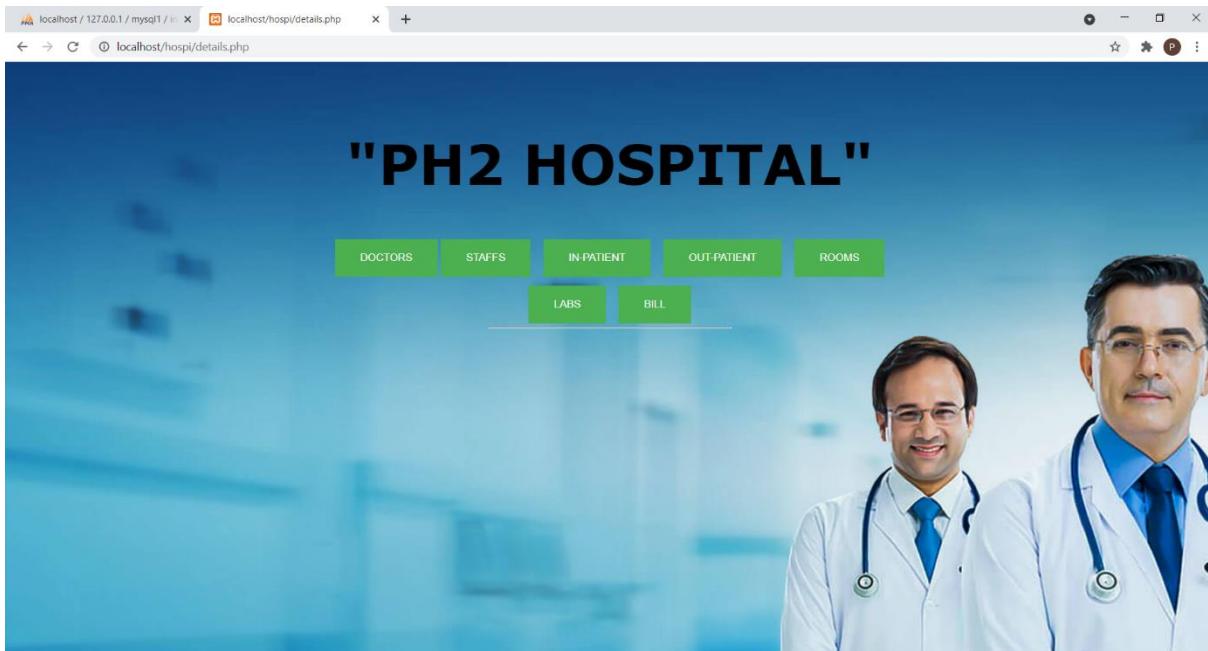
localhost / 127.0.0.1 / mysql1 / □ Room Details

localhost/hospi/room.php

Room Details

updateroom		Go Back	
ROOM-NO	ROOM-TYPE	STATUS	PATIENT-ID
1	NON-AC	NON-AVAILABLE	P19
2	AC	AVAILABLE	
3	NON-AC	AVAILABLE	
4	NON-AC	AVAILABLE	
5	AC	AVAILABLE	
6	NON-AC	AVAILABLE	
7	AC	AVAILABLE	
8	AC	AVAILABLE	
9	NON-AC	AVAILABLE	
10	AC	NON-AVAILABLE	P18
20	AC	NON-AVAILABLE	P12
30	NON-AC	NON-AVAILABLE	P13
40	AC	NON-AVAILABLE	P14
50	NON-AC	NON-AVAILABLE	P15
60	NON-AC	NON-AVAILABLE	P16
70	AC	AVAILABLE	
80	NON-AC	AVAILABLE	
90	NON-AC	AVAILABLE	
100	AC	AVAILABLE	

On clicking the “Go Back” button it takes to the home page:

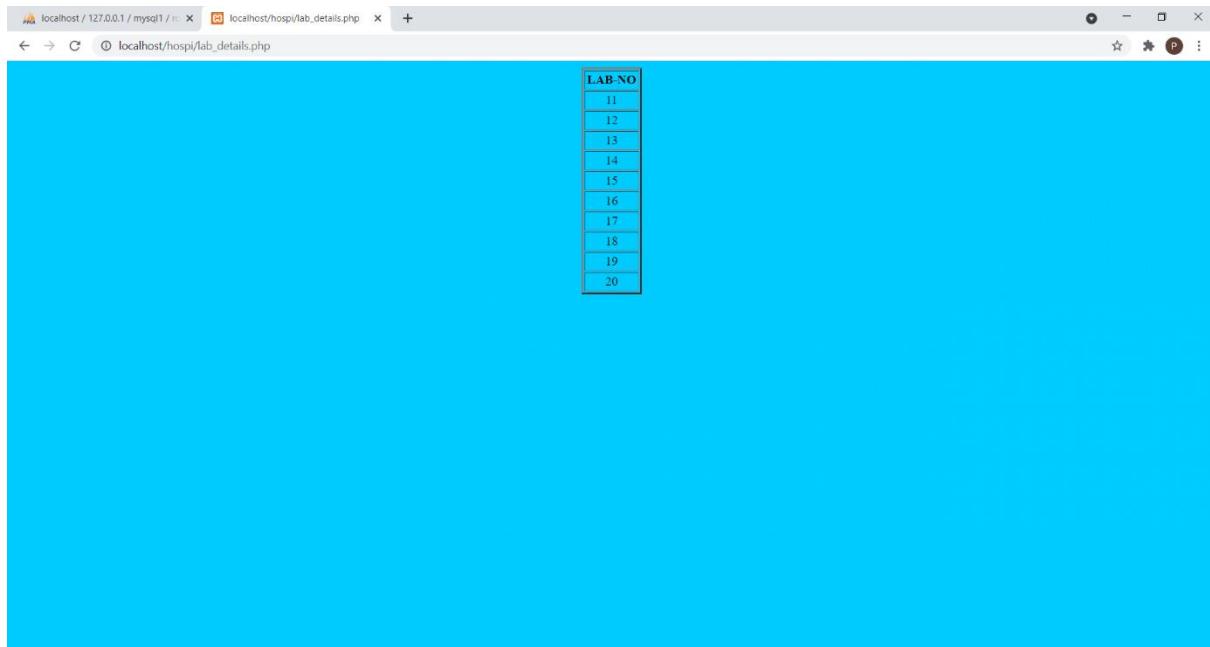


## 6)LABS:

On clicking the “LABS” button it takes to the below page where it displays the details of the labs:

LAB-DETAILS			
<a href="#">lab details</a>	<a href="#">Add lab details</a>	<a href="#">Go Back</a>	
LAB-NO	PATIENT-ID	DOCTOR-ID	LAB-DATE
11	P11	D1	2021-05-12
12	P12	D2	2020-05-11
13	P13	D3	2021-04-12
14	P14	D4	2021-04-15
15	P111	D1	2021-05-03
16	P999	D9	2021-05-07
11	P18	D1	2021-05-19

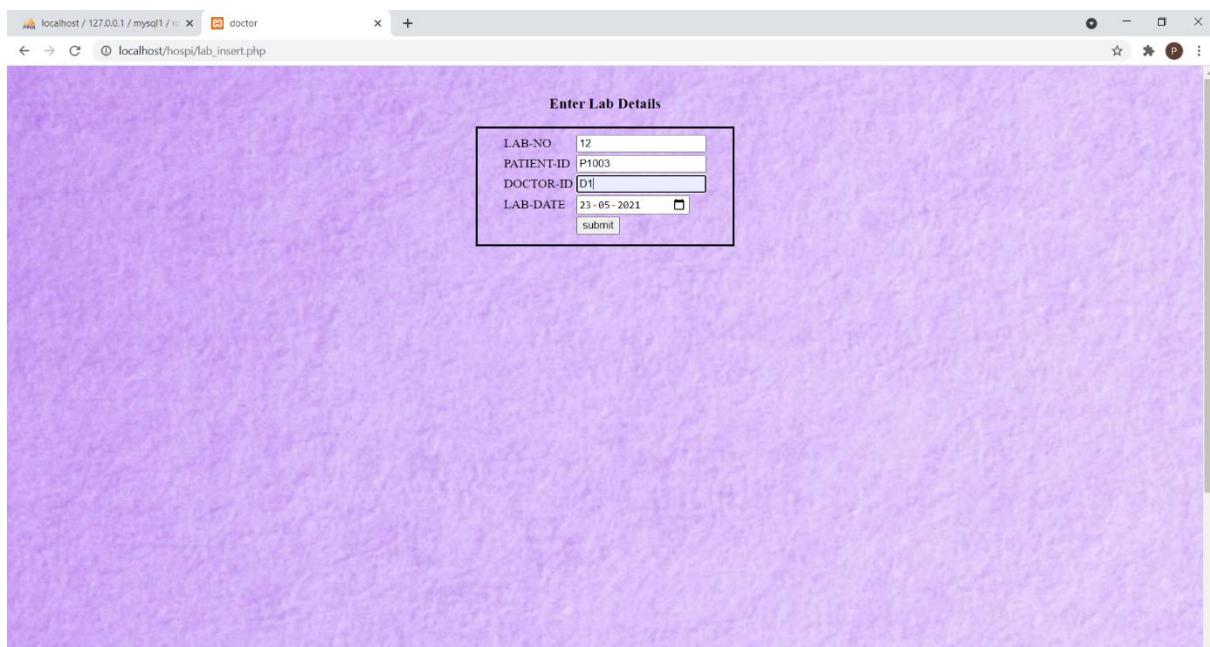
On clicking the “lab details” button it takes to the below page where it lists all the available labs in the hospital:



A screenshot of a web browser window. The address bar shows "localhost / 127.0.0.1 / mysql1 / n" and "localhost/hosp/lab\_details.php". The main content area displays a table with a single column labeled "LAB-NO" containing the numbers 11 through 20.

LAB-NO
11
12
13
14
15
16
17
18
19
20

On clicking the “Add lab details” it takes to the following page where we must enter the new lab details to be added:

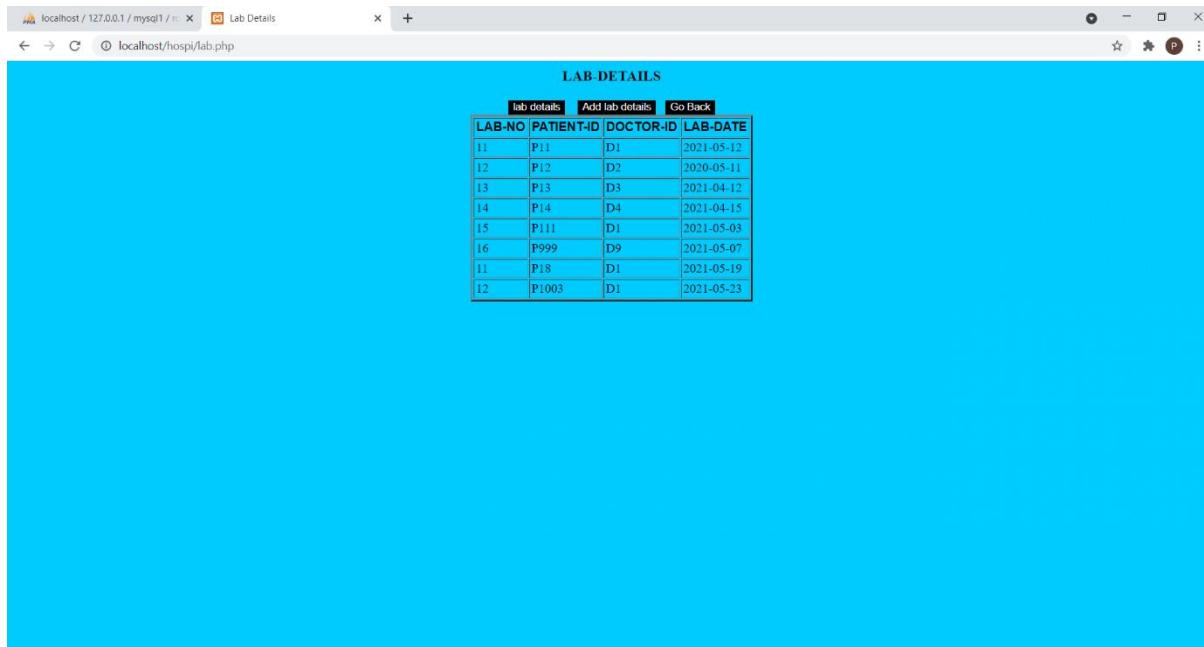


A screenshot of a web browser window. The address bar shows "localhost / 127.0.0.1 / mysql1 / n" and "doctor". The main content area displays a form titled "Enter Lab Details" with fields for LAB-NO, PATIENT-ID, DOCTOR-ID, and LAB-DATE, along with a submit button.

Enter Lab Details

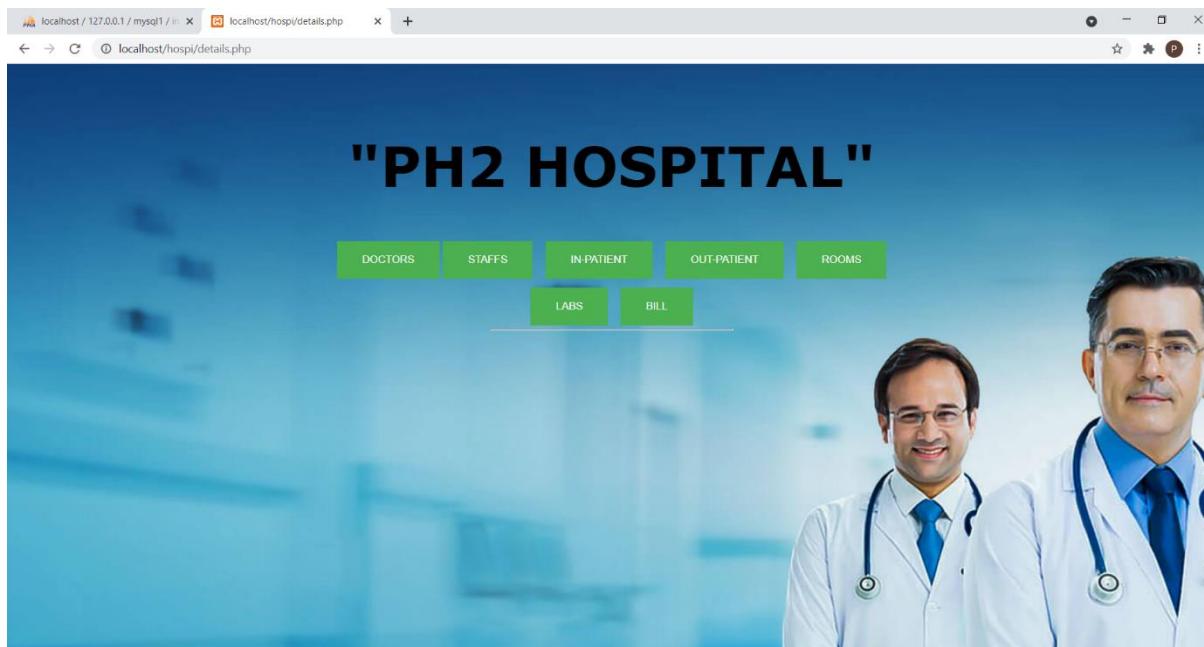
LAB-NO	<input type="text" value="12"/>
PATIENT-ID	<input type="text" value="P1003"/>
DOCTOR-ID	<input type="text" value="D1"/>
LAB-DATE	<input type="text" value="23-05-2021"/> <input type="button" value=""/>
<input type="button" value="submit"/>	

On clicking submit button the newly added record gets added to the lab table and gets displayed along with the LAB-DETAILS:



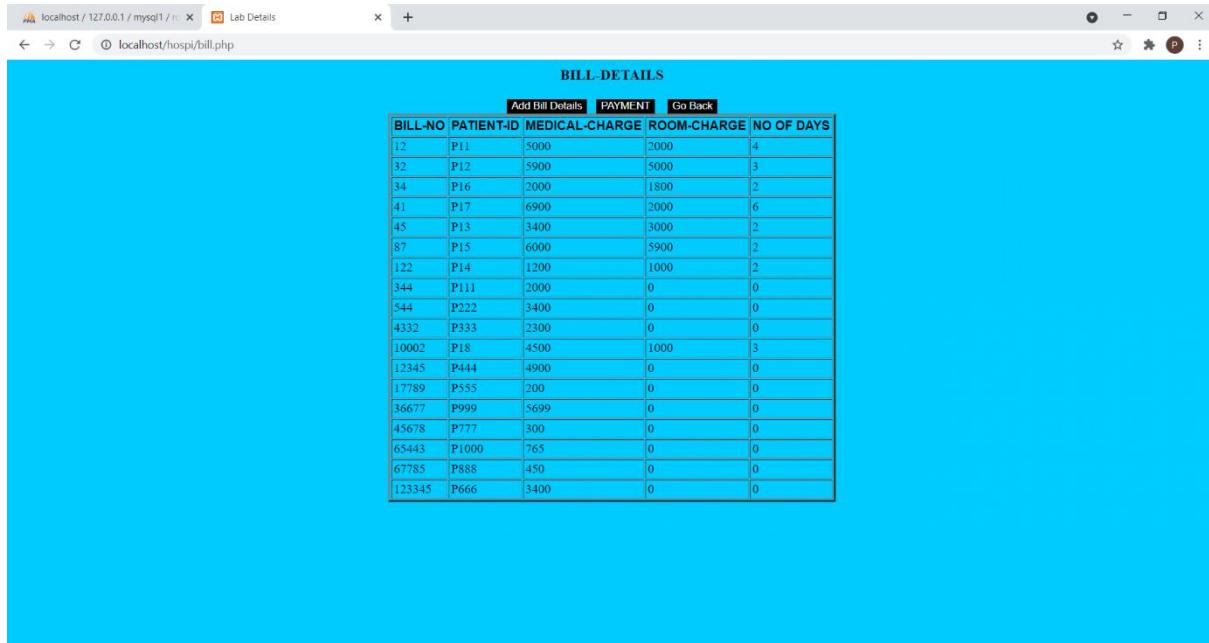
LAB-NO	PATIENT-ID	DOCTOR-ID	LAB-DATE
11	P11	D1	2021-05-12
12	P12	D2	2020-05-11
13	P13	D3	2021-04-12
14	P14	D4	2021-04-15
15	P111	D1	2021-05-03
16	P999	D9	2021-05-07
11	P18	D1	2021-05-19
12	P1003	D1	2021-05-23

On clicking the “Go Back” button it takes to the home page:



## 7)BILL:

On clicking the “BILL” button it takes to the below page where it displays the bill details:

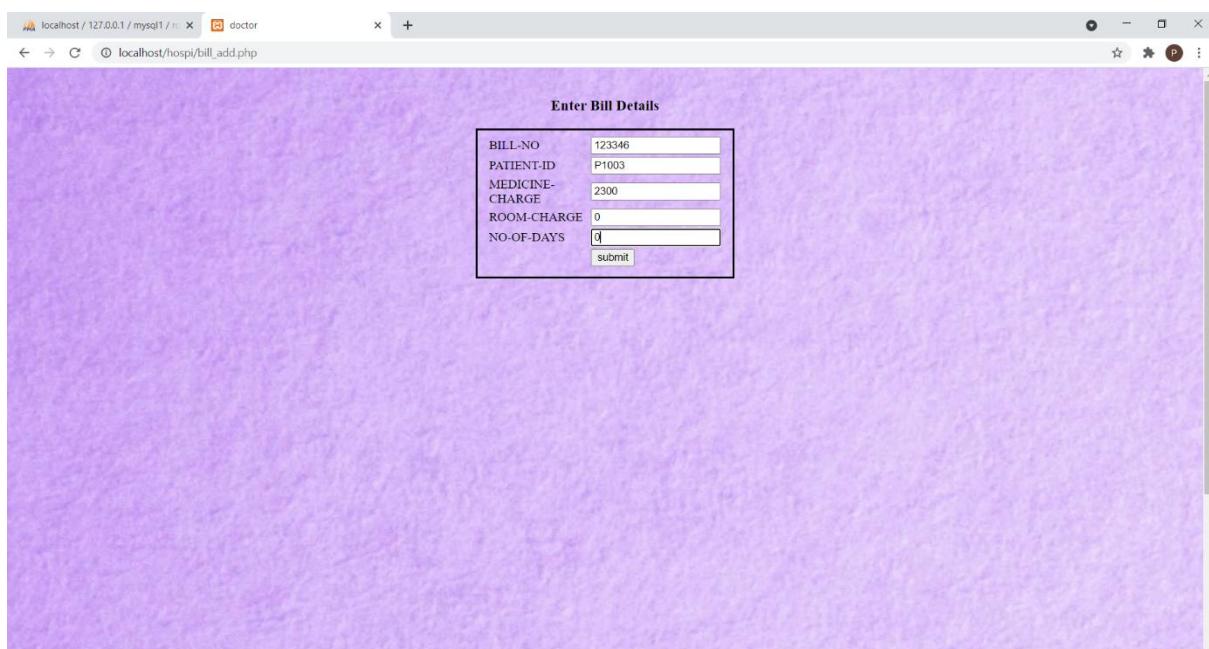


The screenshot shows a web browser window with the URL [localhost/hosp/bill.php](http://localhost/hosp/bill.php). The page title is "Lab Details". Below the title, there is a table with the following columns: BILL-NO, PATIENT-ID, MEDICAL-CHARGE, ROOM-CHARGE, and NO OF DAYS. The table contains 20 rows of data.

BILL-NO	PATIENT-ID	MEDICAL-CHARGE	ROOM-CHARGE	NO OF DAYS
12	P11	5000	2000	4
32	P12	5900	5000	3
34	P16	2000	1800	2
41	P17	6900	2000	6
45	P13	3400	3000	2
87	P15	6000	5900	2
122	P14	1200	1000	2
344	P111	2000	0	0
544	P222	3400	0	0
4332	P333	2300	0	0
10002	P18	4500	1000	3
12345	P444	4900	0	0
17789	P555	200	0	0
36677	P999	5699	0	0
45678	P777	300	0	0
65443	P1000	765	0	0
67785	P888	450	0	0
123345	P666	3400	0	0

A) ADDING OUTPATIENT BILL (give ‘0’ for room charge and for no of days for outpatient):

On clicking the “Add bill details” it takes to the following page where we must enter the new bill details to be added:



The screenshot shows a web browser window with the URL [localhost/hosp/bill\\_add.php](http://localhost/hosp/bill_add.php). The page title is "doctor". Below the title, there is a form titled "Enter Bill Details" with the following fields: BILL-NO, PATIENT-ID, MEDICINE-CHARGE, ROOM-CHARGE, and NO-OF-DAYS. There is also a "submit" button.

Enter Bill Details	
BILL-NO	<input type="text" value="123346"/>
PATIENT-ID	<input type="text" value="P1003"/>
MEDICINE-CHARGE	<input type="text" value="2300"/>
ROOM-CHARGE	<input type="text" value="0"/>
NO-OF-DAYS	<input type="text" value="0"/>
<input type="button" value="submit"/>	

On clicking submit button the newly added record gets added to the bill table and gets displayed along with the BILL-DETAILS:

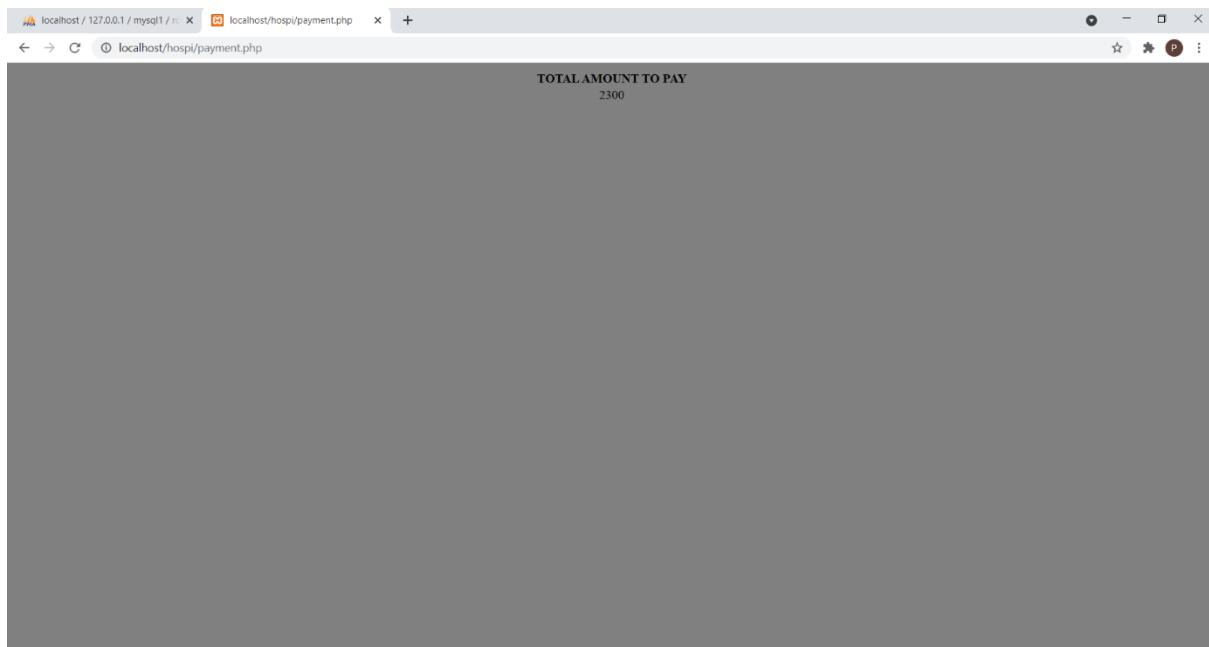
BILL-NO	PATIENT-ID	MEDICAL-CHARGE	ROOM-CHARGE	NO OF DAYS
12	P11	5000	2000	4
32	P12	5900	5000	3
34	P16	2000	1800	2
41	P17	6900	2000	6
45	P13	3400	3000	2
87	P15	6000	5900	2
122	P14	1200	1000	2
344	P111	2000	0	0
544	P222	3400	0	0
4332	P333	2300	0	0
10002	P18	4500	1000	3
12345	P444	4900	0	0
17789	P555	200	0	0
36677	P999	5699	0	0
45678	P777	300	0	0
65443	P1000	765	0	0
67785	P888	450	0	0
123345	P666	3400	0	0
123346	P1003	2300	0	0

On clicking the “PAYMENT” button the following page is directed:

ENTER PATIENT-ID TO CALCULATE THE TOTAL AMOUNT

PATIENT-ID

On clicking the submit button it displays the total amount to pay for that particular patient-id:



B) ADDING INPATIENT BILL (give the necessary amount for room charge and for no of days for inpatient):

A screenshot of a web browser window. The address bar shows 'localhost/hospi/bill\_add.php'. The main content area has a title 'Enter Bill Details' and a form with the following fields:

BILL-NO	1233448
PATIENT-ID	P19
MEDICINE-CHARGE	1000
ROOM-CHARGE	2000
NO-OF-DAYS	1

The form also contains a 'submit' button.

On clicking submit button the newly added record gets added to the bill table and gets displayed along with the BILL-DETAILS:

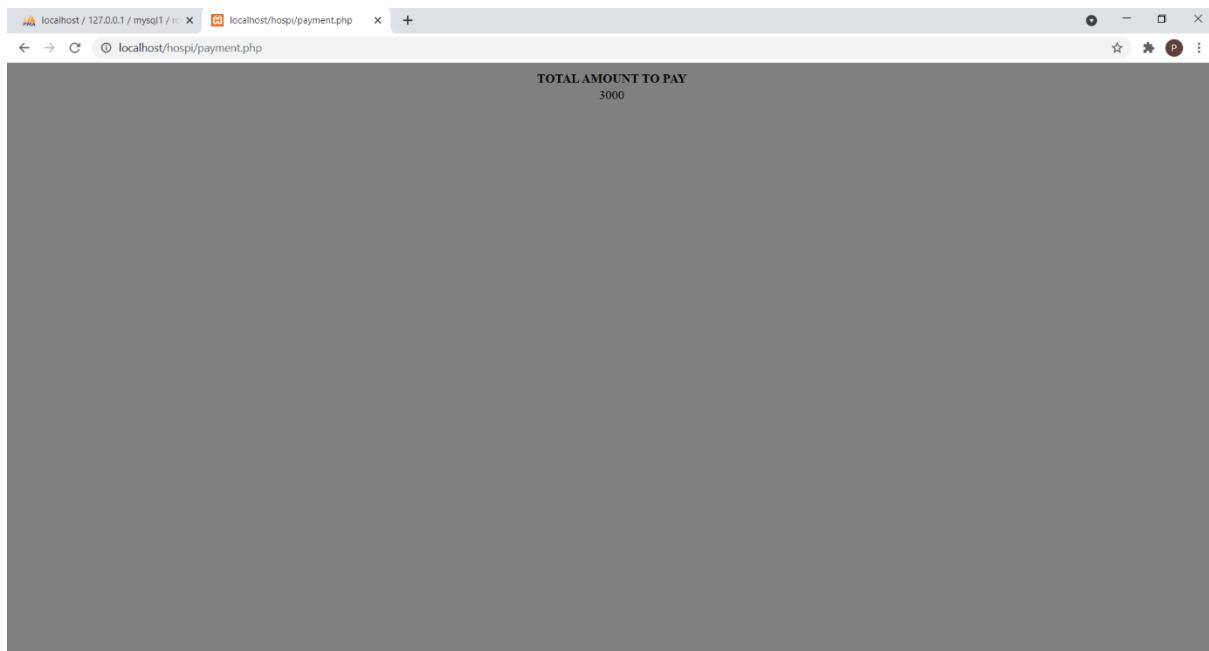
BILL-NO	PATIENT-ID	MEDICAL-CHARGE	ROOM-CHARGE	NO OF DAYS
12	P11	5000	2000	4
32	P12	5900	5000	3
34	P16	2000	1800	2
41	P17	6900	2000	6
45	P13	3400	3000	2
87	P15	6000	5900	2
122	P14	1200	1000	2
344	P111	2000	0	0
544	P222	3400	0	0
4332	P333	2300	0	0
10002	P18	4500	1000	3
12345	P444	4900	0	0
17789	P555	200	0	0
36677	P999	5699	0	0
45678	P777	300	0	0
65443	P1000	765	0	0
67785	P888	450	0	0
123345	P666	3400	0	0
123346	P1003	2300	0	0
1233448	P19	1000	2000	1

On clicking the “PAYMENT” button the following page is directed:

ENTER PATIENT-ID TO CALCULATE THE TOTAL AMOUNT

PATIENT-ID

On clicking the submit button it displays the total amount to pay for that particular patient-id:



## TABLES:

### 1) CREDENTIALS:

The screenshot shows the 'credentials' table structure in MySQL Workbench. The table has three columns: 'Username' (varchar(100), utf8mb4\_general\_ci), 'Password' (varchar(100), utf8mb4\_general\_ci), and 'a\_id' (varchar(50), utf8mb4\_general\_ci). The 'a\_id' column is defined as the primary key (PRIMARY, BTREE, Yes, No, Password, 2, A). Action buttons for each column include Change, Drop, More, and a link to the table structure.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Username	varchar(100)	utf8mb4_general_ci		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
2	Password	varchar(100)	utf8mb4_general_ci		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
3	a_id	varchar(50)	utf8mb4_general_ci		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>

Indexes:

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
<a href="#">Edit</a>	PRIMARY	BTREE	Yes	No	Password	2	A	No	

### 2) STAFF:

The screenshot shows the 'staff' table structure in MySQL Workbench. The table has three columns: 's\_id' (varchar(50), utf8mb4\_general\_ci), 's\_name' (varchar(100), utf8mb4\_general\_ci), and 'salary' (int(11)). The 's\_id' column is defined as the primary key (PRIMARY, BTREE, Yes, No, s\_id, 10, A). Action buttons for each column include Change, Drop, More, and a link to the table structure.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	s_id	varchar(50)	utf8mb4_general_ci		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
2	s_name	varchar(100)	utf8mb4_general_ci		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
3	salary	int(11)			No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>

Indexes:

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
<a href="#">Edit</a>	PRIMARY	BTREE	Yes	No	s_id	10	A	No	

### 3)DOCTOR:

The screenshot shows the MySQL Workbench interface for the 'doctor' table in the 'mysql1' database. The table has three columns: 'doctor\_id', 'doctor\_name', and 'dept'. The 'doctor\_id' column is set as the primary key (PRIMARY). An index named 'doctor\_id' is also present.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	doctor_id	varchar(50)	utf8mb4_general_ci		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
2	doctor_name	varchar(100)	utf8mb4_general_ci		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
3	dept	varchar(100)	utf8mb4_general_ci		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>

**Indexes:**

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
<a href="#">Edit</a>	PRIMARY	BTREE	Yes	No	doctor_id	10	A	No	

### 4)IN-PATIENT:

The screenshot shows the MySQL Workbench interface for the 'in\_patient' table in the 'mysql1' database. The table has eight columns: 'P\_ID', 'Name', 'gender', 'Address', 'R\_NUMBER', 'DOA', 'DOD', and 'Doc\_ID'. The 'P\_ID' column is set as the primary key (PRIMARY). An index named 'P\_ID' is also present.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	P_ID	varchar(20)	utf8mb4_general_ci		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
2	Name	varchar(150)	utf8mb4_general_ci		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
3	gender	varchar(10)	utf8mb4_general_ci		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
4	Address	varchar(500)	utf8mb4_general_ci		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
5	R_NUMBER	int(11)			No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
6	DOA	date			No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
7	DOD	date			No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>
8	Doc_ID	varchar(10)	utf8mb4_general_ci		No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">More</a>

**Indexes:**

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
<a href="#">Edit</a>	PRIMARY	BTREE	Yes	No	P_ID	8	A	No	
<a href="#">Edit</a>	Doc_ID	BTREE	No	No	Doc_ID	8	A	No	

## 5)OUT-PATIENT:

The screenshot shows the MySQL Workbench interface for the 'out\_patient' table. The table has four columns: P\_ID, DATE, name, and doc\_id. The primary key is set to doc\_id. An index named 'room\_no' is also present.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action	
1	P_ID	varchar(20)	utf8mb4_general_ci		No	None				
2	DATE	date			No	None				
3	name	varchar(100)	utf8mb4_general_ci		No	None				
4	doc_id	varchar(10)	utf8mb4_general_ci		No	None				

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
	PRIMARY	BTREE	Yes	No	P_ID	12	A	No	

## 6)ROOM:

The screenshot shows the MySQL Workbench interface for the 'room' table. The table has four columns: room\_no, room\_type, status, and patient\_id. The primary key is set to room\_no. An index named 'room\_no' is also present.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action	
1	room_no	int(11)			No	None				
2	room_type	varchar(50)	utf8mb4_general_ci		No	None				
3	status	varchar(50)	utf8mb4_general_ci		No	None				
4	patient_id	varchar(20)	utf8mb4_general_ci		No	None				

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
	PRIMARY	BTREE	Yes	No	room_no	8	A	No	

## 7)LAB:

Table structure for table lab

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	lab_no	int(11)			No	None		Change  Drop  More	
2	P_id	varchar(20)	utf8mb4_general_ci		No	None		Change  Drop  More	
3	Doc_ID	varchar(20)	utf8mb4_general_ci		No	None		Change  Drop  More	
4	l_date	date			No	None		Change  Drop  More	

Add 1 column(s) after l\_date Go

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit  Rename  Drop	Doc_ID	BTREE	No	No	Doc_ID	7	A	No	

## 8)BILL:

Table structure for table bill

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	bill_no	int(11)			No	None		Change  Drop  More	
2	P_id	varchar(50)	utf8mb4_general_ci		No	None		Change  Drop  More	
3	med_fee	int(11)			No	None		Change  Drop  More	
4	room_fee	int(11)			No	None		Change  Drop  More	
5	days	int(11)			No	None		Change  Drop  More	

Add 1 column(s) after days Go

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit  Rename  Drop	PRIMARY	BTREE	Yes	No	bill_no	18	A	No	

## SCHEMAS:

## CREDENTIALS:

Showing rows 0 - 2 (3 total, Query took 0.0005 seconds.)

```
SELECT * FROM `credentials`
```

Profiling |  Edit inline |  Explain SQL |  Create PHP code |  Refresh

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

				Username	Password	a_id
<input type="checkbox"/>	<input type="button"/> Edit	<input type="button"/> Copy	<input type="button"/> Delete	poojasri	2001	a1
<input type="checkbox"/>	<input type="button"/> Edit	<input type="button"/> Copy	<input type="button"/> Delete	haripriya	2002	a3
<input type="checkbox"/>	<input type="button"/> Edit	<input type="button"/> Copy	<input type="button"/> Delete	hemavarshini	hema	a2

## **STAFF:**

✓ Showing rows 0 - 9 (10 total; Query took 0.0005 seconds.)

**SELECT \* FROM `staff`**

Profiling | Edit inline | Edit | Explain SQL | Create PHP code | Refresh

	<input type="checkbox"/> Show all	Number of rows:	25	Filter rows:	Search this table	Sort by key:	None
+ Options							
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	 Edit	 Copy	 Delete	S1	SATHIYA	23000	
	<input type="checkbox"/>	 Edit	 Copy	 Delete	S10	KRITHI	20000
	<input type="checkbox"/>	 Edit	 Copy	 Delete	S2	HARSHA	10000
	<input type="checkbox"/>	 Edit	 Copy	 Delete	S3	SUGANA	18000
	<input type="checkbox"/>	 Edit	 Copy	 Delete	S4	YAAZHINI	27000
	<input type="checkbox"/>	 Edit	 Copy	 Delete	S5	GAYATHRI	17000
	<input type="checkbox"/>	 Edit	 Copy	 Delete	S6	VENUGOPAL	22000
	<input type="checkbox"/>	 Edit	 Copy	 Delete	S7	NIVI	11000
	<input type="checkbox"/>	 Edit	 Copy	 Delete	S8	KAVIN	16000
	<input type="checkbox"/>	 Edit	 Copy	 Delete	S9	MANISHA	24000

# DOCTOR:

Showing rows 0 - 10 (11 total, Query took 0.0005 seconds.)

**SELECT \* FROM `doctor`**

Profiling | [Edit inline](#) | [Edit](#) | [Explain SQL](#) | [Create PHP code](#) | [Refresh](#)

			doctor_id	doctor_name	dept
<input type="checkbox"/>	<a href="#"></a>	<a href="#"></a>	D1	KARAN	CARDIOLOGIST
<input type="checkbox"/>	<a href="#"></a>	<a href="#"></a>	D10	PRIYA	SURGEON
<input type="checkbox"/>	<a href="#"></a>	<a href="#"></a>	D11	RAMYA S	CARDIOLOGIST
<input type="checkbox"/>	<a href="#"></a>	<a href="#"></a>	D2	SANDIYA	DERMATOLOGIST
<input type="checkbox"/>	<a href="#"></a>	<a href="#"></a>	D3	DHANU	DENTIST
<input type="checkbox"/>	<a href="#"></a>	<a href="#"></a>	D4	DIVYA	GYNECOLOGIST
<input type="checkbox"/>	<a href="#"></a>	<a href="#"></a>	D5	SUBHI	ENT
<input type="checkbox"/>	<a href="#"></a>	<a href="#"></a>	D6	SHANTHANU	PHYSICIAN
<input type="checkbox"/>	<a href="#"></a>	<a href="#"></a>	D7	POOJA	DENTIST
<input type="checkbox"/>	<a href="#"></a>	<a href="#"></a>	D8	HEMA	ENT
<input type="checkbox"/>	<a href="#"></a>	<a href="#"></a>	D9	VARSHINI	CARDIOLOGIST

## IN-PATIENT:

Showing rows 0 - 7 (8 total, Query took 0.0006 seconds.)

```
SELECT * FROM `in_patient`
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

	P_ID	Name	gender	Address	R_NUMBER	DOA	DOD	Doc_ID
<input type="checkbox"/>	P11	HARI R	MALE	ANNA-NAGAR	10	2021-05-01	2021-05-05	D1
<input type="checkbox"/>	P12	KANAGA H	FEMALE	OMR-ROAD	20	2021-04-13	2021-04-16	D2
<input type="checkbox"/>	P13	VINCY K	FEMALE	BESANT-NAGAR	30	2021-04-28	2021-04-30	D3
<input type="checkbox"/>	P14	KRISHNA M	MALE	MADHAVARAM	40	2021-04-20	2021-04-22	D4
<input type="checkbox"/>	P15	AASIQ M	MALE	VELACHERRY	50	2021-04-15	2021-04-17	D4
<input type="checkbox"/>	P16	ABIRAMI P	FEMALE	ADYAR	60	2021-05-04	2021-05-06	D6
<input type="checkbox"/>	P17	SUPRAJA K	FEMALE	MYLAPOOR	70	2021-04-26	2021-05-01	D7
<input type="checkbox"/>	P18	HARSHAT T	MALE	CHINDAMBARAM	10	2021-05-19	2021-05-22	D1

## OUT-PATIENT:

Showing rows 0 - 11 (12 total, Query took 0.0006 seconds.)

```
SELECT * FROM `out_patient`
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

	P_ID	DATE	name	doc_id
<input type="checkbox"/>	P1000	2021-04-25	RAJU S	D1
<input type="checkbox"/>	P1001	2021-05-17	AISHWARYA K	D4
<input type="checkbox"/>	P111	2021-05-03	VAANATHI G	D4
<input type="checkbox"/>	P1200	2021-05-19	ADITYA D	D2
<input type="checkbox"/>	P222	2021-05-05	MITHUN K	D2
<input type="checkbox"/>	P333	2021-04-21	BHARATHI A	D9
<input type="checkbox"/>	P444	2021-04-19	ARUN T	D9
<input type="checkbox"/>	P555	2021-05-09	KISHORE J	D1
<input type="checkbox"/>	P666	2021-04-14	VICKY J	D3
<input type="checkbox"/>	P777	2021-04-28	ARJUN P	D3
<input type="checkbox"/>	P888	2021-05-11	ROJA A	D4
<input type="checkbox"/>	P999	2021-05-07	PREETHI G	D4

## ROOM:

Showing rows 0 - 9 (10 total, Query took 0.0004 seconds.)

```
SELECT * FROM `room`
```

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

	room_no	room_type	status	patient_id
<input type="checkbox"/>	10	AC	NON-AVAILABLE	P18
<input type="checkbox"/>	20	AC	NON-AVAILABLE	P12
<input type="checkbox"/>	30	NON-AC	NON-AVAILABLE	P13
<input type="checkbox"/>	40	AC	NON-AVAILABLE	P14
<input type="checkbox"/>	50	NON-AC	NON-AVAILABLE	P15
<input type="checkbox"/>	60	NON-AC	NON-AVAILABLE	P16
<input type="checkbox"/>	70	AC	AVAILABLE	
<input type="checkbox"/>	80	NON-AC	AVAILABLE	
<input type="checkbox"/>	90	NON-AC	AVAILABLE	
<input type="checkbox"/>	100	AC	AVAILABLE	

LAB:

Showing rows 0 - 6 (7 total, Query took 0.0007 seconds.)

`SELECT * FROM `lab``

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

lab_no	P_id	Doc_ID	I_date
11	P11	D1	2021-05-12
12	P12	D2	2020-05-11
13	P13	D3	2021-04-12
14	P14	D4	2021-04-15
15	P111	D1	2021-05-03
16	P999	D9	2021-05-07
11	P18	D1	2021-05-19

BILL:

Showing rows 0 - 17 (18 total, Query took 0.0005 seconds.)

`SELECT * FROM `bill``

Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

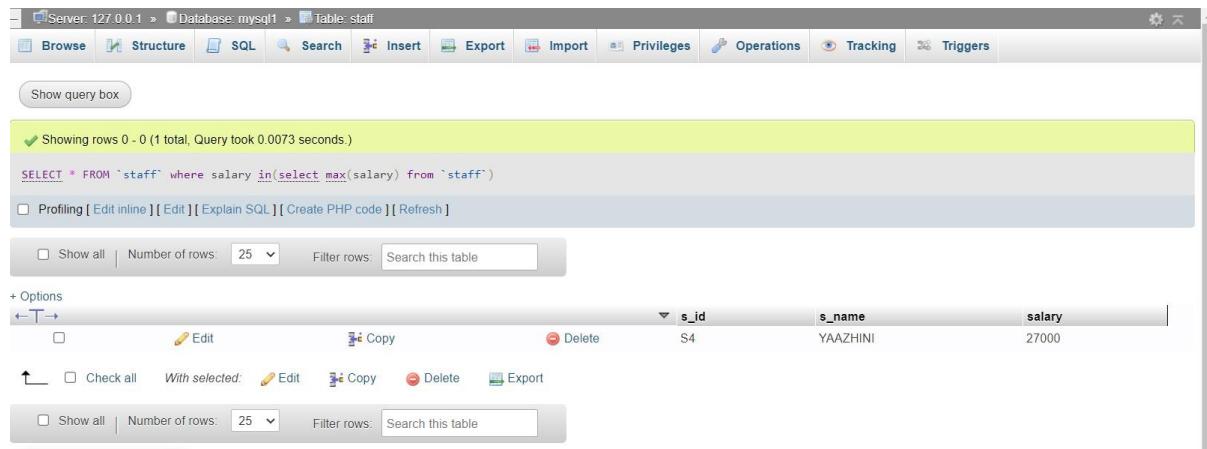
Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	bill_no	P_id	med_fee	room_fee	days
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	12	P11	5000	2000	4
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	32	P12	5900	5000	3
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	34	P16	2000	1800	2
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	41	P17	6900	2000	6
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	45	P13	3400	3000	2
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	87	P15	6000	5900	2
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	122	P14	1200	1000	2
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	344	P111	2000	0	0
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	544	P222	3400	0	0
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	4332	P333	2300	0	0
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	10002	P18	4500	1000	3
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	12345	P444	4900	0	0
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	17789	P555	200	0	0
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	36677	P999	5699	0	0
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	45678	P777	300	0	0
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	65443	P1000	765	0	0
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	67785	P888	450	0	0
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	123345	P666	3400	0	0

## QUERIES AND SUBQUERIES:

- 1) TO DISPLAY THE DETAILS OF THE STAFF WHOSE SALARY IS MAXIMUM:**



The screenshot shows the MySQL Workbench interface with the following details:

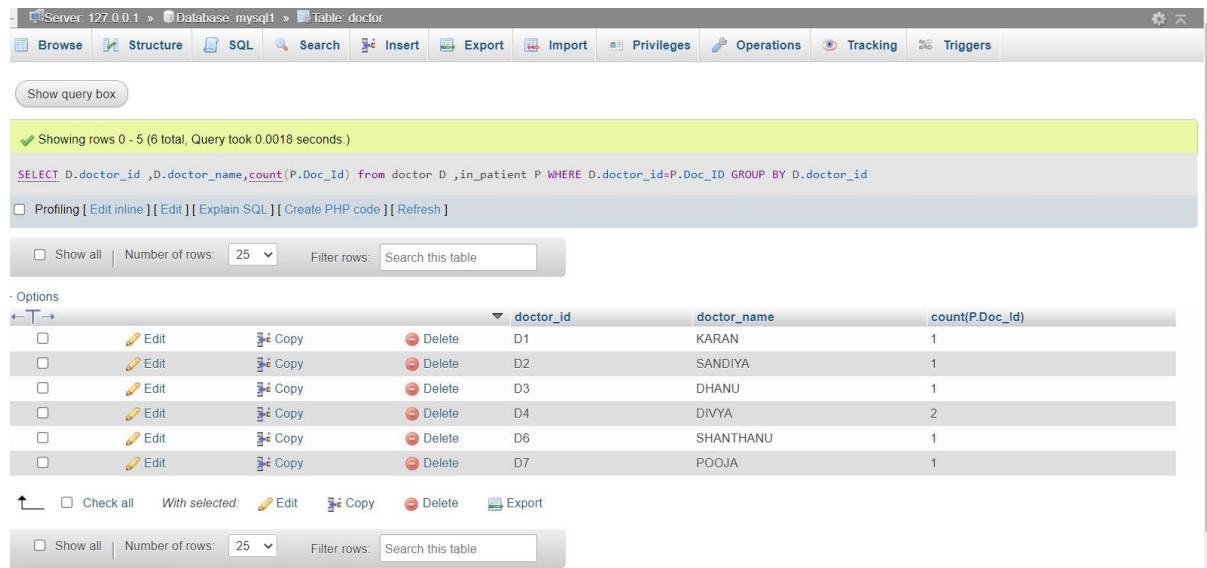
- Server: 127.0.0.1
- Database: mysql1
- Table: staff
- Query results:

```
SELECT * FROM `staff` where salary in(select max(salary) from `staff`)
```

- Result table:

s_id	s_name	salary
S4	YAAZHINI	27000

- 2) TO DISPLAY THE NUMBER OF INPATIENTS TREATED BY EACH DOCTOR:**



The screenshot shows the MySQL Workbench interface with the following details:

- Server: 127.0.0.1
- Database: mysql1
- Table: doctor
- Query results:

```
SELECT D.doctor_id ,D.doctor_name,count(P.Doc_Id) from doctor D ,in_patient P WHERE D.doctor_id=P.Doc_ID GROUP BY D.doctor_id
```

- Result table:

doctor_id	doctor_name	count(P.Doc_Id)
D1	KARAN	1
D2	SANDIYA	1
D3	DHANU	1
D4	DIVYA	2
D6	SHANTHANU	1
D7	POOJA	1

### 3) TO DISPLAY THE NUMBER OF OUTPATIENTS TREATED BY EACH DOCTOR:

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** mysql1
- Table:** doctor
- Query:**

```
SELECT D.doctor_id ,D.doctor_name, count(P.doc_id) from doctor D ,out_patient P WHERE D.doctor_id=P.doc_id GROUP BY D.doctor_id
```
- Results:**

doctor_id	doctor_name	count(P.doc_id)
D1	KARAN	2
D2	SANDIYA	1
D3	DHANU	2
D4	DIVYA	4
D9	VARSHINI	2

### 4) TO DISPLAY THE DOCTOR\_ID, DOCTOR\_NAME WHO HAVE TREATED THE MAXIMUM NUMBER OF OUT\_PATIENTS ALONG WITH THEIR COUNT:

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** mysql1
- Table:** doctor
- Query:**

```
SELECT D.doctor_id ,D.doctor_name, count(P.doc_id) from doctor D ,out_patient P WHERE D.doctor_id=P.doc_id GROUP BY D.doctor_id ORDER BY COUNT(P.doc_id) DESC LIMIT 1
```
- Results:**

doctor_id	doctor_name	count(P.doc_id)
D4	DIVYA	4

## 5) TO COMPUTE AND DISPLAY THE IN\_PATIENT ID, NAME, ROOM NO WITH THEIR NO OF DAYS STAYED IN THE ROOM:

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** mysql1
- Table:** in\_patient
- Query:** SELECT P\_ID, Name, R\_NUMBER ,DATEDIFF(DOD,DOA) AS 'NO OD DAYS STAYED' FROM in\_patient
- Results:** 7 rows returned in 0.0011 seconds.
- Table Headers:** P\_ID, Name, R\_NUMBER, NO OD DAYS STAYED
- Data:**

P_ID	Name	R_NUMBER	NO OD DAYS STAYED
P11	HARI R	10	4
P12	KANAGA H	20	3
P13	VINCY K	30	2
P14	KRISHNA M	40	2
P15	AASIQ M	50	2
P16	ABIRAMI P	60	2
P17	SUPRAJA K	70	5

## 6) TO COMPUTE AND DISPLAY THE INPATIENT's ID, NAME WHO HAVE STAYED MAXIMUM NUMBER OF DAYS IN ROOM ALONG WTH THEIR NO OF DAYS:

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** mysql1
- Table:** in\_patient
- Query:** SELECT P\_ID, Name ,MAX(DATEDIFF(DOD,DOA)) as max\_stay FROM in\_patient GROUP BY DATEDIFF(DOD,DOA) ORDER BY (DATEDIFF(DOD,DOA)) DESC LIMIT 1
- Results:** 1 row returned in 0.0021 seconds.
- Table Headers:** P\_ID, Name, max\_stay
- Data:**

P_ID	Name	max_stay
P17	SUPRAJA K	5

**7) TO DISPLAY THE DOCTOR's ID AND NAME WHO HAVE TREATED THE INPATIENTS BUT NOT THE OUTPATIENT's:**

The screenshot shows the phpMyAdmin interface for a MySQL database named 'mysql1'. The current table is 'doctor'. The query executed is:

```
SELECT D.doctor_id ,D.doctor_name from doctor D WHERE D.doctor_id IN(SELECT in_patient.Doc_Id from in_patient WHERE in_patient.Doc_ID NOT IN(SELECT out_patient.doc_id FROM out_patient))
```

The results show two rows of data:

	doctor_id	doctor_name
<input type="checkbox"/>	D6	SHANTHANU
<input type="checkbox"/>	D7	POOJA

**8) TO DISPLAY THE DOCTOR's ID AND NAME WHO HAVE TREATED THE OUT-PATIENTS BUT NOT THE INPATIENT's:**

The screenshot shows the phpMyAdmin interface for a MySQL database named 'mysql1'. The current table is 'doctor'. The query executed is:

```
SELECT D.doctor_id ,D.doctor_name from doctor D WHERE D.doctor_id IN(SELECT out_patient.doc_id FROM out_patient where out_patient.doc_id NOT IN(SELECT in_patient.Doc_Id from in_patient))
```

The results show one row of data:

	doctor_id	doctor_name
<input type="checkbox"/>	D9	VARSHINI

## 9) TO DISPLAY THE LAB DETAILS WHERE THE LAB DATE IS THE CURRENT DATE

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** mysql1
- Table:** lab
- Toolbar:** Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, Tracking, Triggers.
- Query Editor:**
  - Show query box
  - Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.
  - Showing rows 0 - 0 (1 total, Query took 0.0016 seconds.)
  - SELECT \* FROM lab where l\_date=CURRENT\_DATE
  - Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]
- Table View:**
  - Show all | Number of rows: 25 | Filter rows: Search this table
  - + Options
  - lab\_no P\_id Doc\_ID l\_date
  - 11 P18 D1 2021-05-19

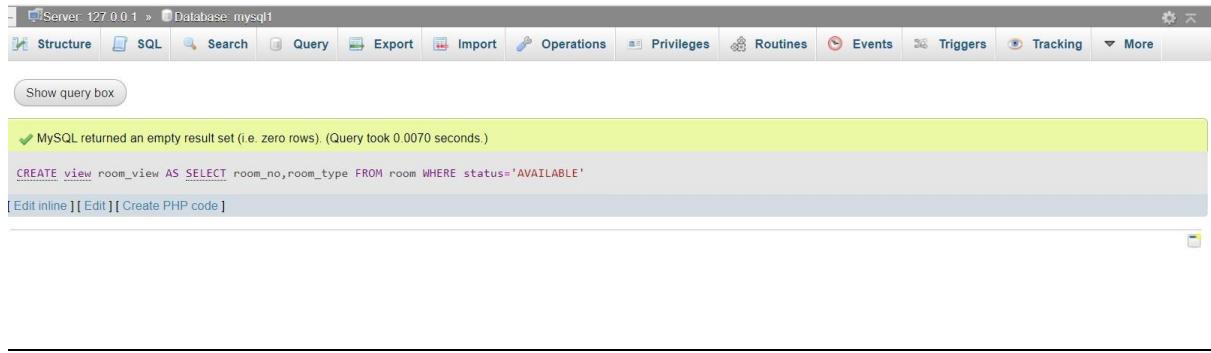
## 10) TO DISPLAY THE DETAILS OF THE BILL WHOSE MEDICAL CHARGE IS MINIMUM:

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** mysql1
- Table:** bill
- Toolbar:** Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, Tracking, Triggers.
- Query Editor:**
  - Show query box
  - Showing rows 0 - 0 (1 total, Query took 0.0012 seconds.)
  - select bill\_no,P\_id, med\_fee as MIN\_AMOUNT FROM bill where med\_fee in(SELECT min(med\_fee) FROM bill)
  - Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]
- Table View:**
  - Show all | Number of rows: 25 | Filter rows: Search this table
  - Options
  - bill\_no P\_id MIN\_AMOUNT
  - 17789 P555 200

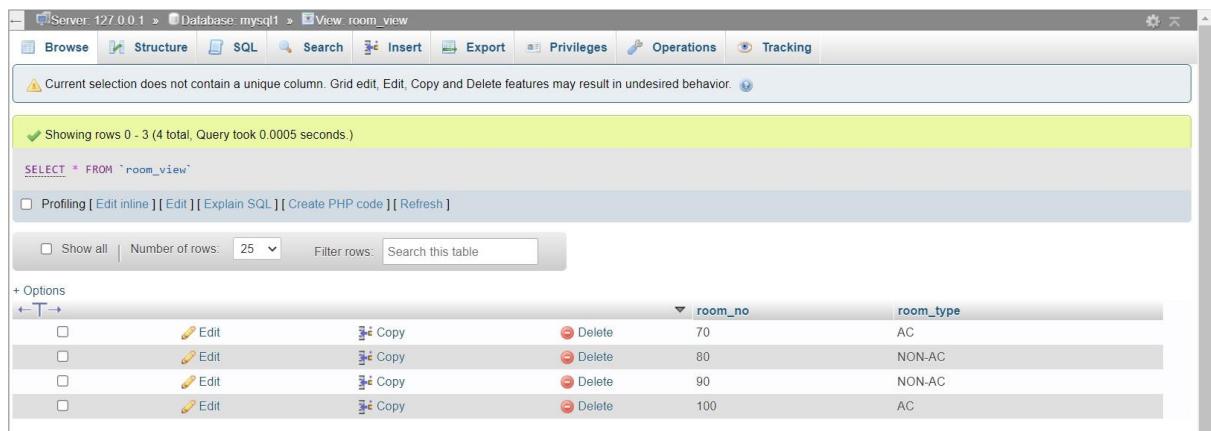
## VIEWS:

### 1)CREATE A VIEW FOR ROOM TO DISPLAY THE ROOM-NO AND ROOMTYPE WHOSE STATUS IS 'AVAILABLE':



```
CREATE view room_view AS SELECT room_no,room_type FROM room WHERE status='AVAILABLE'
```

## OUTPUT:



			room_no	room_type
<input type="checkbox"/>			70	AC
<input type="checkbox"/>			80	NON-AC
<input type="checkbox"/>			90	NON-AC
<input type="checkbox"/>			100	AC

## 2) CREATE A VIEW FOR DOCTOR TO DISPLAY DOCTOR's NAME WHOSE NAME STARTS WITH 'D':

The screenshot shows the MySQL Workbench interface. In the top navigation bar, 'Server: 127.0.0.1' and 'Database: mysql' are selected. Below the navigation bar, there are tabs for Structure, SQL, Search, Query, Export, Import, Operations, Privileges, Routines, Events, Triggers, Tracking, and More. A button 'Show query box' is visible. The main area contains a green message box stating 'MySQL returned an empty result set (i.e. zero rows). (Query took 0.0082 seconds.)'. Below it is a code editor window containing the SQL command: 'CREATE VIEW doctor\_view AS SELECT doctor\_name FROM doctor WHERE doctor\_name LIKE 'D%''. There are buttons for 'Edit inline', 'Edit', and 'Create PHP code'.

## OUTPUT:

The screenshot shows the MySQL Workbench interface with the database set to 'mysql' and the view set to 'doctor\_view'. The top navigation bar includes 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', 'Privileges', 'Operations', and 'Tracking'. A warning message says 'Current selection does not contain a unique column. Grid edit, Edit, Copy and Delete features may result in undesired behavior.' Below the message, a green message box says 'Showing rows 0 - 1 (total: 2 total). Query took 0.0006 seconds.' The SQL query shown is 'SELECT \* FROM `doctor\_view`'. The results table has one column 'doctor\_name' with two rows: 'DHANU' and 'DIVYA'. Each row has checkboxes for 'Edit' and 'Copy', and buttons for 'Delete'.

doctor_name
DHANU
DIVYA

## PROCEDURES:

### 1) CREATE A PROCEDURE TO FIND THE DOCTORS WHO HAVE NOT ATTENDED IN-PATIENTS:

```
Server: 127.0.0.1 » Database: mysql
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Tracking Designer

Your SQL query has been executed successfully.
5 rows affected by the last statement inside the procedure.

CALL `DOCTOR_NOTATTENDED_INPATIENT`();

Execution results of routine `DOCTOR_NOTATTENDED_INPATIENT`

doctor_name
PRIYA
RAMYA S
SUBHI
HEMA
VARSHINI
```

### 2) CREATE A PROCEDURE TO FIND THE DOCTORS WHO HAVE NOT ATTENDED OUT-PATIENTS:

```
Server: 127.0.0.1 » Database: mysql
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Tracking Designer

Your SQL query has been executed successfully.
6 rows affected by the last statement inside the procedure.

CALL `DOCTOR_NOTATTENDED_OUTPATIENT`();

Execution results of routine `DOCTOR_NOTATTENDED_OUTPATIENT`

doctor_name
PRIYA
RAMYA S
SUBHI
SHANTHANU
POOJA
HEMA
```

### 3) CREATE A PROCEDURE TO DISPLAY THE IN-PATIENT's ID, NAME WHO HAVE GIVEN TEST IN LAB ALONG WITH THEIR CORRESPONDING LAB-NO:

```
Server: 127.0.0.1 » Database: mysql
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Tracking Designer

Your SQL query has been executed successfully.
5 rows affected by the last statement inside the procedure.

CALL `lab_inpatient`();

Execution results of routine `lab_inpatient`

P_ID Name lab_no
P11 HARI R 11
P12 KANAGA H 12
P13 VINYC K 13
P14 KRISHNA M 14
P18 HARSHAT T 11
```

**4)CREATE A PROCEDURE TO DISPLAY THE OUT-PATIENT's ID, NAME WHO HAVE GIVEN TEST IN LAB ALONG WITH THEIR CORRESPONDING LAB-NO:**

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** mysql1
- Toolbar:** Structure, SQL, Search, Query, Export, Import, Operations, Privileges, Routines, Events, Triggers, Tracking, Designer.
- Status Bar:** Your SQL query has been executed successfully. 2 rows affected by the last statement inside the procedure.
- Query Editor:** CALL `lab\_outpatient`();
- Execution Results:** Execution results of routine `lab\_outpatient` (Table)
 

P_ID	Name	lab_no
P111	VAANATHI G	15
P999	PREETHI G	16

**5)CREATE A PROCEDURE TO FIND THE AVERAGE BILL AMOUNT:**

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** mysql1
- Toolbar:** Structure, SQL, Search, Query, Export, Import, Operations, Privileges, Routines, Events, Triggers, Tracking, Designer.
- Status Bar:** Your SQL query has been executed successfully. 1 row affected by the last statement inside the procedure.
- Query Editor:** CALL `billtotal\_avg`();
- Execution Results:** Execution results of routine `billtotal\_avg` (Table)
 

average_billamount
6650.7778

**6)CREATE A PROCEDURE TO FIND THE TOTAL SUM OF BILL AMOUNT:**

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** mysql1
- Toolbar:** Structure, SQL, Search, Query, Export, Import, Operations, Privileges, Routines, Events, Triggers, Tracking, Designer.
- Status Bar:** Your SQL query has been executed successfully. 1 row affected by the last statement inside the procedure.
- Query Editor:** CALL `BILL\_SUM`();
- Execution Results:** Execution results of routine `BILL\_SUM` (Table)
 

TOTAL_AMOUNT
119714

## TRIGGER:

The screenshot shows the phpMyAdmin interface with the 'Triggers' tab selected. The database is set to 'mysql'. A single trigger is listed:

Name	Table	Action	Time	Event
delete_lab	out_patient	Edit   Export   Drop	AFTER	DELETE

The left sidebar shows the database structure, including tables like 'bill', 'credentials', 'doctor', 'in\_patient', 'lab', 'out\_patient', 'room', and 'staff'.

## QUERY:

```

CREATE TRIGGER `delete_lab` AFTER DELETE ON `out_patient`
FOR EACH ROW DELETE FROM lab where P_id=old.P_id
  
```

The modal window displays the SQL code for the 'delete\_lab' trigger:

```

CREATE TRIGGER `delete_lab` AFTER DELETE ON `out_patient`
FOR EACH ROW DELETE FROM lab where P_id=old.P_id
  
```

At the bottom right of the modal is a 'Close' button.

## SCHEMA OF OUTPATIENT AT BEGINNING:

localhost / 127.0.0.1 / mysql1 / + localhost/phpmyadmin/index.php?route=/sql&server=1&db=mysql1&table=out\_patient

**phpMyAdmin**

Showing rows 0 - 14 (15 total, Query took 0.0010 seconds.)

SELECT \* FROM `out\_patient`

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

P_ID	DATE	name	doc_id
P1000	2021-04-25	RAJU S	D1
P1001	2021-05-17	AISHWARYA K	D4
P1002	2021-05-24	MALIGA H	D4
P1003	2021-05-23	KARTHIKK D	D1
P111	2021-05-03	VAANATHI G	D4
P1200	2021-05-19	ADITYA D	D2
P222	2021-05-05	MITHUN K	D2
P333	2021-04-21	BHARATHI A	D9
P444	2021-04-19	ARUN T	D9
P555	2021-05-09	KISHORE J	D1
P666	2021-04-14	VICKY J	D3
P777	2021-04-28	ARJUN P	D3
P888	2021-05-11	ROJA A	D4
P991	2021-05-21	ZAARA D	D2
P999	2021-05-07	PREETHI G	D4

Check all With selected: Edit Copy Delete Export

## SCHEMA OF LAB AT BEGINNING:

localhost / 127.0.0.1 / mysql1 / + localhost/phpmyadmin/index.php?route=/sql&server=1&db=mysql1&table=lab&pos=0

**phpMyAdmin**

Showing rows 0 - 8 (9 total, Query took 0.0012 seconds.)

SELECT \* FROM `lab`

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

lab_no	P_Id	Doc_ID	l_date
11	P11	D1	2021-05-12
12	P12	D2	2020-05-11
13	P13	D3	2021-04-12
14	P14	D4	2021-04-15
15	P111	D1	2021-05-03
16	P999	D9	2021-05-07
11	P18	D1	2021-05-19
12	P1003	D1	2021-05-23
13	P991	D2	2021-05-30

Query results operations

Print Copy to clipboard Export Display chart Create view

Bookmark this SQL query

## HERE P991 IS BEING DELETED:

The screenshot shows the phpMyAdmin interface for the 'out\_patient' table. A yellow box highlights the status message '1 row affected. (Query took 0.0165 seconds.)' above the data grid. The data grid shows 15 rows of patient information, including columns: P\_ID, DATE, name, and doc\_id. Row P991 is identified by the status message.

P_ID	DATE	name	doc_id
P1000	2021-04-25	RAJU S	D1
P1001	2021-05-17	AISHWARYA K	D4
P1002	2021-05-24	MALIGA H	D4
P1003	2021-05-23	KARTHIK D	D1
P111	2021-05-03	VAANATHI G	D4
P1200	2021-05-19	ADITYA D	D2
P222	2021-05-05	MITHUN K	D2
P333	2021-04-21	BHARATHI A	D9
P444	2021-04-19	ARUN T	D9
P555	2021-05-09	KISHORE J	D1
P666	2021-04-14	VICKY J	D3
P777	2021-04-28	ARJUN P	D3
P888	2021-05-11	ROJA A	D4
P999	2021-05-07	PREETHI G	D4

## THE DELETE LAB TRIGGER DELETES THE ROW OF THE SAME OUTPATIENT ID:

The screenshot shows the phpMyAdmin interface for the 'lab' table. A yellow box highlights the status message 'Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.' above the data grid. The data grid shows 8 rows of lab test information, including columns: lab\_no, P\_id, Doc\_ID, and I\_date. Row P991 is identified by the status message.

lab_no	P_id	Doc_ID	I_date
11	P11	D1	2021-05-12
12	P12	D2	2020-05-11
13	P13	D3	2021-04-12
14	P14	D4	2021-04-15
15	P111	D1	2021-05-03
16	P999	D9	2021-05-07
11	P18	D1	2021-05-19
12	P1003	D1	2021-05-23

## **CONCLUSION:**

- ❖ A Hospital Management System is a computer or web-based system that facilities managing the functioning of a hospital.
- ❖ It takes care of all the requirements of an average hospital and is capable to provide easy and effective storage of information related to patients that come up to the hospital.
- ❖ It also provides billing facility for both indoor and outdoor patients.
- ❖ In this all the functional and non-functional requirements are specified in order to get a clear-cut idea to develop a project.

## **REFERENCES:**

[https://www.w3schools.com/php/php\\_mysql\\_connect.asp](https://www.w3schools.com/php/php_mysql_connect.asp)

<https://www.geeksforgeeks.org/how-to-insert-form-data-into-database-using-php/>

[https://www.w3schools.com/php/func mysqli\\_query.asp](https://www.w3schools.com/php/func mysqli_query.asp)

<https://www.php.net/manual/en/mysqli.query.php>