





Assignment-2

DBMS



Zunaira Abdul Aziz
BSSE23058
Section A



BSSE23058
ZUNAIRA ABDUL AZIZ
SECTION A

Importing Data from Excel Sheet

1. Patient

The first screenshot shows the phpMyAdmin interface with the 'Import' tab selected. The 'patient' table is selected in the database 'assignment2_bsse23058'. The 'Import' button is clicked, and a CSV file is imported. The SQL console shows the following queries:

```
CREATE TABLE IF NOT EXISTS `assignment2_bsse23058`.`patient` (`Patient_id` int(2), `name` varchar(15), `date_of_birth` varchar(10), `ailment` varchar(14)) DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;
```

```
INSERT INTO `assignment2_bsse23058`.`patient` (`Patient_id`, `name`, `date_of_birth`, `ailment`) VALUES (1, 'Waqas Masood', '01-Jan-63', 'diabetes'), (2, 'Jawad Rasheed', '01-Jul-33', 'Paralysis'), (3, 'Mussarat Sarwar', '26-Jun-71', 'cancer'), (4, 'Nazia Junaid', '16-Oct-78', 'PTSD'), (5, 'Taimur Anwar', '08-Mar-88', 'gastro'), (6, 'Daniyal Ajmal', '26-Sept-59', 'kidney failure'), (7, 'Umair Suleman', '04-May-53', 'angina'), (8, 'Osaid Javaid', '02-Jul-83', 'gastro'), (9, 'Sarmad Khan', '26-Jun-71', 'bronchitis'), (10, 'Ammar Nabeel', '29-Oct-77', 'PTSD'), (11, 'Muhammad Salman', '08-Mar-85', 'cancer'), (12, 'Khurram Mehmood', '26-Sept-88', 'Paralysis'), (13, 'Wajid Haider', '01-Jan-93', 'cancer'), (14, 'Kamran Ali', '04-Aug-48', 'angina'), (15, 'Namra Nadeem', '26-Jun-55', 'kidney failure'), (16, 'Affan Ahmed', '16-Oct-75', 'epilepsy'), (17, 'Ali Sikander', '08-Feb-88', 'angina'), (18, 'Salman Khalid', '26-Sept-90', 'gastro'), (19, 'Aamir Shaikat', '26-Sept-88', 'gastro');
```

The second screenshot shows the 'Table' view of the 'patient' table. The table contains 26 rows of data:

| Patient_id | name | date_of_birth | ailment |
|------------|-----------------|---------------|----------------|
| 1 | Waqas Masood | 01-Jan-63 | diabetes |
| 2 | Jawad Rasheed | 01-Jul-33 | Paralysis |
| 3 | Mussarat Sarwar | 26-Jun-71 | cancer |
| 4 | Nazia Junaid | 16-Oct-78 | PTSD |
| 5 | Taimur Anwar | 08-Mar-88 | gastro |
| 6 | Daniyal Ajmal | 26-Sept-59 | kidney failure |
| 7 | Umair Suleman | 04-May-53 | angina |
| 8 | Osaid Javaid | 02-Jul-83 | gastro |
| 9 | Sarmad Khan | 26-Jun-71 | bronchitis |
| 10 | Ammar Nabeel | 29-Oct-77 | PTSD |
| 11 | Muhammad Salman | 08-Mar-85 | cancer |
| 12 | Khurram Mehmood | 26-Sept-88 | Paralysis |
| 13 | Wajid Haider | 01-Jan-93 | cancer |
| 14 | Kamran Ali | 04-Aug-48 | angina |
| 15 | Namra Nadeem | 26-Jun-55 | kidney failure |
| 16 | Affan Ahmed | 16-Oct-75 | epilepsy |
| 17 | Ali Sikander | 08-Feb-88 | angina |
| 18 | Salman Khalid | 26-Sept-90 | gastro |
| 19 | Aamir Shaikat | 26-Sept-88 | gastro |

BSSE23058
ZUNAIRA ABDUL AZIZ
SECTION A
2. Doctor

The first screenshot shows the phpMyAdmin interface with the 'doctor' table being created in the 'assignment2_bsse23058' database. The table structure is defined with columns: Doc_id (int(2)), doc_name (varchar(14)), qualification (varchar(4)), fee (int(4)), dep (int(1)), and supervisor (varchar(4)). The table is created with the DEFAULT CHARACTER SET utf8 and COLLATE utf8_general_ci. The second screenshot shows the 'doctor' table with 16 rows of data inserted. The data includes doctor names, qualifications, fees, departments, and supervisors.

Table Structure:

```
CREATE TABLE IF NOT EXISTS `assignment2_bsse23058`.`doctor` (
  `Doc_id` int(2),
  `doc_name` varchar(14),
  `qualification` varchar(4),
  `fee` int(4),
  `dep` int(1),
  `supervisor` varchar(4)
) DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;
```

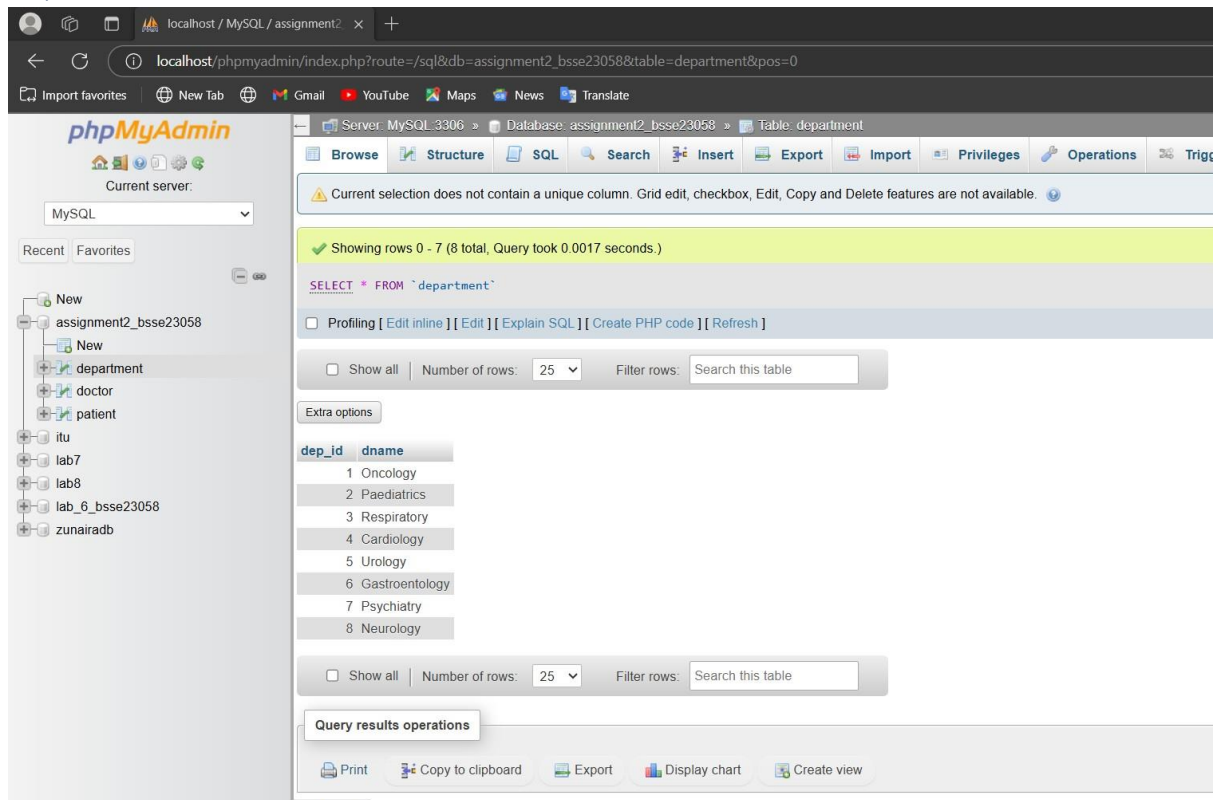
Data Inserted:

```
INSERT INTO `assignment2_bsse23058`.`doctor` (
  `Doc_id`, `doc_name`, `qualification`, `fee`, `dep`, `supervisor`
) VALUES (
  1, 'Tareen Ali', 'MRCP', 1000, 7, 'null',
  2, 'Aamna Khalid', 'MBBS', 500, 7, '1',
  3, 'Amir Ikram', 'MRCP', 1000, 8, '2',
  4, 'Naeem Kasuri', 'FRCP', 1200, 8, 'null',
  5, 'Jamshed Khan', 'MD', 1200, 1, 'null',
  6, 'Fatima Saif', 'MRCP', 1000, 1, '5',
  7, 'Iftikhar Ali', 'FCPS', 800, 1, '6',
  8, 'Tayyaba Khawar', 'FRCP', 1200, 2, 'null',
  9, 'Ayesha Hanif', 'MRCP', 1000, 2, '8',
  10, 'Saqib Saeed', 'FRCP', 1200, 3, 'null',
  11, 'Adnan Younus', 'MRCP', 1000, 3, '10',
  12, 'Ali Kiyani', 'FRCP', 1200, 5, 'null',
  13, 'Rida Gillani', 'FRCP', 1200, 4, '12',
  14, 'Bilal Zakriyah', 'FRCP', 1200, 4, 'null',
  15, 'Yousaf Iqbal', 'MRCP', 1000, 6, '14',
  16, 'Fahmida Raza', 'FRCP', 1200, 6, 'null'
);
```

Table Data:

| Doc_id | doc_name | qualification | fee | dep | supervisor |
|--------|----------------|---------------|------|-----|------------|
| 1 | Tareen Ali | MRCP | 1000 | 7 | null |
| 2 | Aamna Khalid | MBBS | 500 | 7 | 1 |
| 3 | Amir Ikram | MRCP | 1000 | 8 | 2 |
| 4 | Naeem Kasuri | FRCP | 1200 | 8 | null |
| 5 | Jamshed Khan | MD | 1200 | 1 | null |
| 6 | Fatima Saif | MRCP | 1000 | 1 | 5 |
| 7 | Iftikhar Ali | FCPS | 800 | 1 | 6 |
| 8 | Tayyaba Khawar | FRCP | 1200 | 2 | null |
| 9 | Ayesha Hanif | MRCP | 1000 | 2 | 8 |
| 10 | Saqib Saeed | FRCP | 1200 | 3 | null |
| 11 | Adnan Younus | MRCP | 1000 | 3 | 10 |
| 12 | Ali Kiyani | FRCP | 1200 | 5 | null |
| 13 | Rida Gillani | FRCP | 1200 | 4 | 12 |
| 14 | Bilal Zakriyah | FRCP | 1200 | 4 | null |
| 15 | Yousaf Iqbal | MRCP | 1000 | 6 | 14 |
| 16 | Fahmida Raza | FRCP | 1200 | 6 | null |

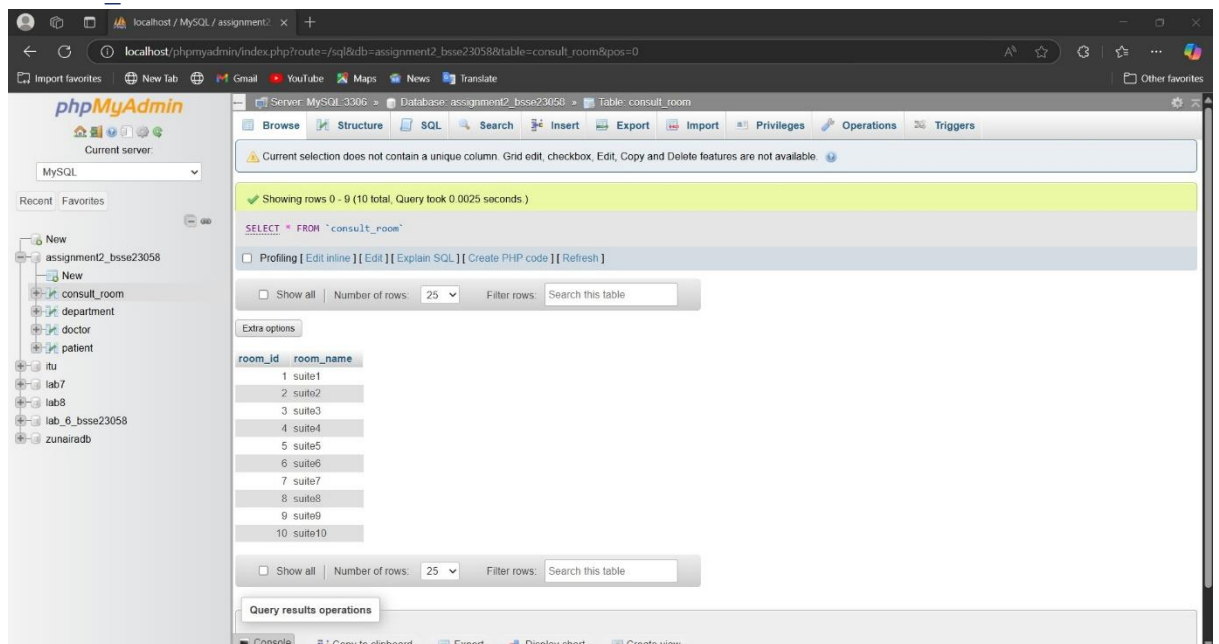
3. Department



The screenshot shows the phpMyAdmin interface for a MySQL database named 'assignment2_bsse23058'. The 'department' table is selected, displaying 8 rows of data. The table structure shows columns 'dep_id' and 'dname'.

| dep_id | dname |
|--------|----------------|
| 1 | Oncology |
| 2 | Paediatrics |
| 3 | Respiratory |
| 4 | Cardiology |
| 5 | Urology |
| 6 | Gastroentology |
| 7 | Psychiatry |
| 8 | Neurology |

4. Consult_room



The screenshot shows the phpMyAdmin interface for a MySQL database named 'assignment2_bsse23058'. The 'consult_room' table is selected, displaying 10 rows of data. The table structure shows columns 'room_id' and 'room_name'.

| room_id | room_name |
|---------|-----------|
| 1 | suite1 |
| 2 | suite2 |
| 3 | suite3 |
| 4 | suite4 |
| 5 | suite5 |
| 6 | suite6 |
| 7 | suite7 |
| 8 | suite8 |
| 9 | suite9 |
| 10 | suite10 |

BSSE23058

ZUNAIRA ABDUL AZIZ

SECTION A

5. Doctor_consult

The screenshot shows the phpMyAdmin web interface in a browser. The URL is `localhost/phpmyadmin/index.php?route=/sql&db=assignment2_bsse23058&table=doctor_consult&pos=0`. The interface displays the 'doctor_consult' table from the 'assignment2_bsse23058' database. A message indicates that the current selection does not contain a unique column, so grid edit, checkbox, Edit, Copy, and Delete features are not available. The table shows 16 rows of data, with columns: Doc_id, Room_id, Day_of_week, Start_time, and End_time. The data is as follows:

| Doc_id | Room_id | Day_of_week | Start_time | End_time |
|--------|---------|-------------|------------|----------|
| 1 | 1 | Thursday | 12:00:00 | 15:00:00 |
| 2 | 2 | Monday | 12:00:00 | 15:00:00 |
| 3 | 3 | Thursday | 12:00:00 | 15:00:00 |
| 4 | 4 | Monday | 12:00:00 | 15:00:00 |
| 5 | 5 | Thursday | 12:00:00 | 15:00:00 |
| 6 | 6 | Monday | 12:00:00 | 15:00:00 |
| 7 | 7 | Monday | 12:00:00 | 15:00:00 |
| 8 | 8 | Thursday | 12:00:00 | 15:00:00 |
| 9 | 1 | Thursday | 12:00:00 | 15:00:00 |
| 10 | 2 | Monday | 12:00:00 | 15:00:00 |
| 11 | 3 | Monday | 12:00:00 | 15:00:00 |
| 12 | 4 | Thursday | 12:00:00 | 15:00:00 |
| 13 | 5 | Thursday | 12:00:00 | 15:00:00 |
| 14 | 6 | Monday | 12:00:00 | 15:00:00 |
| 15 | 7 | Thursday | 12:00:00 | 15:00:00 |
| 16 | 8 | Thursday | 12:00:00 | 15:00:00 |

BSSE23058
ZUNAIRA ABDUL AZIZ
SECTION A

6. Patient_appointment

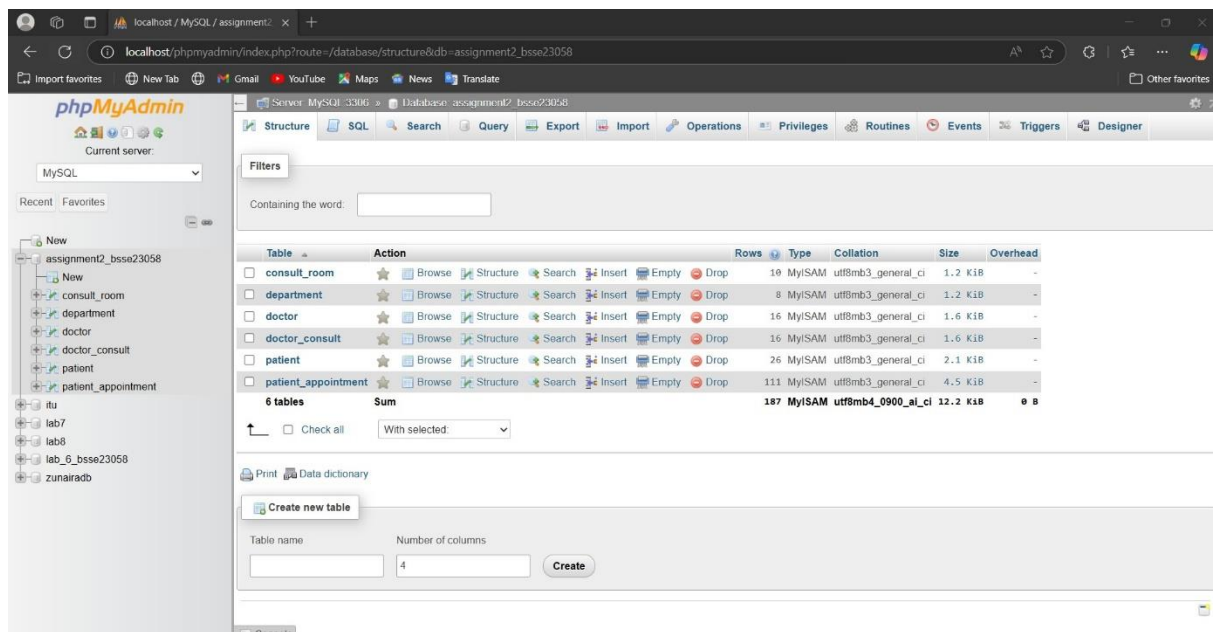
The first screenshot shows the phpMyAdmin interface with the 'patient_appointment' table selected. The table structure is as follows:

| Doc_id | Pat_id | date | Start_time |
|--------|-----------|-----------|------------|
| 1 | 4 | 05-Nov-15 | 12:00:00 |
| 1 | 4 | 19-Nov-15 | 13:00:00 |
| 1 | 10 | 05-Nov-15 | 12:30:00 |
| 1 | 20 | 12-Nov-15 | 12:00:00 |
| 2 | 20 | 02-Nov-15 | 12:00:00 |
| 2 | 10 | 12-Nov-15 | 12:00:00 |
| 15 | 1 | 05-Nov-15 | 12:00:00 |
| 15 | 5 | 05-Nov-15 | 12:30:00 |
| 15 | 8 | 05-Nov-15 | 13:00:00 |
| 15 | 18 | 05-Nov-15 | 13:30:00 |
| 15 | 1 | 26-Nov-15 | 12:00:00 |
| 15 | 5 | 26-Nov-15 | 12:30:00 |
| 15 | 18 | 26-Nov-15 | 13:30:00 |
| 16 | 8 | 19-Nov-15 | 12:00:00 |
| 16 | 5 | 26-Nov-15 | 12:00:00 |
| 21 | 05-Nov-15 | 12:00:00 | |

The second screenshot shows the phpMyAdmin interface with the 'Patient_appointment' table selected. The table structure is as follows:

| Doc_id | Pat_id | date | Start_time |
|--------|-----------|-----------|------------|
| 1 | 4 | 05-Nov-15 | 12:00:00 |
| 1 | 4 | 19-Nov-15 | 13:00:00 |
| 1 | 10 | 05-Nov-15 | 12:30:00 |
| 1 | 20 | 12-Nov-15 | 12:00:00 |
| 2 | 20 | 02-Nov-15 | 12:00:00 |
| 2 | 10 | 12-Nov-15 | 12:00:00 |
| 15 | 1 | 05-Nov-15 | 12:00:00 |
| 15 | 5 | 05-Nov-15 | 12:30:00 |
| 15 | 8 | 05-Nov-15 | 13:00:00 |
| 15 | 18 | 05-Nov-15 | 13:30:00 |
| 15 | 1 | 26-Nov-15 | 12:00:00 |
| 15 | 5 | 26-Nov-15 | 12:30:00 |
| 15 | 18 | 26-Nov-15 | 13:30:00 |
| 16 | 8 | 19-Nov-15 | 12:00:00 |
| 16 | 5 | 26-Nov-15 | 12:00:00 |
| 21 | 05-Nov-15 | 12:00:00 | |

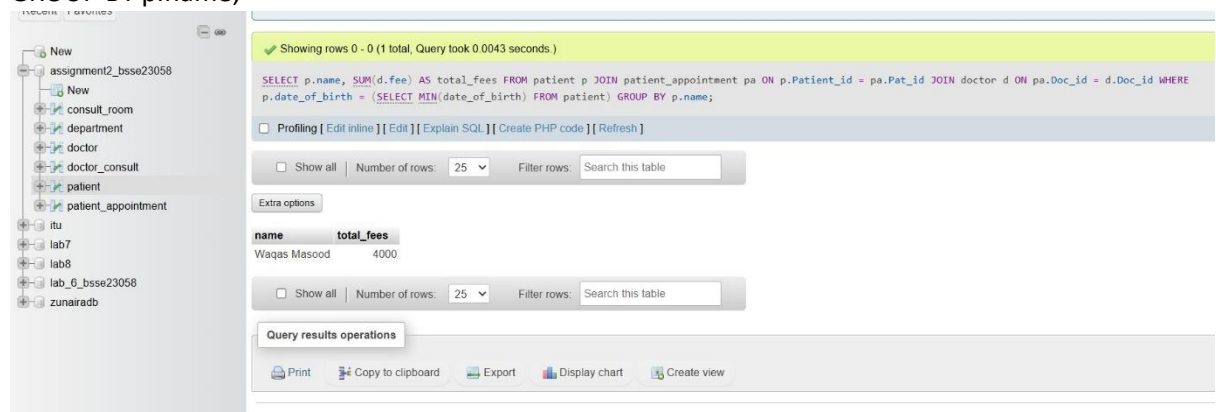
7. Database + All Tables



Question 1

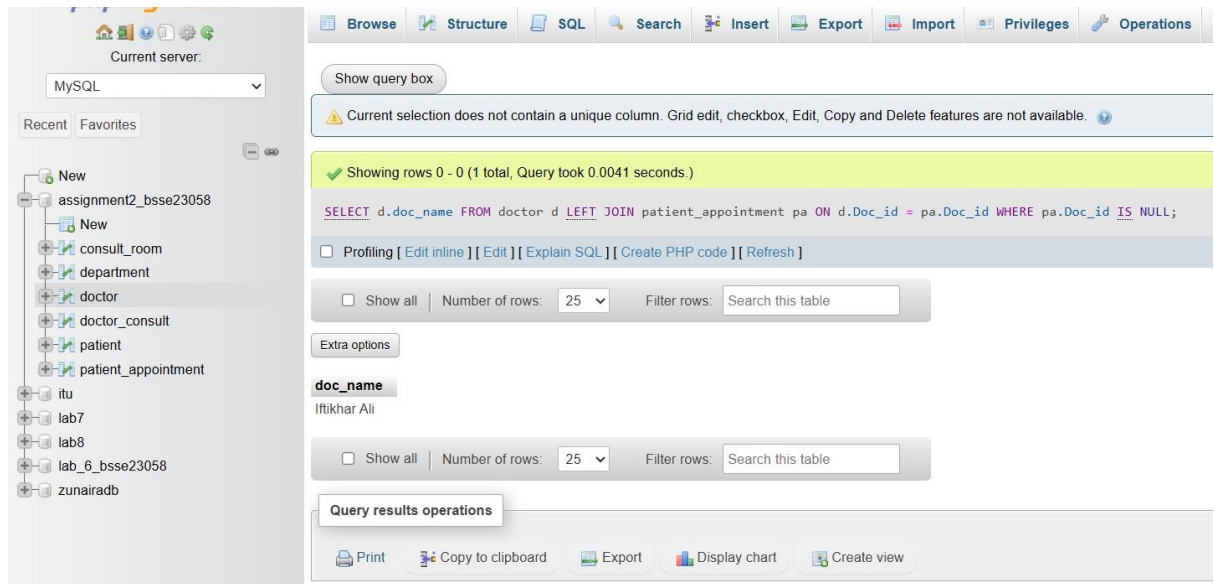
1.

```
SELECT p.name, SUM(d.fee) AS total_fees
FROM patient p
JOIN patient_appointment pa ON p.Patient_id = pa.Pat_id
JOIN doctor d ON pa.Doc_id = d.Doc_id
WHERE p.date_of_birth = (SELECT MIN(date_of_birth) FROM patient)
GROUP BY p.name;
```



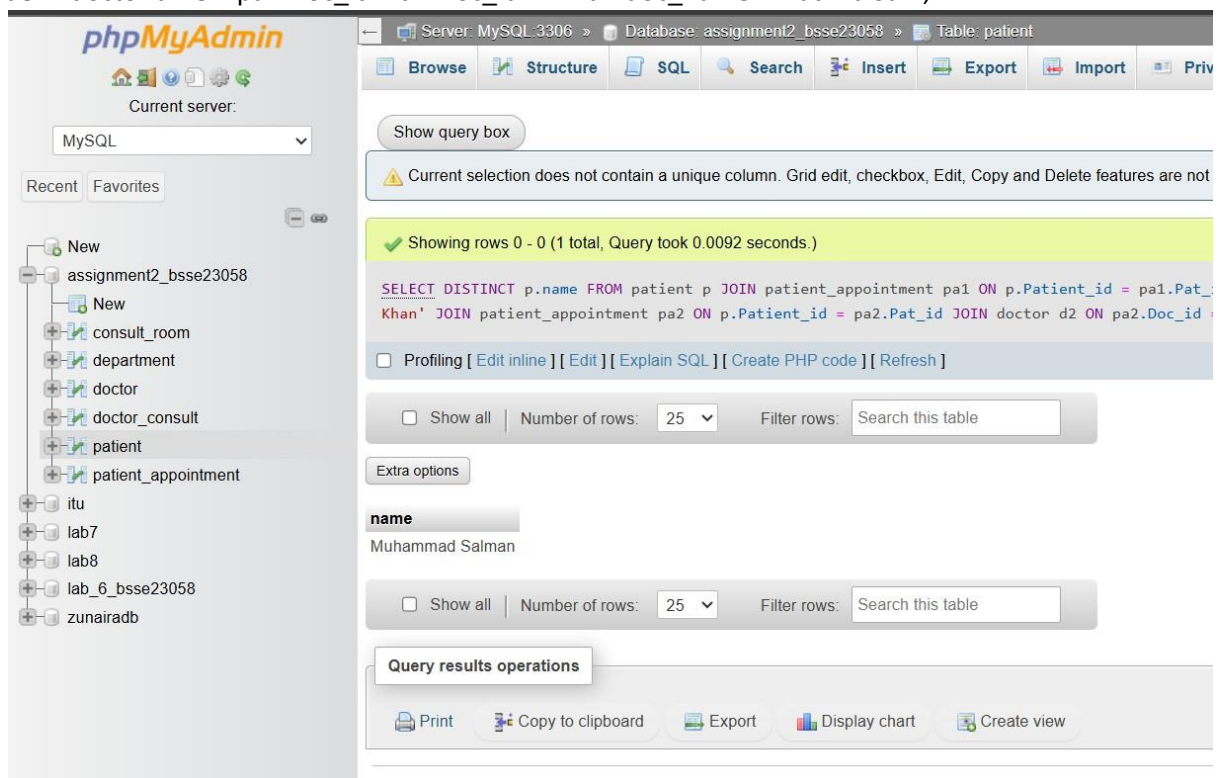
2.

```
SELECT d.doc_name
FROM doctor d
LEFT JOIN patient_appointment pa ON d.Doc_id = pa.Doc_id
WHERE pa.Doc_id IS NULL;
```



3.

```
SELECT DISTINCT p.name
FROM patient p
JOIN patient_appointment pa1 ON p.Patient_id = pa1.Pat_id
JOIN doctor d1 ON pa1.Doc_id = d1.Doc_id AND d1.doc_name = 'Jamshed Khan'
JOIN patient_appointment pa2 ON p.Patient_id = pa2.Pat_id
JOIN doctor d2 ON pa2.Doc_id = d2.Doc_id AND d2.doc_name = 'Fatima Saif';
```



4.

```
SELECT DISTINCT p.name
FROM patient p
JOIN patient_appointment pa1 ON p.Patient_id = pa1.Pat_id
```



```
JOIN doctor d1 ON pa1.Doc_id = d1.Doc_id AND d1.doc_name = 'Jamshed Khan'  
WHERE NOT EXISTS (  
    SELECT 1  
    FROM patient_appointment pa2  
    JOIN doctor d2 ON pa2.Doc_id = d2.Doc_id  
    WHERE pa2.Pat_id = p.Patient_id AND d2.doc_name = 'Fatima Saif'  
);
```

The screenshot shows the phpMyAdmin interface. On the left is the database navigation tree with 'assignment2_bsse23058' selected. The main panel displays a SQL query in the 'SQL' tab. The query is: `SELECT DISTINCT p.name FROM patient p JOIN patient_appointment pa1 ON p.Patient_id = pa1.Pat_id : Khan' WHERE NOT EXISTS (SELECT 1 FROM patient_appointment pa2 JOIN doctor d2 ON pa2.Doc_id = d2);`. Below the query, it shows 'Showing rows 0 - 1 (2 total, Query took 0.0083 seconds.)'. The results table has two rows: 'Mussarat Sarwar' and 'Wajid Haider'. The interface also includes a 'Query results operations' bar at the bottom with options like 'Print', 'Copy to clipboard', 'Export', 'Display chart', and 'Create view'.

5.

```
SELECT d.doc_name  
FROM doctor d  
LEFT JOIN patient_appointment pa ON d.Doc_id = pa.Doc_id  
WHERE d.supervisor IS NULL  
OR d.Doc_id IN (  
    SELECT Doc_id  
    FROM patient_appointment  
    GROUP BY Doc_id  
    HAVING COUNT(DISTINCT Pat_id) > 3  
);
```

The screenshot shows the phpMyAdmin interface. On the left is a sidebar with a database tree. The main panel displays a SQL query and its results for the 'doctor' table.

Database: assignment2_bsse23058
Table: doctor

Query:

```
SELECT d.doc_name FROM doctor d LEFT JOIN patient_appointment pa ON d.Doc_id = pa.Doc_id WHERE d.supervi
patient_appointment GROUP BY Doc_id HAVING COUNT(DISTINCT Pat_id) > 3 );
```

Results: Showing rows 0 - 13 (14 total, Query took 0.0060 seconds.)

| doc_name |
|--------------|
| Yousaf Iqbal |
| Yousaf Iqbal |
| Yousaf Iqbal |
| Yousaf Iqbal |
| Yousaf Iqbal |
| Yousaf Iqbal |
| Yousaf Iqbal |
| Yousaf Iqbal |
| Yousaf Iqbal |
| Yousaf Iqbal |
| Yousaf Iqbal |
| Yousaf Iqbal |
| Yousaf Iqbal |
| Yousaf Iqbal |

6.

```
SELECT d.doc_name, COUNT(pa.doc_id) * d.fee AS total_earnings
FROM doctor d
JOIN patient_appointment pa ON d.doc_id = pa.doc_id
WHERE EXTRACT(MONTH FROM pa.date) = 10 AND EXTRACT(YEAR FROM pa.date)
=2015
GROUP BY d.doc_name, d.fee
ORDER BY total_earnings DESC
LIMIT 1;
```

The screenshot shows the phpMyAdmin interface for a MySQL server. The left sidebar displays the database structure, including a database named 'assignment2_bsse23058' with several tables. The main panel shows the 'Table: doctor' view. A query has been executed, resulting in an empty result set. The query is as follows:

```
SELECT d.doc_name, SUM(d.fee) AS total_earnings FROM doctor d JOIN patient_appointment pa ON d.Doc_id = pa.doc_id ORDER BY total_earnings DESC LIMIT 1;
```

The result set is empty, and the message states: "MySQL returned an empty result set (i.e. zero rows). (Query took 0.0021 seconds.)".

7.

```
SELECT d.doc_name, COUNT(DISTINCT pa.pat_id) AS patient_count
FROM doctor d
JOIN patient_appointment pa ON d.doc_id = pa.doc_id
GROUP BY d.doc_name
ORDER BY patient_count DESC
LIMIT 1;
```

The screenshot shows the phpMyAdmin interface for a MySQL server. The left sidebar displays the database structure, including a database named 'assignment2_bsse23058' with several tables. The main panel shows the 'Table: doctor' view. A query has been executed, resulting in one row of data. The query is as follows:

```
SELECT d.doc_name, COUNT(DISTINCT pa.pat_id) AS patient_count FROM doctor d JOIN patient_appointment pa ON d.doc_id = pa.doc_id GROUP BY d.doc_name ORDER BY patient_count DESC LIMIT 1;
```

The result set contains one row with the following data:

| doc_name | patient_count |
|--------------|---------------|
| Yousaf Iqbal | 4 |

8.

```
SELECT p.name
FROM patient p
WHERE NOT EXISTS (
```

BSSE23058
ZUNAIRA ABDUL AZIZ
SECTION A

```
SELECT d.doc_id
FROM doctor d
JOIN department dept ON d.dep = dept.dep_id
WHERE dept.dname = 'Gastroentology'
AND NOT EXISTS (
    SELECT 1
    FROM patient_appointment pa
    WHERE pa.doc_id = d.doc_id AND pa.pat_id = p.patient_id
)
);
```

The screenshot shows the phpMyAdmin interface for a MySQL server. The left sidebar displays the database structure, including a database named 'assignment2_bsse23058' with several tables. The main panel shows the 'Table: patient' view. A SQL query has been executed, and the results are displayed as a table with 2 rows. The query is a complex SELECT statement with a WHERE clause that uses a NOT EXISTS subquery to filter results. The results table has a single column named 'name' with two entries: 'Taimur Anwar' and 'Osaid Javaid'.

| name |
|--------------|
| Taimur Anwar |
| Osaid Javaid |

9.

```
SELECT p.name
FROM patient p
JOIN patient_appointment pa ON p.patient_id = pa.pat_id
JOIN doctor d ON pa.doc_id = d.doc_id
WHERE d.doc_name = 'Saqib Saeed'
AND NOT EXISTS (
    SELECT 1
    FROM patient_appointment pa2
    JOIN doctor d2 ON pa2.doc_id = d2.doc_id
    WHERE pa2.pat_id = p.patient_id AND d2.doc_name != 'Saqib Saeed'
);
```

The screenshot shows the phpMyAdmin interface for a MySQL database named 'assignment2_bsse23058'. The 'patient' table is selected, and its structure is displayed. The table has columns: name, patient_id, doc_id, and appointment_id. A query is executed, showing rows 0 to 3 (4 total). The query is:

```
SELECT p.name FROM patient p JOIN patient_appointment pa ON p.patient_id = pa.pat_id JOIN doctor d ON pa.doc_id = d.doc_id
EXISTS ( SELECT 1 FROM patient_appointment pa2 JOIN doctor d2 ON pa2.doc_id = d2.doc_id WHERE pa2.pat_id = p.patient_id )
```

The query result shows 4 rows, all with the name 'Aamir Shaukat'.

10.

SELECT cr.room_name, COALESCE(string_agg(d.doc_name, ', '), 'No doctor') AS doctors
FROM consult_room cr
LEFT JOIN doctor_consult dc ON cr.room_id = dc.room_id
LEFT JOIN doctor d ON dc.doc_id = d.doc_id
GROUP BY cr.room_name;

The screenshot shows the phpMyAdmin interface for a MySQL database named 'assignment2_bsse23058'. The 'consult_room' table is selected, and its structure is displayed. The table has columns: room_name, room_id, and doctor_id. A query is executed, showing rows 0 to 9 (10 total). The query is:

```
SELECT cr.room_name, COALESCE(GROUP_CONCAT(DISTINCT d.doc_name SEPARATOR ', '), 'No doctor') AS doctors
FROM consult_room cr LEFT JOIN doctor d ON cr.room_id = d.doc_id GROUP BY cr.room_name LIMIT 0, 25;
```

The query result shows 10 rows, listing room names and their corresponding doctors. The results are as follows:

| room_name | doctors |
|-----------|------------------------------|
| suite1 | Ayesha Hanif, Tareen Ali |
| suite10 | No doctor |
| suite2 | Amna Khalid, Saqib Saeed |
| suite3 | Adnan Younus, Amir Ikram |
| suite4 | Ali Kiyani, Naeem Kasuri |
| suite5 | Jamshed Khan, Rida Gillani |
| suite6 | Bilal Zakriyah, Fatima Saif |
| suite7 | Iftekhar Ali, Yousaf Iqbal |
| suite8 | Fahmida Raza, Tayyaba Khawar |
| suite9 | No doctor |

QUESTION 2

1.

DELIMITER //
CREATE FUNCTION CalculateAverageFeeByDepartment(dept_id INT)
RETURNS DECIMAL(10,2)
DETERMINISTIC

BSSE23058
ZUNAIRA ABDUL AZIZ
SECTION A

BEGIN

DECLARE avg_fee DECIMAL(10,2);

SELECT AVG(fee) INTO avg_fee

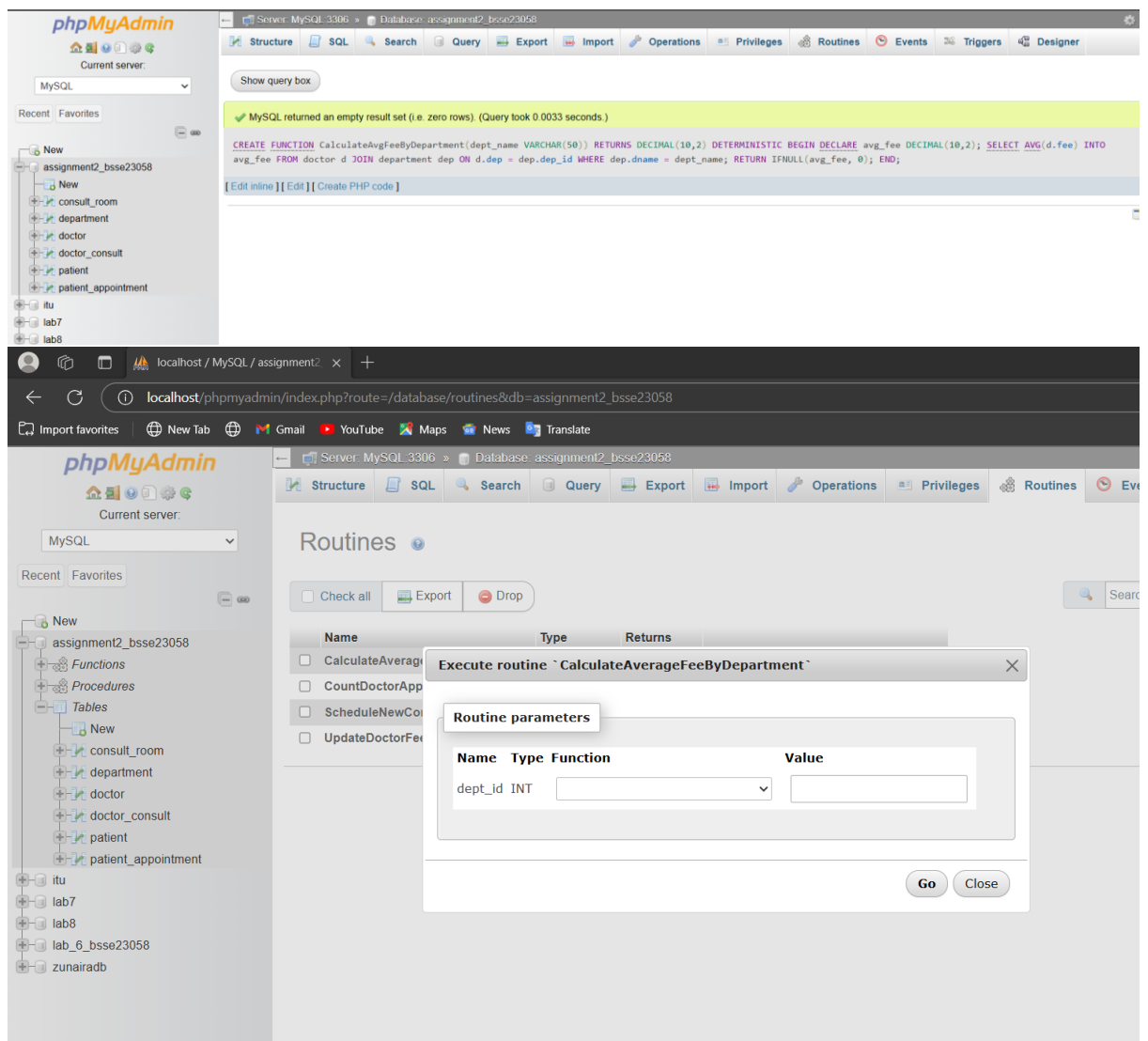
FROM doctor

WHERE dep = dept_id;

RETURN IFNULL(avg_fee, 0);

END //

DELIMITER ;



2.

DELIMITER //

CREATE FUNCTION CountDoctorAppointments(doc_id INT)

RETURNS INT

DETERMINISTIC

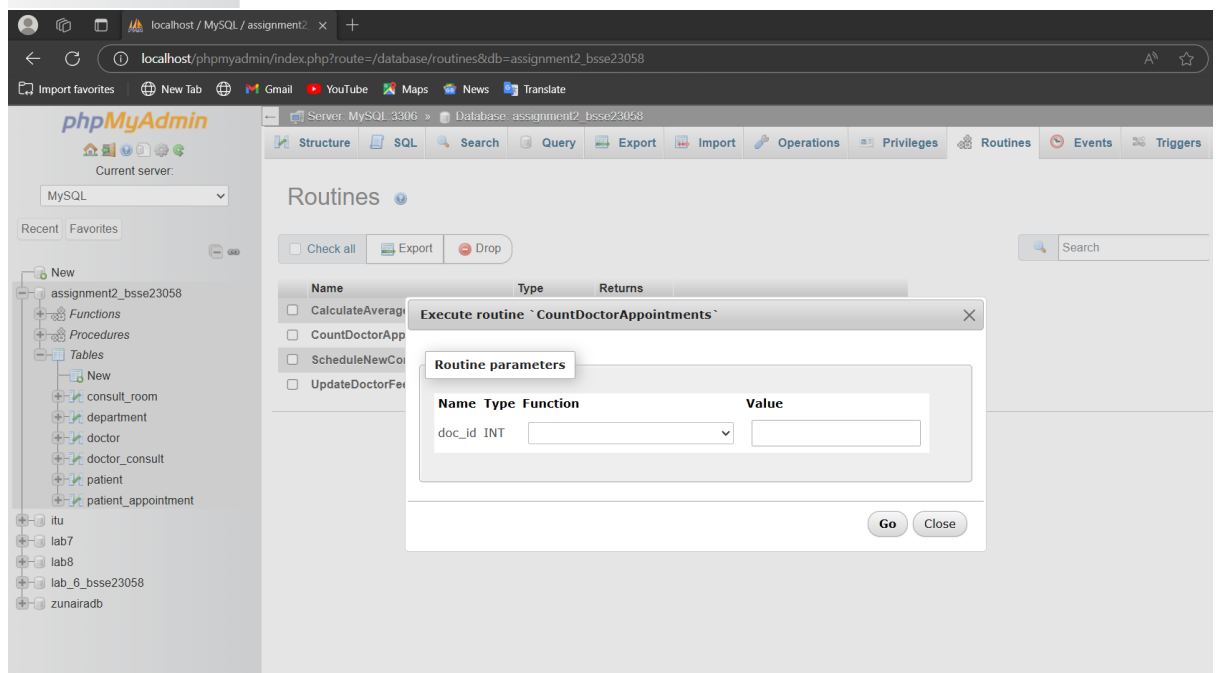
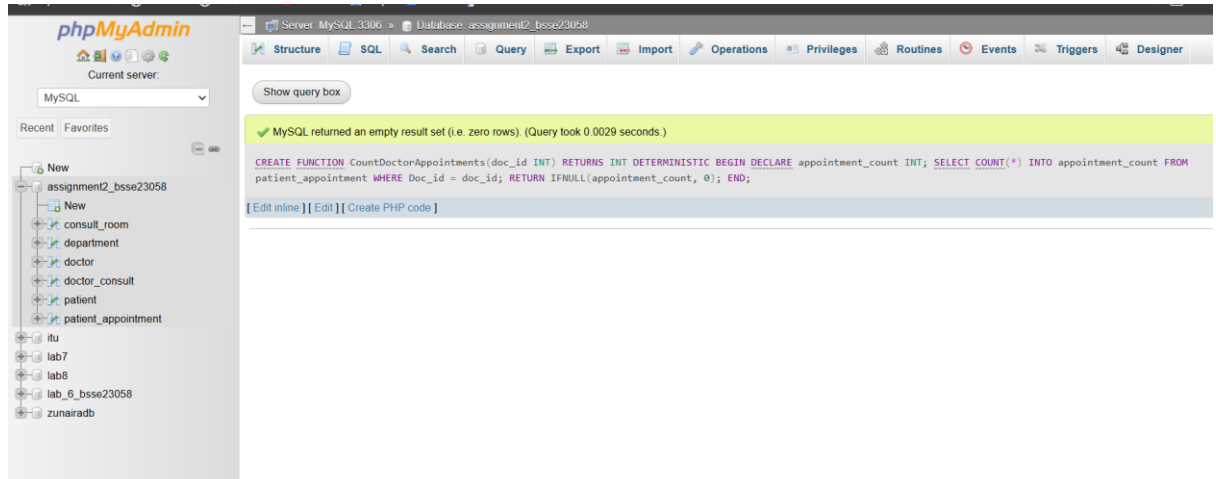
BEGIN

DECLARE appointment_count INT;

BSSE23058
ZUNAIRA ABDUL AZIZ
SECTION A

```
SELECT COUNT(*) INTO appointment_count
FROM patient_appointment
WHERE Doc_id = doc_id;

RETURN IFNULL(appointment_count, 0);
END //
DELIMITER ;
```



QUESTION 3

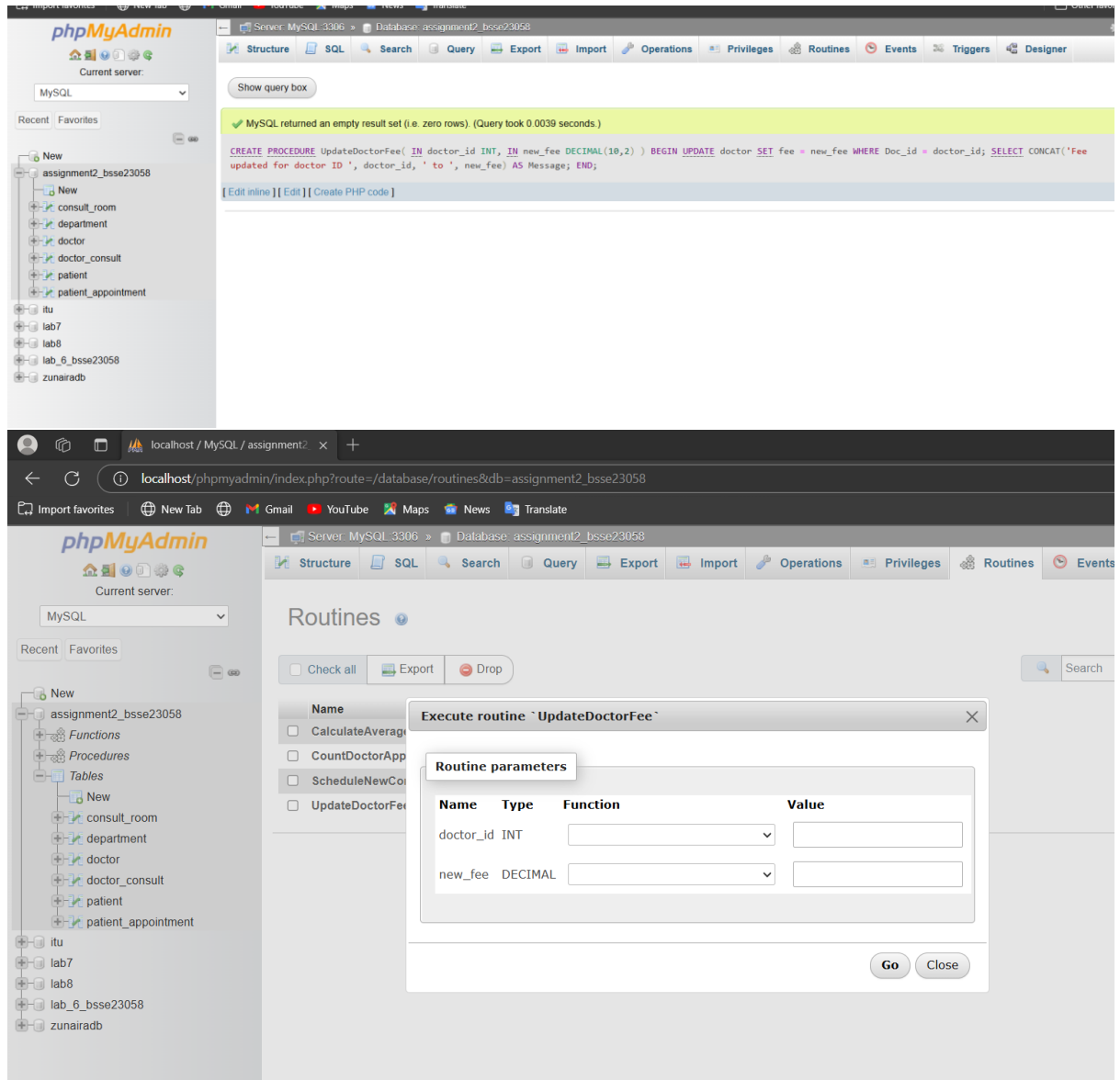
3.

```
DELIMITER //
CREATE PROCEDURE UpdateDoctorFee(
    IN doctor_id INT,
    IN new_fee DECIMAL(10,2)
)
BEGIN
    UPDATE doctor
```

BSSE23058
ZUNAIRA ABDUL AZIZ
SECTION A

```
SET fee = new_fee  
WHERE Doc_id = doctor_id;
```

```
SELECT CONCAT('Fee updated for doctor ID ', doctor_id, ' to ', new_fee) AS Message;  
END //  
DELIMITER ;
```



4.

```
DELIMITER //  
CREATE PROCEDURE ScheduleNewConsultation(  
    IN doctor_id INT,  
    IN patient_id INT,  
    IN app_date DATE,  
    IN start_time TIME)  
BEGIN  
    -- Check if doctor exists  
    DECLARE doctor_exists INT;
```

```
DECLARE patient_exists INT;
```

```
SELECT COUNT(*) INTO doctor_exists FROM doctor WHERE Doc_id = doctor_id;  
SELECT COUNT(*) INTO patient_exists FROM patient WHERE Patient_id = patient_id;
```

```
IF doctor_exists = 0 THEN
```

```
    SELECT 'Error: Doctor does not exist' AS Message;
```

```
ELSEIF patient_exists = 0 THEN
```

```
    SELECT 'Error: Patient does not exist' AS Message;
```

```
ELSE
```

```
    -- Insert the new appointment
```

```
    INSERT INTO patient_appointment (Doc_id, Pat_id, date, Start_time)
```

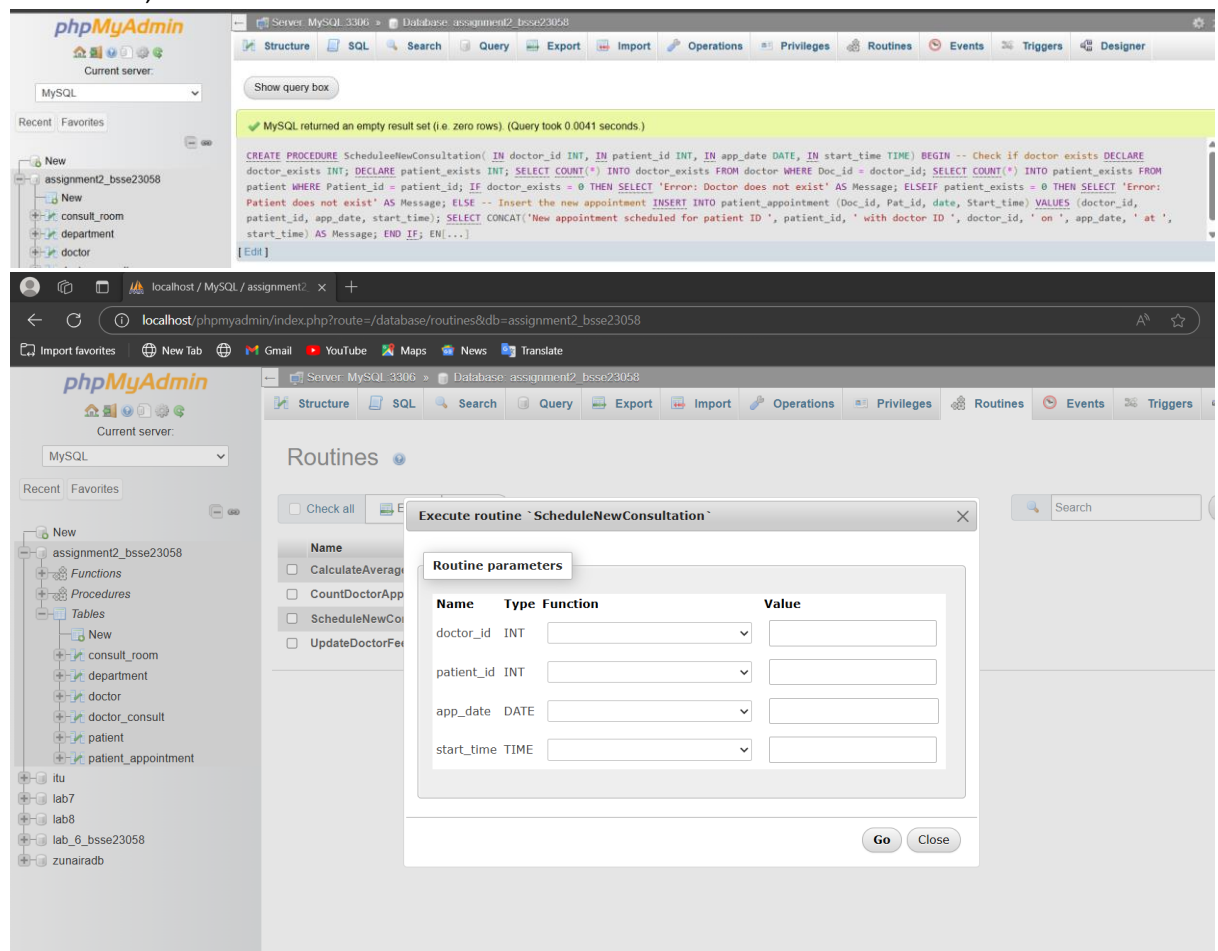
```
    VALUES (doctor_id, patient_id, app_date, start_time);
```

```
    SELECT CONCAT('New appointment scheduled for patient ID ', patient_id,  
        ' with doctor ID ', doctor_id, ' on ', app_date, ' at ', start_time) AS Message;
```

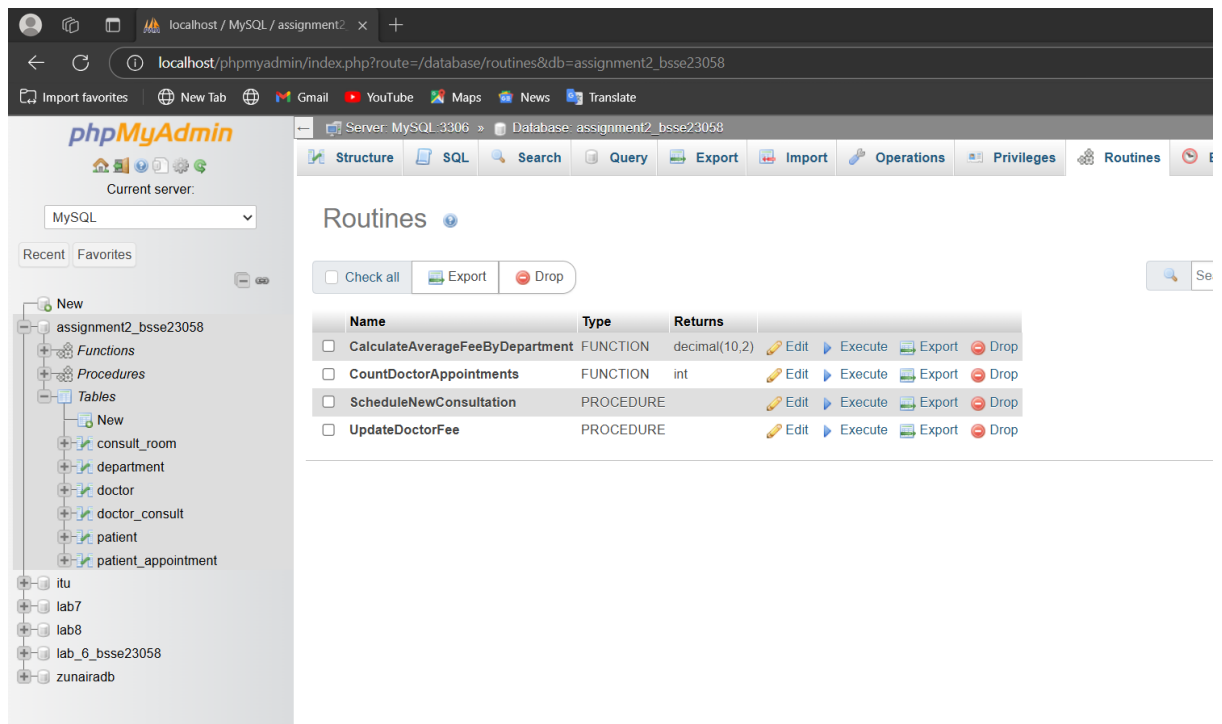
```
END IF;
```

```
END //
```

```
DELIMITER ;
```



BSSE23058
ZUNAIRA ABDUL AZIZ
SECTION A
ROUTINES



The screenshot shows the phpMyAdmin web interface. The browser address bar indicates the URL: `localhost/phpmyadmin/index.php?route=/database/routines&db=assignment2_bsse23058`. The interface is divided into a sidebar on the left and a main content area on the right.

Sidebar: The 'Current server' is set to 'MySQL'. The database tree shows a 'New' button and a list of databases: 'assignment2_bsse23058', 'itu', 'lab7', 'lab8', 'lab_6_bsse23058', and 'zunairadb'. The 'assignment2_bsse23058' database is expanded, showing 'Functions', 'Procedures', and 'Tables'.

Main Content Area: The 'Routines' page is displayed. It includes a toolbar with 'Check all', 'Export', and 'Drop' buttons. Below the toolbar is a table listing the routines in the database.

| Name | Type | Returns | |
|--|-----------|---------------|--|
| <input type="checkbox"/> CalculateAverageFeeByDepartment | FUNCTION | decimal(10,2) | Edit Execute Export Drop |
| <input type="checkbox"/> CountDoctorAppointments | FUNCTION | int | Edit Execute Export Drop |
| <input type="checkbox"/> ScheduleNewConsultation | PROCEDURE | | Edit Execute Export Drop |
| <input type="checkbox"/> UpdateDoctorFee | PROCEDURE | | Edit Execute Export Drop |