AMERICAN INTERNATIONAL UNIVERSITY BANGLADESH (AIUB)

FACULTY OF SCIENCE & TECHNOLOGY

Course Title

INTRODUCTION TO DATABASE (CSC2108)

Semester: Fall 24-25

Section: [M]

TITLE

Healthcare Management System

Supervised By

Jubayer Ahamed

Submitted By: Group no: 03

Name	ID
Tahniq Rahman Tian	23-55347-3
Yasir Arafat	23-55352-3
Md. Saad Arefin	23-55381-3
Md. Samiul Kabir Mishel	23-54058-3

TABLE OF CONTENTS

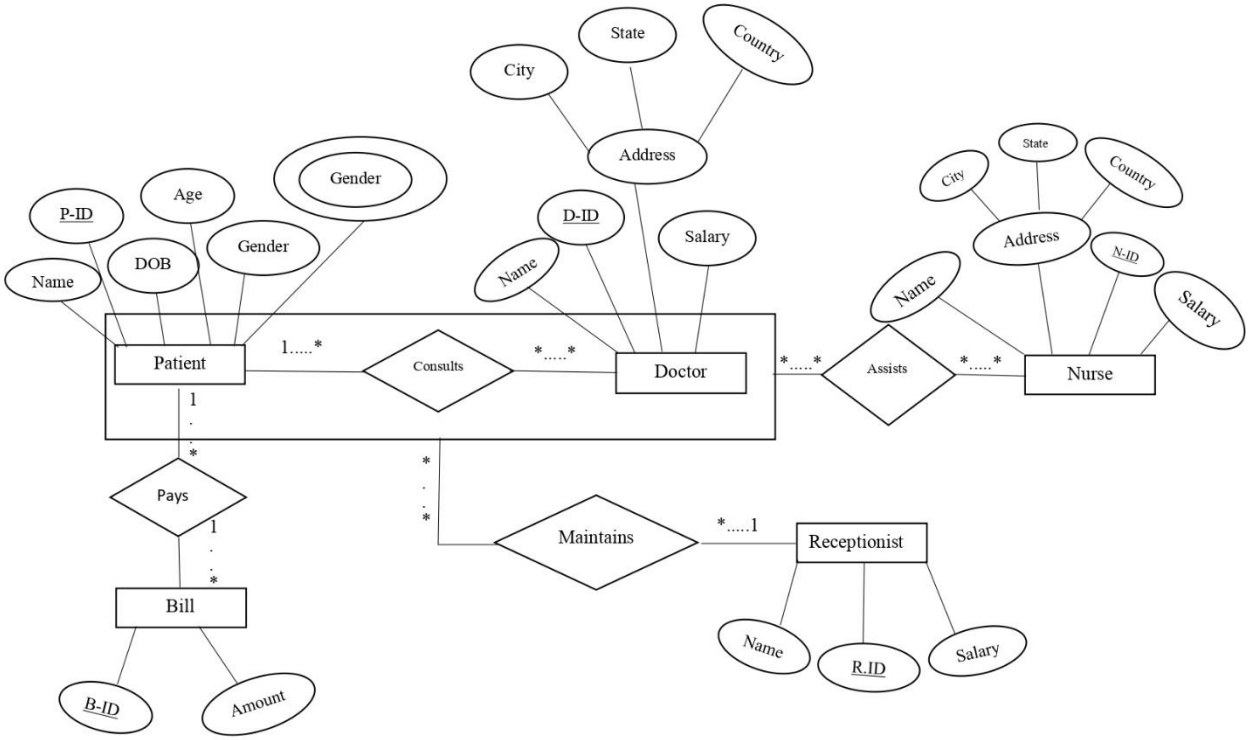
Introduction

A healthcare management system (HMS) is a software application designed to improve the management of healthcare services. It helps hospitals, clinics, and other medical institutions efficiently organize patient data, appointments, medical records, billing, and administrative tasks. By integrating various functions, an HMS ensures that healthcare providers can focus more on patient care rather than paperwork. One of the primary goals of an HMS is to provide a centralized system where all patient-related information is stored securely and can be accessed easily by authorized personnel. This reduces errors, saves time, and enhances communication between doctors, nurses, and administrative staff. Additionally, such systems support maintaining accurate medical histories, enabling better diagnosis and treatment plans. An HMS also plays a vital role in managing hospital resources, such as beds, equipment, and staff schedules. It simplifies billing processes, ensuring transparency for patients and reducing the chances of billing errors. For patients, it offers features like online appointment booking, access to medical reports, and reminders for checkups or medication, enhancing their overall experience. With advancements in technology, modern healthcare management systems now include features like telemedicine, electronic health records (EHR), and integration with wearable devices for real-time health monitoring. These innovations aim to improve the quality of care while reducing operational costs.In conclusion, a healthcare management system is an essential tool for modern healthcare providers, ensuring efficient, secure, and patient-centric care delivery. It not only streamlines daily operations but also supports the long-term growth and improvement of healthcare facilities.

Case Study / Scenario

A hospital management system is designed to manage operations involving patients, doctors, nurses, receptionists, and billing efficiently. Patients are uniquely identified by a P-ID and have attributes such as name, date of birth, age, gender, and contact number. Each patient can consult one or many doctors, and a doctor can provide consultations to multiple patients. Patients can also generate bills for the services received, where each bill is uniquely identified by a B-ID and contains the amount. Each bill is linked to only one patient. Doctors are identified by a D-ID and have attributes such as name, address (city, state, country), and salary. They provide medical consultations to patients. Nurses, identified by a N-ID, assist doctors and oversee patient care. Each nurse can assist multiple patients, and a patient can be assisted by multiple nurses. Receptionists are identified by a *R-ID and are responsible for maintaining patient records. They have attributes such as name and salary. A receptionist can manage records for multiple patients, and a patient's records can be maintained by multiple receptionists. This system ensures efficient management of consultations, billing, and patient records, while maintaining clear relationships between all entities.

ER Diagram


State
Country
City
Address
Gender
Age
P-ID
Gender
DOB
Name
D-ID
Salary
Name
City
State
Country
Address
N-ID
Name
Salary
Patient
1.....*
Consults
.....
Doctor
.....
Assists
.....
Nurse
1
.
.
*
Pays
1
.
.
*
Bill
B-ID
Amount
*
.
.
*
Maintains
*.....1
Receptionist
Name
R.ID
Salary

Normalization

Show the process of normalization up to 3NF for all the relations in the ER diagram

<table>
<tr>
<td>

CONSULTS

UNF (Unnormalized Form):
P-ID, DOB, Age, Name, Gender, C Number, Name, D-ID, Address, City, State, Country, Salary

1NF (First Normal Form):
P-ID, DOB, Age, Name, Gender, C Number, D-ID, City, State, Country, Salary

2NF (Second Normal Form):

1. P-ID (PK), DOB, Age, Name, Gender, C Number
2. D-ID (PK), City, State, Country, Salary

3NF (Third Normal Form):

1. P-ID (PK), DOB, Age, Name, Gender, C Number, D-ID (FK)
2. D-ID (PK), City, Salary
3. City, State, Country

</td>
<td>

MAINTENANCE

UNF (Unnormalized Form):
P-ID, DOB, Age, Name, Gender, C Number, Name, , D-ID, Address, City,State ,Country ,Salary ,R-ID, Salary

1NF (First Normal Form):
P-ID, DOB, Age, Gender, C Number, Name, D-ID, City,State ,Country ,Salary, Name, R-ID, Salary

2NF (Second Normal Form):

1. P-ID (PK), DOB, Age, Gender, C Number
2. D-ID (PK), Name, City,State,Country ,Salary
3. R-ID (PK), Name Salary

3NF (Third Normal Form):

1. P-ID (FK), DOB, Age, Gender, C Number,R-ID(PK)
2. D-ID (FK), Name, City,Salary
3. R-ID (PK), Name Salary

City,State,Country

</td>
</tr>
<tr>
<td>

PAYS

UNF (Unnormalized Form):
P-ID, DOB, Age, Name, Gender, C Number, B-ID, Amount

1NF (First Normal Form):
P-ID, DOB, Age, Name, Gender, C Number, B-ID, Amount

</td>
<td>

ASSISTS

UNF (Unnormalized Form):
P-ID, DOB, Age, Name, Gender, C Number, Name, D-ID, Address, City, State, Country, Salary, Name, Address, City, State, Country, NID, Salary

1NF (First Normal Form):
P-ID, DOB, Age, Name, Gender, C Number, D-ID, City, State, Country, Salary, NID

</td>
</tr>
</table>

2NF (Second Normal Form): 1. P-ID(PK), DOB, Age, Name, Gender, C Number 2. B-ID(PK), Amount **3NF (Third Normal Form):** 1. P-ID(FK), DOB, Age, Name, Gender, C Number,B-ID(PK) B-ID(PK), Amount	**2NF (Second Normal Form):** 1.P-ID(PK), DOB, Age, Name, Gender, C Number, 2. D-ID(PK), City, State, Country, Salary 3. NID(PK) **3NF (Third Normal Form):** 1.P-ID(FK), DOB, Age, Name, Gender, C Number, N-ID(PK) 2. D-ID(FK), City,Salary, N-ID(PK) 3. NID(PK) 4.City,State ,Country

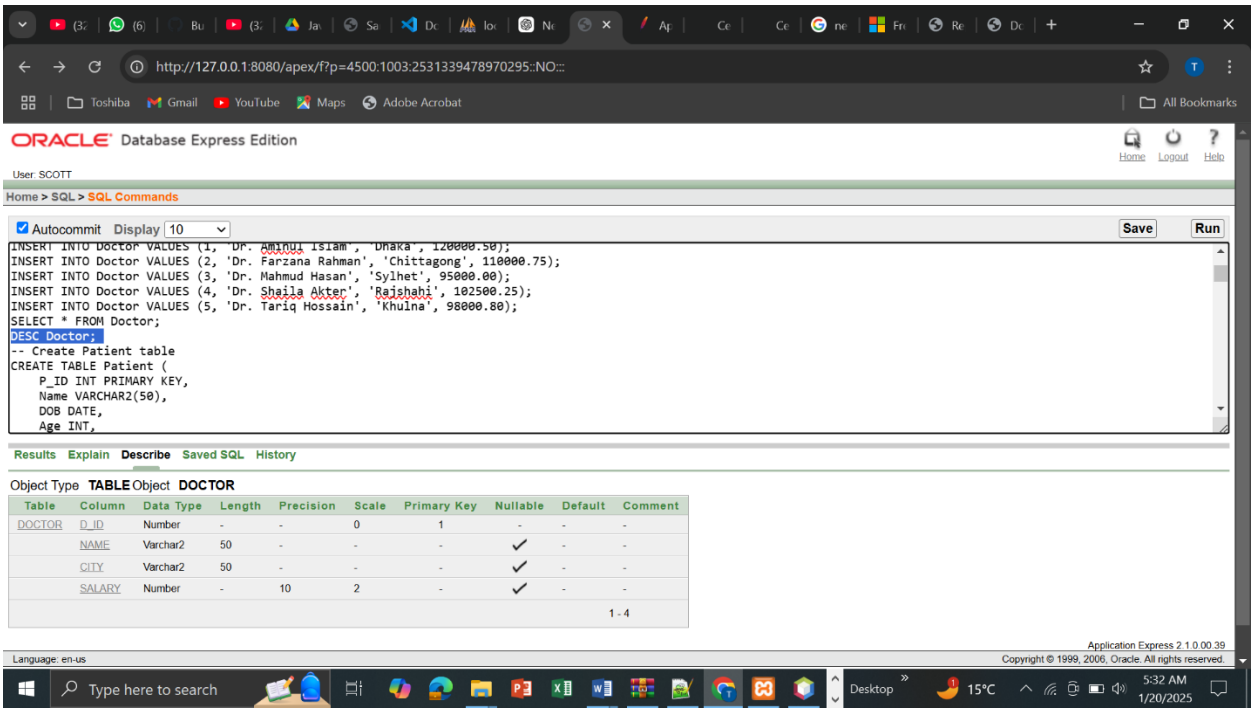
Finalization

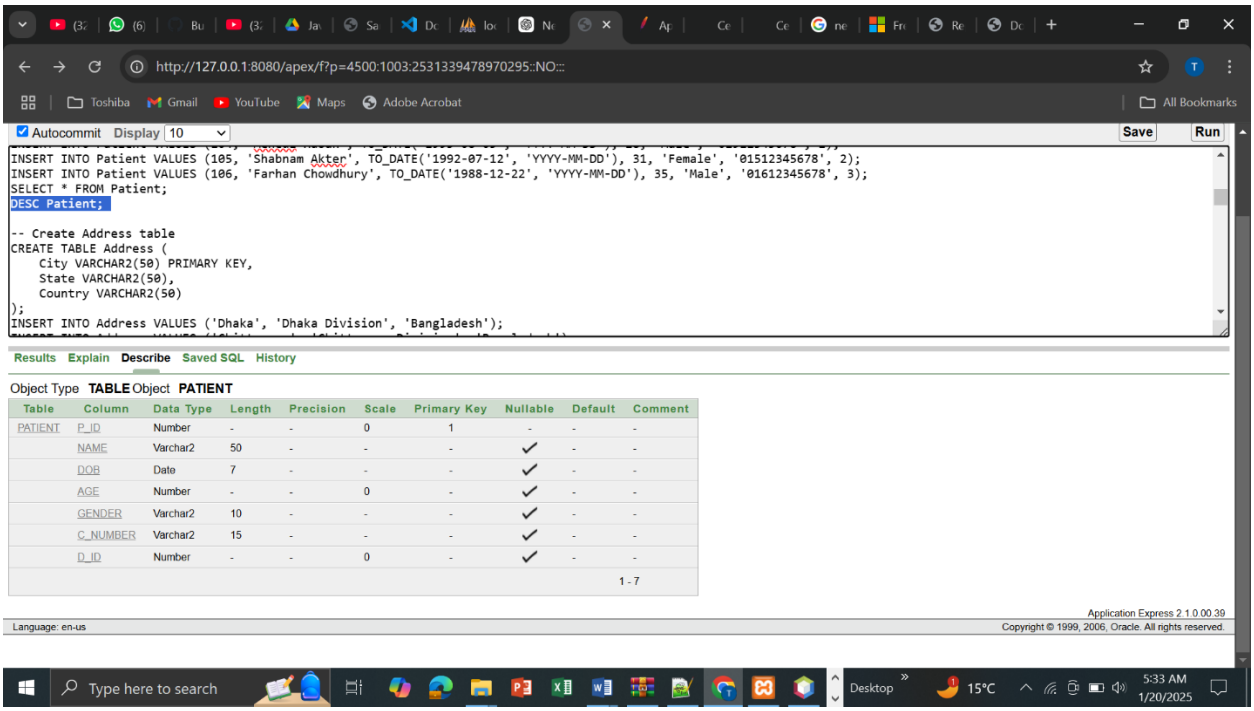
Finalization:

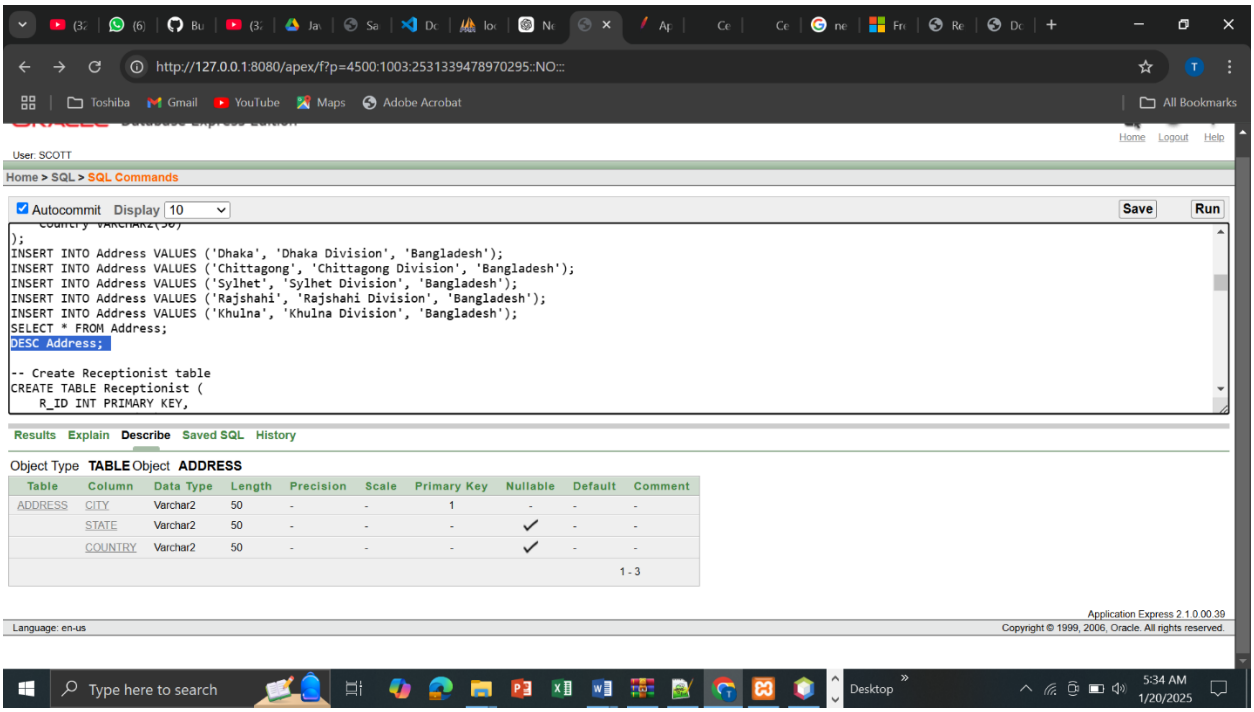
1. P-ID (PK), DOB, Age, Name, Gender, C Number, D-ID (FK)
2. D-ID (PK), City, Salary
3. City, State, Country
4. P-ID (FK), DOB, Age, Gender, C Number,R-ID(PK)
5. D-ID (FK), Name, City,Salary
6. R-ID (PK), Name Salary
7. P-ID(FK), DOB, Age, Name, Gender, C Number,B-ID(PK)
8. B-ID(PK), Amount
9. P-ID(FK), DOB, Age, Name, Gender, C Number, N-ID(PK)
10. D-ID(FK), City, Salary, N-ID(PK)

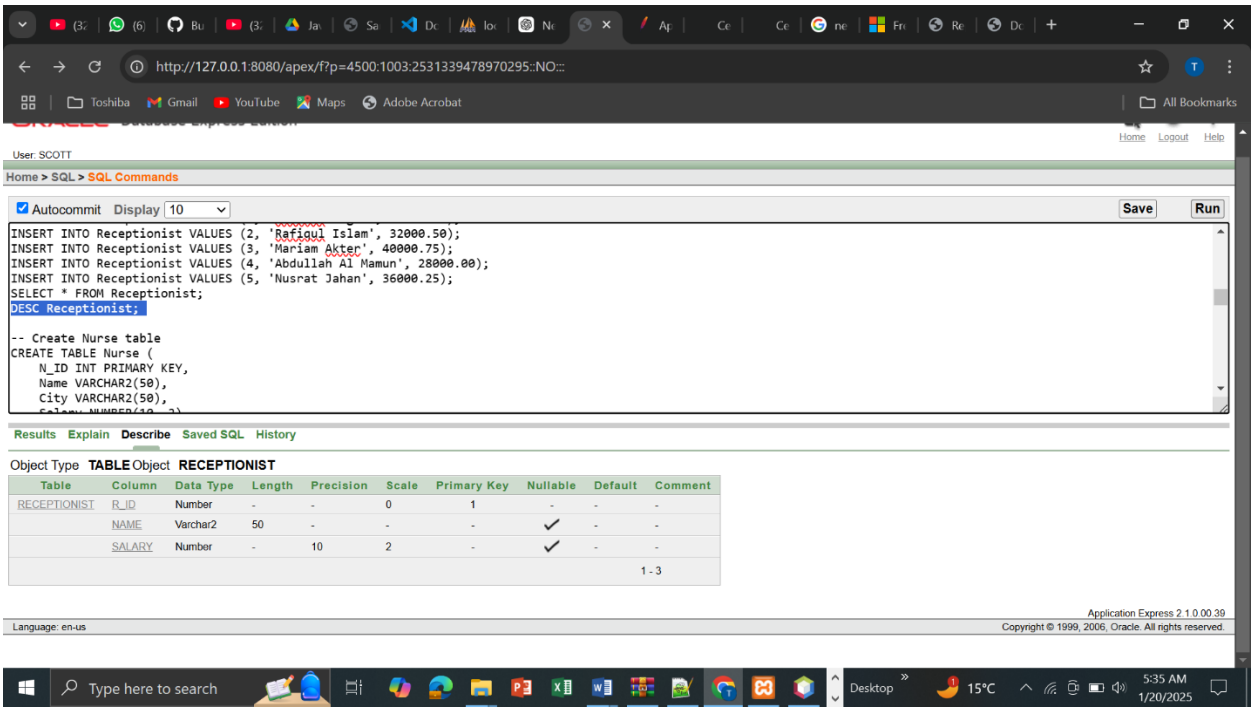
Table Creation (DDL Operations)

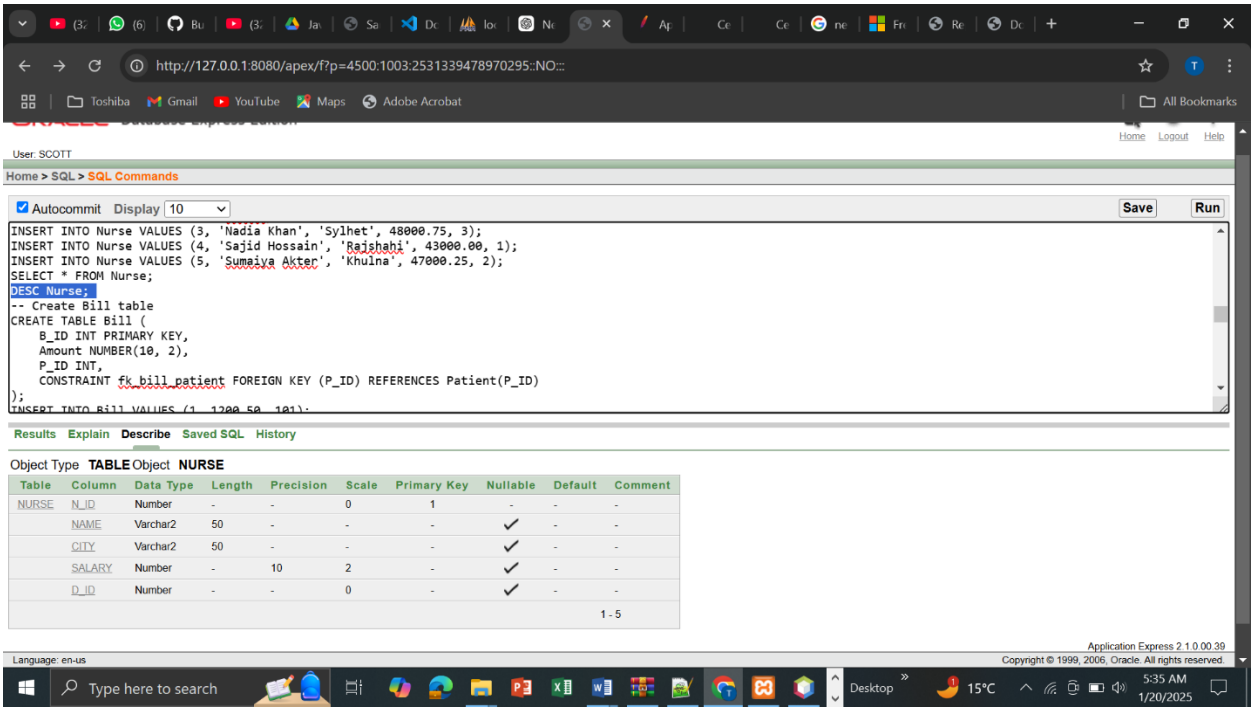
StudentID1: 23-55347-3 Name: Tahniq Rahman Tian	StudentID3: 23-55381-3 Name: Md. Saad Arefin
StudentID2: 23-55352-3 Name: Yasir Arafat	StudentID4: 23-54580-3 Name: Md. Samiul Kabir Mishel
CO4: Creating DML, DDL using Oracle and connection with ODBC/JDBC for existing JAVA application	
PO-e-2: Use modern engineering and IT tools for prediction and modeling of complex computer science and engineering problem	Marks

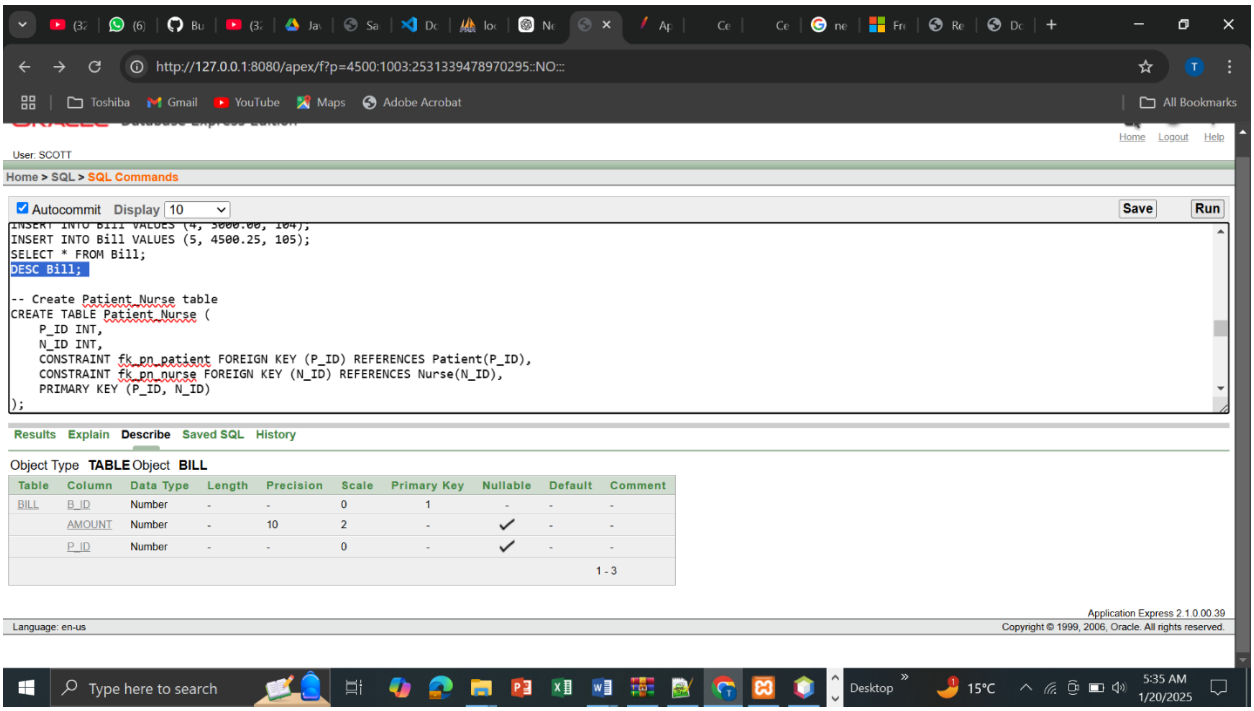


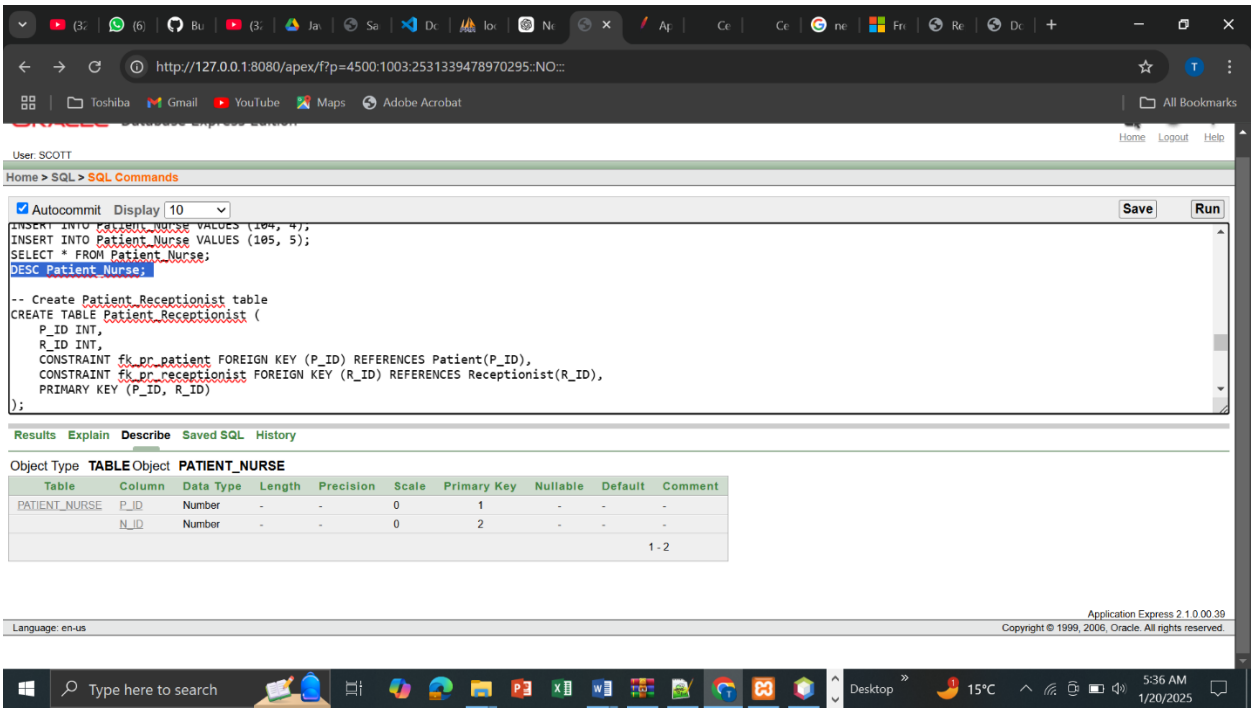

http://127.0.0.1:8080/apex/f?p=4500:1003:25313394789702 95::NO:::
Toshiba
Gmail
YouTube
Maps
Adobe Acrobat
All Bookmarks
Autocommit
Display 10
Save
Run
INSERT INTO Patient VALUES (105, 'Shabnam Akter', TO_DATE('1992-07-12', 'YYYY-MM-DD'), 31, 'Female', '01512345678', 2);
INSERT INTO Patient VALUES (106, 'Farhan Chowdhury', TO_DATE('1988-12-22', 'YYYY-MM-DD'), 35, 'Male', '01612345678', 3);
SELECT * FROM Patient;
DESC Patient;
-- Create Address table
CREATE TABLE Address (
City VARCHAR2(50) PRIMARY KEY,
State VARCHAR2(50),
Country VARCHAR2(50)
);
INSERT INTO Address VALUES ('Dhaka', 'Dhaka Division', 'Bangladesh');
Results
Explain
Describe
Saved SQL
History
Object Type TABLE Object PATIENT
Table
Column
Data Type
Length
Precision
Scale
Primary Key
Nullable
Default
Comment
PATIENT
P_ID
Number
NAME
Varchar2
50
DOB
Date
7
AGE
GENDER
C_NUMBER
D_ID
1 - 7
Application Express 2.1.0.00.39
Language: en-us
Copyright © 1999, 2006, Oracle. All rights reserved.
Type here to search
Desktop
15°C
5:33 AM
1/20/2025

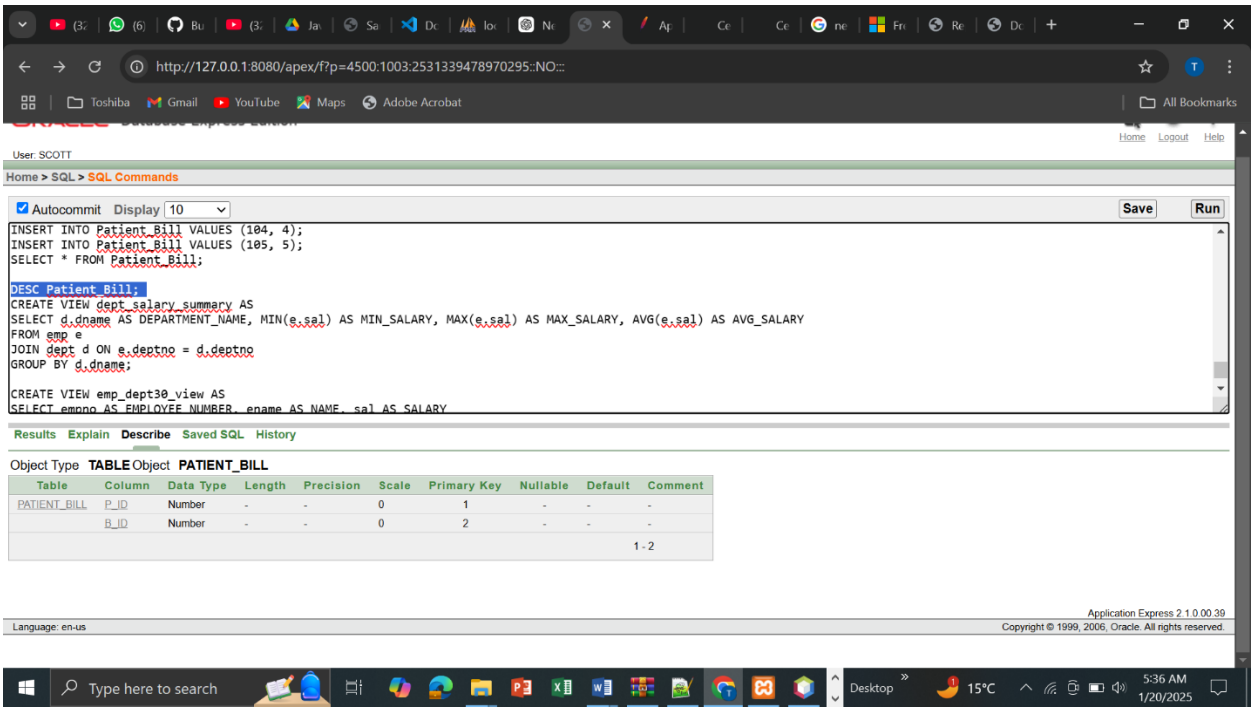

http://127.0.0.1:8080/apex/f?p=4500:1003:25313394789702 95::NO:::
Toshiba
Gmail
YouTube
Maps
Adobe Acrobat
All Bookmarks
Home
Logout
Help
User: SCOTT
Home > SQL > SQL Commands
Autocommit
Display 10
Save
Run
);
INSERT INTO Address VALUES ('Dhaka', 'Dhaka Division', 'Bangladesh');
INSERT INTO Address VALUES ('Chittagong', 'Chittagong Division', 'Bangladesh');
INSERT INTO Address VALUES ('Sylhet', 'Sylhet Division', 'Bangladesh');
INSERT INTO Address VALUES ('Rajshahi', 'Rajshahi Division', 'Bangladesh');
INSERT INTO Address VALUES ('Khulna', 'Khulna Division', 'Bangladesh');
SELECT * FROM Address;
DESC Address;
-- Create Receptionist table
CREATE TABLE Receptionist (
R_ID INT PRIMARY KEY,
Results
Explain
Describe
Saved SQL
History
Object Type TABLE Object ADDRESS
Table
Column
Data Type
Length
Precision
Scale
Primary Key
Nullable
Default
Comment
ADDRESS
CITY
Varchar2
50
STATE
COUNTRY
1 - 3
Application Express 2.1.0.00.39
Language: en-us
Copyright © 1999, 2006, Oracle. All rights reserved.
Type here to search
Desktop
5:34 AM
1/20/2025


http://127.0.0.1:8080/apex/f?p=4500:1003:25313394789702 95::NO:::
Toshiba
Gmail
YouTube
Maps
Adobe Acrobat
All Bookmarks
User: SCOTT
Home > SQL > SQL Commands
Autocommit
Display 10
Save
Run
INSERT INTO Receptionist VALUES (2, 'Rafiqul Islam', 32000.50);
INSERT INTO Receptionist VALUES (3, 'Mariam Akter', 40000.75);
INSERT INTO Receptionist VALUES (4, 'Abdullah Al Mamun', 28000.00);
INSERT INTO Receptionist VALUES (5, 'Nusrat Jahan', 36000.25);
SELECT * FROM Receptionist;
DESC Receptionist;
-- Create Nurse table
CREATE TABLE Nurse (
N_ID INT PRIMARY KEY,
Name VARCHAR2(50),
City VARCHAR2(50),
Results
Explain
Describe
Saved SQL
History
Object Type TABLE Object RECEPTIONIST
Table
Column
Data Type
Length
Precision
Scale
Primary Key
Nullable
Default
Comment
RECEPTIONIST
R_ID
Number
0
1
NAME
Varchar2
50
SALARY
Number
10
2
1 - 3
Application Express 2.1.0.00.39
Language: en-us
Copyright © 1999, 2006, Oracle. All rights reserved.
Type here to search
Desktop
15°C
5:35 AM
1/20/2025

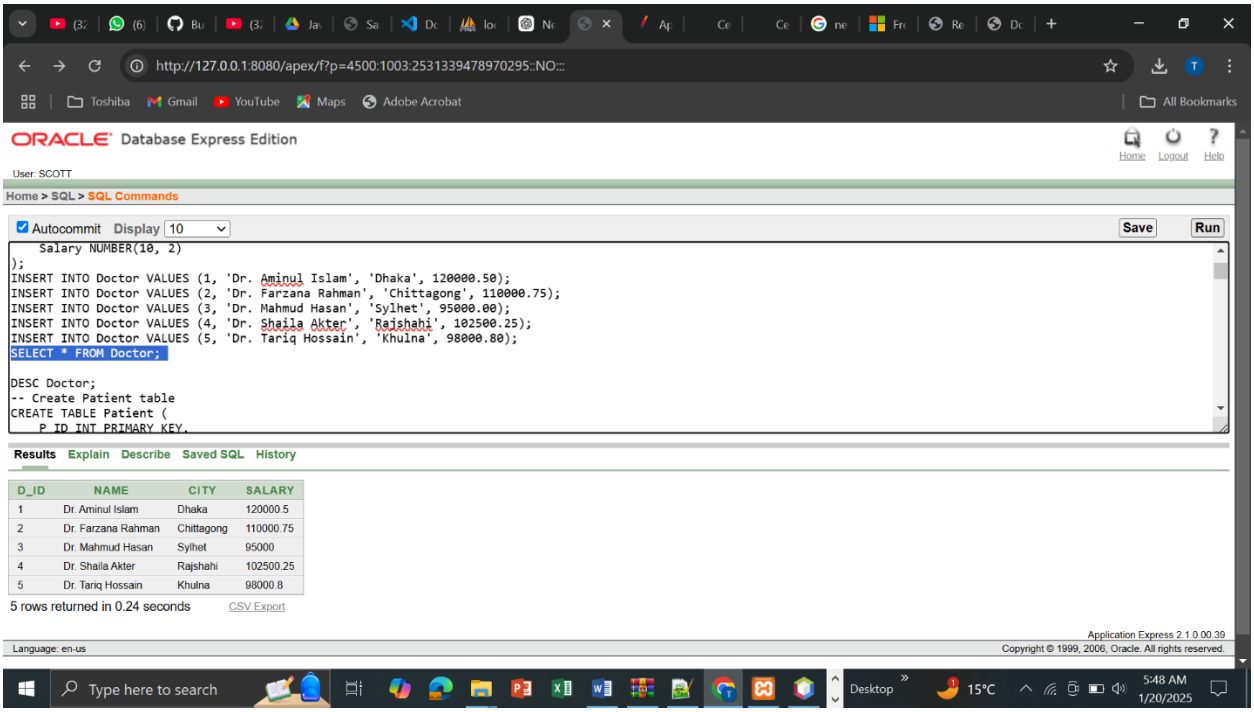

http://127.0.0.1:8080/apex/f?p=4500:1003:25313394789702 95::NO:::
Toshiba
Gmail
YouTube
Maps
Adobe Acrobat
All Bookmarks
User: SCOTT
Home > SQL > SQL Commands
Autocommit
Display 10
Save
Run
INSERT INTO Nurse VALUES (3, 'Nadia Khan', 'Sylhet', 48000.75, 3);
INSERT INTO Nurse VALUES (4, 'Sajid Hossain', 'Rajshahi', 43000.00, 1);
INSERT INTO Nurse VALUES (5, 'Sumaiya Akter', 'Khulna', 47000.25, 2);
SELECT * FROM Nurse;
DESC Nurse;
-- Create Bill table
CREATE TABLE Bill (
B_ID INT PRIMARY KEY,
Amount NUMBER(10, 2),
P_ID INT,
CONSTRAINT fk_bill_patient FOREIGN KEY (P_ID) REFERENCES Patient(P_ID)
);
Results
Explain
Describe
Saved SQL
History
Object Type TABLE Object NURSE
Table
Column
Data Type
Length
Precision
Scale
Primary Key
Nullable
Default
Comment
NURSE
N_ID
Number
0
1
NAME
Varchar2
50
CITY
Varchar2
50
SALARY
Number
10
2
D_ID
Number
0
1 - 5
Application Express 2.1.0.00.39
Language: en-us
Copyright © 1999, 2006, Oracle. All rights reserved.
Type here to search
Desktop
15°C
5:35 AM
1/20/2025

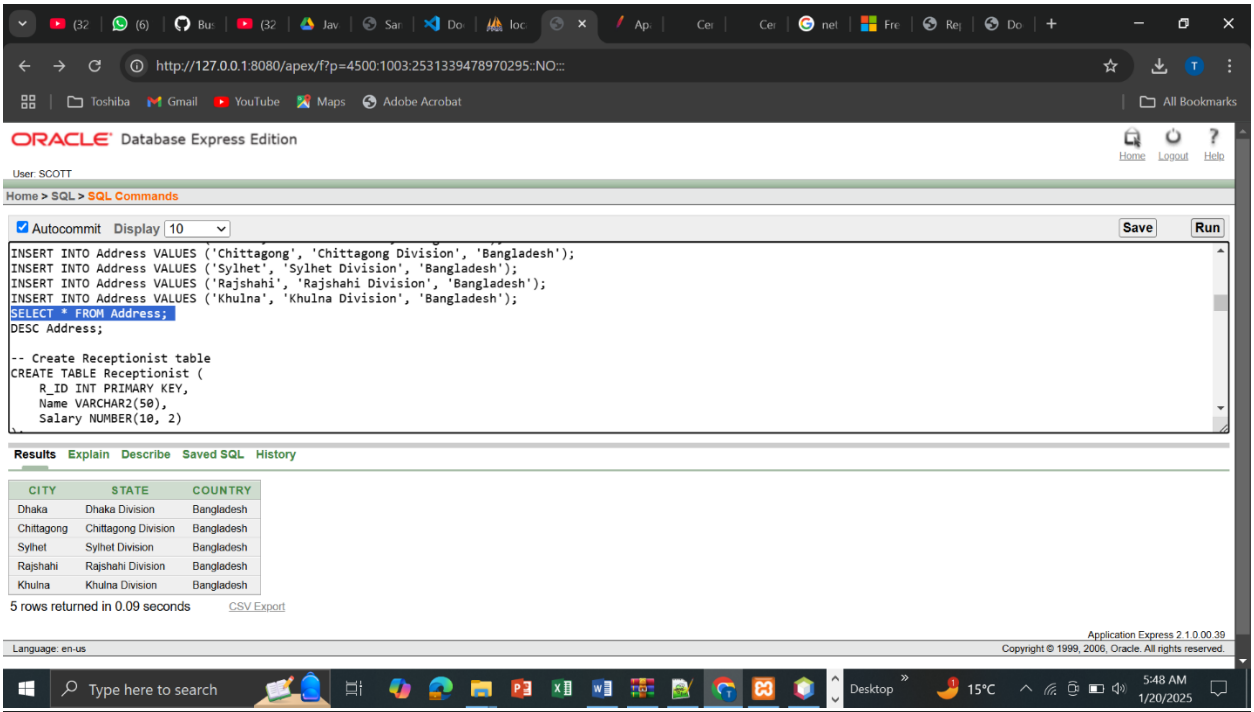

http://127.0.0.1:8080/apex/f?p=4500:1003:25313394789702 95::NO:::
Toshiba
Gmail
YouTube
Maps
Adobe Acrobat
All Bookmarks
Home
Logout
Help
User: SCOTT
Home > SQL > SQL Commands
Autocommit
Display 10
Save
Run
INSERT INTO Bill VALUES (5, 4500.25, 105);
SELECT * FROM Bill;
DESC Bill;
-- Create Patient_Nurse table
CREATE TABLE Patient_Nurse (
P_ID INT,
N_ID INT,
CONSTRAINT fk_pn_patient FOREIGN KEY (P_ID) REFERENCES Patient(P_ID),
CONSTRAINT fk_pn_nurse FOREIGN KEY (N_ID) REFERENCES Nurse(N_ID),
PRIMARY KEY (P_ID, N_ID)
);
Results
Explain
Describe
Saved SQL
History
Object Type TABLE Object BILL
Table
Column
Data Type
Length
Precision
Scale
Primary Key
Nullable
Default
Comment
BILL
B_ID
Number
0
1
AMOUNT
Number
10
2
P_ID
Number
0
1 - 3
Application Express 2.1.0.00.39
Language: en-us
Copyright © 1999, 2006, Oracle. All rights reserved.
Type here to search
Desktop
15°C
5:35 AM
1/20/2025

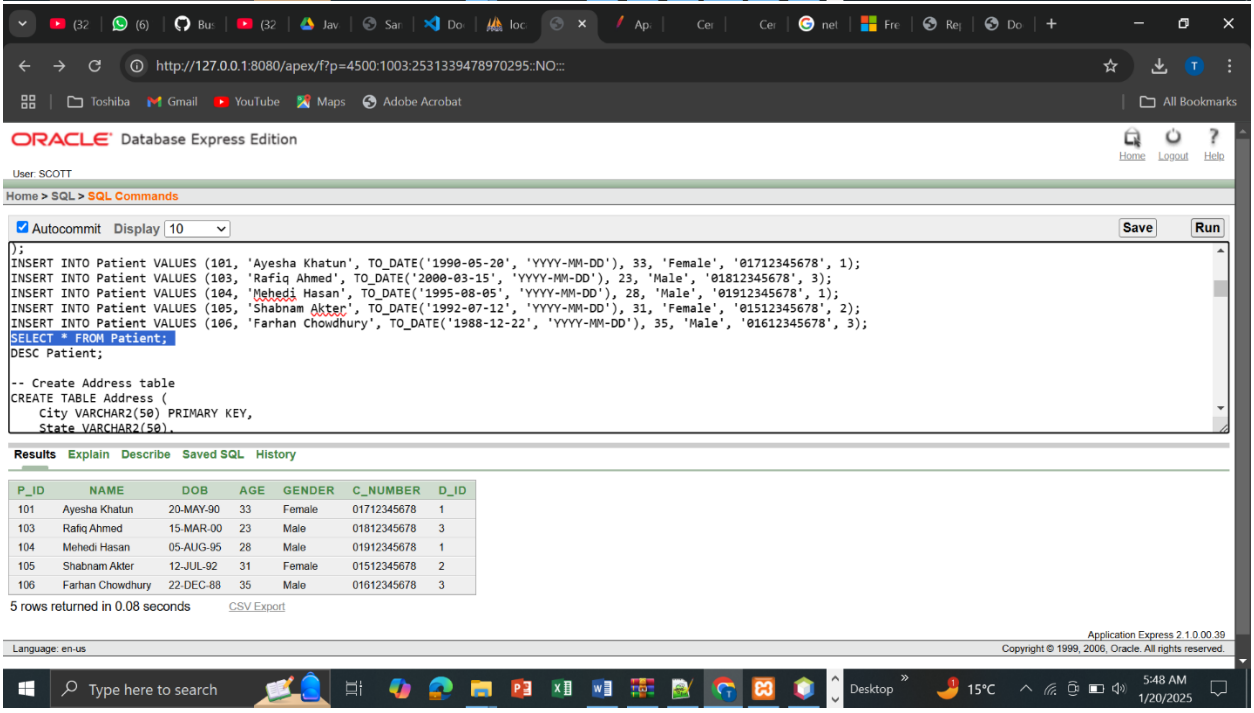

http://127.0.0.1:8080/apex/f?p=4500:1003:25313394789702 95::NO:::
Toshiba
Gmail
YouTube
Maps
Adobe Acrobat
All Bookmarks
Home
Logout
Help
User: SCOTT
Home > SQL > SQL Commands
Autocommit
Display 10
Save
Run
INSERT INTO Patient_Nurse VALUES (105, 5);
SELECT * FROM Patient_Nurse;
DESC Patient_Nurse;
-- Create Patient_Receptionist table
CREATE TABLE Patient_Receptionist (
P_ID INT,
R_ID INT,
CONSTRAINT fk_pr_patient FOREIGN KEY (P_ID) REFERENCES Patient(P_ID),
CONSTRAINT fk_pr_receptionist FOREIGN KEY (R_ID) REFERENCES Receptionist(R_ID),
PRIMARY KEY (P_ID, R_ID)
);
Results
Explain
Describe
Saved SQL
History
Object Type TABLE Object PATIENT_NURSE
Table
Column
Data Type
Length
Precision
Scale
Primary Key
Nullable
Default
Comment
PATIENT_NURSE
P_ID
Number
0
1
N_ID
Number
0
2
1 - 2
Application Express 2.1.0.00.39
Language: en-us
Copyright © 1999, 2006, Oracle. All rights reserved.
Type here to search
Desktop
15°C
5:36 AM
1/20/2025

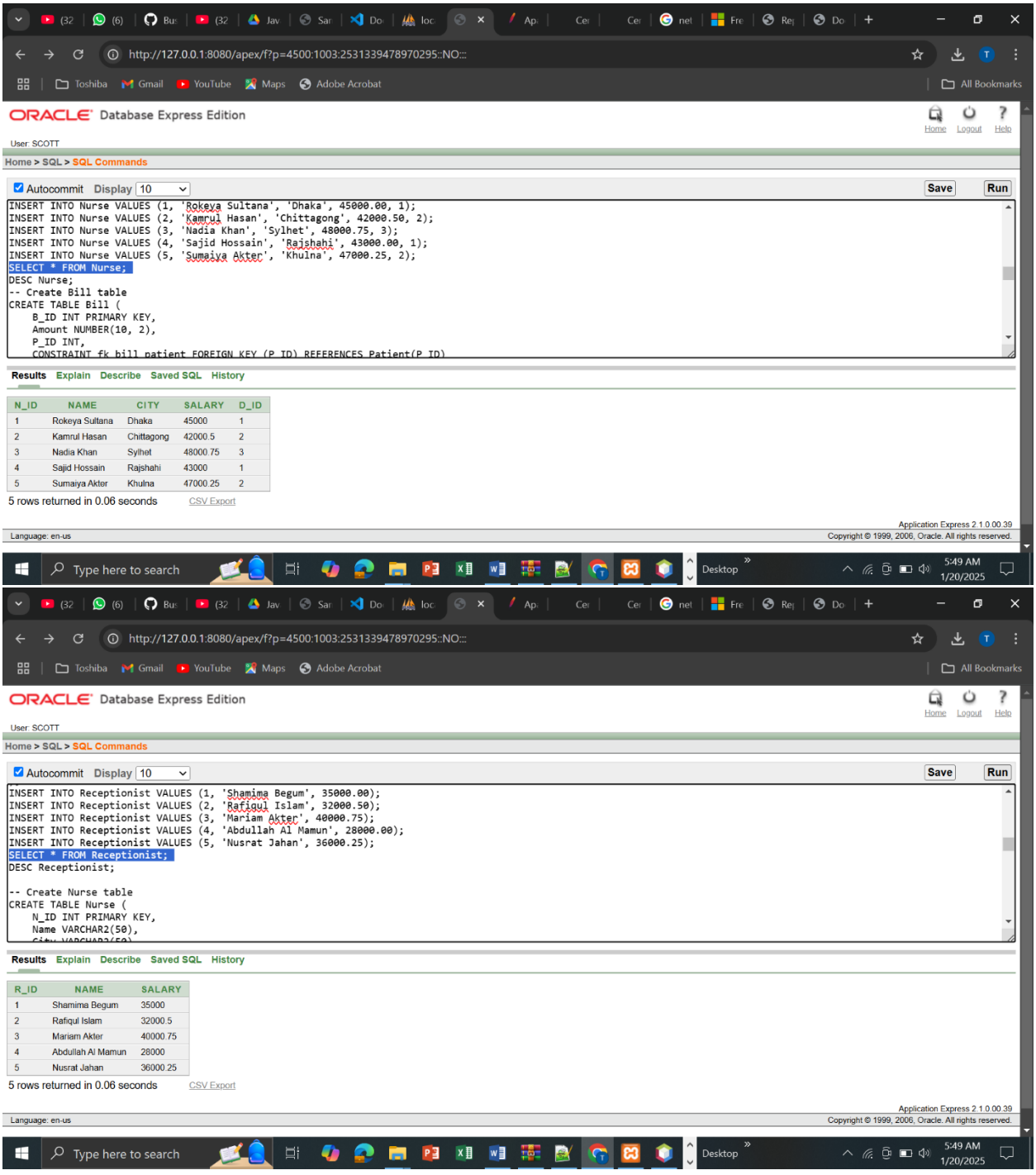


Inserted Values in the tables

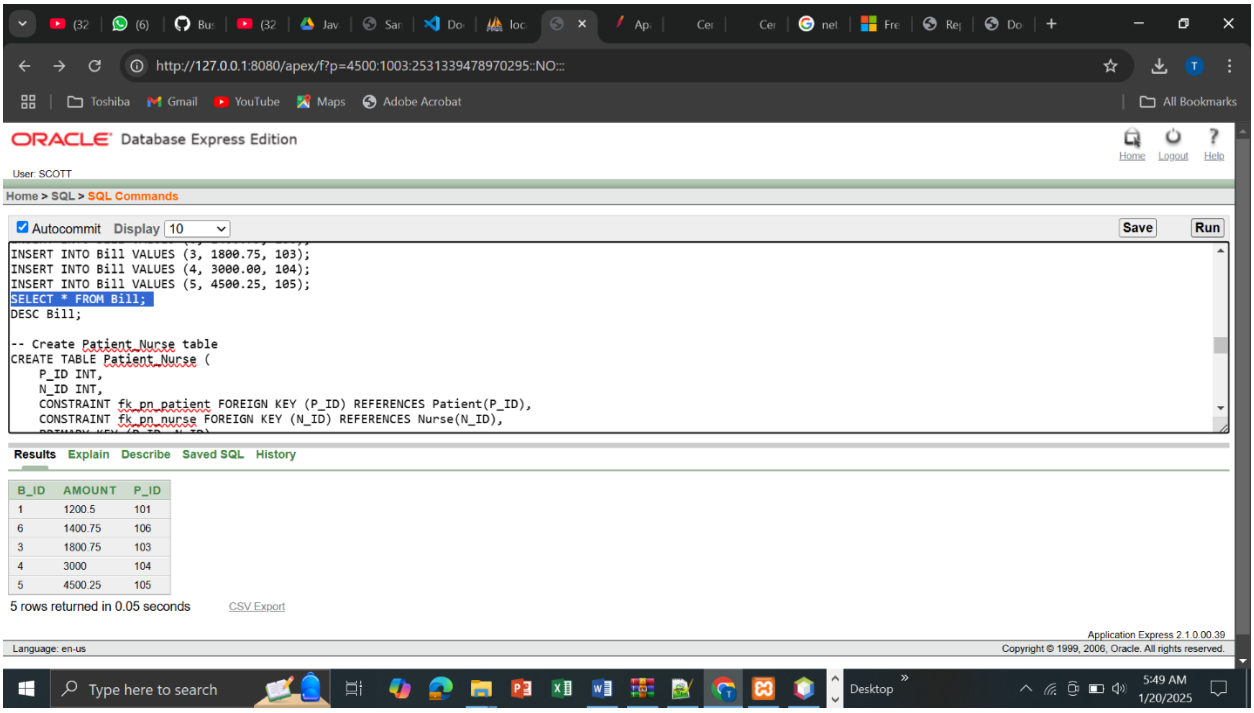


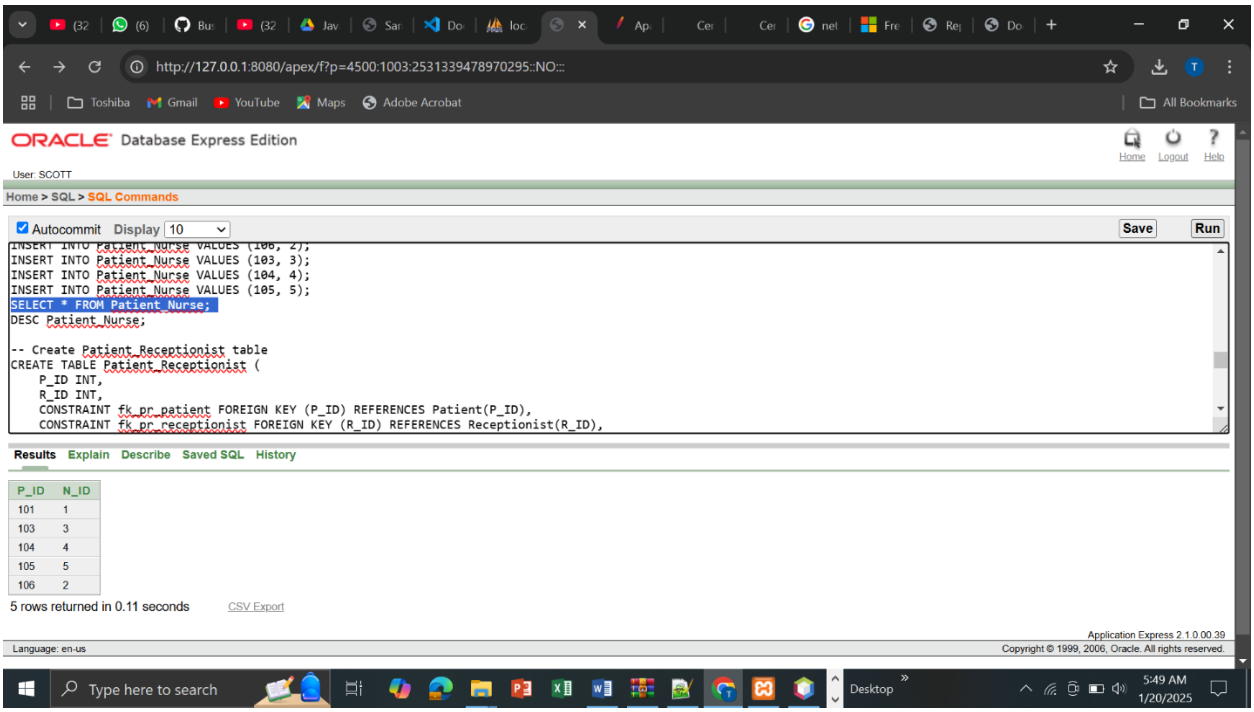

http://127.0.0.1:8080/apex/f?p=4500:1003:2531339478970295::NO:::
ORACLE Database Express Edition
User: SCOTT
Home > SQL > SQL Commands
Autocommit Display 10
Save
Run
INSERT INTO Address VALUES ('Chittagong', 'Chittagong Division', 'Bangladesh');
INSERT INTO Address VALUES ('Sylhet', 'Sylhet Division', 'Bangladesh');
INSERT INTO Address VALUES ('Rajshahi', 'Rajshahi Division', 'Bangladesh');
INSERT INTO Address VALUES ('Khulna', 'Khulna Division', 'Bangladesh');
SELECT * FROM Address;
DESC Address;
-- Create Receptionist table
CREATE TABLE Receptionist (
R_ID INT PRIMARY KEY,
Name VARCHAR2(50),
Salary NUMBER(10, 2)
Results Explain Describe Saved SQL History
CITY STATE COUNTRY
Dhaka Dhaka Division Bangladesh
Chittagong Chittagong Division Bangladesh
Sylhet Sylhet Division Bangladesh
Rajshahi Rajshahi Division Bangladesh
Khulna Khulna Division Bangladesh
5 rows returned in 0.09 seconds
CSV Export
Application Express 2.1.0.00.39
Language: en-us
Copyright © 1999, 2006, Oracle. All rights reserved.

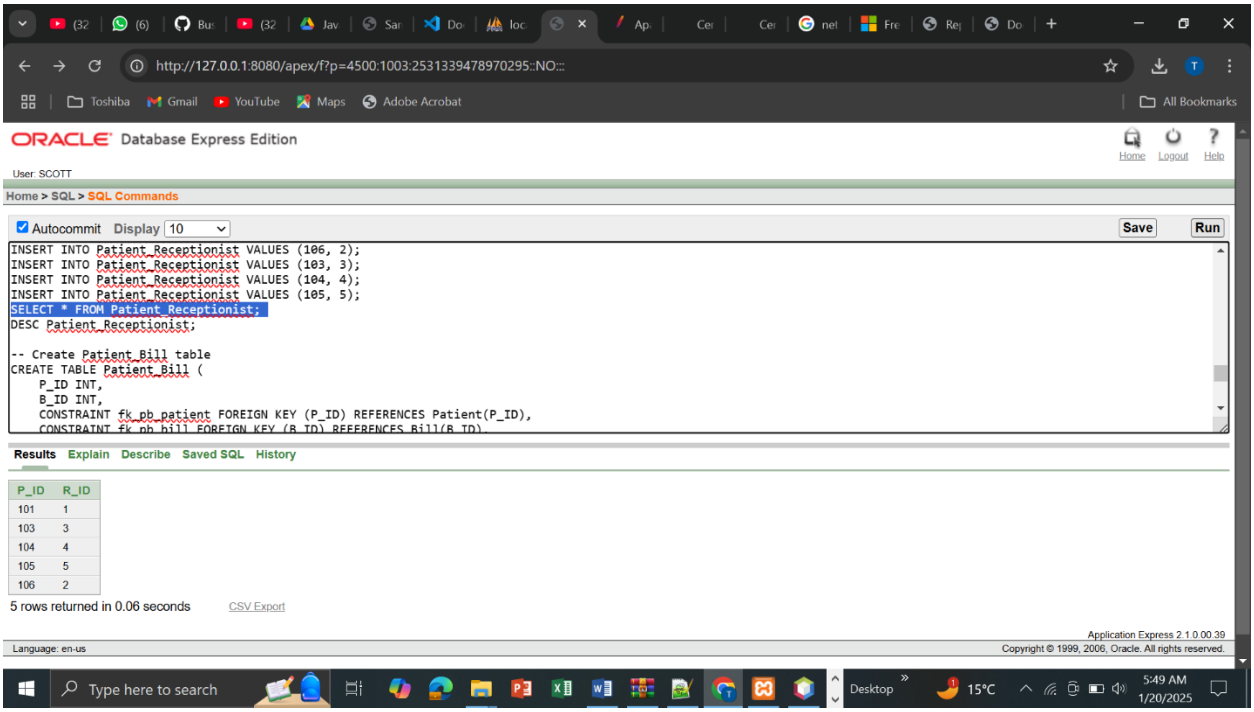

http://127.0.0.1:8080/apex/f?p=4500:1003:2531339478970295::NO:::
ORACLE Database Express Edition
User: SCOTT
Home > SQL > SQL Commands
Autocommit Display 10
Save
Run
);
INSERT INTO Patient VALUES (101, 'Ayesha Khatun', TO_DATE('1990-05-20', 'YYYY-MM-DD'), 33, 'Female', '01712345678', 1);
INSERT INTO Patient VALUES (103, 'Rafiq Ahmed', TO_DATE('2000-03-15', 'YYYY-MM-DD'), 23, 'Male', '01812345678', 3);
INSERT INTO Patient VALUES (104, 'Mehedi Hasan', TO_DATE('1995-08-05', 'YYYY-MM-DD'), 28, 'Male', '01912345678', 1);
INSERT INTO Patient VALUES (105, 'Shabnam Akter', TO_DATE('1992-07-12', 'YYYY-MM-DD'), 31, 'Female', '01512345678', 2);
INSERT INTO Patient VALUES (106, 'Farhan Chowdhury', TO_DATE('1988-12-22', 'YYYY-MM-DD'), 35, 'Male', '01612345678', 3);
SELECT * FROM Patient;
DESC Patient;
-- Create Address table
CREATE TABLE Address (
City VARCHAR2(50) PRIMARY KEY,
State VARCHAR2(50),
Results Explain Describe Saved SQL History
P_ID NAME DOB AGE GENDER C_NUMBER D_ID
101 Ayesha Khatun 20-MAY-90 33 Female 01712345678 1
103 Rafiq Ahmed 15-MAR-00 23 Male 01812345678 3
104 Mehedi Hasan 05-AUG-95 28 Male 01912345678 1
105 Shabnam Akter 12-JUL-92 31 Female 01512345678 2
106 Farhan Chowdhury 22-DEC-88 35 Male 01612345678 3
5 rows returned in 0.08 seconds
CSV Export
Application Express 2.1.0.00.39
Language: en-us
Copyright © 1999, 2006, Oracle. All rights reserved.

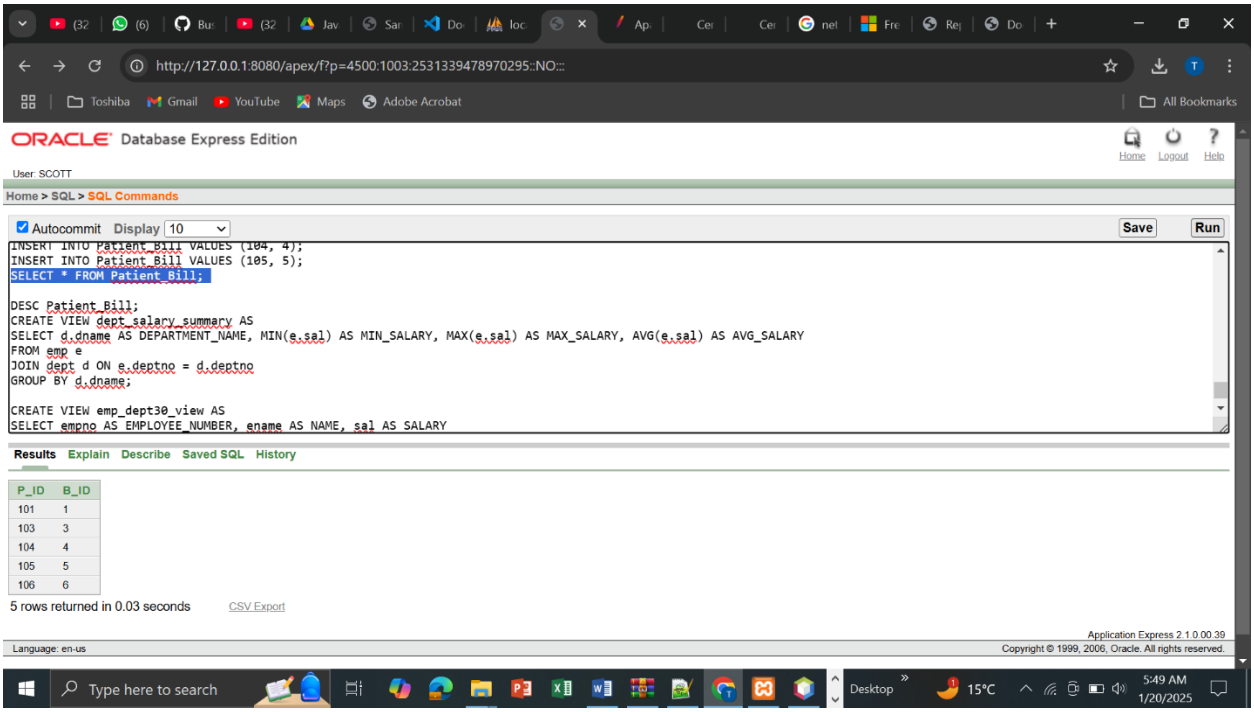

http://127.0.0.1:8080/apex/f?p=4500:1003:2531339478970295::NO:::
Toshiba
Gmail
YouTube
Maps
Adobe Acrobat
All Bookmarks
ORACLE Database Express Edition
Home
Logout
Help
User: SCOTT
Home > SQL > SQL Commands
Autocommit
Display 10
Save
Run
INSERT INTO Nurse VALUES (1, 'Rokeya Sultana', 'Dhaka', 45000.00, 1);
INSERT INTO Nurse VALUES (2, 'Kamrul Hasan', 'Chittagong', 42000.50, 2);
INSERT INTO Nurse VALUES (3, 'Nadia Khan', 'Sylhet', 48000.75, 3);
INSERT INTO Nurse VALUES (4, 'Sajid Hossain', 'Rajshahi', 43000.00, 1);
INSERT INTO Nurse VALUES (5, 'Sumaiya Akter', 'Khulna', 47000.25, 2);
SELECT * FROM Nurse;
DESC Nurse;
-- Create Bill table
CREATE TABLE Bill (
B_ID INT PRIMARY KEY,
Amount NUMBER(10, 2),
P_ID INT,
CONSTRAINT fk_bill_patient FOREIGN KEY (P_ID) REFERENCES Patient(P_ID)
Results
Explain
Describe
Saved SQL
History
N_ID NAME CITY SALARY D_ID
1 Rokeya Sultana Dhaka 45000 1
2 Kamrul Hasan Chittagong 42000.5 2
3 Nadia Khan Sylhet 48000.75 3
4 Sajid Hossain Rajshahi 43000 1
5 Sumaiya Akter Khulna 47000.25 2
5 rows returned in 0.06 seconds
CSV Export
Application Express 2.1.0.00.39
Language: en-us
Copyright © 1999, 2006, Oracle. All rights reserved.
Type here to search
Desktop
5:49 AM
1/20/2025

http://127.0.0.1:8080/apex/f?p=4500:1003:2531339478970295::NO:::
Toshiba
Gmail
YouTube
Maps
Adobe Acrobat
All Bookmarks
ORACLE Database Express Edition
Home
Logout
Help
User: SCOTT
Home > SQL > SQL Commands
Autocommit
Display 10
Save
Run
INSERT INTO Receptionist VALUES (1, 'Shamima Begum', 35000.00);
INSERT INTO Receptionist VALUES (2, 'Rafiqul Islam', 32000.50);
INSERT INTO Receptionist VALUES (3, 'Mariam Akter', 40000.75);
INSERT INTO Receptionist VALUES (4, 'Abdullah Al Mamun', 28000.00);
INSERT INTO Receptionist VALUES (5, 'Nusrat Jahan', 36000.25);
SELECT * FROM Receptionist;
DESC Receptionist;
-- Create Nurse table
CREATE TABLE Nurse (
N_ID INT PRIMARY KEY,
Name VARCHAR2(50),
Results
Explain
Describe
Saved SQL
History
R_ID NAME SALARY
1 Shamima Begum 35000
2 Rafiqul Islam 32000.5
3 Mariam Akter 40000.75
4 Abdullah Al Mamun 28000
5 Nusrat Jahan 36000.25
5 rows returned in 0.06 seconds
CSV Export
Application Express 2.1.0.00.39
Language: en-us
Copyright © 1999, 2006, Oracle. All rights reserved.
Type here to search
Desktop
5:49 AM
1/20/2025


http://127.0.0.1:8080/apex/f?p=4500:1003:25313394789702 95::NO:::
Toshiba
Gmail
YouTube
Maps
Adobe Acrobat
All Bookmarks
ORACLE Database Express Edition
Home
Logout
Help
User: SCOTT
Home > SQL > SQL Commands
Autocommit
Display 10
Save
Run
INSERT INTO Bill VALUES (3, 1800.75, 103);
INSERT INTO Bill VALUES (4, 3000.00, 104);
INSERT INTO Bill VALUES (5, 4500.25, 105);
SELECT * FROM Bill;
DESC Bill;
-- Create Patient_Nurse table
CREATE TABLE Patient_Nurse (
P_ID INT,
N_ID INT,
CONSTRAINT fk_pn_patient FOREIGN KEY (P_ID) REFERENCES Patient(P_ID),
CONSTRAINT fk_pn_nurse FOREIGN KEY (N_ID) REFERENCES Nurse(N_ID),
Results
Explain
Describe
Saved SQL
History
B_ID AMOUNT P_ID
1 1200.5 101
6 1400.75 106
3 1800.75 103
4 3000 104
5 4500.25 105
5 rows returned in 0.05 seconds
CSV Export
Application Express 2.1.0.00.39
Language: en-us
Copyright © 1999, 2006, Oracle. All rights reserved.
Type here to search
Desktop
5:49 AM
1/20/2025


http://127.0.0.1:8080/apex/f?p=4500:1003:25313394789702 95::NO:::
Toshiba
Gmail
YouTube
Maps
Adobe Acrobat
All Bookmarks
ORACLE Database Express Edition
Home
Logout
Help
User: SCOTT
Home > SQL > SQL Commands
Autocommit
Display 10
Save
Run
INSERT INTO Patient_Nurse VALUES (103, 3);
INSERT INTO Patient_Nurse VALUES (104, 4);
INSERT INTO Patient_Nurse VALUES (105, 5);
SELECT * FROM Patient_Nurse;
DESC Patient_Nurse;
-- Create Patient_Receptionist table
CREATE TABLE Patient_Receptionist (
P_ID INT,
R_ID INT,
CONSTRAINT fk_pr_patient FOREIGN KEY (P_ID) REFERENCES Patient(P_ID),
CONSTRAINT fk_pr_receptionist FOREIGN KEY (R_ID) REFERENCES Receptionist(R_ID),
Results
Explain
Describe
Saved SQL
History
P_ID N_ID
101 1
103 3
104 4
105 5
106 2
5 rows returned in 0.11 seconds
CSV Export
Application Express 2.1.0.00.39
Language: en-us
Copyright © 1999, 2006, Oracle. All rights reserved.
Type here to search
Desktop
5:49 AM
1/20/2025


http://127.0.0.1:8080/apex/f?p=4500:1003:25313394789702 95::NO:::
Toshiba
Gmail
YouTube
Maps
Adobe Acrobat
All Bookmarks
ORACLE Database Express Edition
Home
Logout
Help
User: SCOTT
Home > SQL > SQL Commands
Autocommit
Display 10
Save
Run
INSERT INTO Patient_Receptionist VALUES (106, 2);
INSERT INTO Patient_Receptionist VALUES (103, 3);
INSERT INTO Patient_Receptionist VALUES (104, 4);
INSERT INTO Patient_Receptionist VALUES (105, 5);
SELECT * FROM Patient_Receptionist;
DESC Patient_Receptionist;
-- Create Patient_Bill table
CREATE TABLE Patient_Bill (
P_ID INT,
B_ID INT,
CONSTRAINT fk_pb_patient FOREIGN KEY (P_ID) REFERENCES Patient(P_ID),
Results
Explain
Describe
Saved SQL
History
P_ID R_ID
101 1
103 3
104 4
105 5
106 2
5 rows returned in 0.06 seconds
CSV Export
Application Express 2.1.0.00.39
Language: en-us
Copyright © 1999, 2006, Oracle. All rights reserved.
Type here to search
Desktop
15°C
5:49 AM
1/20/2025


http://127.0.0.1:8080/apex/f?p=4500:1003:25313394789702 95::NO:::
Toshiba
Gmail
YouTube
Maps
Adobe Acrobat
All Bookmarks
ORACLE Database Express Edition
Home
Logout
Help
User: SCOTT
Home > SQL > SQL Commands
Autocommit
Display 10
Save
Run
INSERT INTO Patient_Bill VALUES (104, 4);
INSERT INTO Patient_Bill VALUES (105, 5);
SELECT * FROM Patient_Bill;
DESC Patient_Bill;
CREATE VIEW dept_salary_summary AS
SELECT d.dname AS DEPARTMENT_NAME, MIN(e.sal) AS MIN_SALARY, MAX(e.sal) AS MAX_SALARY, AVG(e.sal) AS AVG_SALARY
FROM emp e
JOIN dept d ON e.deptno = d.deptno
GROUP BY d.dname;
CREATE VIEW emp_dept30_view AS
SELECT empno AS EMPLOYEE_NUMBER, ename AS NAME, sal AS SALARY
Results
Explain
Describe
Saved SQL
History
P_ID B_ID
101 1
103 3
104 4
105 5
106 6
5 rows returned in 0.03 seconds
CSV Export
Application Express 2.1.0.00.39
Language: en-us
Copyright © 1999, 2006, Oracle. All rights reserved.
Type here to search
Desktop
15°C
5:49 AM
1/20/2025

Query Test in DB

a) Basic Query (4 Queries)

SELECT Name, City, Salary FROM Doctor;

SELECT Name, Age FROM Patient WHERE Age > 30;

SELECT * FROM Bill WHERE Amount > 2000;

SELECT Name, Salary FROM Nurse WHERE City = 'Dhaka';

b) Two queries using the Like operator

SELECT * FROM Patient WHERE Name LIKE 'A%';

SELECT * FROM Nurse WHERE Name LIKE '%a%';

c) Two queries using the Character Manipulation Functions

SELECT UPPER(Name) AS Upper_Name FROM Receptionist;

SELECT SUBSTR(Name, 1, 3) AS Short_Name FROM Doctor;

d) Two queries using the Character Manipulation Functions

SELECT Name, DECODE(SIGN(Salary - 100000), 1, 'Above 100K', -1, 'Below 100K', 'Exactly 100K') AS Salary_Status FROM Doctor;

SELECT Amount, DECODE(SIGN(Amount - 3000), 1, 'High', 0, 'Medium', -1, 'Low') AS Bill_Category FROM Bill;

e) 3 kinds of joining (1 query for equijoin ; 1 non-equijoin; 1 outer join)

SELECT P.Name AS Patient_Name, D.Name AS Doctor_Name FROM Patient P, Doctor D WHERE P.D_ID = D.D_ID;

SELECT P.Name AS Patient_Name, B.Amount FROM Patient P, Bill B WHERE P.P_ID = B.P_ID AND B.Amount BETWEEN 1000 AND 3000;

SELECT D.Name AS Doctor_Name, N.Name AS Nurse_Name FROM Doctor D, Nurse N WHERE D.D_ID = N.D_ID(+);

CREATE VIEW dept_sum_vu 2 (name, minsal, maxsal, avgsal) 3 AS SELECT d.dname, MIN(e.sal), MAX(e.sal), 4 AVG(e.sal) 5 FROM emp e, dept d 6 WHERE e.deptno = d.deptno 7 GROUP BY d.dname;

CREATE VIEW PatientDetails AS SELECT P.Name AS Patient_Name, P.Age, P.Gender, D.Name AS Doctor_Name, N.Name AS Nurse_Name FROM Patient P JOIN Doctor D ON P.D_ID = D.D_ID JOIN Patient_Nurse PN ON P.P_ID = PN.P_ID JOIN Nurse N ON PN.N_ID = N.N_ID; CREATE VIEW salvu30 AS SELECT empno EMPLOYEE_NUMBER, ename NAME, sal SALARY FROM emp WHERE deptno = 30;

Description of a Successful DB connection

9.1 Patient Table:

Name: Yasir Arafat ID: 23-55352-3

1. Software Installation and JAR File Download:

● Installed Visual Studio Code for code editing purposes.

● Installed Xampp Server to set up a local development environment.

● Downloaded the mysql-connector-java-8.0.30.jar file to facilitate database connectivity.

2. I began by starting the Apache Web Server and MySQL Database through the XAMPP control panel. Afterward, I navigated to “http://localhost/phpmyadmin” to create a new database named "healthcare_management_system". Within this newly created database, I create a table titled "Patient" and then add columns according to our project. After that, I put some data into the table.

	P_ID	NAME	DOB	AGE	GENDER	C_NUMBER	D_ID
Edit Copy Delete	101	Ayesha Khatun	0000-00-00	33	Female	1712345678	1
Edit Copy Delete	103	Rafiq Ahmed	0000-00-00	23	Male	1812345678	3
Edit Copy Delete	104	Mehedi Hasan	0000-00-00	28	Male	1912345678	1
Edit Copy Delete	105	Shabnam Akter	0000-00-00	31	Female	1512345678	2
Edit Copy Delete	106	Farhan Chowdhury	0000-00-00	35	Male	1612345678	3

Fig 9.1.1 Create and Insert Value in Officers Table

3. After making my database and table, I chose to use VSCODE editor for my project. I added the mysql-connector-java-8.0.30.jar library to my project. Then, in the main file, I wrote the code to connect to my Xampp-made database. Finally, I fetched and printed the columns of the Officers table.

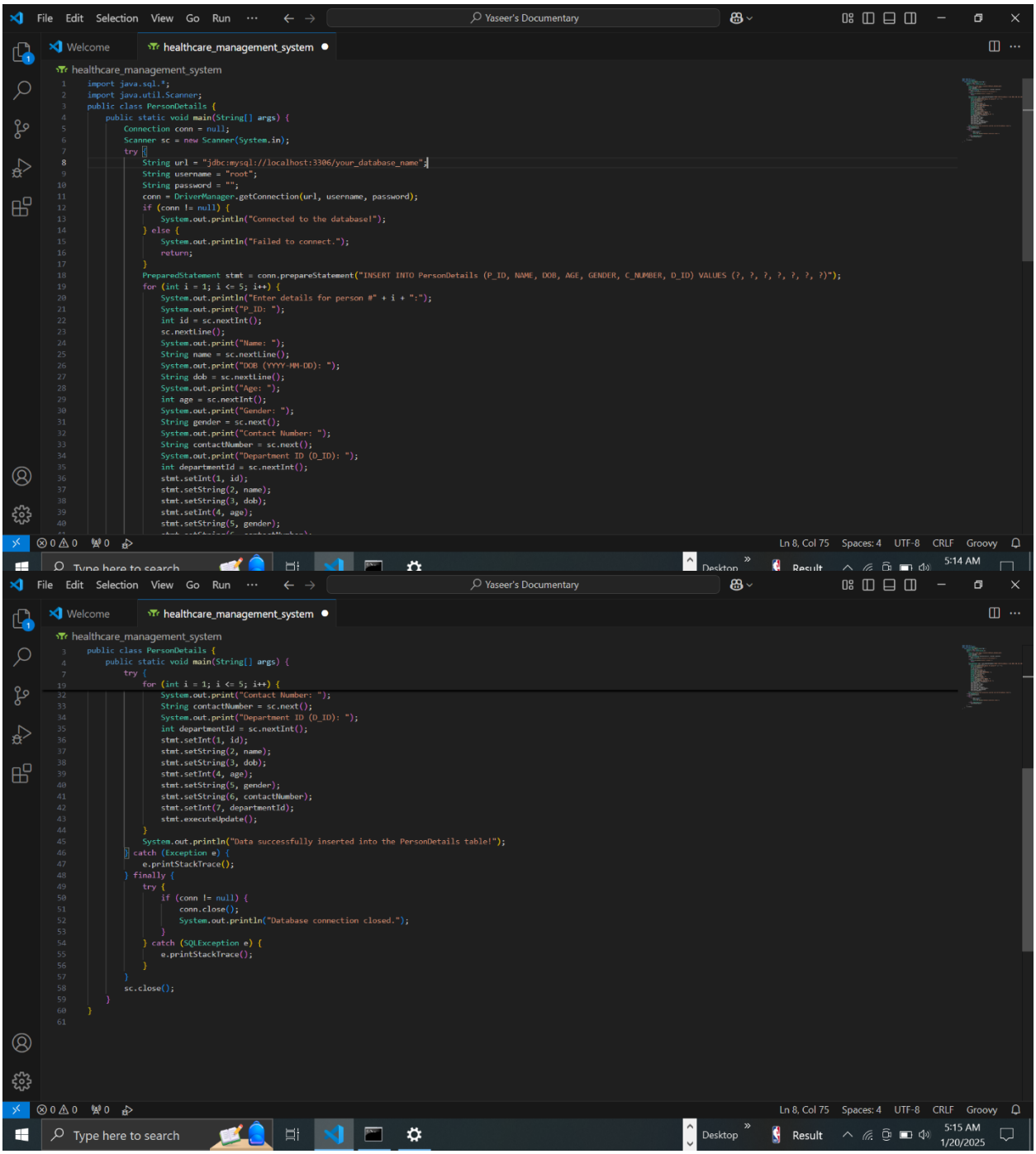


Fig 9.1.2 DB Connection Code

```
P_ID: 101,Name: Ayesha Khatun,DOB : 1991-05-15,Age: 33,Gender:Female,Contact Number: 1712345678, D_ID: 1
P_ID: 103,Name: Rafiq Ahmed DOB : 2001-02-20,Age: 23,Gender: Male,Contact Number: 1812345678,D_ID: 3
P_ID: 104,Name: Mehedi Hasan DOB : 1996-08-05,Age: 28,Gender: Male,Contact Number: 1912345678,D_ID: 1
P_ID: 105,Name: Shabnam Akter,DOB : 1992-07-22,Age: 31,Gender: Female,Contact Number: 1512345678,D_ID: 2
P_ID: 106,Name: Farhan Chowdhury,DOB : 1988-12-30,Age: 35,Gender: Male,Contact Number: 1612345678,D_ID: 3
```

Description of a Successful DB connection

9.2 Receptionist Table:

Name: Tahniq Rahman Tian ID: 23-55347-3

1. Software Installation and JAR File Download:

● Installed Visual Studio Code for code editing purposes.

● Installed Xampp Server to set up a local development environment.

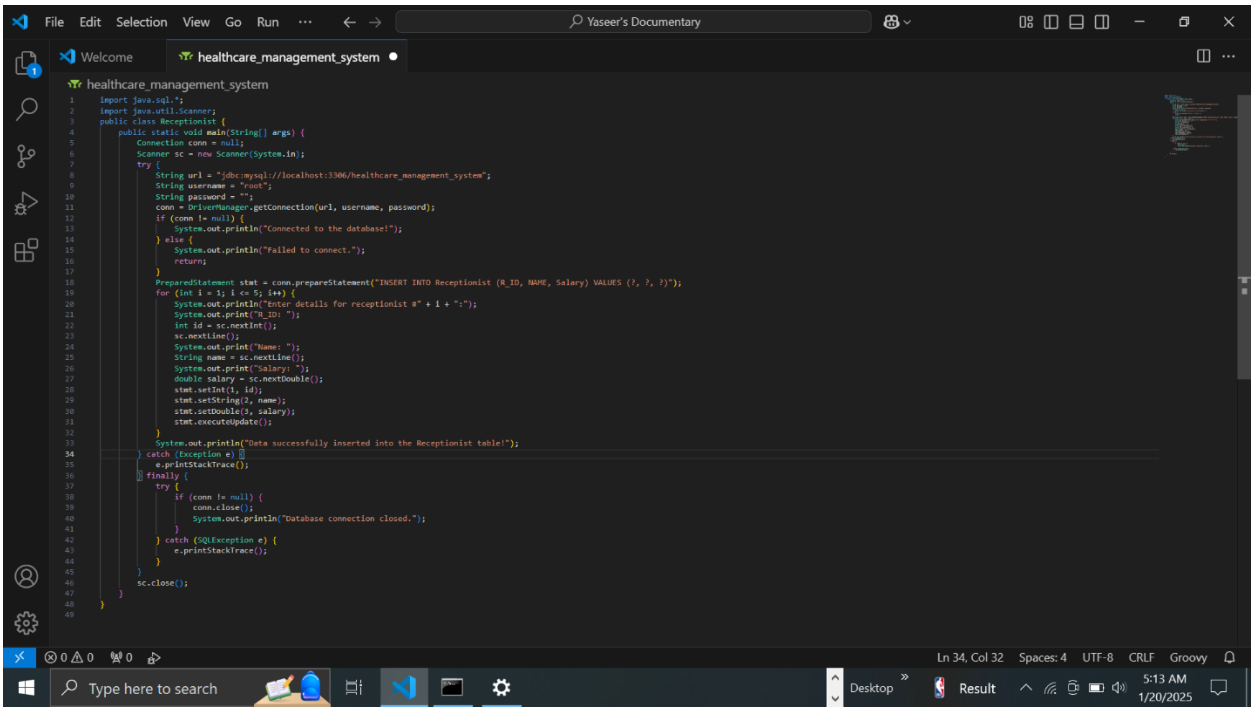
● Downloaded the mysql-connector-java-8.0.30.jar file to facilitate database connectivity.

2. I began by starting the Apache Web Server and MySQL Database through the XAMPP control panel. Afterward, I navigated to “http://localhost/phpmyadmin” to create a new database named "healthcare_management_system". Within this newly created database, I create a table titled "Receptionist" and then add columns according to our project. After that, I put some data into the table.

R_ID	Name	Salary
1	Shamima Begum	35000
2	Rafiqul Islam	32001
3	Mariam Akter	40001
4	Abdullah Al Mamun	28000
5	Nusrat Jahan	36000

Fig 9.2.1 Create and Insert Value in Officers Table

3. After making my database and table, I chose to use VSCODE editor for my project. I added the mysql-connector-java-8.0.30.jar library to my project. Then, in the main file, I wrote the code to connect to my Xampp-made database. Finally, I fetched and printed the columns of the Officers table.



```
import java.sql.*;
import java.util.Scanner;
public class Receptionist {
    public static void main(String[] args) {
        Connection conn = null;
        Scanner sc = new Scanner(System.in);
        try {
            String url = "jdbc:mysql://localhost:3306/healthcare_management_system";
            String username = "root";
            String password = "";
            conn = DriverManager.getConnection(url, username, password);
            if (conn != null) {
                System.out.println("Connected to the database!");
            } else {
                System.out.println("Failed to connect.");
                return;
            }
            PreparedStatement stmt = conn.prepareStatement("INSERT INTO Receptionist (R_ID, NAME, Salary) VALUES (?, ?, ?)");
            for (int i = 1; i <= 5; i++) {
                System.out.println("Enter details for receptionist #" + i + ":");
                System.out.print("R_ID: ");
                int id = sc.nextInt();
                sc.nextLine();
                System.out.print("Name: ");
                String name = sc.nextLine();
                System.out.print("Salary: ");
                double salary = sc.nextDouble();
                stmt.setInt(1, id);
                stmt.setString(2, name);
                stmt.setDouble(3, salary);
                stmt.executeUpdate();
            }
            System.out.println("Data successfully inserted into the Receptionist table!");
        } catch (Exception e) {
            e.printStackTrace();
        } finally {
            try {
                if (conn != null) {
                    conn.close();
                    System.out.println("Database connection closed.");
                }
            } catch (SQLException e) {
                e.printStackTrace();
            }
        }
        sc.close();
    }
}
```

Fig 9.2.2 DB Connection Code

9.2 doctors Table:

Name: Md.Samiul Kabir Mishel
ID: 23-54058-3

1. Software Installation and JAR File Download:
    - Installed Netbeans for code editing purposes.
    - Installed Xampp Server to set up a local development environment.
    - Downloaded the mysql-connector-java-8.0.30.jar file to facilitate database connectivity.

2. I began by starting the Apache Web Server and MySQL Database through the XAMPP control panel. Afterward, I navigated to "jdbc:mysql://localhost:3306/health_care_management_system" to create a new database named "health_care_management_system". Within this newly created database, I create a table titled "doctors", and then add columns according to our project. After that, I put some data into the table.

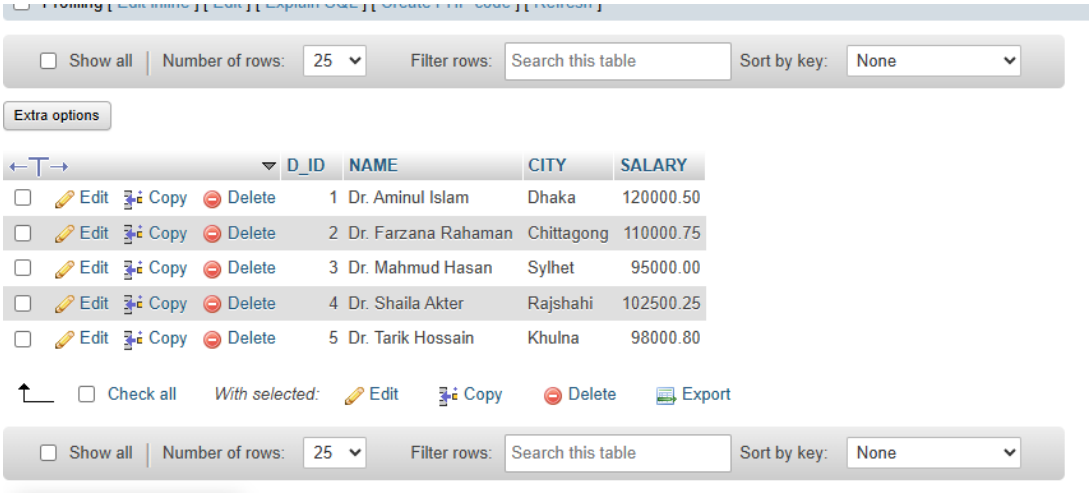
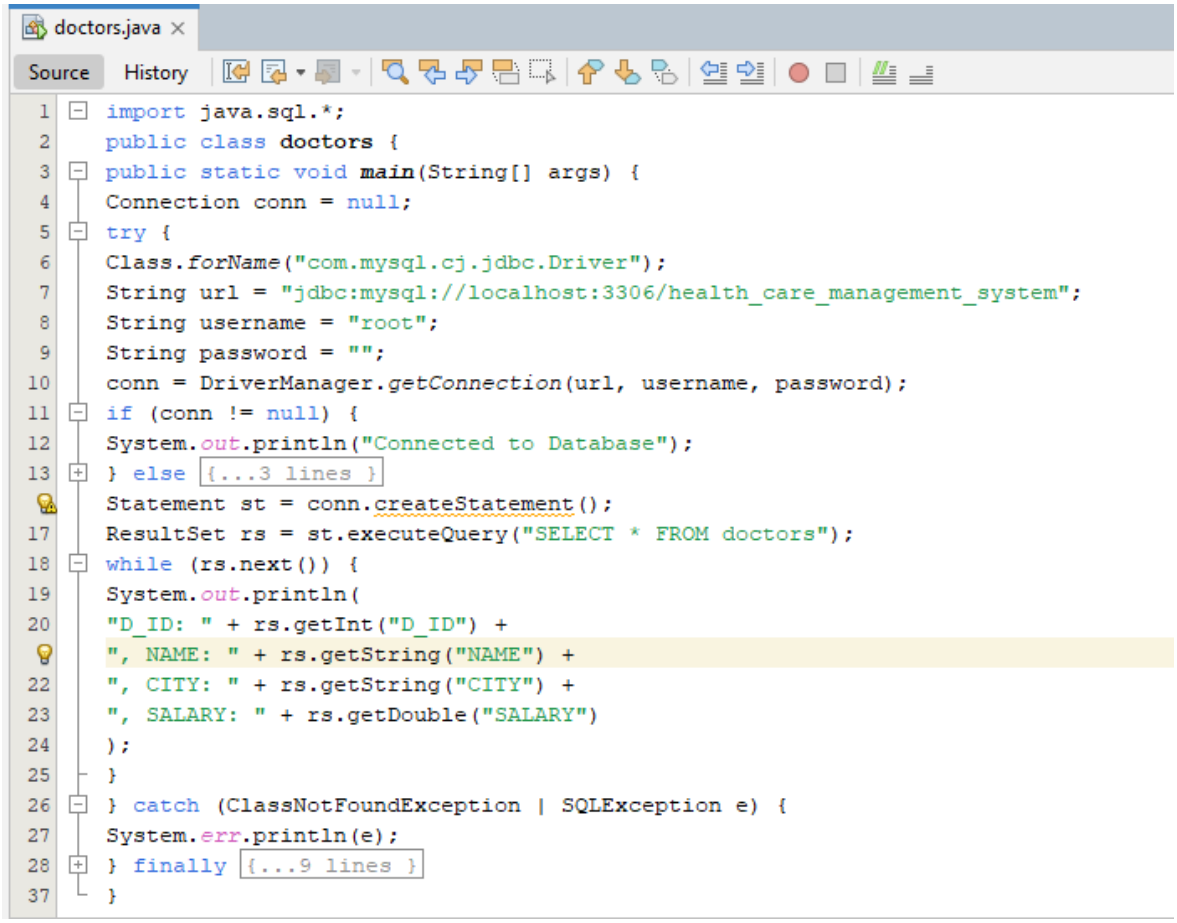


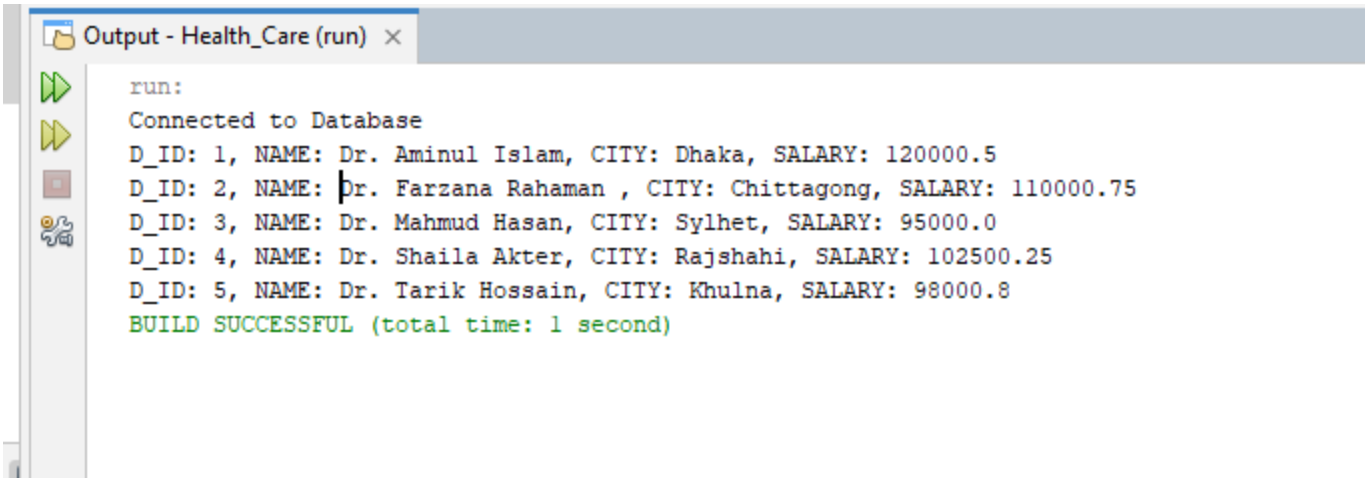
Fig 9.2.1 Create and Insert Value in doctors Table

3. After making my database and table, I chose to use Netbeans editor for my project. I added the mysql-connector-java-9.1.0.jar library to my project. Then, in the main file, I wrote the code to connect to my Xampp-made database. Finally, I fetched and printed the columns of the Officers table.



```java
import java.sql.*;
public class doctors {
public static void main(String[] args) {
Connection conn = null;
try {
Class.forName("com.mysql.cj.jdbc.Driver");
String url = "jdbc:mysql://localhost:3306/health_care_management_system";
String username = "root";
String password = "";
conn = DriverManager.getConnection(url, username, password);
if (conn != null) {
System.out.println("Connected to Database");
} else {...3 lines }
Statement st = conn.createStatement();
ResultSet rs = st.executeQuery("SELECT * FROM doctors");
while (rs.next()) {
System.out.println(
"D_ID: " + rs.getInt("D_ID") +
", NAME: " + rs.getString("NAME") +
", CITY: " + rs.getString("CITY") +
", SALARY: " + rs.getDouble("SALARY")
);
}
} catch (ClassNotFoundException | SQLException e) {
System.err.println(e);
} finally {...9 lines }
}
```

Fig 9.2.2 DB Connection Code



```
run:
Connected to Database
D_ID: 1, NAME: Dr. Aminul Islam, CITY: Dhaka, SALARY: 120000.5
D_ID: 2, NAME: Dr. Farzana Rahaman , CITY: Chittagong, SALARY: 110000.75
D_ID: 3, NAME: Dr. Mahmud Hasan, CITY: Sylhet, SALARY: 95000.0
D_ID: 4, NAME: Dr. Shaila Akter, CITY: Rajshahi, SALARY: 102500.25
D_ID: 5, NAME: Dr. Tarik Hossain, CITY: Khulna, SALARY: 98000.8
BUILD SUCCESSFUL (total time: 1 second)
```

Fig 9.2.3: Output

Conclusion

In this project, we started by writing a case study to understand the system requirements. Based on this, we designed an ER diagram to visualize the relationships between different entities. After that, we normalized the data and finalized the structure of the database. From this, we created nine tables: Doctor, Patient, Address, Receptionist, Nurse, Bill, Patient_Nurse, Patient_Receptionist, and Patient_Bill. Each table was populated with five rows of sample data to test functionality.Next, we performed five test queries to ensure the database was working as expected. After testing, we downloaded and installed XAMPP and the required JDBC jar file to manage thedatabase. We also installed Visual Studio Code (VS Code) to write and run our application code. Using XAMPP, we hosted the database and connected it to our application through VS Code.In VS Code, we wrote and tested code to retrieve and modify data in the database. This helped us verify that the application could interact with the database efficiently. The project taught us how to create, organize, and manage a database from scratch. It also showed us how to connect a database to an application using tools like XAMPP and VS Code.

Overall, this project provided practical experience in database design, normalization, and application integration. It helped us apply theoretical knowledge to real-world tasks, ensuring we understood every step of building and managing a functional database system.