



AMERICAN INTERNATIONAL UNIVERSITY–BANGLADESH (AIUB)
FACULTY OF SCIENCE & TECHNOLOGY

Introduction to Database

SPRING 2023-2024

Section: F, Group: 3

PROJECT ON
COACHING MANAGEMENT SYSTEM

Supervised By

SIFAT RAHMAN AHONA

Submitted By

Name	ID
MD MERAZ UDDIN	23-50322-1
SAKIB HOSSAIN ABIR	22-49309-3
SAMIA JANNAT LIZA	23-51085-1

Date of Submission: MAY 21, 2024

TABLE OF CONTENTS

TOPICS	<i>Page no.</i>
I. Title Page	1
II. Table of Content	2
1. Introduction	3
2. Scenario Description	4-5
3. ER Diagram	5
4. Normalization	6-15
5. Finalization	16-17
6. Table Creation and Data insertion	18-47
7. Querry Test	48-60
8. Relational Algebra	61-64
9. Conclusion	65

Introduction

The Coaching Management System aims to enhance communication and information handling within coaching centers. This system simplifies administrative tasks and ensures prompt updates for students, teachers, and parents through an easy-to-use online platform. In today's dynamic educational environment, effective communication among students, teachers, and parents is vital. However, traditional methods often lead to missed information on classes, exams, and other significant events, hindering students' progress and creating unnecessary hurdles. To overcome this challenge, we developed the Coaching Management System to keep all stakeholders well-informed. This system offers a user-friendly solution for efficient information management and sharing. It delivers real-time updates on classes, exams, and events, enabling students to stay updated on their academic commitments. Teachers can utilize the system to update schedules, share results, and communicate essential information directly with students and parents. Moreover, a dedicated section for parents enables them to track their child's academic performance, encouraging greater parental involvement in their child's education. The primary goal of the Coaching Management System is to provide a seamless and accessible platform for sharing crucial information among students, teachers, and parents. Through the integration of technology, we aim to foster a more interconnected educational community, promoting academic success and collaboration of all stakeholders involved.

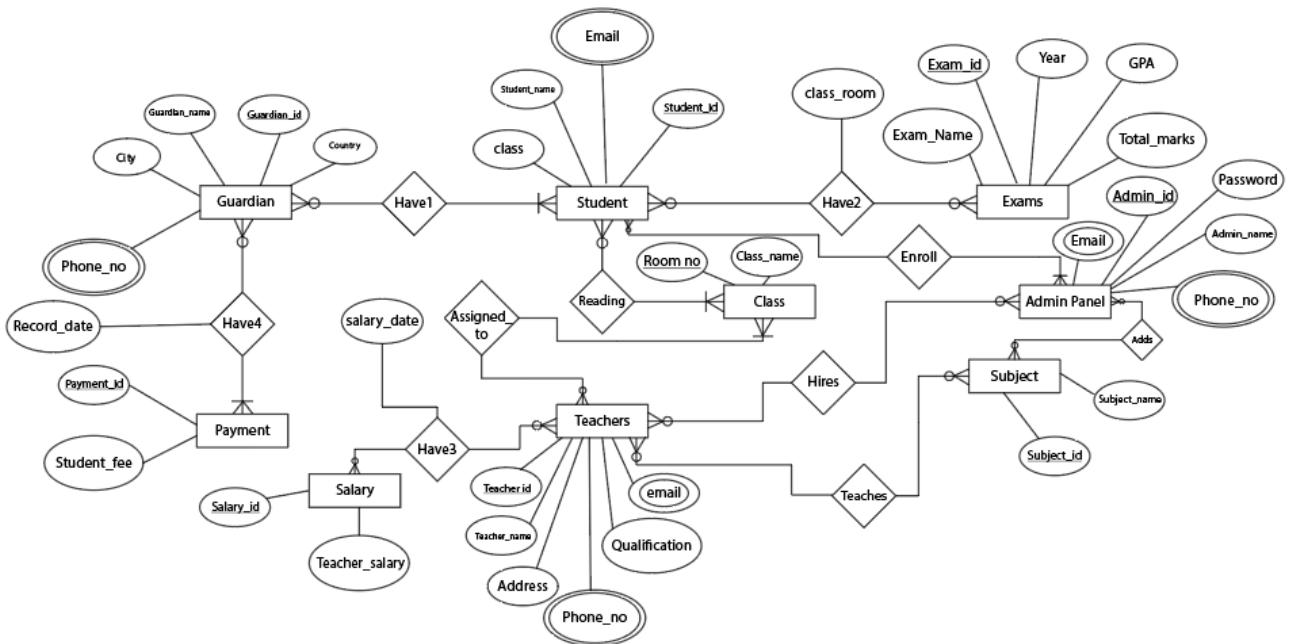
StudentID1: 23-50322-1 Name: MD MERAZ UDDIN	StudentID3: 23-51085-1 Name: SAMIA JANNAT LIZA
StudentID2: 22-49309-3 Name: SAKIB HOSSAIN ABIR	
CO2: Understand the fundamental concepts underlying database systems and gain hands-on experience with ER diagram Case study	
PO-e2: Develop process for complex computer science and engineering problems considering cultural and societal factors.	Marks

Scenario Description

In our Student Coaching Management System, each student is uniquely identified by the primary key, the Student ID, serving as the cornerstone for all student-related activities and information. Students also have some attributes like name, course, exam, phone number. This identifier plays a pivotal role in maintaining data integrity and facilitating efficient retrieval of student-specific details, forming the linchpin for various interactions within the system. Students, positioned as key entities, are intricately connected to diverse aspects crucial to their educational journey. The student-guardian relationship remains fundamental, with guardians uniquely identified by a Guardian ID. This connection ensures that students have dedicated support systems for their overall well-being, with guardian attributes, including Name, Phone Number, and Email (used as a password), intricately linked to corresponding student profiles. Moreover, students actively participate in academic assessments through exams, each uniquely distinguished by an Exam ID. This relationship empowers the systematic tracking of student performance and achievements across various examinations. The attributes within the student entity, including Name, Classroom, and Phone Number, collectively contribute to a comprehensive understanding of each student's academic profile. As we delve into the financial aspects, the Teachers Salary entity takes center stage, offering a nuanced perspective on the financial intricacies of the teaching faculty. Each teacher Salary record, identified by a unique ID, is seamlessly linked to specific teachers through the Teacher ID, teacher also has phone number, name, address, qualification, email establishing a robust foreign key relationship. The inclusion of the Salary Date ensures meticulous record-keeping, contributing to the transparency and accountability of our salary management system. The Amount attribute further encapsulates the financial dimension, detailing the specific payment amount associated with each salary record, providing a granular approach to salary transactions. Complementing the financial framework, the Admin Panel entity introduces administrative sophistication, defining user roles within the system. While it may not necessitate its own table in the database, the Admin Panel functions as a logical construct, influencing relationships such as "Hired by." This relationship signifies that teachers are recruited into the system by the school administration, represented by the Admin Panel, enforcing organizational structure through user roles. Despite these innovations, core relationships like "Enrolled in," "Has Guardian," "Pays fees," "Teaches," and the recent addition, "Receives Salary," remain pillars defining the interconnected dynamics

of our educational system. As we navigate through these advancements, unresolved aspects like the attributes Reading and Room No. beckon for further clarification to enhance the overall comprehensibility and coherence of our Student Coaching Management System. Nevertheless, our system remains at the forefront, providing a sophisticated and dynamic platform for educators, administrators, and students alike.

ER DIAGRAM



Normalization

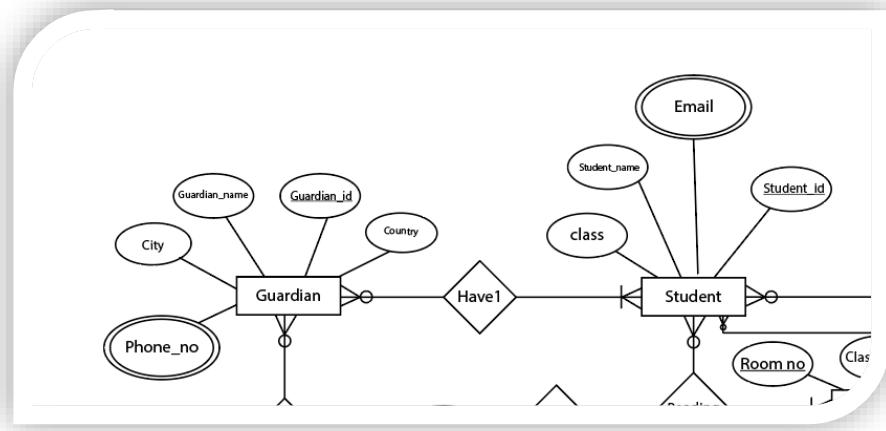


Fig-1: Normalization between Guardian and Student

Have1(Guardian_id, Guardian_name, City, Country, Phone_no, Student_id, Student_name, Class, Email)

1NF: Phone_no, Email multivalued attribute

2NF: Guardian_id, Guardian_name, City, Country, Phone_no
Student_id, Student_name, Class, Email
GS_id, Guardian_id, Student_id

3NF: Guardian_id, Guardian_name, ACid, Phone_no
Student_id, Student_name, Class, Email
GS_id, Guardian_id, Student_id
ACid, City, Country

Table:

Guardian_id, Guardian_name, ACid, Phone_no
Student_id, Student_name, Class, Email
GS_id, Guardian_id, Student_id
ACid, City, Country

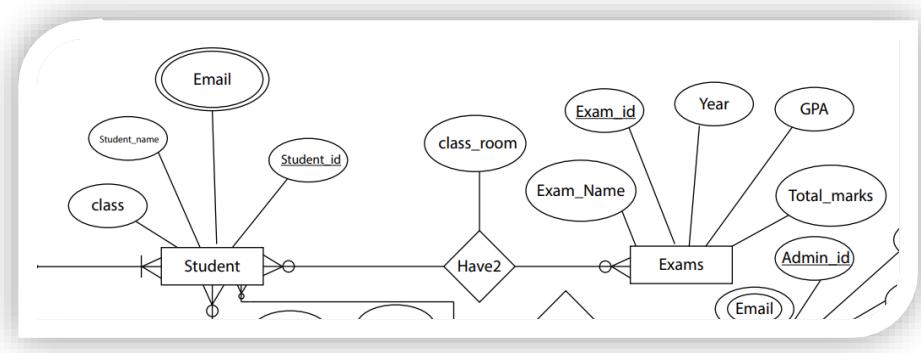


Fig-2: Normalization between Student and Exams

Have2 (Student_id, Student_name, Class, Email, Exam_id, Exam_name, Year, GPA, Total_marks)

1NF: Email multivalued attribute

2NF: Student_id, Student_name, Class, Email
Exam_id, Exam_name, Year, GPA, Total_marks
Exam_id, Student_id, Exam_name

3NF: Student_id, Student_name, Class, Email
Exam_id, Exam_name, Year, GPA, Total_marks
Exam_id, Student_id, Exam_name
No transitive dependency

Table:

Student_id, Student_name, Class, Email
Exam_id, Exam_name, Year, GPA, Total_marks
Exam_id, Student_id, Exam_name

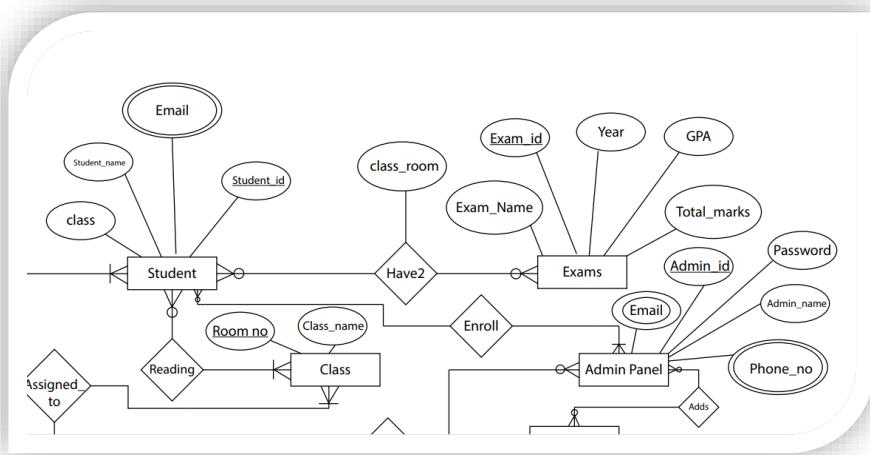


Fig-3: Normalization between Student and Admin Panel

Enroll (Student_id, Student_name, Class, Email, Admin_id, Admin_name, Password, Phone_no, Email)

1NF: Phone_no, Email multivalued attribute

2NF: Student_id, Student_name, Class, Email
Admin_id, Admin_name, Password, Phone_no, Email
SA_id, Student_id, Admin_id

3NF: Student_id, Student_name, Class, Email
Admin_id, Admin_name, Password, Phone_no, Email
SA_id, Student_id, Admin_id
 No transitive dependency

Table:

Student_id, Student_name, Class, Email
Admin_id, Admin_name, Password, Phone_no, Email
SA_id, Student_id, Admin_id

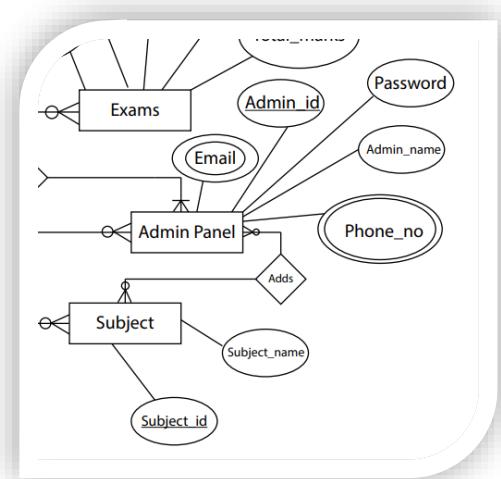


Fig-4: Normalization between Admin panel and Subject

Adds (Admin_id, Admin_name, Password, Phone_no, Email, Subject_id, Subject_name)

1NF: Email, Phone_no multivalued attribute

2NF: Admin_id, Admin_name, Password, Phone_no, Email
Subject_id, Subject_name
AS_id, Admin_id, Subject_id

3NF: Admin_id, Admin_name, Password, Phone_no, Email
Subject_id, Subject_name
AS_id, Admin_id, Subject_id
 No transitive dependency

Table:

Admin_id, Admin_name, Password, Phone_no, Email
Subject_id, Subject_name
AS_id, Admin_id, Subject_id

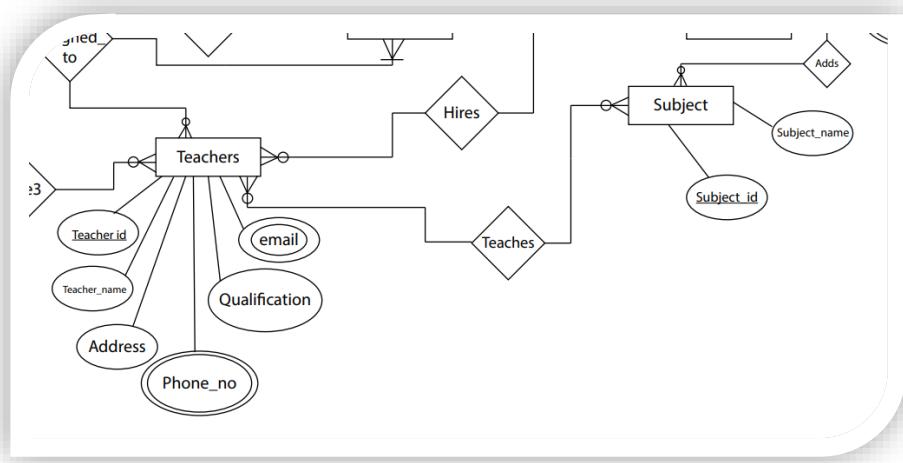


Fig-5: Normalization between Teachers and Subject

Teaches (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email, Subject_id, Subject_name)

1NF: Email,Phone_no multivalued attribute

2NF: Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email
Subject_id, Subject_name
TS_id, Teacher_id, Subject_id

3NF: Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email
Subject_id, Subject_name
TS_id, Teacher_id, Subject_id
 No transitive dependency

Table:

Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email
Subject_id, Subject_name
TS_id, Teacher_id, Subject_id

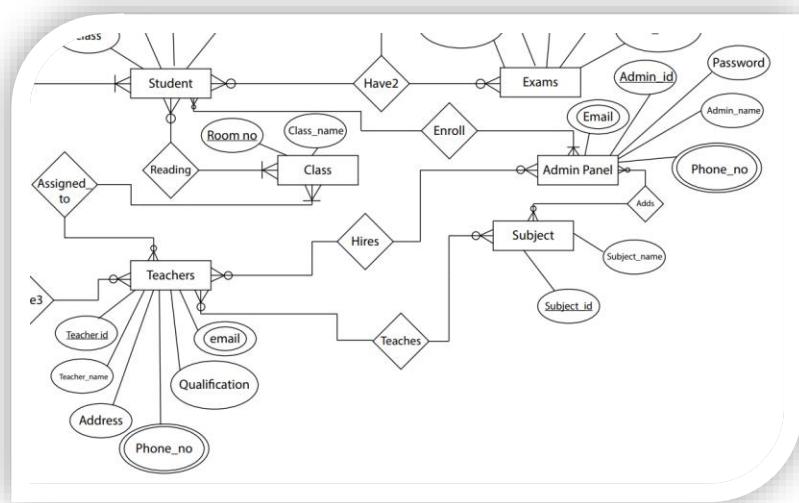


Fig-6: Normalization between Teachers and Admin panel

Hires (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email, Admin_id, Admin_name, Password, Phone_no, Email)

1NF: Email, Phone_no multivalued attribute

2NF: Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email
Admin_id, Admin_name, Password, Phone_no, Email
TA_id, Teacher_id, Admin_id

3NF: Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email
Admin_id, Admin_name, Password, Phone_no, Email
TA_id, Teacher_id, Admin_id
 No transitive dependency

Table:

Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email
Admin_id, Admin_name, Password, Phone_no, Email
TA_id, Teacher_id, Admin_id

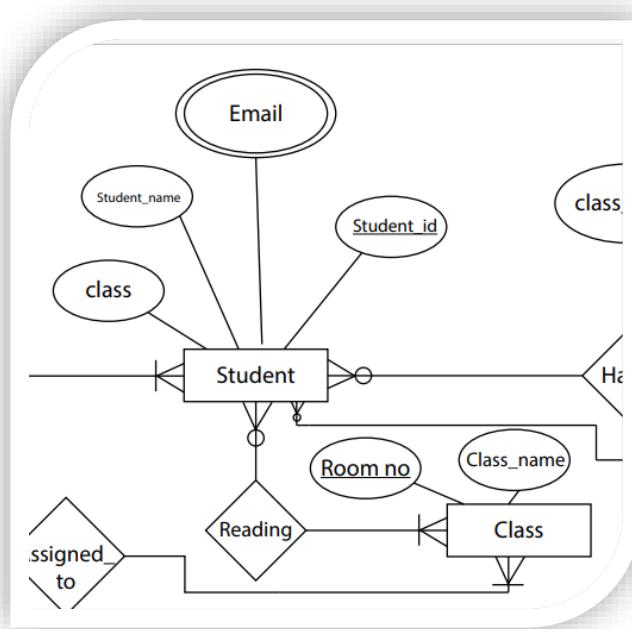


Fig-7: Normalization between Student and Class

Reading (Student_id, Student_name, Class, Email, Room_no, Class_name)

1NF: Email multivalued attribute

2NF: Student_id, Student_name, Class, Email
Room_no, Class_name
SR_id, Student_id, Room_no

3NF: Student_id, Student_name, Class, Email
Room_no, Class_name
SR_id, Student_id, Room_no
 No transitive dependency

Table:

Student_id, Student_name, Class, Email
Room_no, Class_name
SR_id, Student_id, Room_no

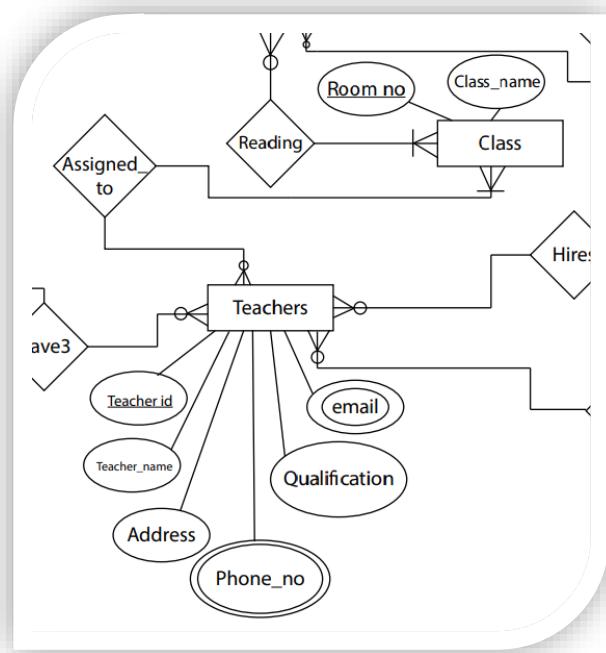


Fig-8: Normalization between Teachers and Class

Assigned_to (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email, Room_no, Class_name)

1NF: Email multivalued attribute

2NF: Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email
Room_no, Class_name
TR_id, Teacher_id, Room_no

3NF: Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email
Room_no, Class_name
TR_id, Teacher_id, Room_no
 No transitive dependency

Table:

Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email
Room_no, Class_name
TR_id, Teacher_id, Room

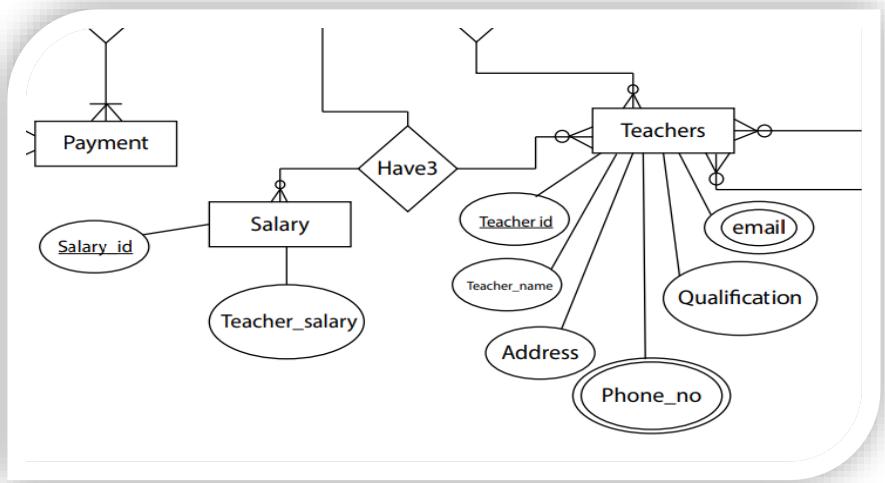


Fig-9: Normalization between Teachers and Salary

Have3 (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email, Salary_id, Teacher_salary)

1NF: Email, Phone_no multivalued attribute

2NF: Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email
Salary_id, Teacher_salary
TS_id, Teacher_id, Salary_id

3NF: Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email
Salary_id, Teacher_salary
TS_id, Teacher_id, Salary_id
 No transitive dependency

Table:

Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email
Salary_id, Teacher_salary
TS_id, Teacher_id, Salary_id

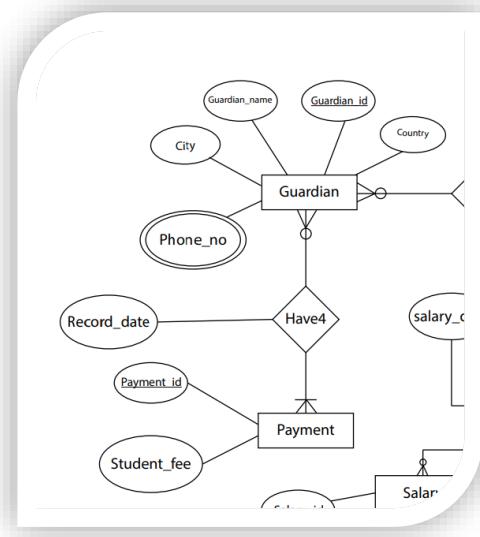


Fig-10: Normalization between Guardian and Payment

Have4 (Guardian_id, Guardian_name, Address, City, Country, Phone_no, Payment_id, Student_fee)

1NF: Phone_no multivalued attribute

2NF: Guardian_id, Guardian_name, Address, City, Country, Phone_no
Payment_id, Student_fee
GP_id, Guardian_id, Payment_id

3NF: Guardian_id, Guardian_name, ACid, Phone_no
Payment_id, Student_fee
GP_id, Guardian_id, Payment_id
ACid, Address, City, Country

Table:

Guardian_id, Guardian_name, ACid, Phone_no
Payment_id, Student_fee
GP_id, Guardian_id, Payment_id
ACid, Address, City, Country

Finalization

Total Table:

✓1. Guardian_id, Guardian_name, ACid, Phone_no
✓2. Student_id, Student_name, Class, Email
✓3. GS_id, Guardian_id, Student_id
✓4. ACid, City, Country
✗5. Student_id, Student_name, Class, Email
✓6. Exam_id, Exam_name, Year, GPA, Total_marks
✓7. SE_id, Student_id, Exam_id
✗8. Student_id, Student_name, Class, Email
✓9. Admin_id, Admin_name, Password, Phone_no, Email
✓10. SA_id, Student_id, Admin_id
✗11. Admin_id, Admin_name, Password, Phone_no, Email
✓12. Subject_id, Subject_name
✓13. AS_id, Admin_id, Subject_id
✓14. Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email
✗15. Subject_id, Subject_name
✓16. TS_id, Teacher_id, Subject_id
✗17. Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email
✗18. Admin_id, Admin_name, Password, Phone_no, Email
✓19. TA_id, Teacher_id, Admin_id
✗20. Student_id, Student_name, Class, Email
✗21. Room_no, Class_name
✓22. SR_id, Student_id, Room_no
✗23. Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email
✓24. Room_no, Class_name
✓25. TR_id, Teacher_id, Room_no
✗26. Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email
✓27. Salary_id, Teacher_salary
✓28. TS_id, Teacher_id, Salary_id
✗29. Guardian_id, Guardian_name, ACid, Phone_no
✓30. Payment_id, Student_fee
✓31. GP_id, Guardian_id, Payment_id
✗32. ACid, Address, City, Country

Final Table:

- 1.Guardian_id, Guardian_name, ACid, Phone_no
- 2.Student_id, Student_name, Class, Email
- 3.GS_id, Guardian_id, Student_id
- 4.ACid, City, Country
- 5.Exam_id, Exam_name, Year, GPA, Total_marks
- 6.SE_id, Student_id, Exam_id
- 7.Admin_id, Admin_name, Password, Phone_no, Email
- 8.SA_id, Student_id, Admin_id
- 9.Subject_id, Subject_name
- 10.AS_id, Admin_id, Subject_id
- 11.Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email
- 12.TS_id, Teacher_id, Subject_id
- 13.TA_id, Teacher_id, Admin_id
- 14.Room_no, Class_name
- 15.TR_id, Teacher_id, Room_no
- 16.SR_id, Student_id, Room_no
- 17.Salary_id, Teacher_salary
- 18.TS_id, Teacher_id, Salary_id
- 19.Payment_id, Student_fee
- 20 .GP_id, Guardian_id, Payment_id

Table Creation (DDL Operations) and Data Insertion

StudentID1: 22-49309-3 Name: SAKIB HOSSAIN ABIR	StudentID3: 23-51085-1 Name: SAMIA JANNAT LIZA
StudentID2: 23-50322-1 Name: MD MERAZ UDDIN	
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

1. Guardian Table:

```
create table Guardian
(
    Guardian_id varchar2(20) constraint Guardian_Gid_pk primary key,
    Guardian_name varchar2(30) not null,
    AC_id varchar2(20) constraint Guardian_acid_fk references AC(AC_id),
    Phone_no varchar2 (11) unique
)
```

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The SQL command window displays the creation of the 'Guardian' table with its constraints and data types. Below the command window, the 'Results' tab is active, showing the table structure with four columns: GUARDIAN_ID, GUARDIAN_NAME, AC_ID, and PHONE_NO. The table has a primary key constraint on GUARDIAN_ID and a foreign key constraint on AC_ID. The 'Comment' column is empty. At the bottom of the results, it says '1 - 4'. The status bar at the bottom right indicates 'Application Express 2.1.0.00.39' and 'Copyright © 1999, 2006, Oracle. All rights reserved.'

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
GUARDIAN	GUARDIAN_ID	Varchar2	20	-	-	1	-	-	-
	GUARDIAN_NAME	Varchar2	30	-	-	-	-	-	-
	AC_ID	Varchar2	20	-	-	-	✓	-	-
	PHONE_NO	Varchar2	11	-	-	-	✓	-	-

Fig-1: Guardian Table Create

```

insert into Guardian (Guardian_id, Guardian_name, AC_id, Phone_no) values ('G001', 'Mohammad Rahman', 'AC001', '01712345678')
insert into Guardian (Guardian_id, Guardian_name, AC_id, Phone_no) values ('G002', 'Fatima Akhtar', 'AC002', '01809876543')
insert into Guardian (Guardian_id, Guardian_name, AC_id, Phone_no) values ('G003', 'Abdul Haque', 'AC003', '01955512345')
insert into Guardian (Guardian_id, Guardian_name, AC_id, Phone_no) values ('G004', 'Aisha Begum', 'AC004', '01799988877')
insert into Guardian (Guardian_id, Guardian_name, AC_id, Phone_no) values ('G005', 'Sajjad Ali', 'AC005', '01811122233')
insert into Guardian (Guardian_id, Guardian_name, AC_id, Phone_no) values ('G006', 'Nazma Islam', 'AC006', '01744455566')
insert into Guardian (Guardian_id, Guardian_name, AC_id, Phone_no) values ('G007', 'Mizanur Rahman', 'AC007', '01977788899')
insert into Guardian (Guardian_id, Guardian_name, AC_id, Phone_no) values ('G008', 'Shahnaz Begum', 'AC008', '01866677788')
insert into Guardian (Guardian_id, Guardian_name, AC_id, Phone_no) values ('G009', 'Kamal Hossain', 'AC009', '01733344455')
insert into Guardian (Guardian_id, Guardian_name, AC_id, Phone_no) values ('G010', 'Farida Akhter', 'AC010', '01922233344')

```

GUARDIAN_ID	GUARDIAN_NAME	AC_ID	PHONE_NO
G001	Mohammad Rahman	AC001	01712345678
G002	Fatima Akhtar	AC002	01809876543
G003	Abdul Haque	AC003	01955512345
G004	Aisha Begum	AC004	01799988877
G005	Sajjad Ali	AC005	01811122233
G006	Nazma Islam	AC006	01744455566
G007	Mizanur Rahman	AC007	01977788899
G008	Shahnaz Begum	AC008	01866677788
G009	Kamal Hossain	AC009	01733344455
G010	Farida Akhter	AC010	01922233344

10 rows returned In 0.03 seconds [CSV Export](#)

Fig-2: Guardian Table Data

2. Student Table:

```
create table Student
(
    Student_id varchar2(20) constraint Students_Sid_pk primary key,
    Student_name varchar2(30) not null,
    Class varchar2(10),
    Email varchar2(20) unique
)
```

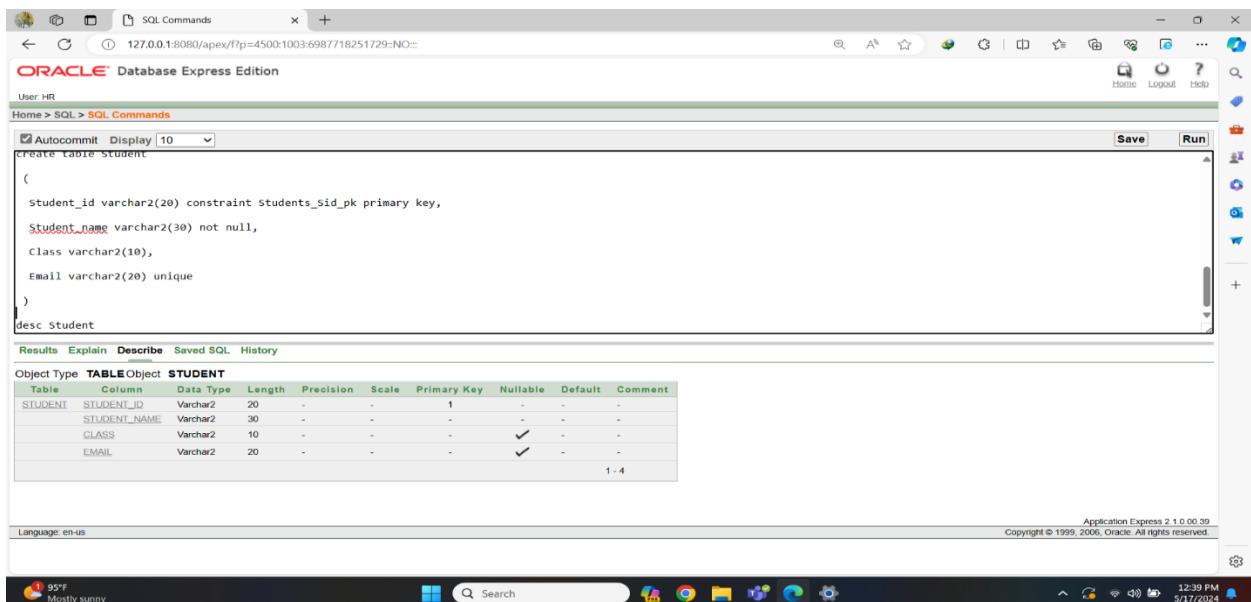


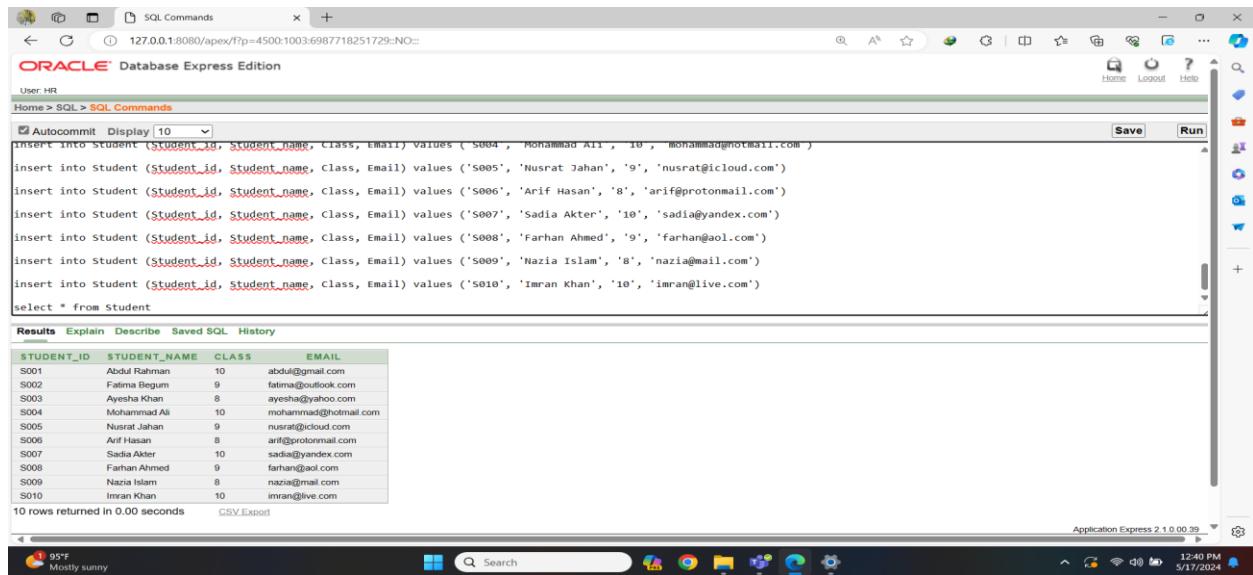
Fig-3: Student Table Create

```
insert into Student (Student_id, Student_name, Class, Email) values ('S001', 'Abdul Rahman', '10', 'abdul@gmail.com')
insert into Student (Student_id, Student_name, Class, Email) values ('S002', 'Fatima Begum', '9', 'fatima@outlook.com')
insert into Student (Student_id, Student_name, Class, Email) values ('S003', 'Ayesha Khan', '8', 'ayesha@yahoo.com')
insert into Student (Student_id, Student_name, Class, Email) values ('S004', 'Mohammad Ali', '10', 'mohammad@hotmail.com')
insert into Student (Student_id, Student_name, Class, Email) values ('S005', 'Nusrat Jahan', '9', 'nusrat@icloud.com')
insert into Student (Student_id, Student_name, Class, Email) values ('S006', 'Arif Hasan', '8', 'arif@protonmail.com')
insert into Student (Student_id, Student_name, Class, Email) values ('S007', 'Sadia Akter', '10', 'sadia@yandex.com')
```

```
insert into Student (Student_id, Student_name, Class, Email) values ('S008', 'Farhan Ahmed', '9', 'farhan@aol.com)
```

```
insert into Student (Student_id, Student_name, Class, Email) values ('S009', 'Nazia Islam', '8', 'nazia@mail.com)
```

```
insert into Student (Student_id, Student_name, Class, Email) values ('S010', 'Imran Khan', '10', 'imran@live.com)
```



STUDENT_ID	STUDENT_NAME	CLASS	EMAIL
S001	Abdul Rahman	10	abdul@gmail.com
S002	Fatima Begum	9	fatima@outlook.com
S003	Ayesha Khan	8	ayesha@yahoo.com
S004	Mohammad Ali	10	mohammad@hotmail.com
S005	Nusrat Jahan	9	nusrat@cloud.com
S006	Arif Hasan	8	arif@protonmail.com
S007	Sadia Akter	10	sadia@yandex.com
S008	Farhan Ahmed	9	farhan@aol.com
S009	Nazia Islam	8	nazia@mail.com
S010	Imran Khan	10	imran@live.com

Fig-4: Student Table Data

3. Guardian Student table:

```
create table GuardianStudent
(
    GS_id varchar2(20) constraint GuardianStudent_GSid_pk primary key,
    Guardian_id varchar2(20) constraint GuardianStudent_Guardianid_fk references
    Guardian(Guardian_id),
    Student_id varchar2(20) constraint GuardianStudent_Studentid_fk references
    Student(Student_id)
)
```

```

create table GUARDIANSTUDENT
(
    GS_ID varchar2(20) constraint GUARDIANSTUDENT_GSID_pk primary key,
    GUARDIAN_ID varchar2(20) constraint GUARDIANSTUDENT_Guardianid_fk references GUARDIAN(GUARDIAN_id),
    STUDENT_ID varchar2(20) constraint GUARDIANSTUDENT_Studentid_fk references STUDENT(STUDENT_id)
)
desc GUARDIANSTUDENT

```

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
GUARDIANSTUDENT	GS_ID	VARCHAR2	20	-	-	1	-	-	
GUARDIANSTUDENT	GUARDIAN_ID	VARCHAR2	20	-	-	-	✓	-	
GUARDIANSTUDENT	STUDENT_ID	VARCHAR2	20	-	-	-	✓	-	

Fig-5: Guardian-Student Table Create

insert into GuardianStudent (GS_id, Guardian_id, Student_id) values ('GS001', 'G001', 'S001')
 insert into GuardianStudent (GS_id, Guardian_id, Student_id) values ('GS002', 'G002', 'S002')
 insert into GuardianStudent (GS_id, Guardian_id, Student_id) values ('GS003', 'G003', 'S003')
 insert into GuardianStudent (GS_id, Guardian_id, Student_id) values ('GS004', 'G004', 'S004')
 insert into GuardianStudent (GS_id, Guardian_id, Student_id) values ('GS005', 'G005', 'S005')
 insert into GuardianStudent (GS_id, Guardian_id, Student_id) values ('GS006', 'G006', 'S006')
 insert into GuardianStudent (GS_id, Guardian_id, Student_id) values ('GS007', 'G007', 'S007')
 insert into GuardianStudent (GS_id, Guardian_id, Student_id) values ('GS008', 'G008', 'S008')
 insert into GuardianStudent (GS_id, Guardian_id, Student_id) values ('GS009', 'G009', 'S009')
 insert into GuardianStudent (GS_id, Guardian_id, Student_id) values ('GS010', 'G010', 'S010')

```

insert into GUARDIANSTUDENT (GS_id, GUARDIAN_id, STUDENT_id) values ('GS005', 'G005', 'S005')
insert into GUARDIANSTUDENT (GS_id, GUARDIAN_id, STUDENT_id) values ('GS006', 'G006', 'S006')
insert into GUARDIANSTUDENT (GS_id, GUARDIAN_id, STUDENT_id) values ('GS007', 'G007', 'S007')
insert into GUARDIANSTUDENT (GS_id, GUARDIAN_id, STUDENT_id) values ('GS008', 'G008', 'S008')
insert into GUARDIANSTUDENT (GS_id, GUARDIAN_id, STUDENT_id) values ('GS009', 'G009', 'S009')
insert into GUARDIANSTUDENT (GS_id, GUARDIAN_id, STUDENT_id) values ('GS010', 'G010', 'S010')
select * from GUARDIANSTUDENT

```

GS_ID	GUARDIAN_ID	STUDENT_ID
GS001	G001	S001
GS002	G002	S002
GS003	G003	S003
GS004	G004	S004
GS005	G005	S005
GS006	G006	S006
GS007	G007	S007
GS008	G008	S008
GS009	G009	S009
GS010	G010	S010

Fig-6: Guardian-Student Tabel Data

4. Address City Table:

```
create table AddressCity
(
    AC_id varchar(20) constraint AddressCity_acid_pk primary key,
    City varchar2(20) not null,
    Country varchar2(20) default 'Bangladesh'
)
```

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
create table AddressCity
(
    AC_id varchar(20) constraint AddressCity_acid_pk primary key,
    City varchar2(20) not null,
    Country varchar2(20) default 'Bangladesh'
)
desc AddressCity
```

Below the code, the results of the DESCRIBE command are displayed in a table:

Object Type	TABLE	Object	ADDRESSCITY						
ADDRESSCITY	AC_ID	VARCHAR2	20	-	-	1	-	-	-
	CITY	VARCHAR2	20	-	-	-	-	-	-
	COUNTRY	VARCHAR2	20	-	-	-	✓	Bangladesh	-

At the bottom of the interface, there is a status bar showing the language as "en-us", the application version as "Application Express 2.1.0.00.39", and the copyright notice "Copyright © 1999, 2006, Oracle. All rights reserved."

Fig-7:Address-City Table Create

```
insert into AddressCity (AC_id, City, Country) values ('AC001', 'Dhaka', 'Bangladesh')
insert into AddressCity (AC_id, City, Country) values ('AC002', 'Chittagong', 'Bangladesh')
insert into AddressCity (AC_id, City, Country) values ('AC003', 'Sylhet', 'Bangladesh')
insert into AddressCity (AC_id, City, Country) values ('AC004', 'Khulna', 'Bangladesh')
insert into AddressCity (AC_id, City, Country) values ('AC005', 'Rajshahi', 'Bangladesh')
insert into AddressCity (AC_id, City, Country) values ('AC006', 'Barisal', 'Bangladesh')
insert into AddressCity (AC_id, City, Country) values ('AC007', 'Comilla', 'Bangladesh')
insert into AddressCity (AC_id, City, Country) values ('AC008', 'Mymensingh', 'Bangladesh')
insert into AddressCity (AC_id, City, Country) values ('AC009', 'Rangpur', 'Bangladesh')
insert into AddressCity (AC_id, City, Country) values ('AC010', 'Dinajpur', 'Bangladesh')
```

```

SQL Commands
127.0.0.1:8080/apex/F?p=4500:1003:6987718251729::NO::
ORACLE Database Express Edition
User HR
Home > SQL > SQL Commands
Autocommit Display | 10 | Save Run
insert into AddressCity (AC_id, city, Country) values ('AC004', 'Khulna', 'Bangladesh')
insert into AddressCity (AC_id, city, Country) values ('AC005', 'Rajshahi', 'Bangladesh')
insert into AddressCity (AC_id, city, Country) values ('AC006', 'Barisal', 'Bangladesh')
insert into AddressCity (AC_id, city, Country) values ('AC007', 'Comilla', 'Bangladesh')
insert into AddressCity (AC_id, city, Country) values ('AC008', 'Mymensingh', 'Bangladesh')
insert into AddressCity (AC_id, city, Country) values ('AC009', 'Rangpur', 'Bangladesh')
insert into AddressCity (AC_id, city, Country) values ('AC010', 'Dinajpur', 'Bangladesh')
select * from AddressCity
Results Explain Describe Saved SQL History
AC_ID CITY COUNTRY
AC003 Sylhet Bangladesh
AC004 Khulna Bangladesh
AC005 Rajshahi Bangladesh
AC001 Dhaka Bangladesh
AC002 Chittagong Bangladesh
AC006 Barisal Bangladesh
AC007 Comilla Bangladesh
AC008 Mymensingh Bangladesh
AC009 Rangpur Bangladesh
AC010 Dinajpur Bangladesh
10 rows returned in 0.00 seconds CSV Export

```

Fig-8: Address-City Table Data

5. Exam Table:

create table Exam

(

Exam_id varchar2(20) constraint Exam_Eid_pk primary key,
 Exam_name varchar2(30) unique,
 Year varchar2(10),
 GPA number(3,2),
 Total_marks number(10)

)

```

SQL Commands
127.0.0.1:8080/apex/F?p=4500:1003:6987718251729::NO::
ORACLE Database Express Edition
User HR
Home > SQL > SQL Commands
Autocommit Display | 10 | Save Run
create table Exam
(
    Exam_id varchar2(20) constraint Exam_Eid_pk primary key,
    Exam_name varchar2(30) unique,
    Year varchar2(10),
    GPA number(3,2),
    Total_marks number(10)
)
desc Exam
Results Explain Describe Saved SQL History
Object Type TABLE Object EXAM
Table Column Data Type Length Precision Scale Primary Key Nullable Default Comment
EXAM EXAM_ID Varchar2 20 - - 1 - - -
EXAM_NAME Varchar2 30 - - - - -
YEAR Varchar2 10 - - - ✓ - -
GPA Number - 3 2 - - ✓ - -
TOTAL_MARKS Number - 10 0 - - ✓ - -
1 - 5
Language: en-us
Application Express 2.1.0.0.39
Copyright © 1999, 2006, Oracle. All rights reserved.

```

Fig-9: Exam Table Create

```

insert into Exam (Exam_id, Exam_name, Year, GPA, Total_marks) values ('E001',
'Mathematics', '2023', 5.00, 1000)
insert into Exam (Exam_id, Exam_name, Year, GPA, Total_marks) values ('E002', 'Science',
'2023', 4.80, 1200)
insert into Exam (Exam_id, Exam_name, Year, GPA, Total_marks) values ('E003', 'English',
'2023', 4.20, 800)
insert into Exam (Exam_id, Exam_name, Year, GPA, Total_marks) values ('E004', 'History',
'2023', 3.00, 900)
insert into Exam (Exam_id, Exam_name, Year, GPA, Total_marks) values ('E005', 'Geography',
'2023', 3.60, 1100)
insert into Exam (Exam_id, Exam_name, Year, GPA, Total_marks) values ('E006', 'Biology',
'2023', 4.50, 1300)
insert into Exam (Exam_id, Exam_name, Year, GPA, Total_marks) values ('E007', 'Chemistry',
'2023', 4.20, 1250)
insert into Exam (Exam_id, Exam_name, Year, GPA, Total_marks) values ('E008', 'Physics',
'2023', 4.30, 1150)
insert into Exam (Exam_id, Exam_name, Year, GPA, Total_marks) values ('E009', 'Economics',
'2023', 3.60, 950)
insert into Exam (Exam_id, Exam_name, Year, GPA, Total_marks) values ('E010', 'Database',
'2023', 4.80, 1350)

```

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The SQL editor contains the following code:

```


Autocommit Display 10
Insert into Exam (Exam_id, Exam_name, Year, GPA, Total_marks) Values ('E004', 'History', 2023, 3.00, 900)
insert into Exam (Exam_id, Exam_name, Year, GPA, Total_marks) values ('E005', 'Geography', '2023', 3.60, 1100)
insert into Exam (Exam_id, Exam_name, Year, GPA, Total_marks) values ('E006', 'Biology', '2023', 4.50, 1300)
insert into Exam (Exam_id, Exam_name, Year, GPA, Total_marks) values ('E007', 'Chemistry', '2023', 4.20, 1250)
insert into Exam (Exam_id, Exam_name, Year, GPA, Total_marks) values ('E008', 'Physics', '2023', 4.30, 1150)
insert into Exam (Exam_id, Exam_name, Year, GPA, Total_marks) values ('E009', 'Economics', '2023', 3.60, 950)
insert into Exam (Exam_id, Exam_name, Year, GPA, Total_marks) values ('E010', 'Database', '2023', 4.80, 1350)
select * from Exam


```

The Results tab displays the following data:

EXAM_ID	EXAM_NAME	YEAR	GPA	TOTAL_MARKS
E001	Mathematics	2023	5	1000
E002	Science	2023	4.8	1200
E003	English	2023	4.2	800
E004	History	2023	3	900
E005	Geography	2023	3.6	1100
E006	Biology	2023	4.5	1300
E007	Chemistry	2023	4.2	1250
E008	Physics	2023	4.3	1150
E009	Economics	2023	3.6	950
E010	Database	2023	4.8	1350

10 rows returned in 0.00 seconds [CSV Export](#)

Fig-10: Exam Table Data

6. Student Exam Table:

```
create table StudentExam
```

```
(
```

```
SE_id varchar2(20) constraint StudentExam_SEid_pk primary key,  
Student_id varchar2(20) constraint StudentExam_Studentid_fk references Student(Student_id),  
Exam_id varchar2(20) constraint StudentExam_Examid_fk references Exam(Exam_id)
```

```
)
```

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
create table StudentExam  
(  
SE_id varchar2(20) constraint StudentExam_SEid_pk primary key,  
Student_id varchar2(20) constraint StudentExam_Studentid_fk references Student(Student_id),  
Exam_id varchar2(20) constraint StudentExam_Examid_fk references Exam(Exam_id)  
)  
  
desc StudentExam
```

Below the code, the 'Describe' tab is selected, showing the table structure:

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
STUDENTEXAM	SE_ID	Varchar2	20	-	-	1	-	-	-
	STUDENT_ID	Varchar2	20	-	-	-	✓	-	-
	EXAM_ID	Varchar2	20	-	-	-	✓	-	-

At the bottom of the interface, the status bar indicates "Application Express 2.1.0.00.39" and "Copyright © 1999, 2006, Oracle. All rights reserved."

Fig-11: Student_Exam Table Create

```
insert into StudentExam (SE_id, Student_id, Exam_id) values ('se001', 's001', 'e001')  
insert into StudentExam (SE_id, Student_id, Exam_id) values ('se002', 's002', 'e002')  
insert into StudentExam (SE_id, Student_id, Exam_id) values ('se003', 's003', 'e003')  
insert into StudentExam (SE_id, Student_id, Exam_id) values ('se004', 's004', 'e004')  
insert into StudentExam (SE_id, Student_id, Exam_id) values ('se005', 's005', 'e005')  
insert into StudentExam (SE_id, Student_id, Exam_id) values ('se006', 's006', 'e006')  
insert into StudentExam (SE_id, Student_id, Exam_id) values ('se007', 's007', 'e007')  
insert into StudentExam (SE_id, Student_id, Exam_id) values ('se008', 's008', 'e008')  
insert into StudentExam (SE_id, Student_id, Exam_id) values ('se009', 's009', 'e009')  
insert into StudentExam (SE_id, Student_id, Exam_id) values ('se010', 's010', 'e010')
```

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
Insert into STUDENTEXAM (SE_id, STUDENT_id, EXAM_id) values ('se004', 's004', 'e004')
insert into StudentExam (SE_id, Student_id, Exam_id) values ('se005', 's005', 'e005')
insert into StudentExam (SE_id, Student_id, Exam_id) values ('se006', 's006', 'e006')
insert into StudentExam (SE_id, Student_id, Exam_id) values ('se007', 's007', 'e007')
insert into StudentExam (SE_id, Student_id, Exam_id) values ('se008', 's008', 'e008')
insert into StudentExam (SE_id, Student_id, Exam_id) values ('se009', 's009', 'e009')
insert into StudentExam (SE_id, Student_id, Exam_id) values ('se010', 's010', 'e010')
select * from StudentExam
```

The Results tab shows the output of the query:

SE_ID	STUDENT_ID	EXAM_ID
SE001	S001	E001
SE002	S002	E002
SE003	S003	E003
SE004	S004	E004
SE005	S005	E005
SE006	S006	E006
SE007	S007	E007
SE008	S008	E008
SE009	S009	E009
SE010	S010	E010

10 rows returned in 0.00 seconds

Fig-12: Student-Exam Table Data

7. Admin Table:

create table Admin

(

 Admin_id varchar2(20) constraint Admin_Adminid_pk primary key,
 Admin_name varchar2(30) not null,
 Password varchar2(20),
 Phone_no number(11),
 Email varchar2(30) unique

)

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
(  

    Admin_id varchar2(20) constraint Admin_Adminid_pk primary key,  

    Admin_name varchar2(30) not null,  

    Password varchar2(20),  

    Phone_no number(11),  

    Email varchar2(30) unique  

)desc Admin
```

The Results tab shows the output of the query:

Object Type	Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ADMIN	ADMIN_ID	VARCHAR2	20	-	-	-	1	-	-	-
ADMIN	ADMIN_NAME	VARCHAR2	30	-	-	-	-	-	-	-
ADMIN	PASSWORD	VARCHAR2	20	-	-	-	-	✓	-	-
ADMIN	PHONE_NO	NUMBER	-	11	0	-	-	✓	-	-
ADMIN	EMAIL	VARCHAR2	30	-	-	-	-	✓	-	-

1 - 5

Language: en-us

Fig-13: Admin Table Create

```

insert into Admin (Admin_id, Admin_name, Password, Phone_no, Email) values ('A001', 'Sohel Ahmed', 'password123', 01712345678, 'sohel@example.com')
insert into Admin (Admin_id, Admin_name, Password, Phone_no, Email) values ('A002', 'Nazneen Khan', 'admin123', 01809876543, 'nazneen@example.com')
insert into Admin (Admin_id, Admin_name, Password, Phone_no, Email) values ('A003', 'Imran Chowdhury', 'admin1234', 01955512345, 'imran@example.com')
insert into Admin (Admin_id, Admin_name, Password, Phone_no, Email) values ('A004', 'Sumaiya Islam', 'securepass', 01799988877, 'sumaiya@example.com')
insert into Admin (Admin_id, Admin_name, Password, Phone_no, Email) values ('A005', 'Rahat Rahman', 'password', 01811122233, 'rahat@example.com')
insert into Admin (Admin_id, Admin_name, Password, Phone_no, Email) values ('A006', 'Tasnim Ahmed', 'adminpass', 01744455566, 'tasnim@example.com')
insert into Admin (Admin_id, Admin_name, Password, Phone_no, Email) values ('A007', 'Nafisa Khan', 'passadmin', 01977788899, 'nafisa@example.com')
insert into Admin (Admin_id, Admin_name, Password, Phone_no, Email) values ('A008', 'Sarwar Islam', 'admin@123', 01866677788, 'sarwar@example.com')
insert into Admin (Admin_id, Admin_name, Password, Phone_no, Email) values ('A009', 'Anika Chowdhury', 'admin12345', 01733344455, 'anika@example.com')
insert into Admin (Admin_id, Admin_name, Password, Phone_no, Email) values ('A010', 'Tahmid Akhter', 'admin12', 01922233344, 'tahmid@example.com')

```

ADMIN_ID	ADMIN_NAME	PASSWORD	PHONE_NO	EMAIL
A001	Sohel Ahmed	password123	1712345678	sohel@example.com
A002	Nazneen Khan	admin123	1809876543	nazneen@example.com
A003	Imran Chowdhury	admin1234	1955512345	imran@example.com
A004	Sumaiya Islam	securepass	1799988877	sumaiya@example.com
A005	Rahat Rahman	password	1811122233	rahat@example.com
A006	Tasnim Ahmed	adminpass	1744455566	tasnim@example.com
A007	Nafisa Khan	passadmin	1977788899	nafisa@example.com
A008	Sarwar Islam	admin@123	1866677788	sarwar@example.com
A009	Anika Chowdhury	admin12345	1733344455	anika@example.com
A010	Tahmid Akhter	admin12	1922233344	tahmid@example.com

10 rows returned in 0.00 seconds [CSV Export](#)

Application Express 2.1.0.0.39

Fig-14: Admin Table Data

8. Student Admin Table:

```
create table StudentAdmin
(
    SA_id varchar2(20) constraint StudentAdmin _SAid_pk primary key,
    Student_id varchar2(20) constraint StudentAdmin _Studentid_fk references
Student(Student_id),
    Admin_id varchar2(20) constraint StudentAdmin _Adminid_fk references Admin/Admin_id)
)
```

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The SQL editor window contains the DDL code for creating the StudentAdmin table. Below the editor is a results grid showing the table structure with three columns: SA_ID, STUDENT_ID, and ADMIN_ID. The bottom status bar indicates the application version (Application Express 2.1.0.00.39), copyright information (Copyright © 1999, 2006, Oracle. All rights reserved.), and the system date and time (5/17/2024 12:54 PM).

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
STUDENTADMIN	SA_ID	Varchar2	20	-	-	1	-	-	-
	STUDENT_ID	Varchar2	20	-	-	-	✓	-	-
	ADMIN_ID	Varchar2	20	-	-	-	✓	-	-

Fig-15: Student-Admin Table Create

```
insert into StudentAdmin (SA_id, Student_id, Admin_id) values ('SA001', 'S001', 'A001')
insert into StudentAdmin (SA_id, Student_id, Admin_id) values ('SA002', 'S002', 'A002')
insert into StudentAdmin (SA_id, Student_id, Admin_id) values ('SA003', 'S003', 'A003')
insert into StudentAdmin (SA_id, Student_id, Admin_id) values ('SA004', 'S004', 'A004')
insert into StudentAdmin (SA_id, Student_id, Admin_id) values ('SA005', 'S005', 'A005')
insert into StudentAdmin (SA_id, Student_id, Admin_id) values ('SA006', 'S006', 'A006')
insert into StudentAdmin (SA_id, Student_id, Admin_id) values ('SA007', 'S007', 'A007')
insert into StudentAdmin (SA_id, Student_id, Admin_id) values ('SA008', 'S008', 'A008')
insert into StudentAdmin (SA_id, Student_id, Admin_id) values ('SA009', 'S009', 'A009')
insert into StudentAdmin (SA_id, Student_id, Admin_id) values ('SA010', 'S010', 'A010')
```

```

SQL Commands
127.0.0.1:8080/apex/F?p=4500:1003:6987718251729::NO::
ORACLE Database Express Edition
User HR
Home > SQL > SQL Commands
Autocommit Display 10 Save Run
insert into StudentAdmin (SA_id, student_id, Admin_id) values ('SA004', 'S004', 'A004')
insert into StudentAdmin (SA_id, student_id, Admin_id) values ('SA005', 'S005', 'A005')
insert into StudentAdmin (SA_id, student_id, Admin_id) values ('SA006', 'S006', 'A006')
insert into StudentAdmin (SA_id, student_id, Admin_id) values ('SA007', 'S007', 'A007')
insert into StudentAdmin (SA_id, student_id, Admin_id) values ('SA008', 'S008', 'A008')
insert into StudentAdmin (SA_id, student_id, Admin_id) values ('SA009', 'S009', 'A009')
insert into StudentAdmin (SA_id, student_id, Admin_id) values ('SA010', 'S010', 'A010')
select * from StudentAdmin
Results Explain Describe Saved SQL History
SA_ID STUDENT_ID ADMIN_ID
SA001 S001 A001
SA002 S002 A002
SA003 S003 A003
SA004 S004 A004
SA005 S005 A005
SA006 S006 A006
SA007 S007 A007
SA008 S008 A008
SA009 S009 A009
SA010 S010 A010
10 rows returned in 0.01 seconds CSV Export
Application Express 2.1.0.00.39

```

Fig-16: Student-Admin Table Data

9. Subject Table:

create table Subject

(

 Subject_id varchar2(20) constraint Subject_Subjectid_pk primary key,
 Subject_name varchar2(30) not null

)

```

SQL Commands
127.0.0.1:8080/apex/F?p=4500:1003:6987718251729::NO::
ORACLE Database Express Edition
User HR
Home > SQL > SQL Commands
Autocommit Display 10 Save Run
insert into StudentAdmin (SA_id, student_id, Admin_id) Values ('SA010', 'S010', 'A010')
select * from StudentAdmin
create table Subject
(
    Subject_id varchar2(20) constraint Subject_Subjectid_pk primary key,
    Subject_name varchar2(30) not null
)
desc Subject
Results Explain Describe Saved SQL History
Object Type TABLE Object SUBJECT
Table Column Data Type Length Precision Scale Primary Key Nullable Default Comment
SUBJECT SUBJECT_ID Varchar2 20 - - 1 - - -
SUBJECT_NAME Varchar2 30 - - - - - -
1-2
Language: en-us
Application Express 2.1.0.00.39
Copyright © 1999, 2006, Oracle. All rights reserved.

```

Fig-17: Subject Table Create

```

insert into Subject (Subject_id, Subject_name) values ('Sub001', 'Mathematics')
insert into Subject (Subject_id, Subject_name) values ('Sub002', 'Physics')
insert into Subject (Subject_id, Subject_name) values ('Sub003', 'Chemistry')
insert into Subject (Subject_id, Subject_name) values ('Sub004', 'Biology')
insert into Subject (Subject_id, Subject_name) values ('Sub005', 'English')
insert into Subject (Subject_id, Subject_name) values ('Sub006', 'History')
insert into Subject (Subject_id, Subject_name) values ('Sub007', 'Geography')
insert into Subject (Subject_id, Subject_name) values ('Sub008', 'Computer Science')
insert into Subject (Subject_id, Subject_name) values ('Sub009', 'Economics')
insert into Subject (Subject_id, Subject_name) values ('Sub010', 'Accounting')

```

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, a script is run:

```

Insert into Subject (Subject_id, Subject_name) Values ('Sub004', 'Biology')
insert into Subject (Subject_id, Subject_name) values ('Sub005', 'English')
insert into Subject (Subject_id, Subject_name) values ('Sub006', 'History')
insert into Subject (Subject_id, Subject_name) values ('Sub007', 'Geography')
insert into Subject (Subject_id, Subject_name) values ('Sub008', 'Computer Science')
insert into Subject (Subject_id, Subject_name) values ('Sub009', 'Economics')
insert into Subject (Subject_id, Subject_name) values ('Sub010', 'Accounting')
select * from Subject

```

The Results tab displays the following data:

SUBJECT_ID	SUBJECT_NAME
Sub003	Chemistry
Sub004	Biology
Sub005	English
Sub001	Mathematics
Sub002	Physics
Sub006	History
Sub007	Geography
Sub008	Computer Science
Sub009	Economics
Sub010	Accounting

10 rows returned in 0.00 seconds

Fig-18: Subject Table Data

10. Admin Subject Table:

```
create table AdminSubject
```

```
(
```

```

AS_id varchar2(20) constraint AdminSubject_id_pk primary key,
Admin_id varchar2(20) constraint AdminSubject_Adminid_fk references Admin(Admin_id),
Subject_id varchar2(20) constraint AdminSubject_Subjectid_fk references
Subject(Subject_id)

```

```
)
```

```

CREATE TABLE ADMINSUBJECT (
    AS_id varchar2(20) constraint AdminSubject_id_pk primary key,
    Admin_id varchar2(20) constraint AdminSubject_Adminid_fk references Admin/Admin_id,
    Subject_id varchar2(20) constraint AdminSubject_Subjectid_fk references Subject/Subject_id
)
desc AdminSubject

```

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ADMINSUBJECT	AS_ID	VARCHAR2	20	-	-	1	✓	-	-
	ADMIN_ID	VARCHAR2	20	-	-	-	✓	-	-
	SUBJECT_ID	VARCHAR2	20	-	-	-	✓	-	-

Fig-19: Admin-Subject Table Create

```

insert into AdminSubject (AS_id, Admin_id, Subject_id) values ('AS001', 'A001', 'Sub001')
insert into AdminSubject (AS_id, Admin_id, Subject_id) values ('AS002', 'A002', 'Sub002')
insert into AdminSubject (AS_id, Admin_id, Subject_id) values ('AS003', 'A003', 'Sub003')
insert into AdminSubject (AS_id, Admin_id, Subject_id) values ('AS004', 'A004', 'Sub004')
insert into AdminSubject (AS_id, Admin_id, Subject_id) values ('AS005', 'A005', 'Sub005')
insert into AdminSubject (AS_id, Admin_id, Subject_id) values ('AS006', 'A006', 'Sub006')
insert into AdminSubject (AS_id, Admin_id, Subject_id) values ('AS007', 'A007', 'Sub007')
insert into AdminSubject (AS_id, Admin_id, Subject_id) values ('AS008', 'A008', 'Sub008')
insert into AdminSubject (AS_id, Admin_id, Subject_id) values ('AS009', 'A009', 'Sub009')
insert into AdminSubject (AS_id, Admin_id, Subject_id) values ('AS010', 'A010', 'Sub010')

```

AS_ID	ADMIN_ID	SUBJECT_ID
AS001	A001	Sub001
AS002	A002	Sub002
AS003	A003	Sub003
AS004	A004	Sub004
AS005	A005	Sub005
AS006	A006	Sub006
AS007	A007	Sub007
AS008	A008	Sub008
AS009	A009	Sub009
AS010	A010	Sub010

Fig-20: Admin-Subject Table Data

11. Teacher Table:

```
create table Teacher
```

```
(
```

```
    Teacher_id varchar2(20) constraint Teacher_Teacherid_pk primary key,  
    Teacher_name varchar2(30) not null,  
    Address varchar2(30),  
    Qualification varchar2(30),  
    Phone_no number(11),  
    Email varchar2(30) unique
```

```
)
```

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
Teacher_id varchar2(20) constraint Teacher_Teacherid_pk primary key,  
Teacher_name varchar2(30) not null,  
Address varchar2(30),  
Qualification varchar2(30),  
Phone_no number(11),  
Email varchar2(30) unique
```

The code is preceded by a brace `)` indicating the end of the table definition. Below the code, the command `desc Teacher` is shown. At the bottom of the SQL window, there are tabs for Results, Explain, Describe, Saved SQL, and History. The Results tab is selected.

Below the SQL window, a table definition for the TEACHER object is displayed:

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
TEACHER	TEACHER_ID	Varchar2	20	-	-	1	-	-	-
TEACHER	TEACHER_NAME	Varchar2	30	-	-	-	-	-	-
ADDRESS	Address	Varchar2	30	-	-	-	✓	-	-
QUALIFICATION	Qualification	Varchar2	30	-	-	-	✓	-	-
PHONE_NO	Phone_no	Number	-	11	0	-	✓	-	-
EMAIL	Email	Varchar2	30	-	-	-	✓	-	-

At the bottom of the interface, it says "Language: en-us" and "Application Express 2.1.0.00.39 Copyright © 1999, 2006, Oracle. All rights reserved." The system tray at the bottom right shows the date and time as 5/17/2024 10:02 PM.

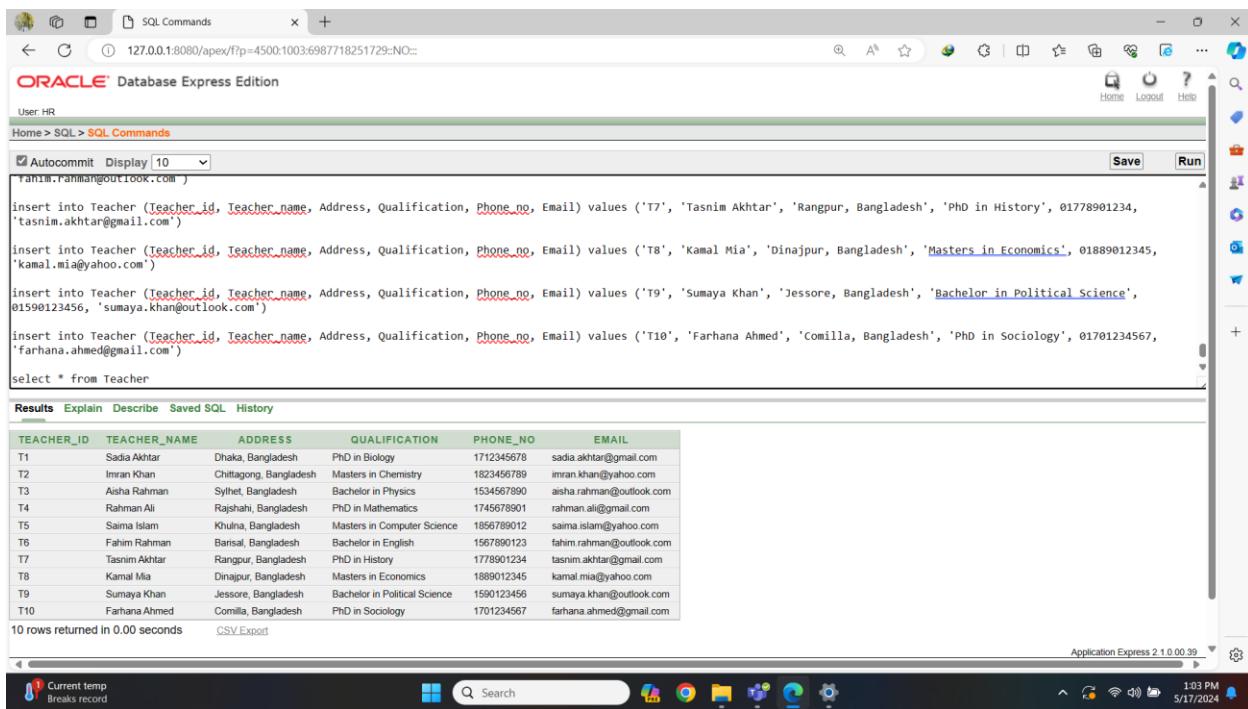
Fig-21: Teacher Table Create

```
insert into Teacher (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email)  
values ('T1', 'Sadia Akhtar', 'Dhaka, Bangladesh', 'PhD in Biology', 01712345678,  
'sadia.akhtar@gmail.com')
```

```
insert into Teacher (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email)  
values ('T2', 'Imran Khan', 'Chittagong, Bangladesh', 'Masters in Chemistry', 01823456789,  
'imran.khan@yahoo.com')
```

```
insert into Teacher (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email)  
values ('T3', 'Aisha Rahman', 'Sylhet, Bangladesh', 'Bachelor in Physics', 01534567890,  
'aisha.rahman@outlook.com')
```

insert into Teacher (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email)
 values ('T4', 'Rahman Ali', 'Rajshahi, Bangladesh', 'PhD in Mathematics', 01745678901,
['rahman.ali@gmail.com'\)](mailto:rahman.ali@gmail.com)
 insert into Teacher (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email)
 values ('T5', 'Saima Islam', 'Khulna, Bangladesh', 'Masters in Computer Science', 01856789012,
['saima.islam@yahoo.com'\)](mailto:saima.islam@yahoo.com)
 insert into Teacher (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email)
 values ('T6', 'Fahim Rahman', 'Barisal, Bangladesh', 'Bachelor in English', 01567890123,
['fahim.rahman@outlook.com'\)](mailto:fahim.rahman@outlook.com)
 insert into Teacher (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email)
 values ('T7', 'Tasnim Akhtar', 'Rangpur, Bangladesh', 'PhD in History', 01778901234,
['tasnim.akhtar@gmail.com'\)](mailto:tasnim.akhtar@gmail.com)
 insert into Teacher (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email)
 values ('T8', 'Kamal Mia', 'Dinajpur, Bangladesh', 'Masters in Economics', 01889012345,
['kamal.mia@yahoo.com'\)](mailto:kamal.mia@yahoo.com)
 insert into Teacher (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email)
 values ('T9', 'Sumaya Khan', 'Jessore, Bangladesh', 'Bachelor in Political Science', 01590123456,
['sumaya.khan@outlook.com'\)](mailto:sumaya.khan@outlook.com)
 insert into Teacher (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email)
 values ('T10', 'Farhana Ahmed', 'Comilla, Bangladesh', 'PhD in Sociology', 01701234567,
['farhana.ahmed@gmail.com'\)](mailto:farhana.ahmed@gmail.com)



The screenshot shows the Oracle Database Express Edition interface. The SQL Commands window displays the following SQL code and its execution results:

```

SQL Commands
127.0.0.1:8080/apex/f?p=4500:1003:6987718251729::NO::
ORACLE Database Express Edition
User HR
Home > SQL > SQL Commands
Autocommit Display | 10 | Save | Run
Tahim.Rahman@outlook.com

insert into Teacher (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email) values ('T7', 'Tasnim Akhtar', 'Rangpur, Bangladesh', 'PhD in History', 01778901234, 'tasnim.akhtar@gmail.com')
insert into Teacher (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email) values ('T8', 'Kamal Mia', 'Dinajpur, Bangladesh', 'Masters in Economics', 01889012345, 'kamal.mia@yahoo.com')
insert into Teacher (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email) values ('T9', 'Sumaya Khan', 'Jessore, Bangladesh', 'Bachelor in Political Science', 01590123456, 'sumaya.khan@outlook.com')
insert into Teacher (Teacher_id, Teacher_name, Address, Qualification, Phone_no, Email) values ('T10', 'Farhana Ahmed', 'Comilla, Bangladesh', 'PhD in Sociology', 01701234567, 'farhana.ahmed@gmail.com')
select * from Teacher

```

The Results tab shows the data inserted into the Teacher table:

TEACHER_ID	TEACHER_NAME	ADDRESS	QUALIFICATION	PHONE_NO	EMAIL
T1	Sadia Akhtar	Dhaka, Bangladesh	PhD in Biology	1712345678	sadia.akhtar@gmail.com
T2	Imran Khan	Chittagong, Bangladesh	Masters in Chemistry	1823456789	imran.khan@yahoo.com
T3	Aisha Rahman	Sylhet, Bangladesh	Bachelor in Physics	1534567890	aisha.rahman@outlook.com
T4	Rahman Ali	Rajshahi, Bangladesh	PhD in Mathematics	1745678901	rahman.ali@gmail.com
T5	Saima Islam	Khulna, Bangladesh	Masters in Computer Science	1856789012	saima.islam@yahoo.com
T6	Fahim Rahman	Barisal, Bangladesh	Bachelor in English	1567890123	fahim.rahman@outlook.com
T7	Tasnim Akhtar	Rangpur, Bangladesh	PhD in History	1778901234	tasnim.akhtar@gmail.com
T8	Kamal Mia	Dinajpur, Bangladesh	Masters in Economics	1889012345	kamal.mia@yahoo.com
T9	Sumaya Khan	Jessore, Bangladesh	Bachelor in Political Science	1590123456	sumaya.khan@outlook.com
T10	Farhana Ahmed	Comilla, Bangladesh	PhD in Sociology	1701234567	farhana.ahmed@gmail.com

10 rows returned in 0.00 seconds [CSV Export](#)

Application Express 2.1.0.00.39

Fig-22: Teacher Table Data

12. Teacher Subject Table:

```
create table TeacherSubject
```

```
(
```

```
    TS_id varchar2(20) constraint TeacherSubject_TSid_pk primary key,  
    Teacher_id varchar2(20) constraint TeacherSubject_Tid_fk references Teacher(Teacher_id),  
    Subject_id varchar2(20) constraint TeacherSubject_Sid_fk references Subject(Subject_id)
```

```
)
```

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The SQL editor window contains the following code:

```
select * from Teacher  
create table TeacherSubject  
(  
    TS_id varchar2(20) constraint TeacherSubject_TSid_pk primary key,  
    Teacher_id varchar2(20) constraint TeacherSubject_Tid_fk references Teacher(Teacher_id),  
    Subject_id varchar2(20) constraint TeacherSubject_Sid_fk references Subject(Subject_id)  
)  
desc TeacherSubject
```

Below the code, the 'Results' tab is selected, showing the table definition:

Object Type	Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
TABLE Object	TEACHERSUBJECT	TS_ID	Varchar2	20	-	-	1	-	-	-
		TEACHER_ID	Varchar2	20	-	-	-	✓	-	-
		SUBJECT_ID	Varchar2	20	-	-	-	✓	-	-

At the bottom of the interface, there is a status bar with the following information:

Language: en-us Application Express 2.1.0.00.39
Copyright © 1999, 2006, Oracle. All rights reserved.

System status icons: 100°F Mostly sunny, Search, Home, Logout, Help, etc.

Fig-23: Teacher-Subject Table Create

```
insert into TeacherSubject (TS_id, Teacher_id, Subject_id) values ('TS1', 'T1', 'Sub001')  
insert into TeacherSubject (TS_id, Teacher_id, Subject_id) values ('TS2', 'T2', 'Sub002')  
insert into TeacherSubject (TS_id, Teacher_id, Subject_id) values ('TS3', 'T3', 'Sub003')  
insert into TeacherSubject (TS_id, Teacher_id, Subject_id) values ('TS4', 'T4', 'Sub004')  
insert into TeacherSubject (TS_id, Teacher_id, Subject_id) values ('TS5', 'T5', 'Sub005')  
insert into TeacherSubject (TS_id, Teacher_id, Subject_id) values ('TS6', 'T6', 'Sub006')  
insert into TeacherSubject (TS_id, Teacher_id, Subject_id) values ('TS7', 'T7', 'Sub007')  
insert into TeacherSubject (TS_id, Teacher_id, Subject_id) values ('TS8', 'T8', 'Sub008')  
insert into TeacherSubject (TS_id, Teacher_id, Subject_id) values ('TS9', 'T9', 'Sub009')  
insert into TeacherSubject (TS_id, Teacher_id, Subject_id) values ('TS10', 'T10', 'Sub010')
```

```

SQL Commands
127.0.0.1:8080/apex/f?p=4500:1003:6987718251729::NO:::
ORACLE Database Express Edition
User: HR
Home > SQL > SQL Commands
Autocommit Display 10 Save Run
Insert into TEACHERSUBJECT (TS_ID, Teacher_id, Subject_id) values ('TS4', 'T4', 'Sub004')
insert into TeacherSubject (TS_id, Teacher_id, Subject_id) values ('TS5', 'T5', 'Sub005')
insert into TeacherSubject (TS_id, Teacher_id, Subject_id) values ('TS6', 'T6', 'Sub006')
insert into TeacherSubject (TS_id, Teacher_id, Subject_id) values ('TS7', 'T7', 'Sub007')
insert into TeacherSubject (TS_id, Teacher_id, Subject_id) values ('TS8', 'T8', 'Sub008')
insert into TeacherSubject (TS_id, Teacher_id, Subject_id) values ('TS9', 'T9', 'Sub009')
insert into TeacherSubject (TS_id, Teacher_id, Subject_id) values ('TS10', 'T10', 'Sub010')
select * from TeacherSubject
Results Explain Describe Saved SQL History
TS_ID TEACHER_ID SUBJECT_ID
TS1 T1 Sub001
TS2 T2 Sub002
TS3 T3 Sub003
TS4 T4 Sub004
TS5 T5 Sub005
TS6 T6 Sub006
TS7 T7 Sub007
TS8 T8 Sub008
TS9 T9 Sub009
TS10 T10 Sub010
10 rows returned in 0.00 seconds CSV Export
Application Express 2.1.0.0.39 10:04 PM 5/17/2024

```

Fig-24: Teacher-Subject Table Data

13. Teacher Admin Table:

create table TeacherAdmin

(

TA_id varchar2(20) constraint TeacherAdmin_Tid_pk primary key,
 Teacher_id varchar2(20) constraint TeacherAdmin_Tid_fk references Teacher(Teacher_id),
 Admin_id varchar2(20) constraint TeacherAdmin_Aid_fk references Admin(Admin_id)

)

```

SQL Commands
127.0.0.1:8080/apex/f?p=4500:1003:6987718251729::NO:::
ORACLE Database Express Edition
User: HR
Home > SQL > SQL Commands
Autocommit Display 10 Save Run
Select * from TEACHERSUBJECT
create table TeacherAdmin
(
    TA_id varchar2(20) constraint TeacherAdmin_Tid_pk primary key,
    Teacher_id varchar2(20) constraint TeacherAdmin_Tid_fk references Teacher(Teacher_id),
    Admin_id varchar2(20) constraint TeacherAdmin_Aid_fk references Admin(Admin_id)
)
desc TeacherAdmin
Results Explain Describe Saved SQL History
Object Type TABLE Object TEACHERADMIN
Table Column Data Type Length Precision Scale Primary Key Nullable Default Comment
TEACHERADMIN TA_ID Varchar2 20 - - 1 - - -
TEACHERADMIN TEACHER_ID Varchar2 20 - - - ✓ - -
TEACHERADMIN ADMIN_ID Varchar2 20 - - - ✓ - -
1 - 3
Language: en-us Application Express 2.1.0.0.39 Copyright © 1999, 2006, Oracle. All rights reserved.
Air Poor Now
10:05 PM 5/17/2024

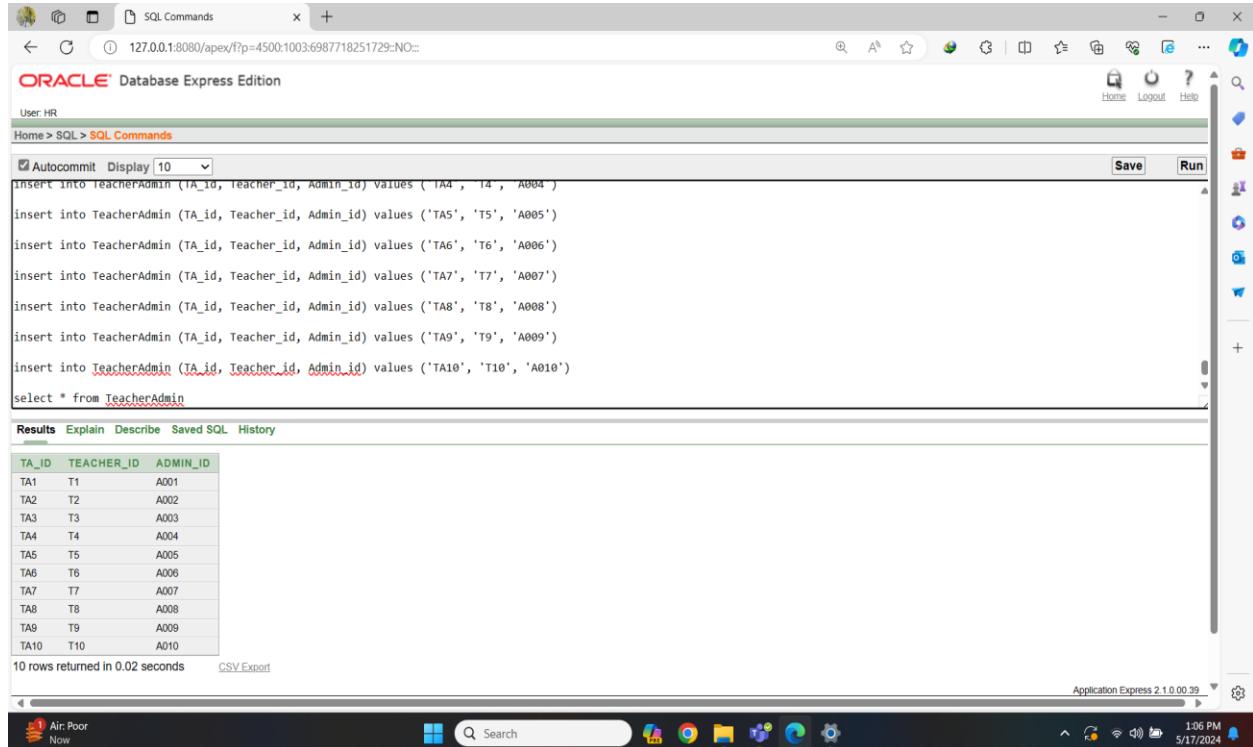
```

Fig-25: Teacher-Admin Table Create

```

insert into TeacherAdmin (TA_id, Teacher_id, Admin_id) values ('TA1', 'T1', 'A001')
insert into TeacherAdmin (TA_id, Teacher_id, Admin_id) values ('TA2', 'T2', 'A002')
insert into TeacherAdmin (TA_id, Teacher_id, Admin_id) values ('TA3', 'T3', 'A003')
insert into TeacherAdmin (TA_id, Teacher_id, Admin_id) values ('TA4', 'T4', 'A004')
insert into TeacherAdmin (TA_id, Teacher_id, Admin_id) values ('TA5', 'T5', 'A005')
insert into TeacherAdmin (TA_id, Teacher_id, Admin_id) values ('TA6', 'T6', 'A006')
insert into TeacherAdmin (TA_id, Teacher_id, Admin_id) values ('TA7', 'T7', 'A007')
insert into TeacherAdmin (TA_id, Teacher_id, Admin_id) values ('TA8', 'T8', 'A008')
insert into TeacherAdmin (TA_id, Teacher_id, Admin_id) values ('TA9', 'T9', 'A009')
insert into TeacherAdmin (TA_id, Teacher_id, Admin_id) values ('TA10', 'T10', 'A010')

```



The screenshot shows the Oracle Database Express Edition interface. The SQL Commands page displays the following SQL code:

```

insert into TeacherAdmin (TA_id, Teacher_id, Admin_id) values ('TA4', 'T4', 'A004')
insert into TeacherAdmin (TA_id, Teacher_id, Admin_id) values ('TA5', 'T5', 'A005')
insert into TeacherAdmin (TA_id, Teacher_id, Admin_id) values ('TA6', 'T6', 'A006')
insert into TeacherAdmin (TA_id, Teacher_id, Admin_id) values ('TA7', 'T7', 'A007')
insert into TeacherAdmin (TA_id, Teacher_id, Admin_id) values ('TA8', 'T8', 'A008')
insert into TeacherAdmin (TA_id, Teacher_id, Admin_id) values ('TA9', 'T9', 'A009')
insert into TeacherAdmin (TA_id, Teacher_id, Admin_id) values ('TA10', 'T10', 'A010')
select * from TeacherAdmin

```

The results section shows a grid of data:

TA_ID	TEACHER_ID	ADMIN_ID
TA1	T1	A001
TA2	T2	A002
TA3	T3	A003
TA4	T4	A004
TA5	T5	A005
TA6	T6	A006
TA7	T7	A007
TA8	T8	A008
TA9	T9	A009
TA10	T10	A010

10 rows returned in 0.02 seconds

Fig-26: Teacher-Admin Table Data

14. Room Table:

```
create table Room
```

```
(
```

```
    Room_no varchar2(20) constraint Room_Roomno_pk primary key,  
    Class_name varchar2(30) not null
```

```
)
```

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The SQL editor window contains the following code:

```
insert into TeacherAdmin (TA_Id, Teacher_id, Admin_id) values ('TA10', 'T10', 'A010')  
select * from TeacherAdmin  
create table Room  
(  
    Room_no varchar2(20) constraint Room_Roomno_pk primary key,  
    Class_name varchar2(30) not null  
)  
desc Room
```

Below the code, there is a table titled "Object Type TABLE Object ROOM" showing the structure of the newly created table:

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ROOM	ROOM_NO	VARCHAR2	20	-	-	1	-	-	-
	CLASS_NAME	VARCHAR2	30	-	-	-	-	-	-

At the bottom of the interface, it says "Language: en-us" and "Copyright © 1999, 2006, Oracle. All rights reserved." The system status bar at the bottom right shows "Application Express 2.1.0.0.39", "106 PM", "5/17/2024", and a battery icon.

Fig-27: Room Table Create

```
insert into Room (Room_no, Class_name) values ('R1', 'class 1A')  
insert into Room (Room_no, Class_name) values ('R2', 'class 1B')  
insert into Room (Room_no, Class_name) values ('R3', 'class 2A')  
insert into Room (Room_no, Class_name) values ('R4', 'class 2B')  
insert into Room (Room_no, Class_name) values ('R5', 'class 3A')  
insert into Room (Room_no, Class_name) values ('R6', 'class 3B')  
insert into Room (Room_no, Class_name) values ('R7', 'class 4A')  
insert into Room (Room_no, Class_name) values ('R8', 'class 4B')  
insert into Room (Room_no, Class_name) values ('R9', 'class 5A')  
insert into Room (Room_no, Class_name) values ('R10', 'class 5B')
```

```

SQL Commands
ORACLE Database Express Edition
User: HR
Home > SQL > SQL Commands
Autocommit Display 10
Insert into ROOM (ROOM_NO, CLASS_NAME) values ('R1', 'class 1A')
insert into ROOM (ROOM_NO, CLASS_NAME) values ('R2', 'class 1B')
insert into ROOM (ROOM_NO, CLASS_NAME) values ('R3', 'class 2A')
insert into ROOM (ROOM_NO, CLASS_NAME) values ('R4', 'class 2B')
insert into ROOM (ROOM_NO, CLASS_NAME) values ('R5', 'class 3A')
insert into ROOM (ROOM_NO, CLASS_NAME) values ('R6', 'class 3B')
insert into ROOM (ROOM_NO, CLASS_NAME) values ('R7', 'class 4A')
insert into ROOM (ROOM_NO, CLASS_NAME) values ('R8', 'class 4B')
insert into ROOM (ROOM_NO, CLASS_NAME) values ('R9', 'class 5A')
insert into ROOM (ROOM_NO, CLASS_NAME) values ('R10', 'class 5B')
select * from ROOM
Results Explain Describe Saved SQL History
ROOM_NO CLASS_NAME
R1 class 1A
R2 class 1B
R3 class 2A
R4 class 2B
R5 class 3A
R6 class 3B
R7 class 4A
R8 class 4B
R9 class 5A
R10 class 5B
10 rows returned in 0.00 seconds
CSV Export

```

Fig-28: Room Table Data

15. Teacher Room Table:

create table TeacherRoom

(

 TR_id varchar2(20) constraint TeacherRoom_TRid_pk primary key,
 Teacher_id varchar2(20) constraint TeacherRoom_Tid_fk references Teacher(Teacher_id),
 Room_no varchar2(20) constraint TeacherRoom_Roomno_fk references Room(Room_no)

)

```

SQL Commands
ORACLE Database Express Edition
User: HR
Home > SQL > SQL Commands
Autocommit Display 10
select * from ROOM
create table TeacherRoom
(
    TR_id varchar2(20) constraint TeacherRoom_TRid_pk primary key,
    Teacher_id varchar2(20) constraint TeacherRoom_Tid_fk references Teacher(Teacher_id),
    Room_no varchar2(20) constraint TeacherRoom_Roomno_fk references Room(Room_no)
)
desc TeacherRoom
Results Explain Describe Saved SQL History
Object Type TABLE Object TEACHERROOM
Table Column Data Type Length Precision Scale Primary Key Nullable Default Comment
TEACHERROOM TR_ID Varchar2 20 - - 1 - -
TEACHERROOM TEACHER_ID Varchar2 20 - - - ✓ - -
TEACHERROOM ROOM_NO Varchar2 20 - - - ✓ - -
1 - 3

```

Fig-29: Teacher-Room Table Create

```

insert into TeacherRoom (TR_id, Teacher_id, Room_no) values ('TR1', 'T1', 'R1')
insert into TeacherRoom (TR_id, Teacher_id, Room_no) values ('TR2', 'T2', 'R2')
insert into TeacherRoom (TR_id, Teacher_id, Room_no) values ('TR3', 'T3', 'R3')
insert into TeacherRoom (TR_id, Teacher_id, Room_no) values ('TR4', 'T4', 'R4')
insert into TeacherRoom (TR_id, Teacher_id, Room_no) values ('TR5', 'T5', 'R5')
insert into TeacherRoom (TR_id, Teacher_id, Room_no) values ('TR6', 'T6', 'R6')
insert into TeacherRoom (TR_id, Teacher_id, Room_no) values ('TR7', 'T7', 'R7')
insert into TeacherRoom (TR_id, Teacher_id, Room_no) values ('TR8', 'T8', 'R8')
insert into TeacherRoom (TR_id, Teacher_id, Room_no) values ('TR9', 'T9', 'R9')
insert into TeacherRoom (TR_id, Teacher_id, Room_no) values ('TR10', 'T10', 'R10')

```

The screenshot shows the Oracle Database Express Edition interface. The top navigation bar includes 'SQL Commands' and a connection URL '127.0.0.1:8080/apex/f?p=4500:1003:6987718251729::NO::'. The main area has tabs for 'Autocommit' and 'Display 10'. Below is a code editor containing SQL statements for inserting data into the TeacherRoom table and a 'select * from TeacherRoom' query. The 'Results' tab is active, displaying a table with 10 rows of data:

TR_ID	TEACHER_ID	ROOM_NO
TR3	T3	R3
TR4	T4	R4
TR5	T5	R5
TR6	T6	R6
TR7	T7	R7
TR8	T8	R8
TR9	T9	R9
TR10	T10	R10
TR1	T1	R1
TR2	T2	R2

Below the table, it says '10 rows returned in 0.00 seconds' and there is a 'CSV Export' link. The bottom status bar shows 'Application Express 2.1.0.0.39' and the system clock '109 PM 5/17/2024'.

Fig-30: Teacher-Room Table Data

16. Student Room Table:

```
create table StudentRoom
```

```
(
```

```
    SR_id varchar2(20) constraint StudentRoom_SRid_pk primary key,  
    Student_id varchar2(20) constraint StudentRoom_Sid_fk references Student(Student_id),  
    Room_no varchar2(20) constraint StudentRoom_Roomno_fk references Room(Room_no)
```

```
)
```

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
create table StudentRoom  
(  
    SR_id varchar2(20) constraint StudentRoom_SRid_pk primary key,  
    Student_id varchar2(20) constraint StudentRoom_Sid_fk references Student(Student_id),  
    Room_no varchar2(20) constraint StudentRoom_Roomno_fk references Room(Room_no)  
)  
  
desc StudentRoom
```

Below the code, the results of the DESCRIBE command are shown:

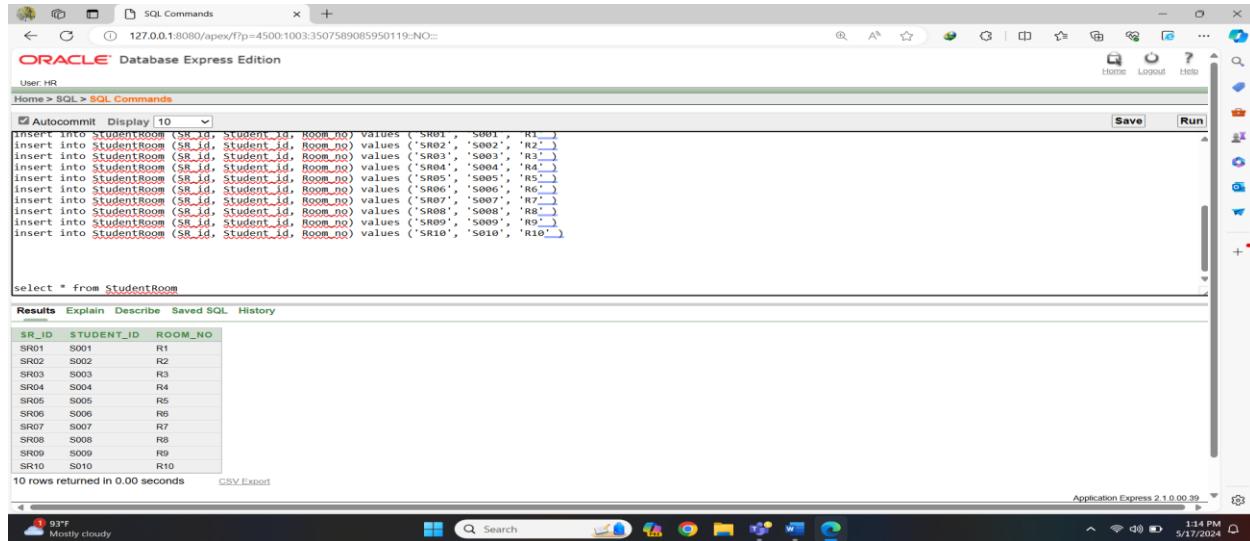
Object Type	Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
STUDENTROOM	SR_ID		VARCHAR2	20	-	-	1	-	-	-
	STUDENT_ID		VARCHAR2	20	-	-	-	✓	-	-
	ROOM_NO		VARCHAR2	20	-	-	-	✓	-	-

At the bottom of the interface, the status bar displays "Language: en-us" and "Application Express 2.1.0.0.39 Copyright © 1999, 2006, Oracle. All rights reserved."

Fig-31: Student-Room Table Create

```
insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR01', 'S001', 'R1')  
insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR02', 'S002', 'R2')  
insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR03', 'S003', 'R3')  
insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR04', 'S004', 'R4')  
insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR05', 'S005', 'R5')  
insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR06', 'S006', 'R6')  
insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR07', 'S007', 'R7')  
insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR08', 'S008', 'R8')
```

insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR09', 'S009', 'R9')
 insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR10', 'S010', 'R10')



The screenshot shows the Oracle Database Express Edition interface. The SQL Commands tab contains the following SQL code:

```

Insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR01', 'S001', 'R1')
Insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR02', 'S002', 'R2')
Insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR03', 'S003', 'R3')
Insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR04', 'S004', 'R4')
Insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR05', 'S005', 'R5')
Insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR06', 'S006', 'R6')
Insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR07', 'S007', 'R7')
Insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR08', 'S008', 'R8')
Insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR09', 'S009', 'R9')
Insert into StudentRoom (SR_id, Student_id, Room_no) values ('SR10', 'S010', 'R10')

select * from StudentRoom
    
```

The Results tab displays the data from the StudentRoom table:

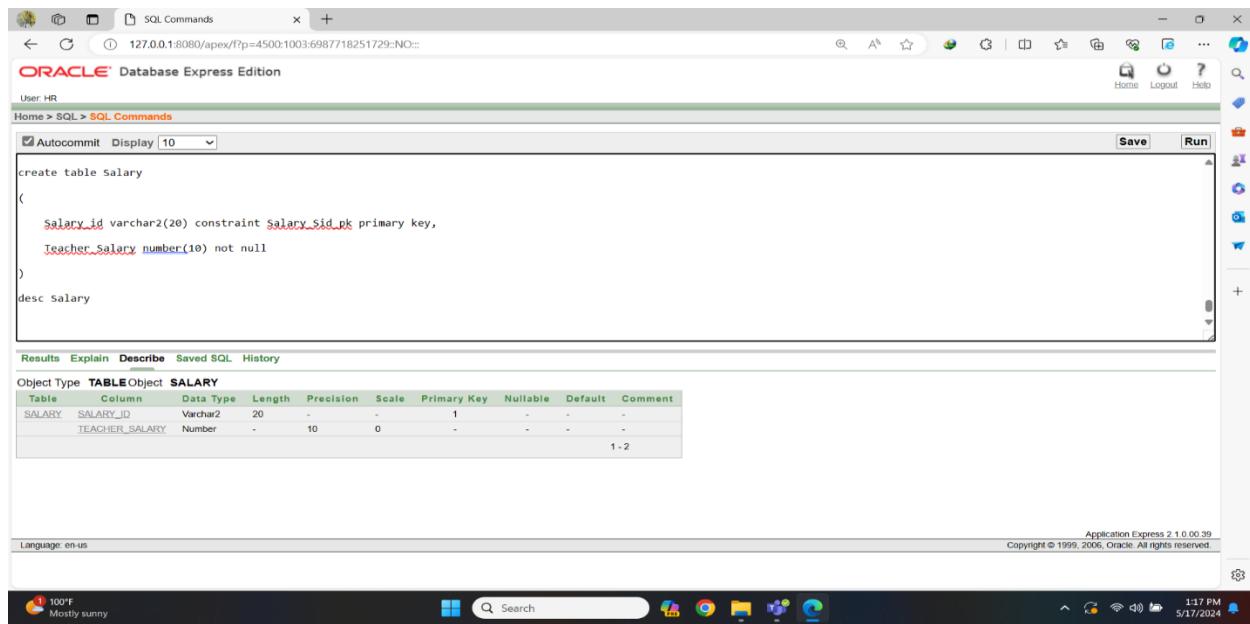
SR_ID	STUDENT_ID	ROOM_NO
SR01	S001	R1
SR02	S002	R2
SR03	S003	R3
SR04	S004	R4
SR05	S005	R5
SR06	S006	R6
SR07	S007	R7
SR08	S008	R8
SR09	S009	R9
SR10	S010	R10

10 rows returned in 0.00 seconds

Fig-32: Student-Room Table Data

17. Salary Table:

create table Salary
 (
 Salary_id varchar2(20) constraint Salary_Sid_pk primary key,
 Teacher_Salary number(10) not null
)



The screenshot shows the Oracle Database Express Edition interface. The SQL Commands tab contains the following SQL code:

```

create table Salary
(
    Salary_id varchar2(20) constraint Salary_Sid_pk primary key,
    Teacher_Salary number(10) not null
)
desc Salary
    
```

The Results tab displays the table structure:

Object Type	TABLE	Object	SALARY						
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
SALARY	Salary_id	Varchar2	20	-	-	1	-	-	-
	Teacher_Salary	Number	-	10	0	-	-	-	-

1 - 2

Language: en-us

Application Express 2.1.0.00.39
 Copyright © 1999, 2006, Oracle. All rights reserved.

Fig-33: Salary Table Create

```

insert into Salary (Salary_id, Teacher_salary) values ('S1', 50000)
insert into Salary (Salary_id, Teacher_salary) values ('S2', 55000)
insert into Salary (Salary_id, Teacher_salary) values ('S3', 60000)
insert into Salary (Salary_id, Teacher_salary) values ('S4', 65000)
insert into Salary (Salary_id, Teacher_salary) values ('S5', 70000)
insert into Salary (Salary_id, Teacher_salary) values ('S6', 75000)
insert into Salary (Salary_id, Teacher_salary) values ('S7', 80000)
insert into Salary (Salary_id, Teacher_salary) values ('S8', 85000)
insert into Salary (Salary_id, Teacher_salary) values ('S9', 90000)
insert into Salary (Salary_id, Teacher_salary) values ('S10', 95000)

```

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The SQL code entered is:

```

insert into Salary (Salary_id, Teacher_salary) values ('S5', 70000)
insert into Salary (Salary_id, Teacher_salary) values ('S6', 75000)
insert into Salary (Salary_id, Teacher_salary) values ('S7', 80000)
insert into Salary (Salary_id, Teacher_salary) values ('S8', 85000)
insert into Salary (Salary_id, Teacher_salary) values ('S9', 90000)
insert into Salary (Salary_id, Teacher_salary) values ('S10', 95000)
select * from Salary

```

The results section displays the data inserted into the Salary table:

SALARY_ID	TEACHER_SALARY
S5	70000
S6	75000
S7	80000
S8	85000
S9	90000
S10	95000
S1	50000
S2	55000

10 rows returned in 0.00 seconds

Fig-34: Salary Table Data

18. Teacher Salary Table:

```
create table TeacherSalary
```

```
(
```

```

TS_id varchar2(20) constraint TeacherSalary_TSid_pk primary key,
Teacher_id varchar2(20) constraint TeacherSalary_Tid_fk references Teacher(Teacher_id),
Salary_id varchar2(20) constraint TeacherSalary_Sid_fk references Salary(Salary_id)
)
```

The screenshot shows the Oracle Database Express Edition SQL Commands interface. In the SQL editor, the following SQL code is entered:

```
create table TeacherSalary
(
    TS_id varchar2(20) constraint Teachersalary_Tsid_pk primary key,
    Teacher_id varchar2(20) constraint Teachersalary_Tid_fk references Teacher(Teacher_id),
    Salary_id varchar2(20) constraint Teachersalary_Sid_fk references Salary(Salary_id)
)
desc TeacherSalary
```

Below the code, the results of the DESCRIBE command are displayed, showing the structure of the TeacherSalary table:

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
TEACHERSALARY	TS_ID	VARCHAR2	20	-	-	1	-	-	
	TEACHER_ID	VARCHAR2	20	-	-	-	✓	-	
	SALARY_ID	VARCHAR2	20	-	-	-	✓	-	

At the bottom right of the window, it says "Application Express 2.1.0.00.39" and "Copyright © 1999, 2006, Oracle. All rights reserved."

Fig-35: Teacher-Salary Table Create

insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS1', 'T1', 'S1')
 insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS2', 'T2', 'S2')
 insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS3', 'T3', 'S3')
 insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS4', 'T4', 'S4')
 insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS5', 'T5', 'S5')
 insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS6', 'T6', 'S6')
 insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS7', 'T7', 'S7')
 insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS8', 'T8', 'S8')
 insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS9', 'T9', 'S9')
 insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS10', 'T10', 'S10')

The screenshot shows the Oracle Database Express Edition SQL Commands interface. In the SQL editor, the following SQL code is entered:

```
insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS1', 'T1', 'S1')
insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS2', 'T2', 'S2')
insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS3', 'T3', 'S3')
insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS4', 'T4', 'S4')
insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS5', 'T5', 'S5')
insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS6', 'T6', 'S6')
insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS7', 'T7', 'S7')
insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS8', 'T8', 'S8')
insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS9', 'T9', 'S9')
insert into TeacherSalary (TS_id, Teacher_id, Salary_id) values ('TS10', 'T10', 'S10')
select * from TeacherSalary
```

Below the code, the results of the SELECT * FROM TeacherSalary command are displayed, showing the data inserted:

TS_ID	TEACHER_ID	SALARY_ID
TS1	T1	S1
TS2	T2	S2
TS3	T3	S3
TS4	T4	S4
TS5	T5	S5
TS6	T6	S6
TS7	T7	S7
TS8	T8	S8
TS9	T9	S9
TS10	T10	S10

At the bottom right of the window, it says "Application Express 2.1.0.00.39" and "Copyright © 1999, 2006, Oracle. All rights reserved."

Fig-36: Teacher-Salary Table Data

19. Payment Table:

```
create table Payment
```

```
(
```

```
    Payment_id varchar2(20) constraint Payment_Paymentid_pk primary key,  
    Student_fee number(10) not null
```

```
)
```

The screenshot shows the Oracle Database Express Edition SQL Commands interface. The SQL editor window contains the following code:

```
create table Payment
(
    Payment_id varchar2(20) constraint Payment_Paymentid_pk primary key,
    Student_fee number(10) not null
)

desc Payment
```

Below the code, there is a table object definition for the PAYMENT table:

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
PAYMENT	PAYMENT_ID	VARCHAR2	20	-	-	1	-	-	-
	STUDENT_FEE	NUMBER	-	10	0	-	-	-	-

The status bar at the bottom indicates the application version and copyright information.

Fig-37: Payment Table Create

```
insert into Payment (Payment_id, Student_fee) values ('P1', 5000)
insert into Payment (Payment_id, Student_fee) values ('P2', 5500)
insert into Payment (Payment_id, Student_fee) values ('P3', 6000)
insert into Payment (Payment_id, Student_fee) values ('P4', 6500)
insert into Payment (Payment_id, Student_fee) values ('P5', 7000)
insert into Payment (Payment_id, Student_fee) values ('P6', 7500)
insert into Payment (Payment_id, Student_fee) values ('P7', 8000)
insert into Payment (Payment_id, Student_fee) values ('P8', 8500)
insert into Payment (Payment_id, Student_fee) values ('P9', 9000)
insert into Payment (Payment_id, Student_fee) values ('P10', 9500)
```

```

SQL Commands
ORACLE Database Express Edition
User HR
Home > SQL > SQL Commands
Autocommit Display | 10 | Save | Run
Insert into Payment (Payment_id, Student_fee) Values ('P4', 6500)
insert into Payment (Payment_id, Student_fee) values ('P5', 7000)
insert into Payment (Payment_id, Student_fee) values ('P6', 7500)
insert into Payment (Payment_id, Student_fee) values ('P7', 8000)
insert into Payment (Payment_id, Student_fee) values ('P8', 8500)
insert into Payment (Payment_id, Student_fee) values ('P9', 9000)
insert into Payment (Payment_id, Student_fee) values ('P10', 9500)
select * from Payment
Results Explain Describe Saved SQL History
PAYMENT_ID STUDENT_FEE
P3          6000
P4          6500
P5          7000
P6          7500
P7          8000
P8          8500
P9          9000
P10         9500
P1          5000
P2          5500
10 rows returned in 0.00 seconds
CSV Export

```

Fig-38: Payment Table Data

20. Guardian Payment Table:

create table GuardianPayment

(

```

GP_id varchar2(20) constraint GuardianPayment_GPid_pk primary key,
Guardian_id varchar2(20) constraint GuardianPayment_Gid_fk references
Guardian(Guardian_id),
Payment_id varchar2(20) constraint GuardianPayment_Pid_fk references
Payment(Payment_id)
)
```

```

SQL Commands
ORACLE Database Express Edition
User HR
Home > SQL > SQL Commands
Autocommit Display | 10 | Save | Run
Select * from Payment
create table GuardianPayment
(
    GP_id varchar2(20) constraint GuardianPayment_GPid_pk primary key,
    Guardian_id varchar2(20) constraint GuardianPayment_Gid_fk references Guardian(Guardian_id),
    Payment_id varchar2(20) constraint GuardianPayment_Pid_fk references Payment(Payment_id)
)
desc GuardianPayment
Results Explain Describe Saved SQL History
Object Type TABLE Object GUARDIANPAYMENT
Table Column Data Type Length Precision Scale Primary Key Nullable Default Comment
GUARDIANPAYMENT GP_ID Varchar2 20 - - 1 - -
GUARDIAN_ID Varchar2 20 - - - ✓ - -
PAYMENT_ID Varchar2 20 - - - ✓ - -

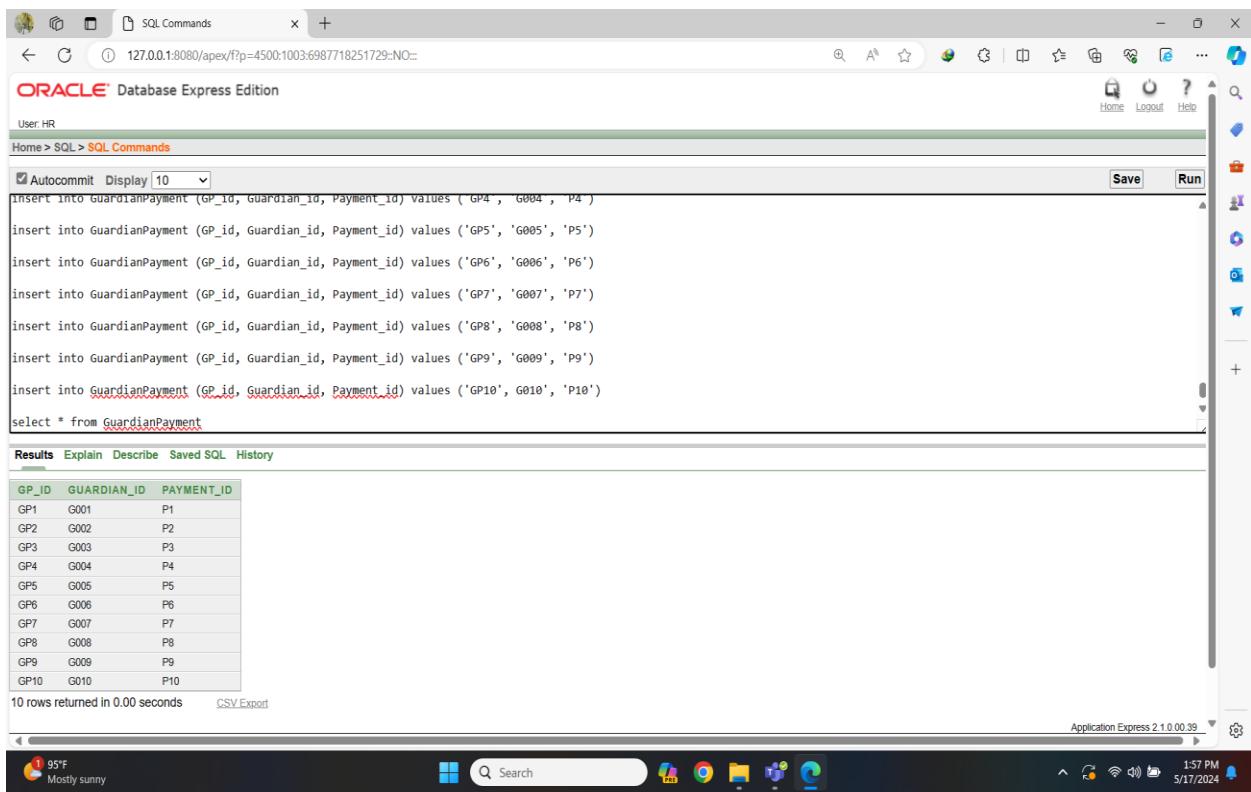
```

Fig-39: Guardian-Payment Table Create

```

insert into GuardianPayment (GP_id, Guardian_id, Payment_id) values ('GP1', 'G001', 'P1')
insert into GuardianPayment (GP_id, Guardian_id, Payment_id) values ('GP2', 'G002', 'P2')
insert into GuardianPayment (GP_id, Guardian_id, Payment_id) values ('GP3', 'G003', 'P3')
insert into GuardianPayment (GP_id, Guardian_id, Payment_id) values ('GP4', 'G004', 'P4')
insert into GuardianPayment (GP_id, Guardian_id, Payment_id) values ('GP5', 'G005', 'P5')
insert into GuardianPayment (GP_id, Guardian_id, Payment_id) values ('GP6', 'G006', 'P6')
insert into GuardianPayment (GP_id, Guardian_id, Payment_id) values ('GP7', 'G007', 'P7')
insert into GuardianPayment (GP_id, Guardian_id, Payment_id) values ('GP8', 'G008', 'P8')
insert into GuardianPayment (GP_id, Guardian_id, Payment_id) values ('GP9', 'G009', 'P9')
insert into GuardianPayment (GP_id, Guardian_id, Payment_id) values ('GP10', 'G010', 'P10')

```



The screenshot shows the Oracle Database Express Edition SQL Commands interface. The URL is 127.0.0.1:8080/apex/f?p=4500:1003:6987718251729::NO:::. The page title is ORACLE Database Express Edition. The user is HR. The main content area contains the following SQL code:

```

Insert Into GuardianPayment (GP_id, Guardian_id, Payment_id) Values ('GP4', 'G004', 'P4')
insert into GuardianPayment (GP_id, Guardian_id, Payment_id) values ('GP5', 'G005', 'P5')
insert into GuardianPayment (GP_id, Guardian_id, Payment_id) values ('GP6', 'G006', 'P6')
insert into GuardianPayment (GP_id, Guardian_id, Payment_id) values ('GP7', 'G007', 'P7')
insert into GuardianPayment (GP_id, Guardian_id, Payment_id) values ('GP8', 'G008', 'P8')
insert into GuardianPayment (GP_id, Guardian_id, Payment_id) values ('GP9', 'G009', 'P9')
insert into GuardianPayment (GP_id, Guardian_id, Payment_id) values ('GP10', 'G010', 'P10')
select * from GuardianPayment

```

The results section shows the following data:

GP_ID	GUARDIAN_ID	PAYMENT_ID
GP1	G001	P1
GP2	G002	P2
GP3	G003	P3
GP4	G004	P4
GP5	G005	P5
GP6	G006	P6
GP7	G007	P7
GP8	G008	P8
GP9	G009	P9
GP10	G010	P10

10 rows returned in 0.00 seconds [CSV Export](#)

The status bar at the bottom shows the weather as 95°F Mostly sunny, the time as 1:57 PM, and the date as 5/17/2024.

Fig-40: Guardian-Payment Table Data

Query Test

A) Conditional Statement:

1. Display the names of all students whose names have 'a' as the second character.
⇒ select Student_name from Student where Student_name like '_a%'

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL query is entered:

```
select Student_name from Student where Student_name like '_a%'
```

The results pane displays the following data:

STUDENT_NAME
Fatima Begum
Sadia Akter
Farhan Ahmed
Nazia Islam

4 rows returned in 0.00 seconds

Language: en-us

Application Express 2.1.0.00.39
Copyright © 1999, 2006, Oracle. All rights reserved.

Fig-1: Conditional Statement Query

2. Display all information for teachers who earn more than 50,000.
⇒ select * from Salary where Teacher_salary > 50000

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL queries are entered:

```
select Student_name from Student where Student_name like '_a%'
```



```
select * from Salary where Teacher_salary > 50000
```

The results pane displays the following data:

SALARY_ID	TEACHER_SALARY
S3	60000
S4	65000
S5	70000
S6	75000
S7	80000
S8	85000
S9	90000
S10	95000
S2	55000

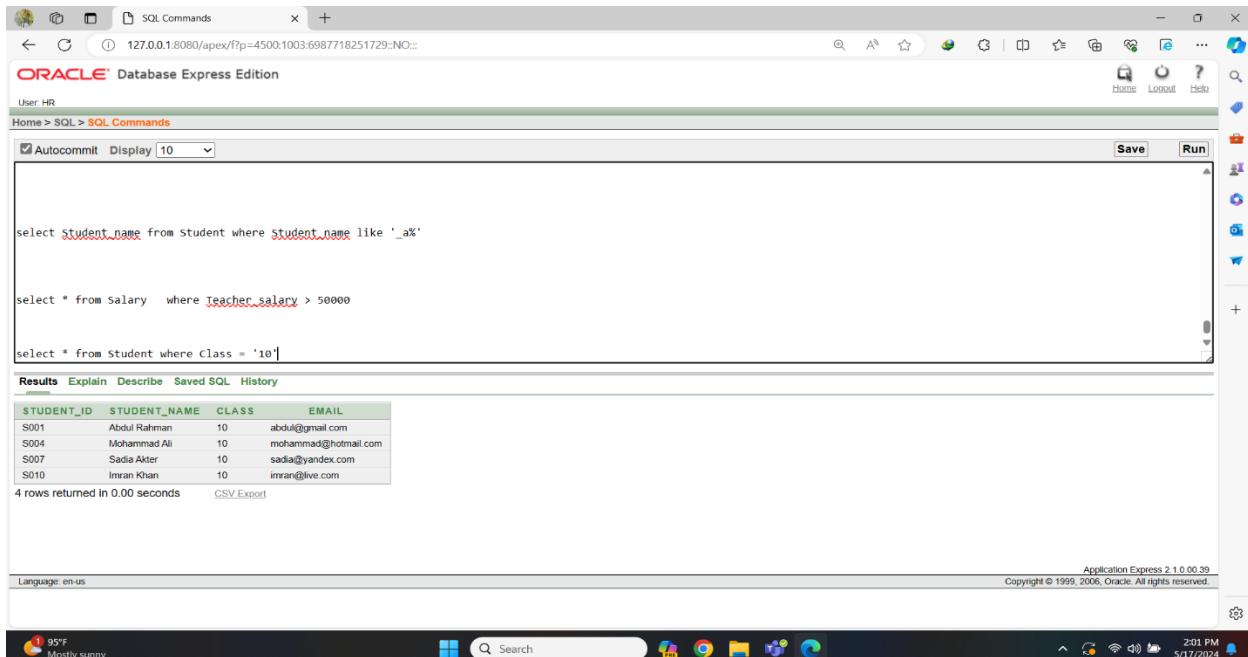
9 rows returned in 0.01 seconds

Language: en-us

Application Express 2.1.0.00.39
Copyright © 1999, 2006, Oracle. All rights reserved.

Fig-2: Conditional Statement Query

3. Display all information for students who are in class '10'.
 ⇒ select * from Student where Class = '10'



The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
select student_name from student where student_name like '_a%'

select * from salary where teacher_salary > 50000

select * from student where class = '10'.
```

Below the code, the results are displayed in a table:

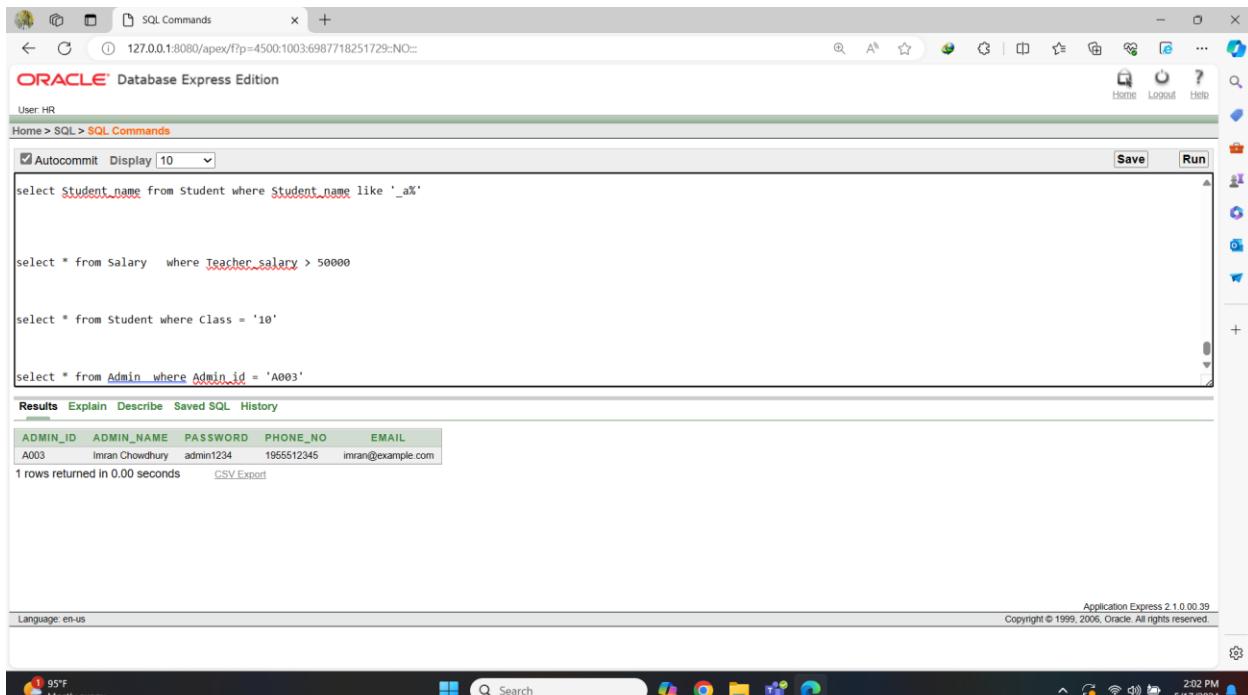
STUDENT_ID	STUDENT_NAME	CLASS	EMAIL
S001	Abdul Rahman	10	abdul@gmail.com
S004	Mohammad Ali	10	mohammad@hotmail.com
S007	Sadia Akter	10	sadia@yandex.com
S010	Imran Khan	10	imran@live.com

4 rows returned in 0.00 seconds [CSV Export](#)

Language: en-us Application Express 2.1.0.00.39 Copyright © 1999, 2006, Oracle. All rights reserved.

Fig-3: Conditional Statement Query

4. Display all information for the administrator with the Admin_id 'A003'.
 ⇒ select * from Admin where Admin_id = 'A003'



The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
select student_name from student where student_name like '_a%'

select * from salary where teacher_salary > 50000

select * from student where class = '10'

select * from admin where admin_id = 'A003'.
```

Below the code, the results are displayed in a table:

ADMIN_ID	ADMIN_NAME	PASSWORD	PHONE_NO	EMAIL
A003	Imran Chowdhury	admin1234	1955512345	imran@example.com

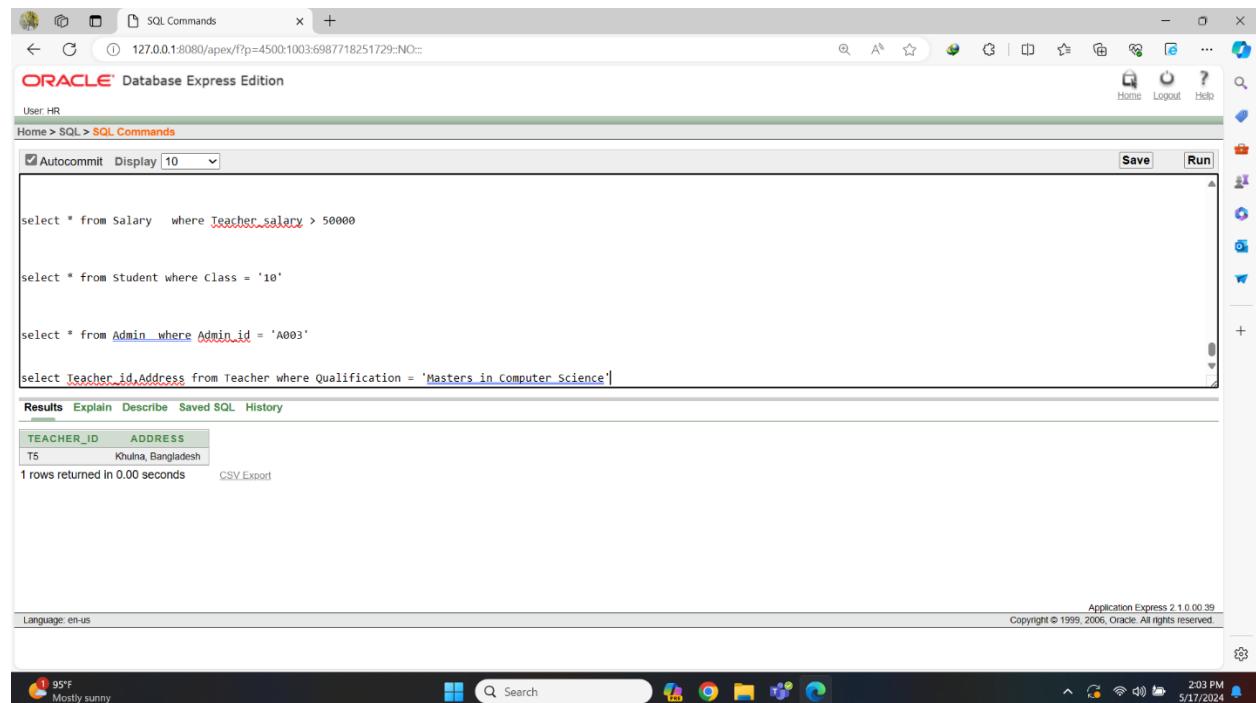
1 rows returned in 0.00 seconds [CSV Export](#)

Language: en-us Application Express 2.1.0.00.39 Copyright © 1999, 2006, Oracle. All rights reserved.

Fig-4: Conditional Statement Query

5. Display the Teacher_id and Address of teachers who have a qualification of 'Masters in Computer Science'.

⇒ select Teacher_id,Address from Teacher where Qualification = 'Masters in Computer Science'



The screenshot shows a Windows desktop environment with a taskbar at the bottom. The taskbar includes icons for weather (95°F Mostly sunny), search, and various system applications. The main window is titled "SQL Commands" and is part of the "ORACLE Database Express Edition" interface. The URL in the address bar is "127.0.0.1:8080/apex/f?p=4500:1003:6987718251729::NO::". The SQL command window contains the following code:

```
select * from Salary where Teacher_salary > 50000  
  
select * from Student where class = '10'  
  
select * from Admin where Admin_id = 'A003'  
  
select Teacher_id,Address from Teacher where Qualification = 'Masters in Computer Science'|
```

The results grid below the command window shows one row of data:

TEACHER_ID	ADDRESS
T5	Khulna, Bangladesh

1 rows returned in 0.00 seconds [CSV Export](#)

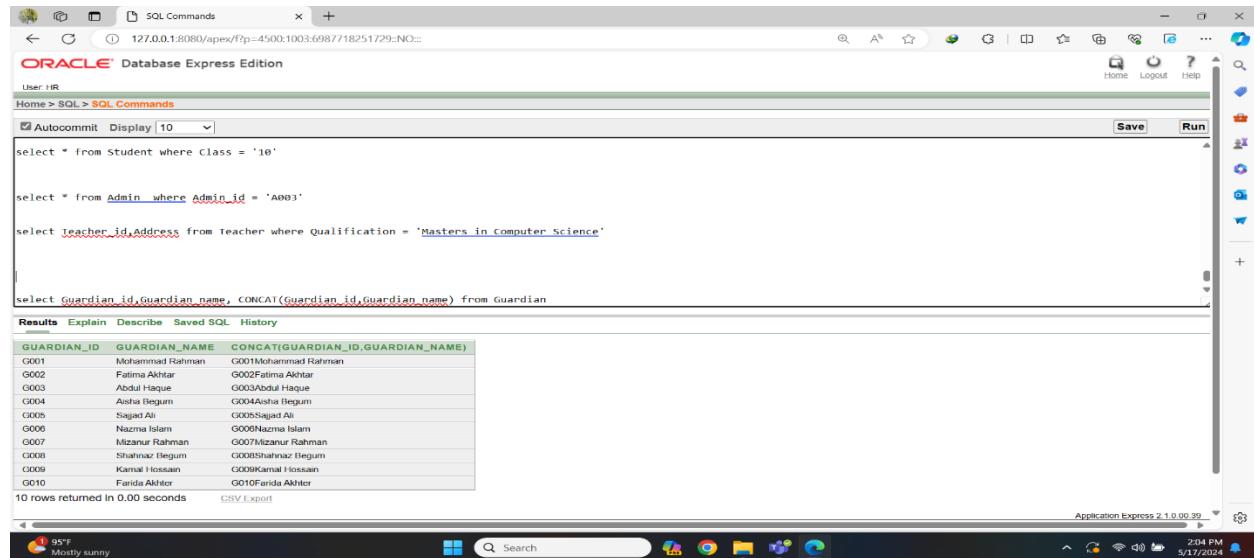
At the bottom right of the application window, it says "Application Express 2.1.0.0.39" and "Copyright © 1999, 2006, Oracle. All rights reserved."

Fig-5: Conditional Statement Query

B) Single row function:

1. Display the Guardian_id, Guardian_name and a concatenated string of Guardian_id and Guardian_name for all guardians.

⇒ select Guardian_id,Guardian_name, CONCAT(Guardian_id,Guardian_name) from Guardian



```
select * from Student where Class = '10'

select * from Admin where Admin_id = 'A003'

select Teacher_id,Address from Teacher where Qualification = 'Masters in Computer Science'

select Guardian_id,Guardian_name, CONCAT(Guardian_id,Guardian_name) from Guardian
```

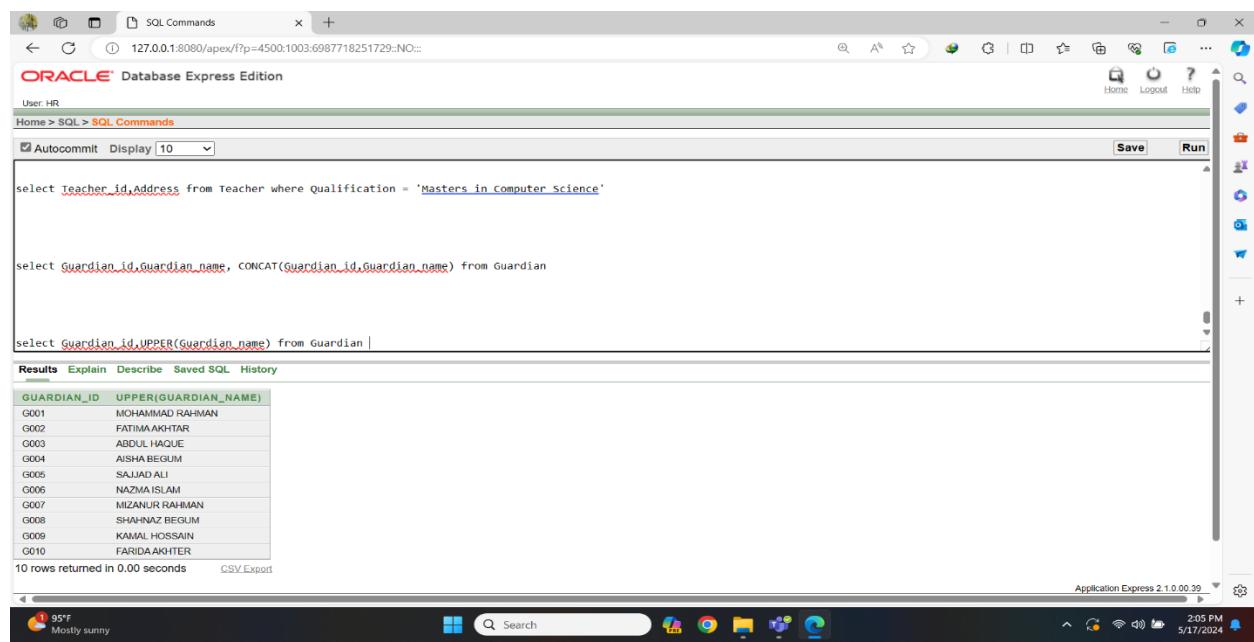
GUARDIAN_ID	GUARDIAN_NAME	CONCAT(GUARDIAN_ID, GUARDIAN_NAME)
G001	Mohammed Rahman	G001Mohammed Rahman
G002	Fatima Akhtar	G002Fatima Akhtar
G003	Abdul Haque	G003Abdul Haque
G004	Aisha Begum	G004Aisha Begum
G005	Sajad Ali	G005Sajad Ali
G006	Nazma Islam	G006Nazma Islam
G007	Mizanur Rahman	G007Mizanur Rahman
G008	Shahnaz Begum	G008Shahnaz Begum
G009	Kamal Hossain	G009Kamal Hossain
G010	Farida Akhter	G010Farida Akhter

10 rows returned in 0.00 seconds CSV Export Application Express 2.1.0.0.39 2:04 PM 5/17/2024

Fig-6: Single Row Function Query

2. Display the Guardian_id and Guardian_name in uppercase for all guardians.

⇒ select Guardian_id,UPPER(Guardian_name) from Guardian



```
select Teacher_id,Address from Teacher where Qualification = 'Masters in Computer Science'

select Guardian_id,Guardian_name, CONCAT(Guardian_id,Guardian_name) from Guardian

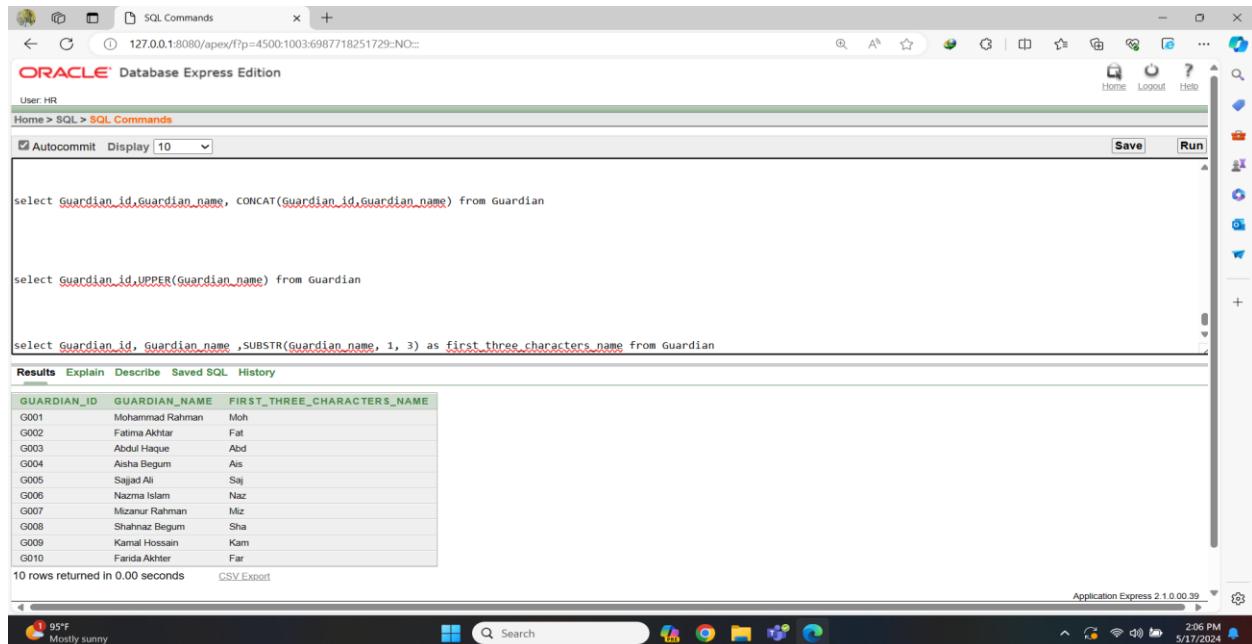
select Guardian_id,UPPER(Guardian_name) from Guardian |
```

GUARDIAN_ID	UPPER(GUARDIAN_NAME)
G001	MOHAMMAD RAHMAN
G002	FATIMA AKHTAR
G003	ABDUL HAQUE
G004	AISHA BEGUM
G005	SAJJAD ALI
G006	NAZMA ISLAM
G007	MIZANUR RAHMAN
G008	SHAHNAZ BEGUM
G009	KAMAL HOSSAIN
G010	FARIDA AKHTER

10 rows returned in 0.00 seconds CSV Export Application Express 2.1.0.0.39 2:05 PM 5/17/2024

Fig-7: Single Row Function Query

3. Display the Guardian_id, Guardian_name, and the first three characters of Guardian_name for all guardians. Alias this substring as first_three_characters_name.
 ⇒ select Guardian_id, Guardian_name ,SUBSTR(Guardian_name, 1, 3) as first_three_characters_name from Guardian



The screenshot shows the Oracle Database Express Edition interface. The SQL Commands window contains the following SQL code:

```

select Guardian_id,Guardian_name, CONCAT(Guardian_id,Guardian_name) from Guardian

select Guardian_id,UPPER(Guardian_name) from Guardian

select Guardian_id, Guardian_name ,SUBSTR(Guardian_name, 1, 3) as first_three_characters_name from Guardian

```

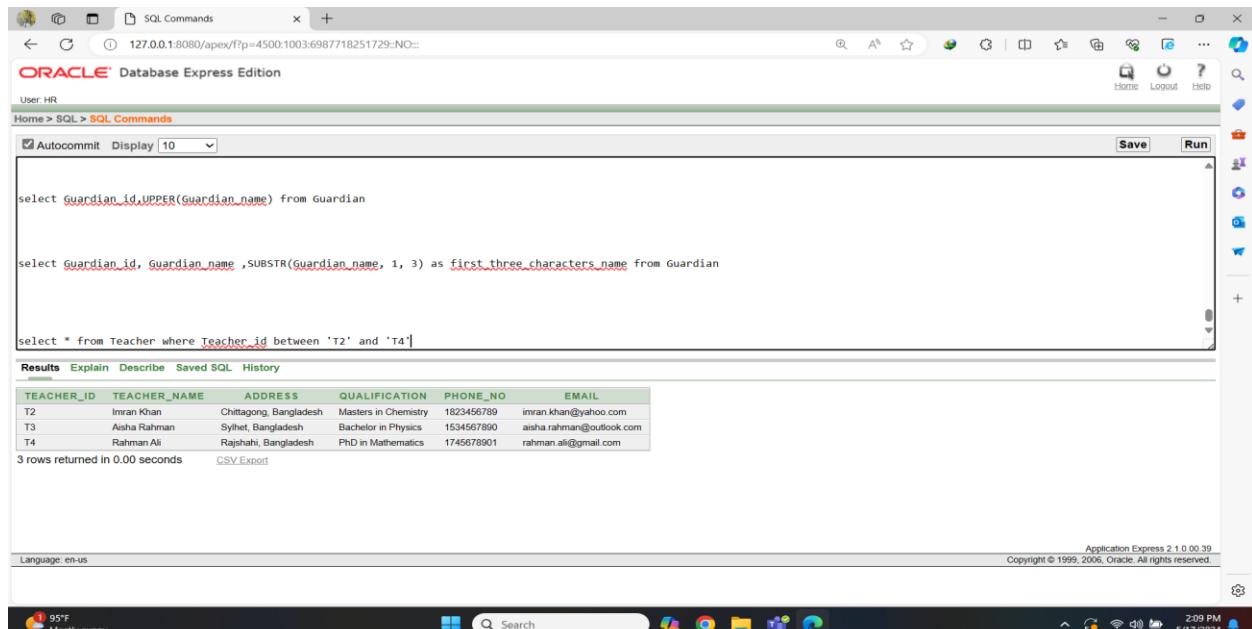
The Results tab displays the output:

GUARDIAN_ID	GUARDIAN_NAME	FIRST_THREE_CHARACTERS_NAME
G001	Mohammad Rahman	Moh
G002	Fatima Akhtar	Fat
G003	Abdul Haque	Abd
G004	Aisha Begum	Ais
G005	Sajad Ali	Saj
G006	Nazma Islam	Naz
G007	Mizanur Rahman	Miz
G008	Shahnaz Begum	Sha
G009	Kamal Hossain	Kam
G010	Farida Akther	Far

10 rows returned in 0.00 seconds [CSV Export](#)

Fig-8: Single Row Function Query

4. Display all columns for teachers whose Teacher_id is between 'T2' and 'T4'.
 ⇒ select * from Teacher where Teacher_id between 'T2' and 'T4'



The screenshot shows the Oracle Database Express Edition interface. The SQL Commands window contains the following SQL code:

```

select Guardian_id,UPPER(Guardian_name) from Guardian

select Guardian_id, Guardian_name ,SUBSTR(Guardian_name, 1, 3) as first_three_characters_name from Guardian

select * from Teacher where Teacher_id between 'T2' and 'T4'

```

The Results tab displays the output:

TEACHER_ID	TEACHER_NAME	ADDRESS	QUALIFICATION	PHONE_NO	EMAIL
T2	Imran Khan	Chittagong, Bangladesh	Masters in Chemistry	1823456789	imran.khan@yahoo.com
T3	Aisha Rahman	Sylhet, Bangladesh	Bachelor in Physics	1534567890	aisha.rahman@outlook.com
T4	Rahman Ali	Rajshahi, Bangladesh	PhD in Mathematics	1745678901	rahman.ali@gmail.com

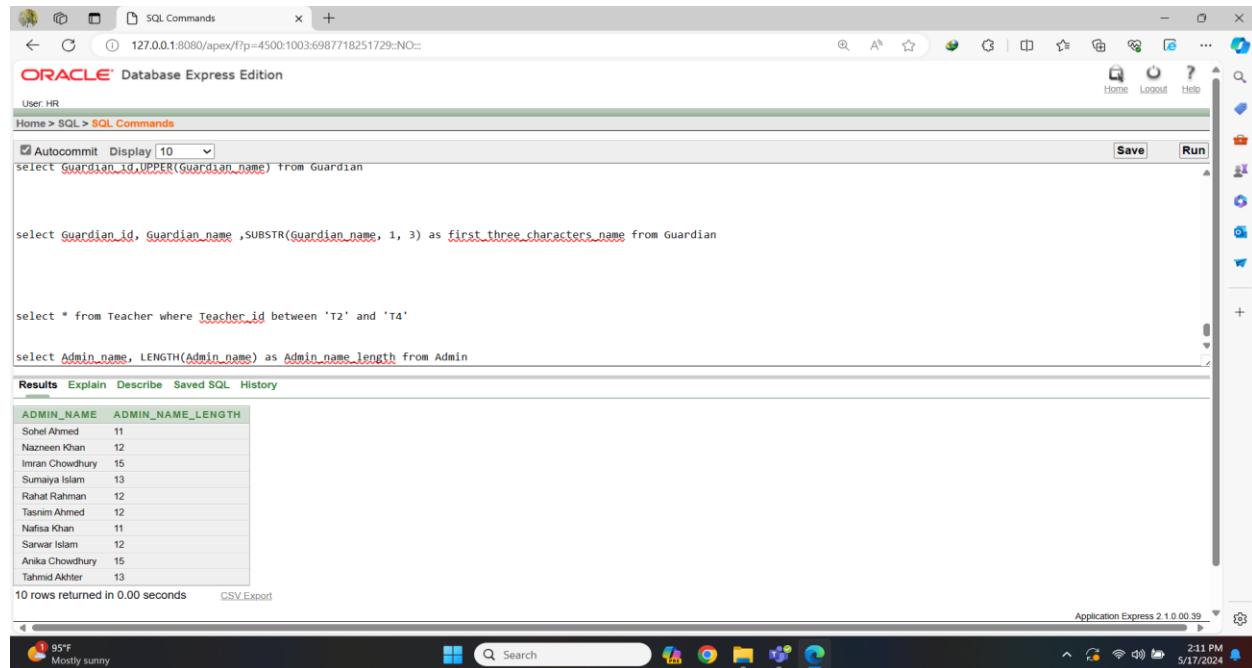
3 rows returned in 0.00 seconds [CSV Export](#)

Language: en-us Application Express 2.1 0.00.39 Copyright © 1999, 2006, Oracle. All rights reserved.

Fig-9: Single Row Function Query

10. Display the Admin_name and the length of Admin_name for all administrators.

⇒ select Admin_name, LENGTH(Admin_name) as Admin_name_length from Admin



The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
select GUARDIAN_ID, UPPER(GUARDIAN_NAME) from GUARDIAN

select GUARDIAN_ID, GUARDIAN_NAME, SUBSTR(GUARDIAN_NAME, 1, 3) as first_three_characters_name from GUARDIAN

select * from Teacher where Teacher_id between 'T2' and 'T4'

select Admin_name, LENGTH(Admin_name) as Admin_name_length from Admin
```

The Results tab displays the output of the last query:

ADMIN_NAME	ADMIN_NAME_LENGTH
Sohel Ahmed	11
Nazneen Khan	12
Imran Chowdhury	15
Sumaya Islam	13
Rahat Rahman	12
Tasnim Ahmed	12
Nafisa Khan	11
Sanwar Islam	12
Anika Chowdhury	15
Tahmid Akther	13

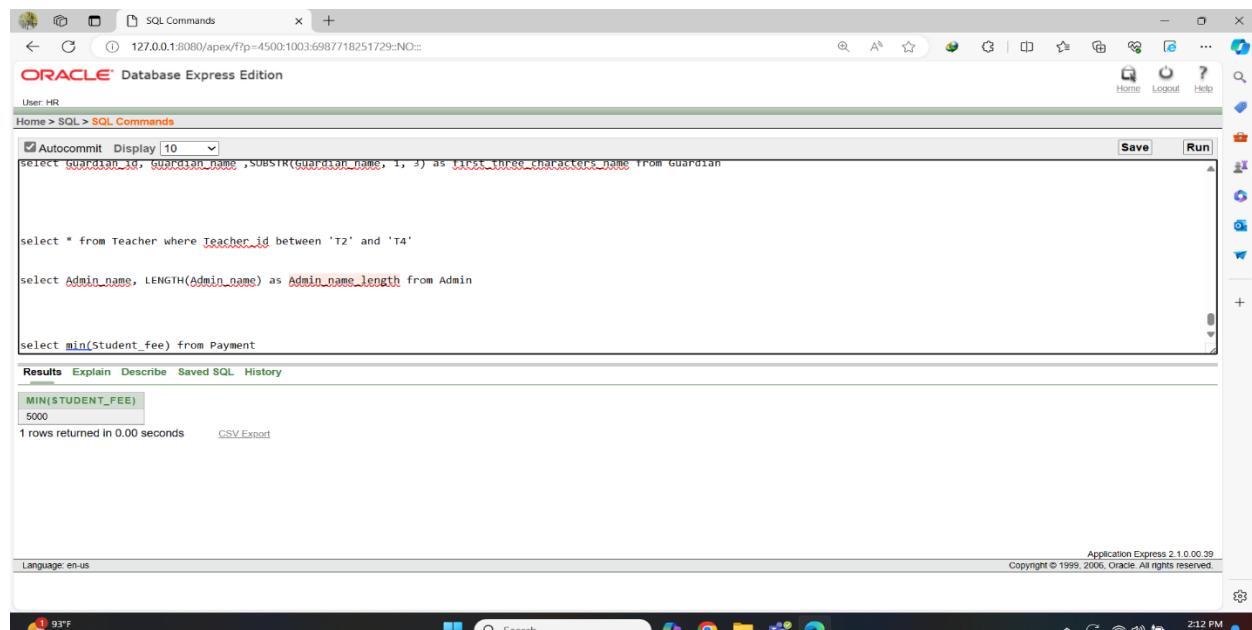
10 rows returned in 0.00 seconds [CSV Export](#)

Fig-10: Single Row Function Query

C) Multiple row function:

1. Display the minimum amount paid by any guardian from payment table.

⇒ select min(Student_fee) from Payment



The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
select GUARDIAN_ID, GUARDIAN_NAME, SUBSTR(GUARDIAN_NAME, 1, 3) as first_three_characters_name from GUARDIAN

select * from Teacher where Teacher_id between 'T2' and 'T4'

select Admin_name, LENGTH(Admin_name) as Admin_name_length from Admin

select min(Student_fee) from Payment
```

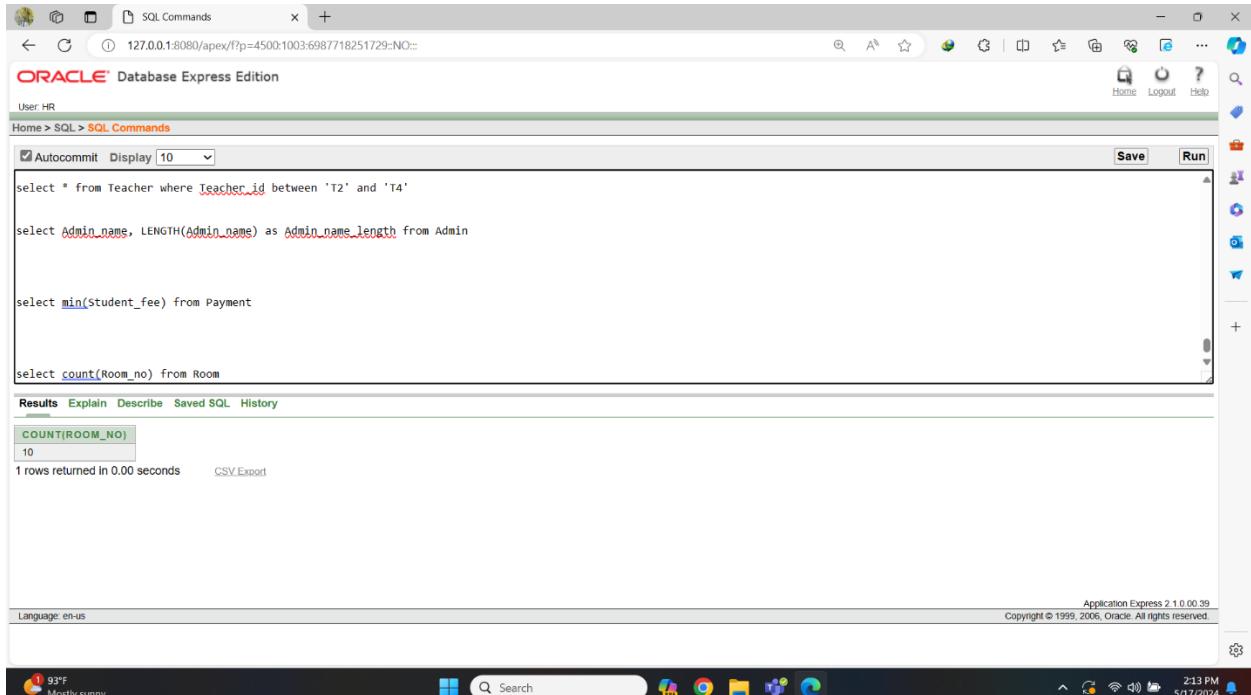
The Results tab displays the output of the last query:

MIN(STUDENT_FEE)
5000

1 rows returned in 0.00 seconds [CSV Export](#)

Fig-11: Multiple Row Function Query

2. Write an SQL query to count the total number of rooms in the room table.
 ⇒ select count(Room_no) from Room



```

SQL Commands
ORACLE Database Express Edition
User: HR
Home > SQL > SQL Commands
Autocommit: Display: 10 | Save | Run
select * from Teacher where Teacher_id between 'T2' and 'T4'

select Admin_name, LENGTH(Admin_name) as Admin_name_length from Admin

select min(Student_fee) from Payment

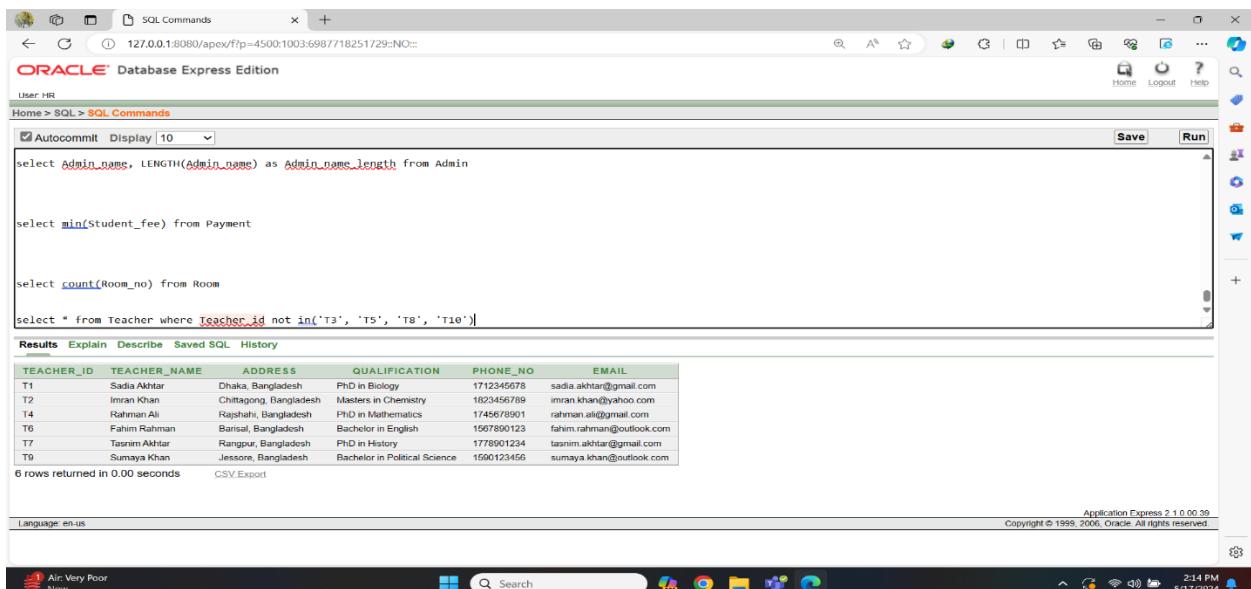
select count(Room_no) from Room

Results Explain Describe Saved SQL History
COUNT(ROOM_NO)
10
1 rows returned in 0.00 seconds CSV Export
Language: en-us Application Express 2.1.0.0.39
Copyright © 1999, 2006, Oracle. All rights reserved.

```

Fig-12: Multiple Row Function Query

3. Display all information for teachers whose Teacher_id is not T3, T5, T8 and T10.
 ⇒ select * from Teacher where Teacher_id not in('T3', 'T5', 'T8', 'T10')



```

SQL Commands
ORACLE Database Express Edition
User: HR
Home > SQL > SQL Commands
Autocommit: Display: 10 | Save | Run
select Admin_name, LENGTH(Admin_name) as Admin_name_length from Admin

select min(Student_fee) from Payment

select count(Room_no) from Room

select * from Teacher where Teacher_id not in('T3', 'T5', 'T8', 'T10')

Results Explain Describe Saved SQL History
TEACHER_ID TEACHER_NAME ADDRESS QUALIFICATION PHONE_NO EMAIL
T1 Sadia Akhtar Dhaka, Bangladesh PhD in Biology 1712345678 sadia.akhtar@gmail.com
T2 Imran Khan Chittagong, Bangladesh Masters in Chemistry 1823456789 imran.khan@yahoo.com
T4 Rahman Ali Rajshahi, Bangladesh PhD in Mathematics 1745678901 rahman.ali@gmail.com
T6 Fahim Rahman Barisal, Bangladesh Bachelor in English 1567890123 fahim.rahman@outlook.com
T7 Tasnim Akhtar Rangpur, Bangladesh PhD in History 1778901234 tasnim.akhtar@gmail.com
T9 Sumaya Khan Jessore, Bangladesh Bachelor in Political Science 1590123456 sumaya.khan@outlook.com
6 rows returned in 0.00 seconds CSV Export
Language: en-us Application Express 2.1.0.0.39
Copyright © 1999, 2006, Oracle. All rights reserved.

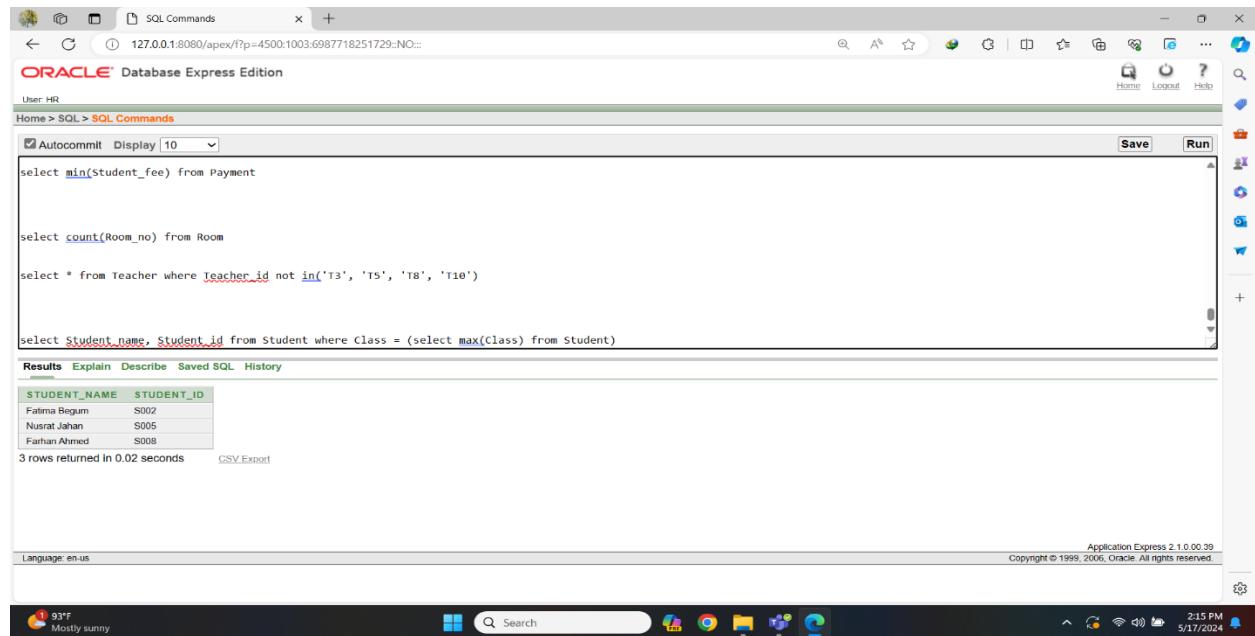
```

Fig-13: Multiple Row Function Query

D) Subquery:

1. Display the Student_name and Student_id of students who belong to the highest class in the school.

⇒ select Student_name, Student_id from Student where Class = (select max(Class) from Student)



```

SQL Commands
ORACLE Database Express Edition
User HR
Home > SQL > SQL Commands
Autocommit Display | 10 | Save | Run
select min(Student_fee) from Payment

select count(Room_no) from Room

select * from Teacher where Teacher_id not in('T3', 'T5', 'T8', 'T10')

select Student_name, Student_id from Student where Class = (select max(Class) from Student)

```

Results

STUDENT_NAME	STUDENT_ID
Fatima Begum	S002
Nusrat Jahan	S005
Faithan Ahmed	S008

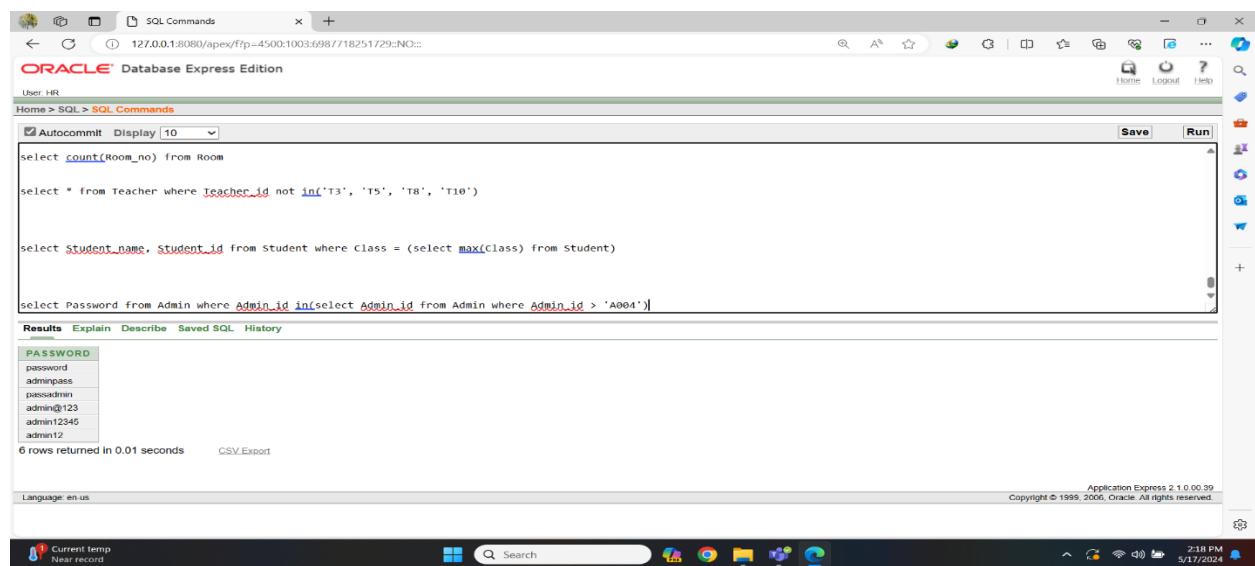
3 rows returned in 0.02 seconds CSV Export

Language: en-us Application Express 2.1.0.0.39 Copyright © 1999, 2006, Oracle. All rights reserved.

Fig-14: Subquery

2. Display the password of administrators whose Admin_id are greater than 'A004'.

⇒ select Password from Admin where Admin_id in(select Admin_id from Admin where Admin_id > 'A004')



```

SQL Commands
ORACLE Database Express Edition
User HR
Home > SQL > SQL Commands
Autocommit Display | 10 | Save | Run
select count(Room_no) from Room

select * from Teacher where Teacher_id not in('T3', 'T5', 'T8', 'T10')

select Student_name, Student_id from Student where Class = (select max(Class) from Student)

select Password from Admin where Admin_id in(select Admin_id from Admin where Admin_id > 'A004')

```

Results

PASSWORD
password
adminpass
passadmin
admin@123
admin12345
admin12

6 rows returned in 0.01 seconds CSV Export

Language: en-us Application Express 2.1.0.0.39 Copyright © 1999, 2006, Oracle. All rights reserved.

Fig-15: Subquery

E) Join Query:

- Equi Join:

Display the Teacher_id, Teacher_name, Qualification, and Subject_id from the Teacher table and TeacherSubject table, where the records in both tables are related by the Teacher_id column.

⇒select t.Teacher_id, t.Teacher_name , t.Qualification , ts.Subject_id from Teacher t, TeacherSubject ts where t.Teacher_id = ts.Teacher_id

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
select student_name, student_id from Student where class = (select max(class) from student)

select Password from Admin where Admin_id in(select Admin_id from Admin where Admin_id > 'A004')

select t.Teacher_id, t.Teacher_name , t.Qualification , ts.Subject_id from Teacher t, TeacherSubject ts where t.Teacher_id = ts.Teacher_id
```

The Results tab displays the output of the query:

TEACHER_ID	TEACHER_NAME	QUALIFICATION	SUBJECT_ID
T1	Sadia Akhtar	PhD in Biology	Sub001
T2	Imran Khan	Masters in Chemistry	Sub002
T3	Aleha Rathman	Bachelor in Physics	Sub003
T4	Rohit Ali	PhD in Mathematics	Sub004
T5	Saima Iqbal	Masters in Computer Science	Sub005
T6	Fahim Hamza	Bachelor in English	Sub006
T7	Tasnim Akhtar	PhD in History	Sub007
T8	Ramya	Masters in Economics	Sub008
T9	Sumayya Khan	Bachelor in Political Science	Sub009
T10	Farhana Ahmed	PhD in Sociology	Sub010

10 rows returned in 0.00 seconds

Fig-16: Equi Join Query

- Non-equi Join:

Display the Teacher_id, Teacher_name, Qualification, and Subject_id for each teacher, excluding cases where a teacher is assigned a subject that doesn't match their own ID.

⇒select t.Teacher_id, t.Teacher_name , t.Qualification , ts.Subject_id from Teacher t, TeacherSubject ts where ts.Teacher_id != t.Teacher_id

The screenshot shows the Oracle Database Express Edition interface. In the SQL Commands window, the following SQL code is entered:

```
select t.Teacher_id, t.Teacher_name , t.Qualification , ts.Subject_id from Teacher t, TeacherSubject ts where t.Teacher_id = ts.Teacher_id

select t.Teacher_id, t.Teacher_name , t.Qualification , ts.Subject_id from Teacher t, TeacherSubject ts where ts.Teacher_id != t.Teacher_id
```

The Results tab displays the output of the second query:

TEACHER_ID	TEACHER_NAME	QUALIFICATION	SUBJECT_ID
T1	Sadia Akhtar	PhD in Biology	Sub002
T1	Sadia Akhtar	PhD in Biology	Sub003
T1	Sadia Akhtar	PhD in Biology	Sub004
T1	Sadia Akhtar	PhD in Biology	Sub005
T1	Sadia Akhtar	PhD in Biology	Sub006
T1	Sadia Akhtar	PhD in Biology	Sub007
T1	Sadia Akhtar	PhD in Biology	Sub008
T1	Sadia Akhtar	PhD in Biology	Sub009
T1	Sadia Akhtar	PhD in Biology	Sub010
T2	Imran Khan	Masters in Chemistry	Sub001

More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.00 seconds

Fig-17: Non-equi Join Query

- Self join:

Display the information of exam id and exam name from the self join of both in Exam table whose gpa is greater than 4.50 and total marks is less than 1000

⇒ select 'Exam id is ' || e1.Exam_id || ' and exam name is ' || e2.Exam_name from Exam e1, Exam e2 where e1.GPA > 4.50 and e2.Total_marks < 1000

The screenshot shows the Oracle Database Express Edition interface. The SQL command window contains the following query:

```
select 'Exam id is ' || e1.Exam_id || ' and exam name is ' || e2.Exam_name from Exam e1, Exam e2 where e1.GPA > 4.50 and e2.Total_marks < 1000
```

The results pane displays the output of the query, which lists nine rows of data. Each row consists of a string concatenation of the exam ID and name from two different rows in the Exam table. The columns shown are 'EXAMID1\$||E1.EXAM_ID||\$AND\$EXAMNAMEIS\$||E2.EXAM_NAME'. The data includes various exam IDs and names such as E001, E002, E003, E004, E005, E006, E007, E008, E009, and E010.

Fig-18: Self join Query

- Outer join:

Display the guardian id, name, address, phone number along with their corresponding payment information payment id from the Guardian and GuardianPayment table

⇒ select g.Guardian_id, g.Guardian_name, g.AC_id, g.Phone_no, gp.GP_id, gp.Payment_id from Guardian g, GuardianPayment gp where g.Guardian_id = gp.GP_id(+) order by g.Guardian_id

The screenshot shows the Oracle Database Express Edition interface. The SQL command window contains the following query:

```
select g.Guardian_id, g.Guardian_name, g.AC_id, g.Phone_no, gp.GP_id, gp.Payment_id from Guardian g, GuardianPayment gp where g.Guardian_id = gp.GP_id(+) order by g.Guardian_id
```

The results pane displays the output of the query, which lists ten rows of data. The columns shown are 'GUARDIAN_ID', 'GUARDIAN_NAME', 'AC_ID', 'PHONE_NO', 'GP_ID', and 'PAYMENT_ID'. The data includes various guardian names like Mohammad Rahman, Fatima Akhtar, Abdul Haque, Aisha Begum, Sajid Ali, Nazma Islam, Mizanur Rahman, Shahna Begum, Kamal Hossain, and Farida Akther, along with their respective AC IDs, phone numbers, and payment IDs.

Fig-19: Outer join Query

F) View:

Create a view for the Guardian_id, Guardian_name, AC_id, Phone_no, Student_id, Student_name, Class , Email of the guardians and students included in the Guardian_Student_View and Guardian_id in ‘G001’, ‘G002’ and Student_id in ‘S001’, ‘S002’

⇒ create view Guardian_Student_View as select g.Guardian_id, g.Guardian_name, g.AC_id, g.Phone_no, s.Student_id, s.Student_name, s.Class, s.Email from Guardian g, Student s where g.Guardian_id in('G001','G002') and s.Student_id in('S001','S002')

```

SQL Commands
127.0.0.1:8080/apex/F?p=4500:1003:2079050842539417::NO::

ORACLE Database Express Edition

User HR
Home > SQL > SQL Commands
Autocommit Display 10
create view Guardian_Student_View as select g.Guardian_id, g.Guardian_name, g.AC_id, g.Phone_no, s.Student_id, s.Student_name, s.Class, s.Email from Guardian g, Student s where g.Guardian_id in('G001','G002') and s.Student_id in('S001','S002')

desc Guardian_Student_View

select * from Guardian_Student_View

Results Explain Describe Saved SQL History
Object Type VIEW Object GUARDIAN_STUDENT_VIEW
Table Column Data Type Length Precision Scale Primary Key Nullable Default Comment
GUARDIAN_STUDENT_VIEW GUARDIAN_ID Varchar2 20 - - - - - - -
GUARDIAN_NAME Varchar2 30 - - - - - - - - -
AC_ID Varchar2 20 - - - - - - - - -
PHONE_NO Varchar2 11 - - - - - - - - -
STUDENT_ID Varchar2 20 - - - - - - - - -
STUDENT_NAME Varchar2 30 - - - - - - - - -
CLASS Varchar2 10 - - - - - - - - -
EMAIL Varchar2 20 - - - - - - - - -
1 - 8

```

Fig-20: View Creation

select * from Guardian_Student_View

```

SQL Commands
127.0.0.1:8080/apex/F?p=4500:1003:2079050842539417::NO::

ORACLE Database Express Edition

User HR
Home > SQL > SQL Commands
Autocommit Display 10
create view Guardian_Student_View as select g.Guardian_id, g.Guardian_name, g.AC_id, g.Phone_no, s.Student_id, s.Student_name, s.Class, s.Email from Guardian g, Student s where g.Guardian_id in('G001','G002') and s.Student_id in('S001','S002')

desc Guardian_Student_View

select * from Guardian_Student_View

Results Explain Describe Saved SQL History
GUARDIAN_ID GUARDIAN_NAME AC_ID PHONE_NO STUDENT_ID STUDENT_NAME CLASS EMAIL
G001 Mohammad Rahman AC001 01712345678 S001 Abdul Rahman 10 abdul@gmail.com
G001 Mohammad Rahman AC001 01712345678 S010 Imran Khan 10 imran@live.com
G002 Farida Akther AC010 01922233344 S001 Abdul Rahman 10 abdul@gmail.com
G002 Farida Akther AC010 01922233344 S010 Imran Khan 10 imran@live.com
4 rows returned in 0.00 seconds CSV Export

Language: en-us Application Express 2.1.0.00.39
Copyright © 1999, 2006, Oracle. All rights reserved.

```

Fig-21: View Data

G) Sequence:

```
create sequence Receipt_id
```

```
minvalue 1  
maxvalue 100  
start with 6  
increment by 5  
nocycle  
nocache
```

```
create table Receipt
```

```
(
```

```
    Receipt_id number(38),  
    Student_id varchar2(20) constraint Receipt_Sid_fk references Student(Student_id),  
    Student_name varchar2(30),  
    Total_Ammount number(10),  
    Receipt_date date
```

```
)
```

The screenshot shows the Oracle Database Express Edition interface. In the top navigation bar, there are tabs for 'SQL Commands', 'student payment money recit - 5', 'receipt of payment of student - 1', and '+'. Below the navigation bar, the title bar displays 'ORACLE Database Express Edition' and the URL '127.0.0.1:8080/apex/f?p=4500:1003:4038932891279944:NO...'.

The main area is divided into two sections: 'SQL Commands' and 'Object Types'.

SQL Commands: This section contains the SQL code for creating a sequence and a table. The sequence 'Receipt_id' is defined with parameters: minvalue 1, maxvalue 100, start with 6, increment by 5, nocycle, and nocache. The table 'Receipt' is created with columns: Receipt_id (number(38)), Student_id (varchar2(20)) with a constraint 'Receipt_Sid_fk' referencing the 'Student' table's 'Student_id' column, Student_name (varchar2(30)), Total_Ammount (number(10)), and Receipt_date (date).

Object Types: This section shows the 'RECEIPT' object type. It lists five columns: RECEIPT_ID, STUDENT_ID, STUDENT_NAME, TOTAL_AMOUNT, and RECEIPT_DATE. The 'TOTAL_AMOUNT' column has a precision of 10 and a scale of 0.

Fig-22: Sequence Create

```

insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date)
values (Receipt_id.nextval,'S001', 'Abdul Rahman', 5000, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date)
values (Receipt_id.nextval,'S002', 'Fatima Begum', 5500, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date)
values (Receipt_id.nextval,'S003', 'Ayesha Khan', 6000, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date)
values (Receipt_id.nextval,'S004', 'Mohammad Ali', 6500, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date)
values (Receipt_id.nextval,'S005', 'Nusrat Jahan', 7000, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date)
values (Receipt_id.nextval,'S006', 'Arif Hasan', 7500, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date)
values (Receipt_id.nextval,'S007', 'Sadia Akter', 8000, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date)
values (Receipt_id.nextval,'S008', 'Farhan Ahmed', 8500, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date)
values (Receipt_id.nextval,'S009', 'Nazia Islam', 9000, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date)
values (Receipt_id.nextval,'S010', 'Imran Khan', 9500, sysdate)

```

The screenshot shows the Oracle Database Express Edition interface. The SQL Commands tab is active, displaying the following SQL code:

```

drop sequence Receipt_Id;
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date) values (Receipt_id.nextval,'S001', 'Abdul Rahman', 5000, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date) values (Receipt_id.nextval,'S002', 'Fatima Begum', 5500, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date) values (Receipt_id.nextval,'S003', 'Ayesha Khan', 6000, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date) values (Receipt_id.nextval,'S004', 'Mohammad Ali', 6500, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date) values (Receipt_id.nextval,'S005', 'Nusrat Jahan', 7000, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date) values (Receipt_id.nextval,'S006', 'Arif Hasan', 7500, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date) values (Receipt_id.nextval,'S007', 'Sadia Akter', 8000, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date) values (Receipt_id.nextval,'S008', 'Farhan Ahmed', 8500, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date) values (Receipt_id.nextval,'S009', 'Nazia Islam', 9000, sysdate)
insert into Receipt (Receipt_id, Student_id, Student_name, Total_Ammount, Receipt_date) values (Receipt_id.nextval,'S010', 'Imran Khan', 9500, sysdate)

select * from Receipt

```

The results section displays the data inserted into the Receipt table:

RECEIPT_ID	STUDENT_ID	STUDENT_NAME	TOTAL_AMMOUNT	RECEIPT_DATE
6	S001	Abdul Rahman	5000	18-MAY-24
11	S002	Fatima Begum	5500	18-MAY-24
16	S003	Ayesha Khan	6000	18-MAY-24
21	S004	Mohammad Ali	6500	18-MAY-24
26	S005	Nusrat Jahan	7000	18-MAY-24
31	S006	Arif Hasan	7500	18-MAY-24
36	S007	Sadia Akter	8000	18-MAY-24
41	S008	Farhan Ahmed	8500	18-MAY-24
46	S009	Nazia Islam	9000	18-MAY-24
51	S010	Imran Khan	9500	18-MAY-24

10 rows returned in 0.00 seconds

Fig-23: Sequence Data

Relational Algebra

1. Display all information from Admin where Admin id is A005.

$$\begin{array}{l} \text{Admin} \\ \sigma \\ \text{Admin_id} = \text{'A005'} \end{array}$$

2. Display Student id, email those class is 9

$$\left[\begin{array}{l} \text{Student} \\ \sigma \\ \text{Class} = \text{'9'} \end{array} \right]$$
 π Student_id, Email

3. Display the Guardian ID, name whose id under G007.

$$\left[\begin{array}{l} \text{Guardian} \\ \sigma \\ \text{Guardian_id} < \text{'G007'} \end{array} \right]$$
 π $\text{Guardian_id, Guardian_name}$

4. Display the Payment id and student fee whose fee between 6000 and 10000

$$\left[\begin{array}{l} \text{Payment} \\ \sigma \\ \text{Student_fee between 6000 and 10000} \end{array} \right]$$

π

Payment_id, Student_fee

5. Display the Teacher maximum and minimum salary

$$\begin{array}{l} \text{Salary} \\ \sigma \\ \max(\text{Teacher_salary}), \min(\text{Teacher_salary}) \end{array}$$

6. Write a query to display the salary id, teacher salary where bonus is 5% of their salary.

$$\left[\begin{array}{l} \text{Salary} \\ \sigma \\ \text{Teacher_salary}*0.05 \text{ as Bonus} \end{array} \right]$$

π

Salary_id, Teacher_salary

7. Display Admin id and name where the third letter of their name is h.

$$\left[\begin{array}{l} \text{Admin} \\ \sigma \\ \text{Admin_name like ' } \underline{} \text{h\%'} \end{array} \right]$$

π

Admin_id, Admin_name

8. Display teacher id , name and salary for all teacher.

$$\left[\begin{array}{l} \text{Teacher} \times \text{TeacherSalary} \times \text{Salary} \\ \sigma \\ \text{t.Teacher_id} = \text{ts.Teacher_id} \text{ and ts.Salary_id} = \text{s.Salary_id} \end{array} \right]$$

π

t.Teacher_id, t.Teacher_name , s.Teacher_salary

9. Display all information of guardian whose id is G002

```
Guardian
σ
Guardian_id = 'G002'
```

10. Display the student maximum and minimum total marks in exam

```
Exam
σ
max(Total_marks), min(Total_marks)
```

Conclusion

In conclusion, adding the student-guardian-teacher examination module to the coaching management system is a major improvement in educational technology. This addition promotes better collaboration, transparency, and accountability among students, parents, and teachers. Using database technology, the system simplifies communication, allows real-time progress tracking, and provides useful insights for making informed decisions. These changes not only enhance the educational experience but also create a solid foundation for ongoing improvements and innovations. By giving everyone involved timely information and a clear understanding of each student's progress, this system fosters a supportive and engaged learning environment. Parents can offer more effective support, teachers can adjust their instruction to meet individual needs, and students can benefit from a more unified and responsive educational framework. The integration of this module helps in identifying trends and areas for improvement, ensuring that the coaching management system adapts to the changing needs of education. Overall, this advancement shows the powerful impact of technology in education, driving continuous improvement and enabling all participants to actively contribute to the learning journey. As a result, the coaching management system is better equipped to prepare students for future challenges, making it a vital tool in the pursuit of academic excellence and overall development.