Hostel Management System

Introduction:

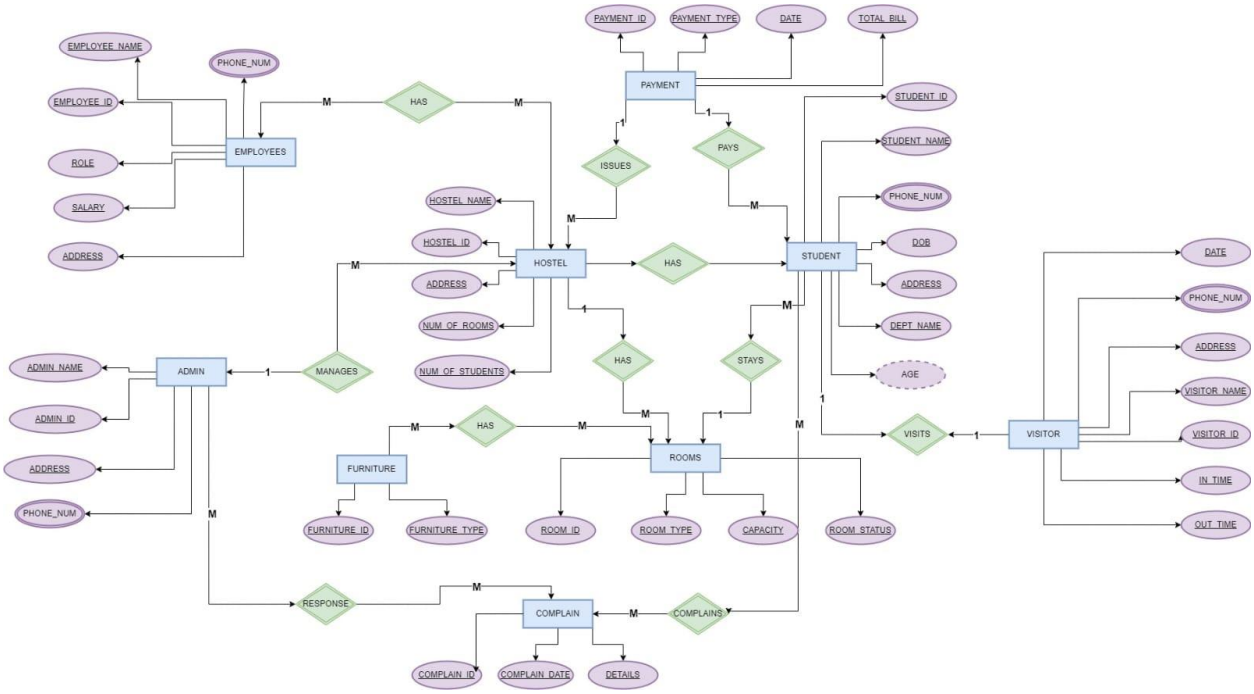
The Hostel Management System (HMS) represents a sophisticated solution designed to tackle the multifaceted challenges associated with managing hostel accommodations in today's fast-paced environment. As the demand for efficient processes continues to rise, the HMS emerges as a pivotal tool for simplifying and optimizing various aspects of hostel management, ranging from student record-keeping to room allocations and administrative tasks. Utilizing cutting-edge technology and robust database management principles, the HMS boasts a user-friendly interface that fosters seamless communication and transparency across all hostel operations. It revolutionizes traditional practices by automating critical processes such as student registration, room allocations based on individual preferences and real-time availability, billing, payment tracking, inventory management, and maintenance scheduling. This automation not only streamlines operations but also minimizes errors and enhances overall efficiency. Moreover, the HMS goes beyond mere automation, offering comprehensive reporting and analytics capabilities. Administrators can leverage these features to gain valuable insights into occupancy trends, revenue generation, resource utilization, and other key performance indicators. Armed with this data-driven intelligence, administrators can make informed decisions, optimize resource allocation, and continuously improve hostel management strategies. Whether managing a small-scale hostel or overseeing accommodations within a large educational institution, the HMS is highly adaptable and customizable to suit specific needs and requirements. By empowering administrators with the tools to deliver a superior living experience for residents while simultaneously optimizing operational efficiency, the HMS stands as a cornerstone in modern hostel management. It not only streamlines operations but also fosters a conducive environment for growth, innovation, and excellence in hostel management practices.

Case Study:

In response to the growing complexities of managing a hostel environment efficiently, XYZ Hostel embarked on a journey to implement a comprehensive Hostel Management System (HMS). The system aimed to streamline various aspects of hostel operations including student accommodation, administration, visitor management, complaint handling, and payment management. Central to this system were several key entities and their attributes. Firstly, the Hostel entity where the name of the hostel, address, number of rooms, and students are stored. Each hostel is identified with a unique Hostel ID. The student entity formed the backbone of the system, encompassing crucial details such as student name, phone number, address, age, department name, and a unique Student ID allotted to each student. A hostel can have many students, but each student can be in only one hostel. A hostel has many rooms, and for each room, there is a unique Room ID as well as room type, room status, and capacity. A student can stay in only one room, but a room can contain many students. A room can have many pieces of furniture, each with a unique Furniture ID, and the furniture type is stored. A visitor can visit a student, and for each visitor, there's a unique Visitor ID. Other details are such as visitor name, visiting date, time to enter and leave, as well as the visitor’s phone number and address. Each student can have one visitor, and one visitor can visit one student. Admins played a pivotal role in overseeing the entire hostel management process. Their details, including Admin ID, admin name, phone number, and address, were integrated into the system. Each admin handles one or more complaints raised by students. Each complaint has a unique complaint ID. The other details such as complaint date and details are stored. One or many students can complain. The system also maintained a comprehensive database of employee information, including employee ID, employee name, address, role, date of birth,

salary, and phone number. Many employees can work in one hostel as well as many employees can work in multiple hostels. The system also stores payment data, where each payment has a unique payment ID. The other important data such as payment type, date, and total bill are also stored. One or many students can make a payment, but one payment can be made by one student. Multiple payments can be made to a single hostel by different students.

ER DIAGRAM:



Normalization:

Has(Emp_id,Emp name,role,salary,address,phone_num,hostel name,hostel_id,address,num of rooms,num of students)

1 NF: Phone_Num Multivalued attribute

2NF:Emp id,Emp name,role,salary,address,phone_num

Hostel_name,Hostel_id,address,num of rooms,num of students

EH_id,Emp_id,Hostel_id

3NF: Emp_id,Emp name, role,salary,address,phone_num

Hostel_name,Hostel-id,address,num of rooms,num of students

EH_id,Emp id,Hostel-id

No transitive dependency

Table:

1. Emp id, Emp name, role,salary,address,phone_num
2. Hostel-id,Hostel-name,address,num of rooms,num of students
3. EH_id,Emp_id,Hostel_id

MANAGES(_Admin ID,Admin name,address,phone_num,Hostel_id,Hostel name,address,num of rooms,num of students)

1NF : Phone_num multivalued attribute

2NF : Admin_id, Admin name,address,phone_num

Hostel_id,Hostel-name,num of rooms,num of students,Admin_id

3NF: Admin_id,Admin name,address,phone_num

Hostelid,Hostel-name,num of rooms,num of students,Admin_id

No transtive dependency

Table:

1. Admin_id,Admin name,address,phone_num
2. Hostel_id,Hostel-name,num of rooms,num of students,Admin_id

ISSUES(Hostel_id,Hostel name,address,num of students,num of rooms,Payment_id,payment type,date,total bill)

1 NF:No multivalued attribute

2 NF: Hostel_id,Hostel name,address,num of students,num of rooms

Payment_id,payment type,date,total bill

HP_id,Hostel_id,Payment_id

3 NF : Hostel_id,Hostel name,address,num of students,num of rooms

Payment_id,payment type,date,total bill

HP_id,Hostel_id,Payment_id

No transtive dependency

Table:

1. Hoste_ld,Hostel name,address,num of students,num of rooms,
2. Payment-id,payment type,date,total bill
3. HP_id,Hostel_id,Payment_id

PAYS(std_id,std name,phone_num,DOB,address,Dept name,age,Payment_id,payment type,date,total bill)

1 NF: Phone_num multivalued attribute

2 NF: std_id,std name,phone_num,DOB,address,Dept name,age

Payment_id,payment type,date,total bill,std_id

3 NF: std_id,std name,phone_num,,address,Dept name,DA_id

Payment_id,payment type,date,total bill

DA_id,DOB,age

Table :

1. std_id,std name,phone_num,,address,Dept name,DA_id
2. Payment_id,payment type,date,total bill
3. DA_id,DOB,age

HAS(Hostel_id,Hostel name,address,num of students,num of rooms, std_d,std name,phone_num,DOB,address,Dept name,age)

1 NF: Phone_num multivalued attribute

2 NF: Hostel_id,Hostel name,address,num of students,num of rooms,

std_id,std name,phone_num,DOB,address,Dept name,age,hostel_id

3 NF:Hostel_d,Hostel name,address,num of students,num of rooms,

std_id,std name,phone_num,address,Dept name,hostel_id,DA_id

DA_id ,DOB,age

TABLE :

1. Hostel_d,Hostel name,address,num of students,num of rooms,
2. std_id,std name,phone_num,address,Dept name,hostel_id,DA_id
3. DA_id ,DOB,age

STAYS(room_id,room type,capacity,room status, std_id,std name,phone_num,DOB,address,Dept name,age)

1 NF :Phone_num multivalued attribute

2 NF : room_id,room type,capacity,room status,

Std_id,std name,phone_num,DOB,address,Dept name,age,room _id

3 NF : room_id,room type,CR_id

std_id,std name,phone_num,address,Dept name,room_id,DA_id

DA_id,DOB,age

CR_id,capacity,room status

Table :

1. room_id,room type,CR_id
2. std_id,std name,phone_num,address,Dept name,room_id,DA_id
3. DA_id,DOB,age
4. CR_id,capacity,room status

HAS(Hostel_id,Hostel_name,address,num of students,num of rooms, room_id,room type,capacity,room status)

1 NF :No multivalued attribute

2 NF : Hostel_id,Hostel_name,address,num of students,num of rooms

room_id,room type,capacity,room status,Hostel_id

3 NF : Hostel_id,Hostel_name,address,num of students,num of rooms

room_id,room typeHostel_id,CR_id

CR_id,capacity ,room status

Table :

1. Hostel_id,Hostel_name,address,num of students,num of rooms
2. room_id,room typeHostel_id,CR_id
3. CR_id,capacity ,room status

VISITS(Visitor_id,visitor name,address,date,phone-num,in time,out time, std_id,std name,phone_num,DOB,address,Dept name,age)

1 NF :Phone _num multivaled attribute

2 NF : Visitor_id,visitor name,address,date,phone_num,in time,out time,

Std_id,std name,phone_num,DOB,address,Dept name,age,visitor_id

3 NF : Visitor_id,visitor name,address,date,phone_num,in time,out time,

Std_id,std name,phone_num,address,Dept name,visitor_id,DA_id

DA_id,DOB,age

Table :

1. Visitor_id,visitor name,address,date,phone_num,in time,out time,
2. Std_id,std name,phone_num,address,Dept name,visitor_id,DA_id
3. DA_id,DOB,age

HAS(room_id,room type,capacity,room status,furniture_id,furniture_name)

1 NF : No multivalued attribute

2 NF : room_id,room type,capacity,room status,

furniture_id,furniture_name,Room_id

3 NF : room_id,room type,CR_id,

furniture_id,furniture_name,Room_id

CR-id,capacity,room status

Table :

1. room_id,room type,CR_id,
2. furniture_id,furniture_name,Room_id
3. CR-id,capacity,room status

RESPONSE(Admin_id,Admin name,address,phone_num,complain_id,complain date,details)

1 NF :Phone_num multivalued attribute

2 NF : Admin_id,Admin name,address,phone_num

complain_id,complain date,details

AC_id,admin_id,complain_id

3 NF : Admin_id,Admin name,address,phone_num,

complain_id,complain date,details

AC_id,admin_id,complain_id

No transitive dependency

Table :

1. Admin_id,Admin name,address,phone_num,
2. complain_id,complain date,details
3. AC_id,admin_id,complain_id

Complains(std_id,std name,phone_num,DOB,address,Dept name,age, complain_id,complain date,details)

1 NF :Phone_num multivalued attribute

2 NF : std_id,std name,phone-num,DOB,address,Dept name,age

complain_id,complain date,details

SC-id,std_id,complain_id

3 NF : std_id,std name,phone_num,address,Dept name,DA_id

complain_id,complain date,details

SC_id,std_id,complain_id

DA_id,DOB,age

Table :

1. std_id,std name,phone_num,address,Dept name,DA_id
2. complain_id,complain date,details
3. SC_id,std_id,complain_id
4. DA_id,DOB,age

Total Table :

1. Emp id, Emp name, role,salary,address,phone_num

2. Hostel-id,Hostel-name,address,num of rooms,num of students(*)
3. EH_id,Emp_id,Hostel_id
4. Admin_id,Admin name,address,phone_num
5. Hostel_id,Hostel-name,num of rooms,num of students,Admin_id
6. Hoste_ld,Hostel name,address,num of students,num of rooms(*)
7. Payment-id,payment type,date,total bill
8. HP_id,Hostel_id,Payment_id
9. std_id,std name,phone_num,,address,Dept name,DA_id
10. Payment_id,payment type,date,total bill(*)
11. DA_id,DOB,age1
12. Hostel_d,Hostel name,address,num of students,num of rooms(*)
13. std_id,std name,phone_num,address,Dept name,hostel_id,DA_id
14. DA_id ,DOB,age(*)
15. room_id,room type,CR_id(*)
16. std_id,std name,phone_num,address,Dept name,room_id,DA_id
17. DA_id,DOB,age(*)
18. CR_id,capacity,room status(*)
19. Hostel_id,Hostel_name,address,num of students,num of rooms(*)
20. room_id,room typeHostel_id,CR_id
21. CR_id,capacity ,room status
22. Visitor_id,visitor name,address,date,phone_num,in time,out time
23. Std_id,std name,phone_num,address,Dept name,visitor_id,DA_id(*)
24. DA_id,DOB,age(*)
25. room_id,room type,CR_id(*)
26. furniture_id,furniture_name,Room_id
27. CR-id,capacity,room status(*)
28. Admin_id,Admin name,address,phone_num(*)
29. complain_id,complain date,details
30. AC_id,admin_id,complain_id
31. std_id,std name,phone_num,address,Dept name,DA_id(*)
32. complain_id,complain date,details(*)
33. SC_id,std_id,complain_id
34. DA_id,DOB,age(*)

Finalization:

1. Emp id, Emp name, role,salary,address,phone_num
2. EH_id,Emp_id,Hostel_id
3. Admin_id,Admin name,address,phone_num
4. Hostel_id,Hostel-name,num of rooms,num of students,Admin_id
5. Payment-id,payment type,date,total bill
6. HP_id,Hostel_id,Payment_id

7. std_id,std name,phone_num,,address,Dept name,DA_id
8. DA_id,DOB,age1
9. std_id,std name,phone_num,address,Dept name,hostel_id,DA_id
10. std_id,std name,phone_num,address,Dept name,room_id,DA_id
11. room_id,room typeHostel_id,CR_id
12. CR_id,capacity ,room status
13. Visitor_id,visitor name,address,date,phone_num,in time,out time
14. furniture_id,furniture_name,Room_id
15. complain_id,complain date,details
16. AC_id,admin_id,complain_id
17. SC_id,std_id,complain_id

Table Creation

Employees:

Autocommit Display 5000

```
CREATE TABLE Employees (
    Emp_id NUMBER(5) PRIMARY KEY,
    Emp_name VARCHAR2(255) NOT NULL,
    Role VARCHAR2(255) NOT NULL,
    Salary NUMBER(5,2) NOT NULL,
    Address VARCHAR2(255) NOT NULL,
    Phone_num VARCHAR2(15) NOT NULL
);
desc employees
```

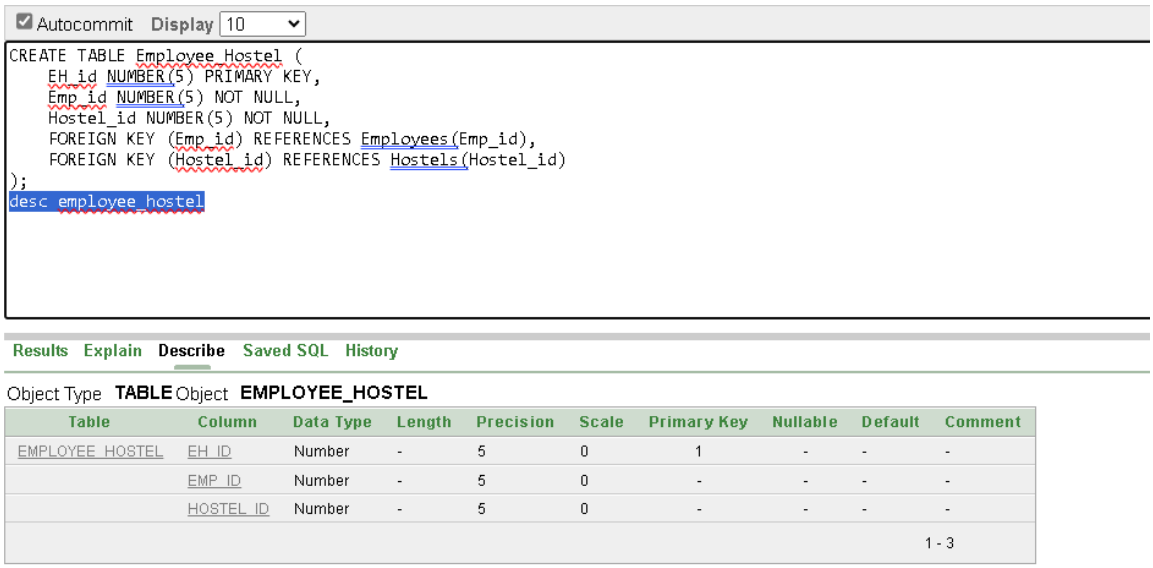
Results Explain Describe Saved SQL History

Object Type **TABLE** Object **EMPLOYEES**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMPLOYEES	EMP_ID	Number	-	5	0	1	-	-	-
	EMP_NAME	Varchar2	255	-	-	-	-	-	-
	ROLE	Varchar2	255	-	-	-	-	-	-
	SALARY	Number	-	10	3	-	-	-	-
	ADDRESS	Varchar2	255	-	-	-	-	-	-
	PHONE_NUM	Varchar2	15	-	-	-	-	-	-

1 - 6

Employee_Hostel:



Autocommit Display 10

```
CREATE TABLE Employee_Hostel (
    EH_id NUMBER(5) PRIMARY KEY,
    Emp_id NUMBER(5) NOT NULL,
    Hostel_id NUMBER(5) NOT NULL,
    FOREIGN KEY (Emp_id) REFERENCES Employees(Emp_id),
    FOREIGN KEY (Hostel_id) REFERENCES Hostels(Hostel_id)
);
desc employee_hostel
```

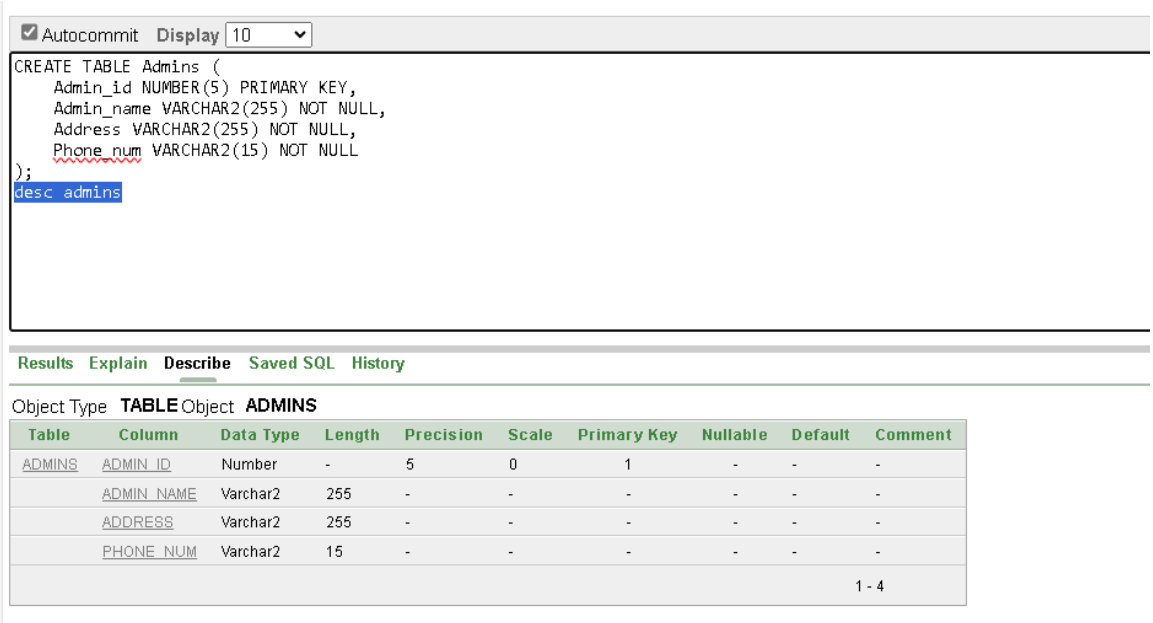
Results Explain Describe Saved SQL History

Object Type **TABLE** Object **EMPLOYEE_HOSTEL**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMPLOYEE_HOSTEL	EH_ID	Number	-	5	0	1	-	-	-
	EMP_ID	Number	-	5	0	-	-	-	-
	HOSTEL_ID	Number	-	5	0	-	-	-	-

1 - 3

Admin:



Autocommit Display 10

```
CREATE TABLE Admins (
    Admin_id NUMBER(5) PRIMARY KEY,
    Admin_name VARCHAR2(255) NOT NULL,
    Address VARCHAR2(255) NOT NULL,
    Phone_num VARCHAR2(15) NOT NULL
);
desc admins
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **ADMINS**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ADMINS	ADMIN_ID	Number	-	5	0	1	-	-	-
	ADMIN_NAME	Varchar2	255	-	-	-	-	-	-
	ADDRESS	Varchar2	255	-	-	-	-	-	-
	PHONE_NUM	Varchar2	15	-	-	-	-	-	-

1 - 4

Hostels:

Autocommit Display 10

```
CREATE TABLE Hostels (
    Hostel_id NUMBER(5) PRIMARY KEY,
    Hostel_name VARCHAR2(255) NOT NULL,
    Address VARCHAR2(255) NOT NULL,
    Num_of_rooms NUMBER(5) NOT NULL,
    Num_of_students NUMBER(5) NOT NULL,
    Admin_id NUMBER(5) NOT NULL,
    FOREIGN KEY (Admin_id) REFERENCES Admins(Admin_id)
);
desc hostels
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **HOSTELS**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
HOSTELS	HOSTEL_ID	Number	-	5	0	1	-	-	-
	HOSTEL_NAME	Varchar2	255	-	-	-	-	-	-
	ADDRESS	Varchar2	255	-	-	-	-	-	-
	NUM_OF_ROOMS	Number	-	5	0	-	-	-	-
	NUM_OF_STUDENTS	Number	-	5	0	-	-	-	-
	ADMIN_ID	Number	-	5	0	-	-	-	-

1 - 6

Payments:

Autocommit Display 10

```
CREATE TABLE Payments (
    Payment_id NUMBER(5) PRIMARY KEY,
    Payment_type VARCHAR2(255) NOT NULL,
    Payment_date DATE NOT NULL,
    Total_bill NUMBER(7,2) NOT NULL
);
desc payments
```

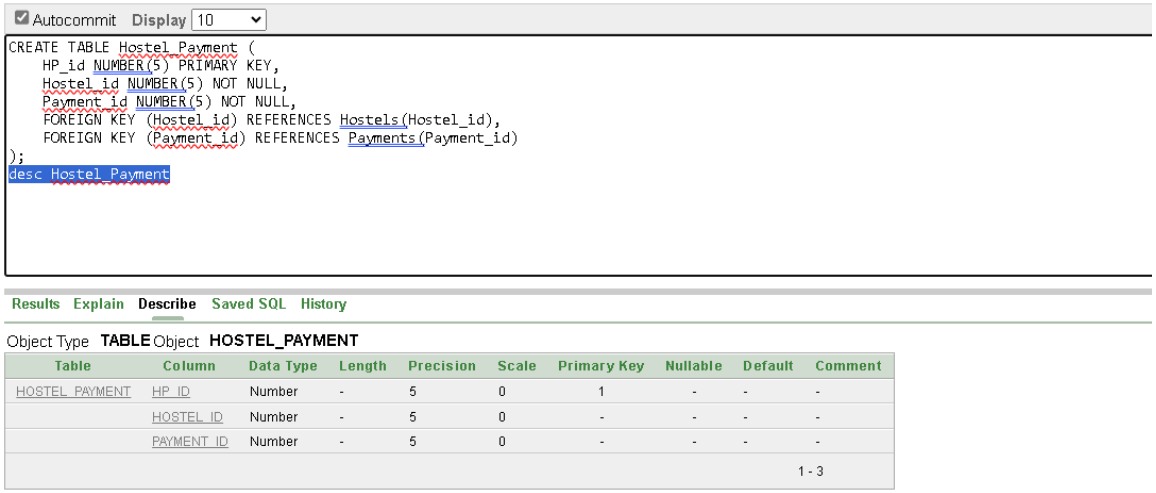
Results Explain Describe Saved SQL History

Object Type **TABLE** Object **PAYMENTS**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
PAYMENTS	PAYMENT_ID	Number	-	5	0	1	-	-	-
	PAYMENT_TYPE	Varchar2	255	-	-	-	-	-	-
	PAYMENT_DATE	Date	7	-	-	-	-	-	-
	TOTAL_BILL	Number	-	7	2	-	-	-	-

1 - 4

Hostels_payments:



Autocommit Display 10

```
CREATE TABLE Hostel_Payment (
    HP_id NUMBER(5) PRIMARY KEY,
    Hostel_id NUMBER(5) NOT NULL,
    Payment_id NUMBER(5) NOT NULL,
    FOREIGN KEY (Hostel_id) REFERENCES Hostels(Hostel_id),
    FOREIGN KEY (Payment_id) REFERENCES Payments(Payment_id)
);
desc Hostel_Payment
```

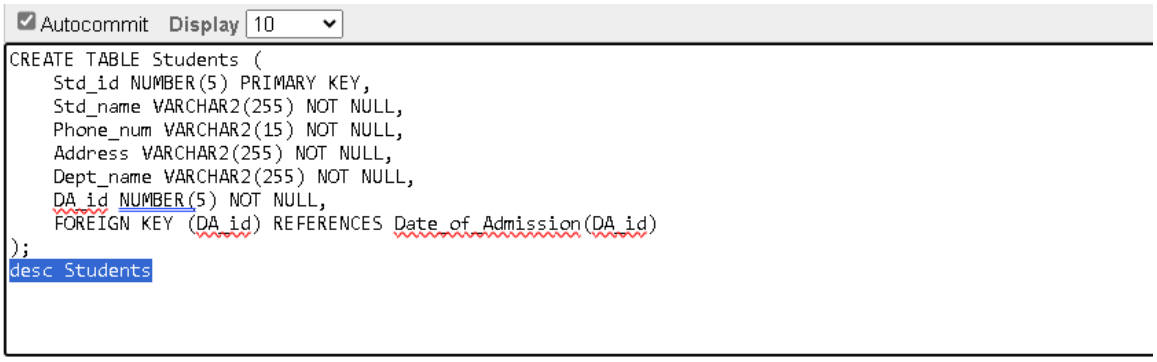
Results Explain Describe Saved SQL History

Object Type **TABLE** Object **HOSTEL_PAYMENT**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
HOSTEL_PAYMENT	HP_ID	Number	-	5	0	1	-	-	-
	HOSTEL_ID	Number	-	5	0	-	-	-	-
	PAYMENT_ID	Number	-	5	0	-	-	-	-

1 - 3

Students:



Autocommit Display 10

```
CREATE TABLE Students (
    Std_id NUMBER(5) PRIMARY KEY,
    Std_name VARCHAR2(255) NOT NULL,
    Phone_num VARCHAR2(15) NOT NULL,
    Address VARCHAR2(255) NOT NULL,
    Dept_name VARCHAR2(255) NOT NULL,
    DA_id NUMBER(5) NOT NULL,
    FOREIGN KEY (DA_id) REFERENCES Date_of_Admission(DA_id)
);
desc Students
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **STUDENTS**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
STUDENTS	STD_ID	Number	-	5	0	1	-	-	-
	STD_NAME	Varchar2	255	-	-	-	-	-	-
	PHONE_NUM	Varchar2	15	-	-	-	-	-	-
	ADDRESS	Varchar2	255	-	-	-	-	-	-
	DEPT_NAME	Varchar2	255	-	-	-	-	-	-
	DA_ID	Number	-	5	0	-	-	-	-

1 - 6

Date_of_Admission:

Autocommit Display 10

```
CREATE TABLE Date_of_Admission (
    DA_id NUMBER(5) PRIMARY KEY,
    DOB DATE NOT NULL,
    Age NUMBER(3) NOT NULL
);
desc Date_of_Admission
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **DATE_OF_ADMISSION**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
DATE_OF_ADMISSION	DA_ID	Number	-	5	0	1	-	-	-
	DOB	Date	7	-	-	-	-	-	-
	AGE	Number	-	3	0	-	-	-	-

1 - 3

Students_Hostels:

Autocommit Display 10

```
CREATE TABLE Students_Hostels (
    SH_id NUMBER(5) PRIMARY KEY,
    Std_id NUMBER(5) NOT NULL,
    Hostel_id NUMBER(5) NOT NULL,
    DA_id NUMBER(5) NOT NULL,
    FOREIGN KEY (Std_id) REFERENCES Students(Std_id),
    FOREIGN KEY (Hostel_id) REFERENCES Hostels(Hostel_id),
    FOREIGN KEY (DA_id) REFERENCES Date_of_Admission(DA_id)
);
desc Students_Hostels
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **STUDENTS_HOSTELS**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
STUDENTS_HOSTELS	SH_ID	Number	-	5	0	1	-	-	-
	STD_ID	Number	-	5	0	-	-	-	-
	HOSTEL_ID	Number	-	5	0	-	-	-	-
	DA_ID	Number	-	5	0	-	-	-	-

1 - 4

Students_Rooms:

Autocommit Display 10

```
CREATE TABLE Students_Rooms (
    SR_id NUMBER(5) PRIMARY KEY,
    Std_id NUMBER(5) NOT NULL,
    Room_id NUMBER(5) NOT NULL,
    DA_id NUMBER(5) NOT NULL,
    FOREIGN KEY (Std_id) REFERENCES Students(Std_id),
    FOREIGN KEY (Room_id) REFERENCES Rooms(Room_id),
    FOREIGN KEY (DA_id) REFERENCES Date_of_Admission(DA_id)
);
desc Students_Rooms
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **STUDENTS_ROOMS**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
STUDENTS_ROOMS	SR_ID	Number	-	5	0	1	-	-	-
	STD_ID	Number	-	5	0	-	-	-	-
	ROOM_ID	Number	-	5	0	-	-	-	-
	DA_ID	Number	-	5	0	-	-	-	-

1 - 4

Rooms:

Autocommit Display 10

```
CREATE TABLE Rooms (
    Room_id NUMBER(5) PRIMARY KEY,
    Room_type VARCHAR2(255) NOT NULL,
    Hostel_id NUMBER(5) NOT NULL,
    CR_id NUMBER(5) NOT NULL,
    FOREIGN KEY (Hostel_id) REFERENCES Hostels(Hostel_id),
    FOREIGN KEY (CR_id) REFERENCES Room_Capacity(CR_id)
);
desc rooms
```

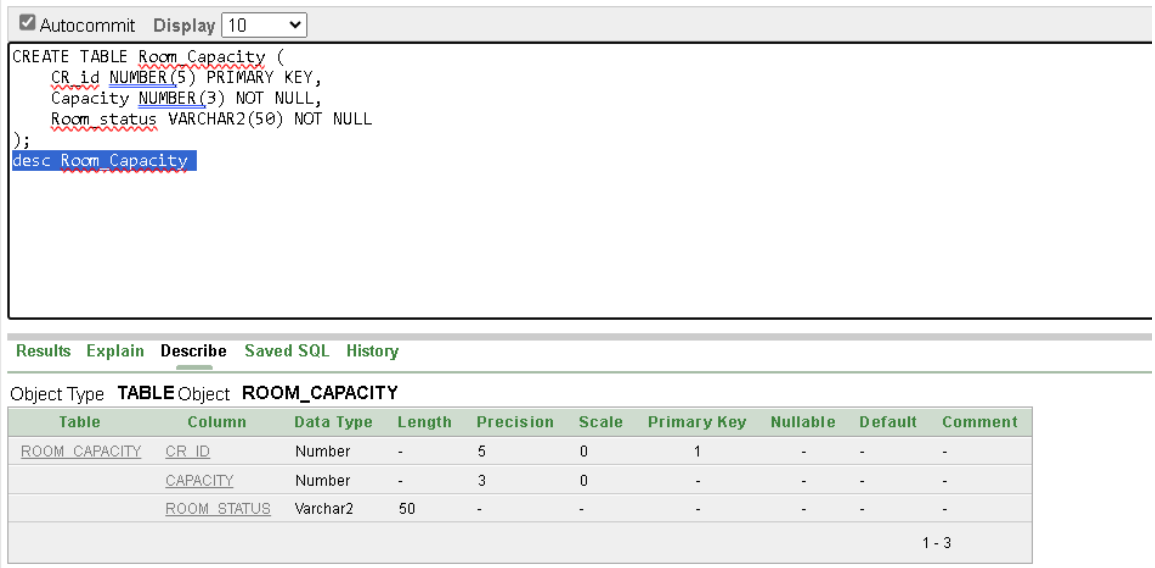
Results Explain Describe Saved SQL History

Object Type **TABLE** Object **ROOMS**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ROOMS	ROOM_ID	Number	-	5	0	1	-	-	-
	ROOM_TYPE	Varchar2	255	-	-	-	-	-	-
	HOSTEL_ID	Number	-	5	0	-	-	-	-
	CR_ID	Number	-	5	0	-	-	-	-

1 - 4

Room_Capacity :



Autocommit Display 10

```
CREATE TABLE Room_Capacity (
    CR_id NUMBER(5) PRIMARY KEY,
    Capacity NUMBER(3) NOT NULL,
    Room_status VARCHAR2(50) NOT NULL
);
desc Room_Capacity
```

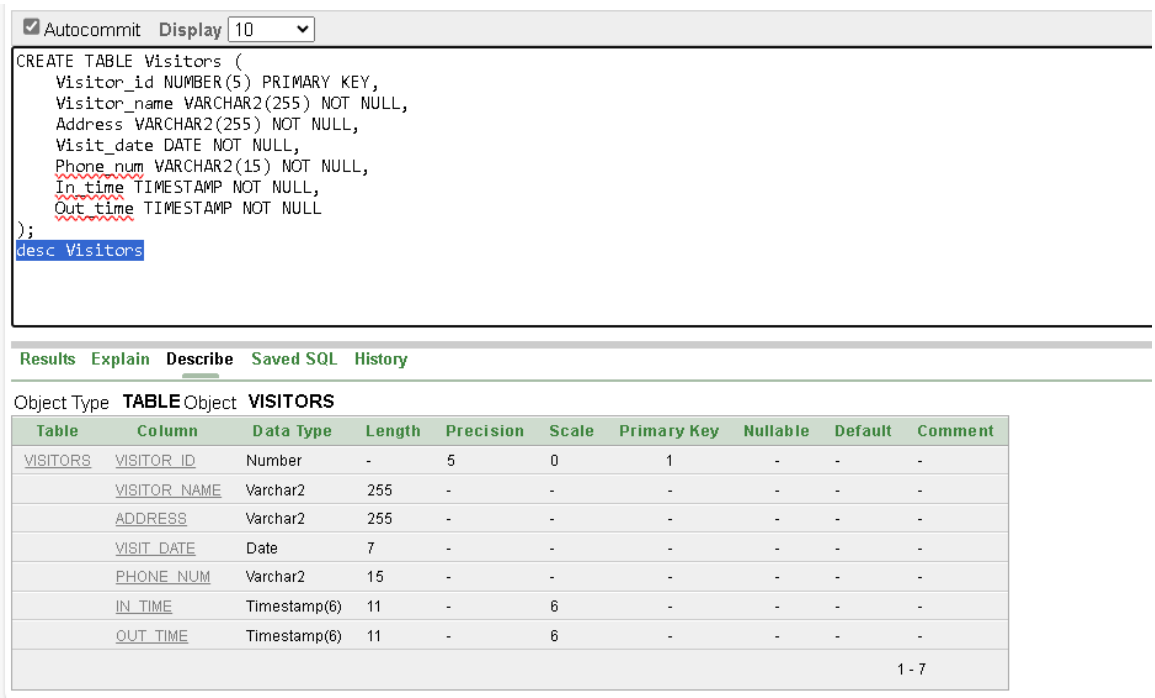
Results Explain Describe Saved SQL History

Object Type **TABLE** Object **ROOM_CAPACITY**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ROOM_CAPACITY	CR_ID	Number	-	5	0	1	-	-	-
	CAPACITY	Number	-	3	0	-	-	-	-
	ROOM_STATUS	Varchar2	50	-	-	-	-	-	-

1 - 3

Visitors:



Autocommit Display 10

```
CREATE TABLE Visitors (
    Visitor_id NUMBER(5) PRIMARY KEY,
    Visitor_name VARCHAR2(255) NOT NULL,
    Address VARCHAR2(255) NOT NULL,
    Visit_date DATE NOT NULL,
    Phone_num VARCHAR2(15) NOT NULL,
    In_time TIMESTAMP NOT NULL,
    Out_time TIMESTAMP NOT NULL
);
desc Visitors
```

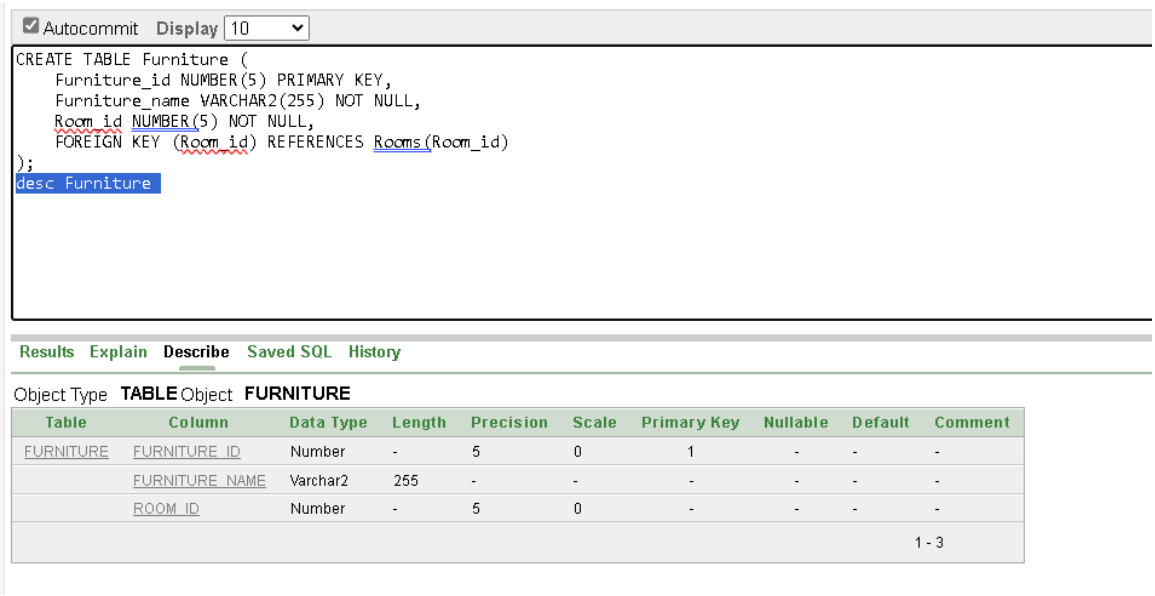
Results Explain Describe Saved SQL History

Object Type **TABLE** Object **VISITORS**

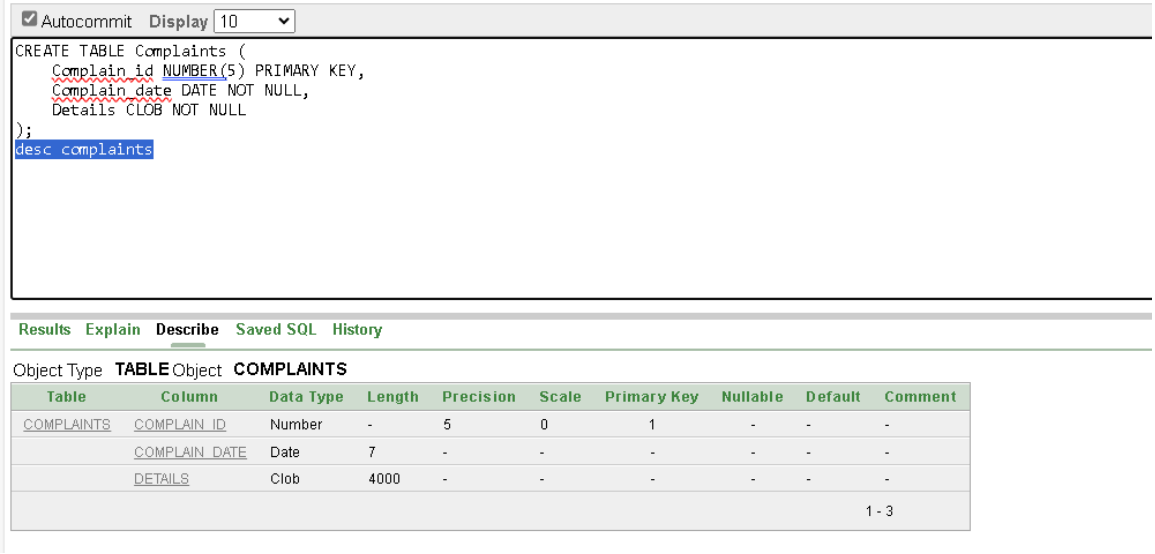
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
VISITORS	VISITOR_ID	Number	-	5	0	1	-	-	-
	VISITOR_NAME	Varchar2	255	-	-	-	-	-	-
	ADDRESS	Varchar2	255	-	-	-	-	-	-
	VISIT_DATE	Date	7	-	-	-	-	-	-
	PHONE_NUM	Varchar2	15	-	-	-	-	-	-
	IN_TIME	Timestamp(6)	11	-	6	-	-	-	-
	OUT_TIME	Timestamp(6)	11	-	6	-	-	-	-

1 - 7

Furniture :



Complaints:



Admin_Complaint:

Autocommit Display 10

```
CREATE TABLE Admin_Complaint (
    AC_id NUMBER(5) PRIMARY KEY,
    Admin_id NUMBER(5) NOT NULL,
    Complain_id NUMBER(5) NOT NULL,
    FOREIGN KEY (Admin_id) REFERENCES Admins(Admin_id),
    FOREIGN KEY (Complain_id) REFERENCES Complaints(Complain_id)
);
desc Admin_Complaint
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **ADMIN_COMPLAINT**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ADMIN_COMPLAINT	AC_ID	Number	-	5	0	1	-	-	-
	ADMIN_ID	Number	-	5	0	-	-	-	-
	COMPLAIN_ID	Number	-	5	0	-	-	-	-

1 - 3

Student_Complaint:

Autocommit Display 10

```
CREATE TABLE Student_Complaint (
    SC_id NUMBER(5) PRIMARY KEY,
    Std_id NUMBER(5) NOT NULL,
    Complain_id NUMBER(5) NOT NULL,
    FOREIGN KEY (Std_id) REFERENCES Students(Std_id),
    FOREIGN KEY (Complain_id) REFERENCES Complaints(Complain_id)
);
desc Student_Complaint
```

Results Explain Describe Saved SQL History

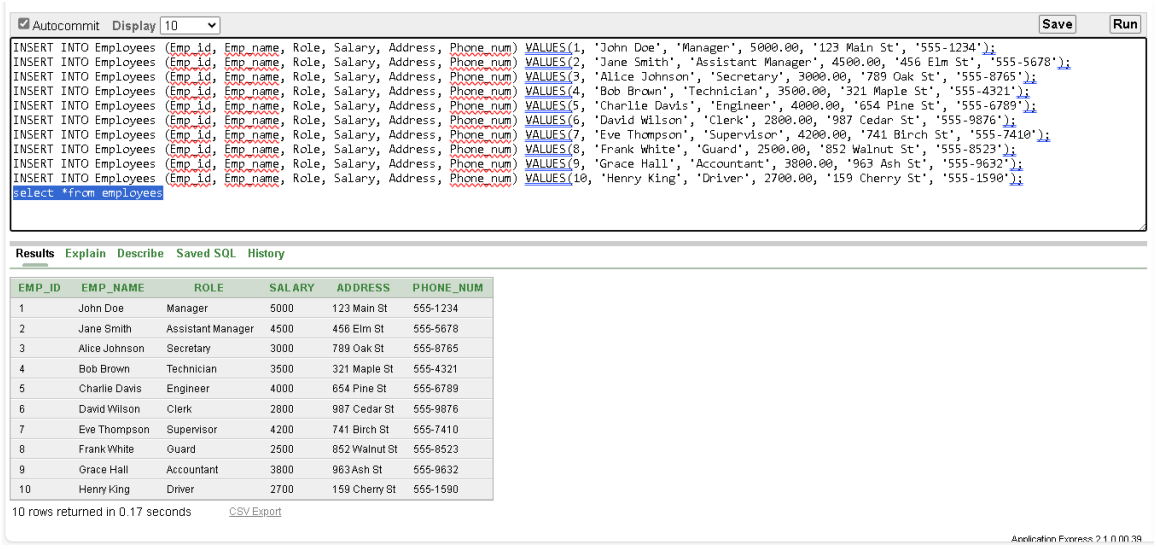
Object Type **TABLE** Object **STUDENT_COMPLAINT**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
STUDENT_COMPLAINT	SC_ID	Number	-	5	0	1	-	-	-
	STD_ID	Number	-	5	0	-	-	-	-
	COMPLAIN_ID	Number	-	5	0	-	-	-	-

1 - 3

Insertion

Employees:



Autocommit Display 10 Save Run

```
INSERT INTO Employees (Emp_id, Emp_name, Role, Salary, Address, Phone_num) VALUES(1, 'John Doe', 'Manager', 5000.00, '123 Main St', '555-1234');
INSERT INTO Employees (Emp_id, Emp_name, Role, Salary, Address, Phone_num) VALUES(2, 'Jane Smith', 'Assistant Manager', 4500.00, '456 Elm St', '555-5678');
INSERT INTO Employees (Emp_id, Emp_name, Role, Salary, Address, Phone_num) VALUES(3, 'Alice Johnson', 'Secretary', 3000.00, '789 Oak St', '555-8765');
INSERT INTO Employees (Emp_id, Emp_name, Role, Salary, Address, Phone_num) VALUES(4, 'Bob Brown', 'Technician', 3500.00, '321 Maple St', '555-4321');
INSERT INTO Employees (Emp_id, Emp_name, Role, Salary, Address, Phone_num) VALUES(5, 'Charlie Davis', 'Engineer', 4000.00, '654 Pine St', '555-6789');
INSERT INTO Employees (Emp_id, Emp_name, Role, Salary, Address, Phone_num) VALUES(6, 'David Wilson', 'Clerk', 2800.00, '987 Cedar St', '555-9876');
INSERT INTO Employees (Emp_id, Emp_name, Role, Salary, Address, Phone_num) VALUES(7, 'Eve Thompson', 'Supervisor', 4200.00, '741 Birch St', '555-7410');
INSERT INTO Employees (Emp_id, Emp_name, Role, Salary, Address, Phone_num) VALUES(8, 'Frank White', 'Guard', 2500.00, '852 Walnut St', '555-8523');
INSERT INTO Employees (Emp_id, Emp_name, Role, Salary, Address, Phone_num) VALUES(9, 'Grace Hall', 'Accountant', 3800.00, '963 Ash St', '555-9632');
INSERT INTO Employees (Emp_id, Emp_name, Role, Salary, Address, Phone_num) VALUES(10, 'Henry King', 'Driver', 2700.00, '159 Cherry St', '555-1590');
select *from employees
```

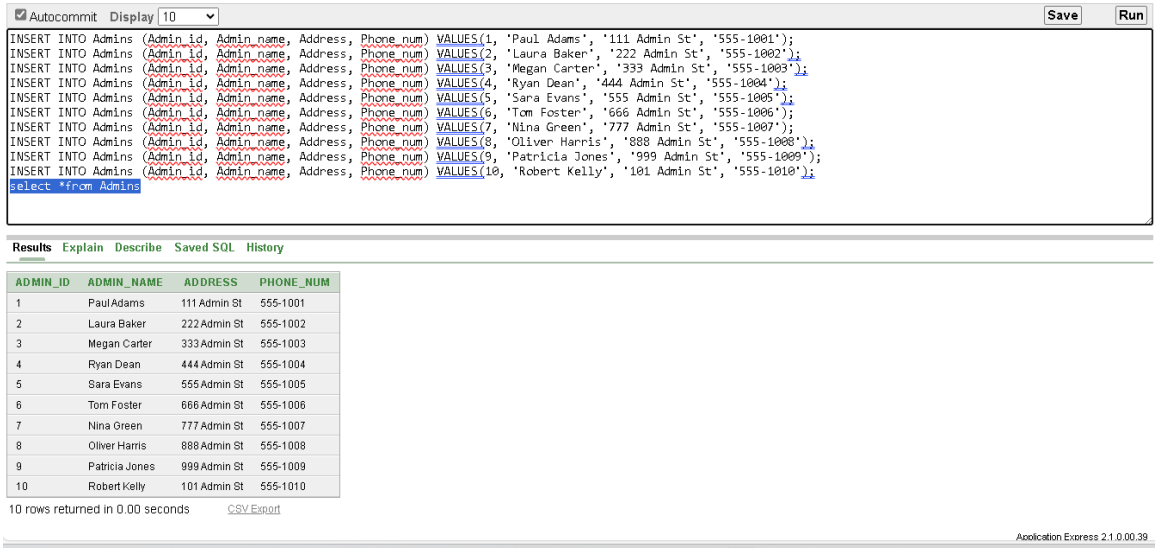
Results Explain Describe Saved SQL History

EMP_ID	EMP_NAME	ROLE	SALARY	ADDRESS	PHONE_NUM
1	John Doe	Manager	5000	123 Main St	555-1234
2	Jane Smith	Assistant Manager	4500	456 Elm St	555-5678
3	Alice Johnson	Secretary	3000	789 Oak St	555-8765
4	Bob Brown	Technician	3500	321 Maple St	555-4321
5	Charlie Davis	Engineer	4000	654 Pine St	555-6789
6	David Wilson	Clerk	2800	987 Cedar St	555-9876
7	Eve Thompson	Supervisor	4200	741 Birch St	555-7410
8	Frank White	Guard	2500	852 Walnut St	555-8523
9	Grace Hall	Accountant	3800	963 Ash St	555-9632
10	Henry King	Driver	2700	159 Cherry St	555-1590

10 rows returned in 0.17 seconds CSV Export

Application Express 21.0.00.39

Admin:



Autocommit Display 10 Save Run

```
INSERT INTO Admins (Admin_id, Admin_name, Address, Phone_num) VALUES(1, 'Paul Adams', '111 Admin St', '555-1001');
INSERT INTO Admins (Admin_id, Admin_name, Address, Phone_num) VALUES(2, 'Laura Baker', '222 Admin St', '555-1002');
INSERT INTO Admins (Admin_id, Admin_name, Address, Phone_num) VALUES(3, 'Megan Carter', '333 Admin St', '555-1003');
INSERT INTO Admins (Admin_id, Admin_name, Address, Phone_num) VALUES(4, 'Ryan Dean', '444 Admin St', '555-1004');
INSERT INTO Admins (Admin_id, Admin_name, Address, Phone_num) VALUES(5, 'Sara Evans', '555 Admin St', '555-1005');
INSERT INTO Admins (Admin_id, Admin_name, Address, Phone_num) VALUES(6, 'Tom Foster', '666 Admin St', '555-1006');
INSERT INTO Admins (Admin_id, Admin_name, Address, Phone_num) VALUES(7, 'Nina Green', '777 Admin St', '555-1007');
INSERT INTO Admins (Admin_id, Admin_name, Address, Phone_num) VALUES(8, 'Oliver Harris', '888 Admin St', '555-1008');
INSERT INTO Admins (Admin_id, Admin_name, Address, Phone_num) VALUES(9, 'Patricia Jones', '999 Admin St', '555-1009');
INSERT INTO Admins (Admin_id, Admin_name, Address, Phone_num) VALUES(10, 'Robert Kelly', '101 Admin St', '555-1010');
select *from Admins
```

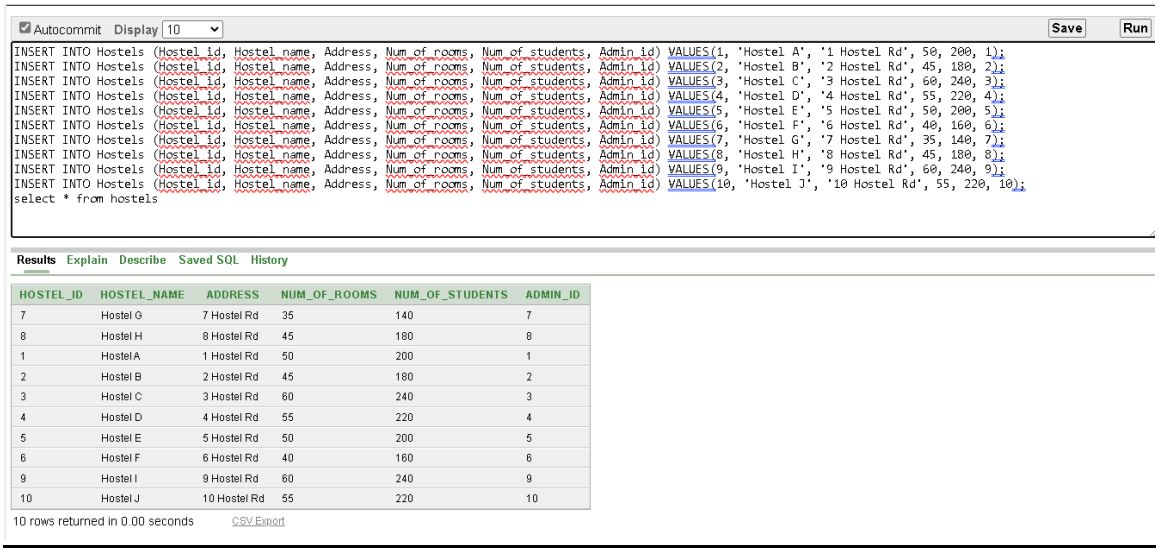
Results Explain Describe Saved SQL History

ADMIN_ID	ADMIN_NAME	ADDRESS	PHONE_NUM
1	Paul Adams	111 Admin St	555-1001
2	Laura Baker	222 Admin St	555-1002
3	Megan Carter	333 Admin St	555-1003
4	Ryan Dean	444 Admin St	555-1004
5	Sara Evans	555 Admin St	555-1005
6	Tom Foster	666 Admin St	555-1006
7	Nina Green	777 Admin St	555-1007
8	Oliver Harris	888 Admin St	555-1008
9	Patricia Jones	999 Admin St	555-1009
10	Robert Kelly	101 Admin St	555-1010

10 rows returned in 0.00 seconds CSV Export

Application Express 21.0.00.39

Hostels:

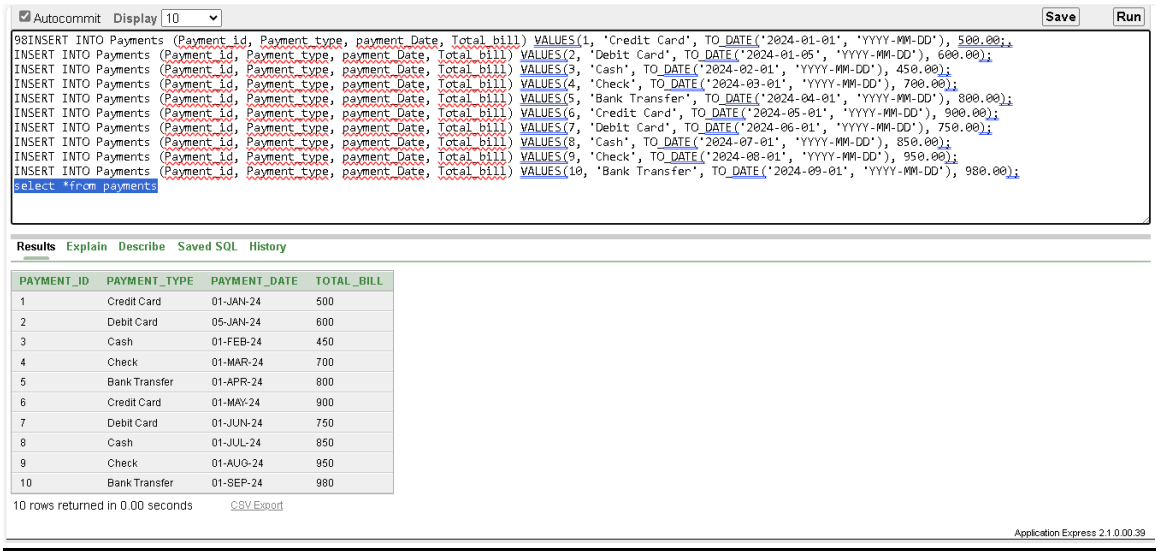


```
INSERT INTO Hostels (Hostel_id, Hostel_name, Address, Num_of_rooms, Num_of_students, Admin_id) VALUES(1, 'Hostel A', '1 Hostel Rd', 50, 200, 1);
INSERT INTO Hostels (Hostel_id, Hostel_name, Address, Num_of_rooms, Num_of_students, Admin_id) VALUES(2, 'Hostel B', '2 Hostel Rd', 45, 180, 2);
INSERT INTO Hostels (Hostel_id, Hostel_name, Address, Num_of_rooms, Num_of_students, Admin_id) VALUES(3, 'Hostel C', '3 Hostel Rd', 60, 240, 3);
INSERT INTO Hostels (Hostel_id, Hostel_name, Address, Num_of_rooms, Num_of_students, Admin_id) VALUES(4, 'Hostel D', '4 Hostel Rd', 55, 220, 4);
INSERT INTO Hostels (Hostel_id, Hostel_name, Address, Num_of_rooms, Num_of_students, Admin_id) VALUES(5, 'Hostel E', '5 Hostel Rd', 50, 200, 5);
INSERT INTO Hostels (Hostel_id, Hostel_name, Address, Num_of_rooms, Num_of_students, Admin_id) VALUES(6, 'Hostel F', '6 Hostel Rd', 40, 160, 6);
INSERT INTO Hostels (Hostel_id, Hostel_name, Address, Num_of_rooms, Num_of_students, Admin_id) VALUES(7, 'Hostel G', '7 Hostel Rd', 35, 140, 7);
INSERT INTO Hostels (Hostel_id, Hostel_name, Address, Num_of_rooms, Num_of_students, Admin_id) VALUES(8, 'Hostel H', '8 Hostel Rd', 45, 180, 8);
INSERT INTO Hostels (Hostel_id, Hostel_name, Address, Num_of_rooms, Num_of_students, Admin_id) VALUES(9, 'Hostel I', '9 Hostel Rd', 60, 240, 9);
INSERT INTO Hostels (Hostel_id, Hostel_name, Address, Num_of_rooms, Num_of_students, Admin_id) VALUES(10, 'Hostel J', '10 Hostel Rd', 55, 220, 10);
select * from hostels
```

HOSTEL_ID	HOSTEL_NAME	ADDRESS	NUM_OF_ROOMS	NUM_OF_STUDENTS	ADMIN_ID
7	Hostel G	7 Hostel Rd	35	140	7
8	Hostel H	8 Hostel Rd	45	180	8
1	Hostel A	1 Hostel Rd	50	200	1
2	Hostel B	2 Hostel Rd	45	180	2
3	Hostel C	3 Hostel Rd	60	240	3
4	Hostel D	4 Hostel Rd	55	220	4
5	Hostel E	5 Hostel Rd	50	200	5
6	Hostel F	6 Hostel Rd	40	160	6
9	Hostel I	9 Hostel Rd	60	240	9
10	Hostel J	10 Hostel Rd	55	220	10

10 rows returned in 0.00 seconds

Payments:

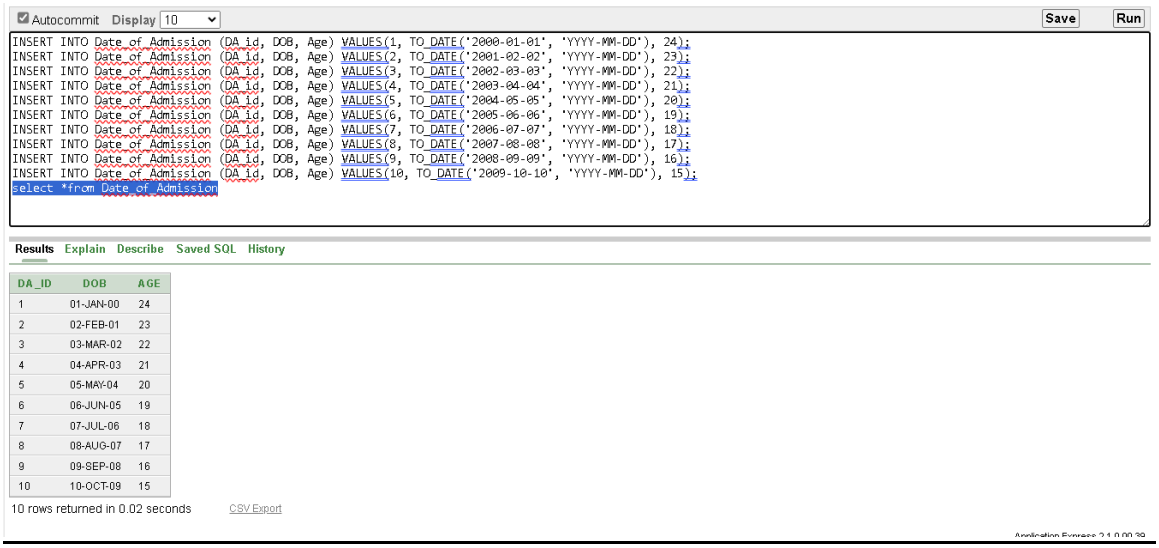


```
98INSERT INTO Payments (Payment_id, Payment_type, payment_Date, Total_bill) VALUES(1, 'Credit Card', TO_DATE('2024-01-01', 'YYYY-MM-DD'), 500.00;
INSERT INTO Payments (Payment_id, Payment_type, payment_Date, Total_bill) VALUES(2, 'Debit Card', TO_DATE('2024-01-05', 'YYYY-MM-DD'), 600.00);
INSERT INTO Payments (Payment_id, Payment_type, payment_Date, Total_bill) VALUES(3, 'Cash', TO_DATE('2024-02-01', 'YYYY-MM-DD'), 450.00);
INSERT INTO Payments (Payment_id, Payment_type, payment_Date, Total_bill) VALUES(4, 'Check', TO_DATE('2024-03-01', 'YYYY-MM-DD'), 700.00);
INSERT INTO Payments (Payment_id, Payment_type, payment_Date, Total_bill) VALUES(5, 'Bank Transfer', TO_DATE('2024-04-01', 'YYYY-MM-DD'), 800.00);
INSERT INTO Payments (Payment_id, Payment_type, payment_Date, Total_bill) VALUES(6, 'Credit Card', TO_DATE('2024-05-01', 'YYYY-MM-DD'), 900.00);
INSERT INTO Payments (Payment_id, Payment_type, payment_Date, Total_bill) VALUES(7, 'Debit Card', TO_DATE('2024-06-01', 'YYYY-MM-DD'), 750.00);
INSERT INTO Payments (Payment_id, Payment_type, payment_Date, Total_bill) VALUES(8, 'Cash', TO_DATE('2024-07-01', 'YYYY-MM-DD'), 850.00);
INSERT INTO Payments (Payment_id, Payment_type, payment_Date, Total_bill) VALUES(9, 'Check', TO_DATE('2024-08-01', 'YYYY-MM-DD'), 950.00);
INSERT INTO Payments (Payment_id, Payment_type, payment_Date, Total_bill) VALUES(10, 'Bank Transfer', TO_DATE('2024-09-01', 'YYYY-MM-DD'), 980.00);
select *from payments
```

PAYMENT_ID	PAYMENT_TYPE	PAYMENT_DATE	TOTAL_BILL
1	Credit Card	01-JAN-24	500
2	Debit Card	05-JAN-24	600
3	Cash	01-FEB-24	450
4	Check	01-MAR-24	700
5	Bank Transfer	01-APR-24	800
6	Credit Card	01-MAY-24	900
7	Debit Card	01-JUN-24	750
8	Cash	01-JUL-24	850
9	Check	01-AUG-24	950
10	Bank Transfer	01-SEP-24	980

10 rows returned in 0.00 seconds

Date_of_Admission:

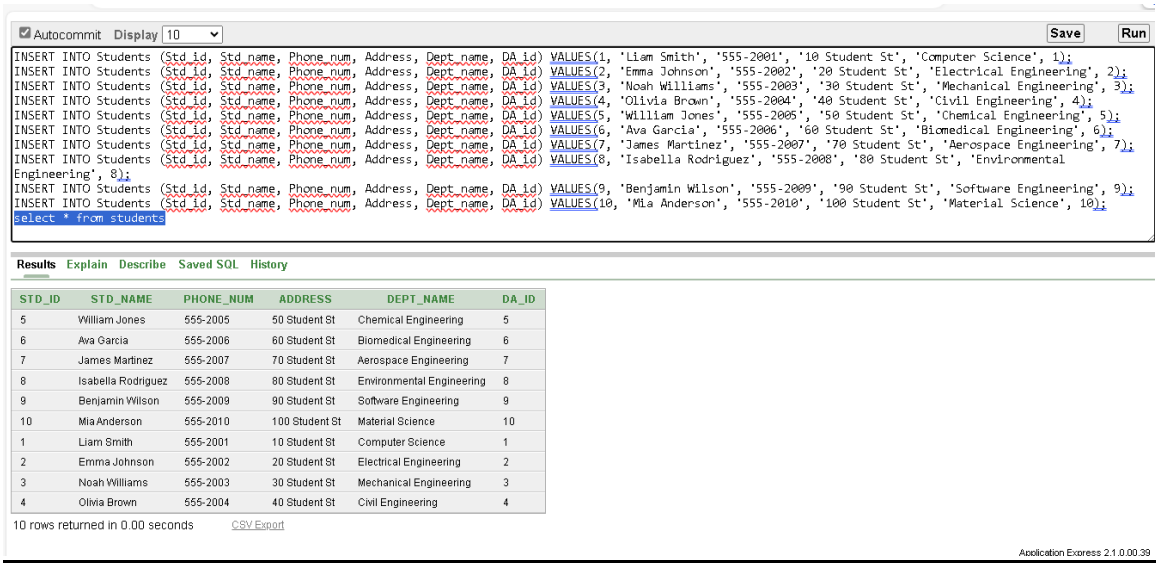


```
INSERT INTO Date_of_Admission (DA_id, DOB, Age) VALUES(1, TO_DATE('2000-01-01', 'YYYY-MM-DD'), 24);
INSERT INTO Date_of_Admission (DA_id, DOB, Age) VALUES(2, TO_DATE('2001-02-02', 'YYYY-MM-DD'), 23);
INSERT INTO Date_of_Admission (DA_id, DOB, Age) VALUES(3, TO_DATE('2002-03-03', 'YYYY-MM-DD'), 22);
INSERT INTO Date_of_Admission (DA_id, DOB, Age) VALUES(4, TO_DATE('2003-04-04', 'YYYY-MM-DD'), 21);
INSERT INTO Date_of_Admission (DA_id, DOB, Age) VALUES(5, TO_DATE('2004-05-05', 'YYYY-MM-DD'), 20);
INSERT INTO Date_of_Admission (DA_id, DOB, Age) VALUES(6, TO_DATE('2005-06-06', 'YYYY-MM-DD'), 19);
INSERT INTO Date_of_Admission (DA_id, DOB, Age) VALUES(7, TO_DATE('2006-07-07', 'YYYY-MM-DD'), 18);
INSERT INTO Date_of_Admission (DA_id, DOB, Age) VALUES(8, TO_DATE('2007-08-08', 'YYYY-MM-DD'), 17);
INSERT INTO Date_of_Admission (DA_id, DOB, Age) VALUES(9, TO_DATE('2008-09-09', 'YYYY-MM-DD'), 16);
INSERT INTO Date_of_Admission (DA_id, DOB, Age) VALUES(10, TO_DATE('2009-10-10', 'YYYY-MM-DD'), 15);
select *from Date_of_Admission
```

DA_ID	DOB	AGE
1	01-JAN-00	24
2	02-FEB-01	23
3	03-MAR-02	22
4	04-APR-03	21
5	05-MAY-04	20
6	06-JUN-05	19
7	07-JUL-06	18
8	08-AUG-07	17
9	09-SEP-08	16
10	10-OCT-09	15

10 rows returned in 0.02 seconds

Students:

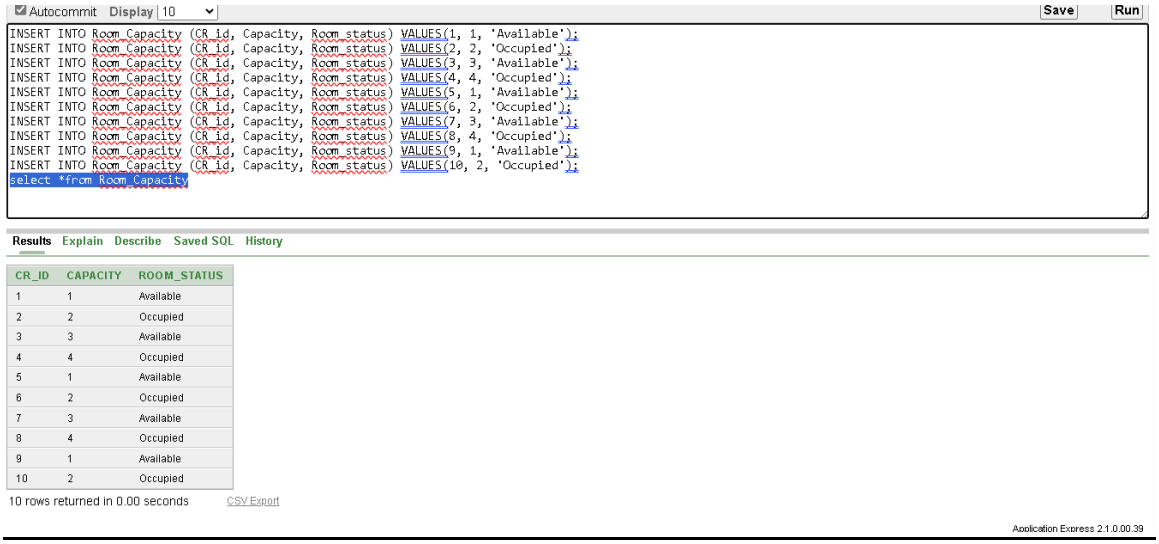


```
INSERT INTO Students (Std_id, Std_name, Phone_num, Address, Dept_name, DA_id) VALUES(1, 'Liam Smith', '555-2001', '10 Student St', 'Computer Science', 1);
INSERT INTO Students (Std_id, Std_name, Phone_num, Address, Dept_name, DA_id) VALUES(2, 'Emma Johnson', '555-2002', '20 Student St', 'Electrical Engineering', 2);
INSERT INTO Students (Std_id, Std_name, Phone_num, Address, Dept_name, DA_id) VALUES(3, 'Noah Williams', '555-2003', '30 Student St', 'Mechanical Engineering', 3);
INSERT INTO Students (Std_id, Std_name, Phone_num, Address, Dept_name, DA_id) VALUES(4, 'Olivia Brown', '555-2004', '40 Student St', 'Civil Engineering', 4);
INSERT INTO Students (Std_id, Std_name, Phone_num, Address, Dept_name, DA_id) VALUES(5, 'William Jones', '555-2005', '50 Student St', 'Chemical Engineering', 5);
INSERT INTO Students (Std_id, Std_name, Phone_num, Address, Dept_name, DA_id) VALUES(6, 'Ava Garcia', '555-2006', '60 Student St', 'Biomedical Engineering', 6);
INSERT INTO Students (Std_id, Std_name, Phone_num, Address, Dept_name, DA_id) VALUES(7, 'James Martinez', '555-2007', '70 Student St', 'Aerospace Engineering', 7);
INSERT INTO Students (Std_id, Std_name, Phone_num, Address, Dept_name, DA_id) VALUES(8, 'Isabella Rodriguez', '555-2008', '80 Student St', 'Environmental
Engineering', 8);
INSERT INTO Students (Std_id, Std_name, Phone_num, Address, Dept_name, DA_id) VALUES(9, 'Benjamin Wilson', '555-2009', '90 Student St', 'Software Engineering', 9);
INSERT INTO Students (Std_id, Std_name, Phone_num, Address, Dept_name, DA_id) VALUES(10, 'Mia Anderson', '555-2010', '100 Student St', 'Material Science', 10);
select * from students
```

STD_ID	STD_NAME	PHONE_NUM	ADDRESS	DEPT_NAME	DA_ID
5	William Jones	555-2005	50 Student St	Chemical Engineering	5
6	Ava Garcia	555-2006	60 Student St	Biomedical Engineering	6
7	James Martinez	555-2007	70 Student St	Aerospace Engineering	7
8	Isabella Rodriguez	555-2008	80 Student St	Environmental Engineering	8
9	Benjamin Wilson	555-2009	90 Student St	Software Engineering	9
10	Mia Anderson	555-2010	100 Student St	Material Science	10
1	Liam Smith	555-2001	10 Student St	Computer Science	1
2	Emma Johnson	555-2002	20 Student St	Electrical Engineering	2
3	Noah Williams	555-2003	30 Student St	Mechanical Engineering	3
4	Olivia Brown	555-2004	40 Student St	Civil Engineering	4

10 rows returned in 0.00 seconds

Room_Capacity:

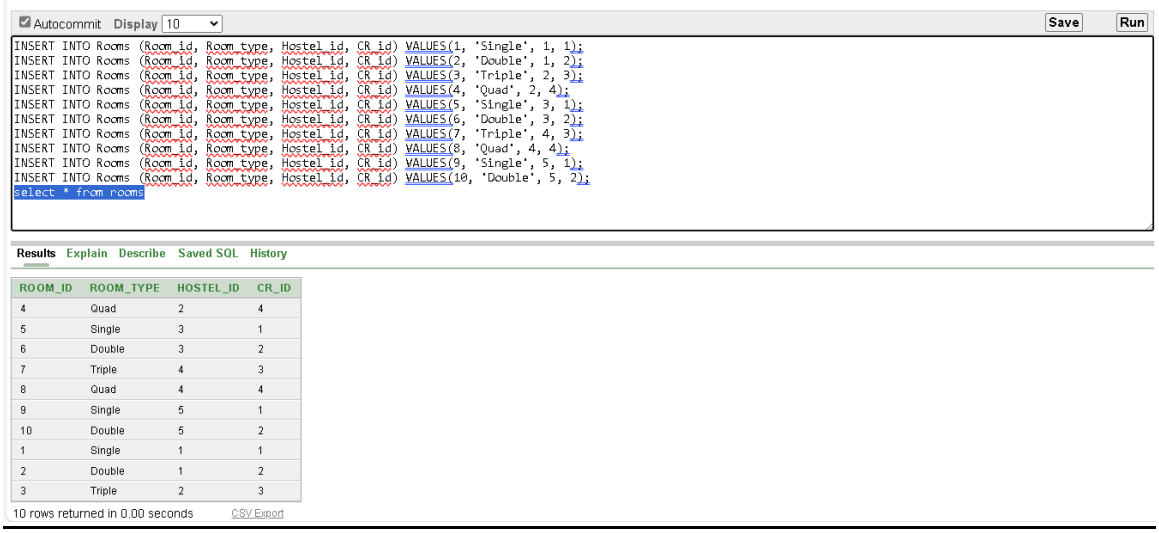


```
INSERT INTO Room_Capacity (CR_id, Capacity, Room_status) VALUES(1, 1, 'Available');
INSERT INTO Room_Capacity (CR_id, Capacity, Room_status) VALUES(2, 2, 'Occupied');
INSERT INTO Room_Capacity (CR_id, Capacity, Room_status) VALUES(3, 3, 'Available');
INSERT INTO Room_Capacity (CR_id, Capacity, Room_status) VALUES(4, 4, 'Occupied');
INSERT INTO Room_Capacity (CR_id, Capacity, Room_status) VALUES(5, 1, 'Available');
INSERT INTO Room_Capacity (CR_id, Capacity, Room_status) VALUES(6, 2, 'Occupied');
INSERT INTO Room_Capacity (CR_id, Capacity, Room_status) VALUES(7, 3, 'Available');
INSERT INTO Room_Capacity (CR_id, Capacity, Room_status) VALUES(8, 4, 'Occupied');
INSERT INTO Room_Capacity (CR_id, Capacity, Room_status) VALUES(9, 1, 'Available');
INSERT INTO Room_Capacity (CR_id, Capacity, Room_status) VALUES(10, 2, 'Occupied');
select *from Room_Capacity
```

CR_ID	CAPACITY	ROOM_STATUS
1	1	Available
2	2	Occupied
3	3	Available
4	4	Occupied
5	1	Available
6	2	Occupied
7	3	Available
8	4	Occupied
9	1	Available
10	2	Occupied

10 rows returned in 0.00 seconds

Room:



```
INSERT INTO Rooms (Room_id, Room_type, Hostel_id, CR_id) VALUES(1, 'Single', 1, 1);
INSERT INTO Rooms (Room_id, Room_type, Hostel_id, CR_id) VALUES(2, 'Double', 1, 2);
INSERT INTO Rooms (Room_id, Room_type, Hostel_id, CR_id) VALUES(3, 'Triple', 2, 3);
INSERT INTO Rooms (Room_id, Room_type, Hostel_id, CR_id) VALUES(4, 'Quad', 2, 4);
INSERT INTO Rooms (Room_id, Room_type, Hostel_id, CR_id) VALUES(5, 'Single', 3, 1);
INSERT INTO Rooms (Room_id, Room_type, Hostel_id, CR_id) VALUES(6, 'Double', 3, 2);
INSERT INTO Rooms (Room_id, Room_type, Hostel_id, CR_id) VALUES(7, 'Triple', 4, 3);
INSERT INTO Rooms (Room_id, Room_type, Hostel_id, CR_id) VALUES(8, 'Quad', 4, 4);
INSERT INTO Rooms (Room_id, Room_type, Hostel_id, CR_id) VALUES(9, 'Single', 5, 1);
INSERT INTO Rooms (Room_id, Room_type, Hostel_id, CR_id) VALUES(10, 'Double', 5, 2);
select * from rooms
```

ROOM_ID	ROOM_TYPE	HOSTEL_ID	CR_ID
4	Quad	2	4
5	Single	3	1
6	Double	3	2
7	Triple	4	3
8	Quad	4	4
9	Single	5	1
10	Double	5	2
1	Single	1	1
2	Double	1	2
3	Triple	2	3

10 rows returned in 0.00 seconds

Visitors:

Autocommit Display 10 Save Run

```
'YYYY-MM-DD'), '555-3005', TO_TIMESTAMP('2024-05-19 14:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('2024-05-19 16:00:00', 'YYYY-MM-DD HH24:MI:SS'));
INSERT INTO Visitors (Visitor_id, Visitor_name, Address, Visit_date, Phone_num, In_time, Out_time) VALUES(6, 'Drew Lee', '106 Visitor Ln', TO_DATE('2024-06-20',
'YYYY-MM-DD'), '555-3006', TO_TIMESTAMP('2024-06-20 15:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('2024-06-20 17:00:00', 'YYYY-MM-DD HH24:MI:SS'));
INSERT INTO Visitors (Visitor_id, Visitor_name, Address, Visit_date, Phone_num, In_time, Out_time) VALUES(7, 'Taylor Chris', '107 Visitor Ln', TO_DATE('2024-07-
21', 'YYYY-MM-DD'), '555-3007', TO_TIMESTAMP('2024-07-21 16:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('2024-07-21 18:00:00', 'YYYY-MM-DD HH24:MI:SS'));
INSERT INTO Visitors (Visitor_id, Visitor_name, Address, Visit_date, Phone_num, In_time, Out_time) VALUES(8, 'Morgan Pat', '108 Visitor Ln', TO_DATE('2024-08-22',
'YYYY-MM-DD'), '555-3008', TO_TIMESTAMP('2024-08-22 17:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('2024-08-22 19:00:00', 'YYYY-MM-DD HH24:MI:SS'));
INSERT INTO Visitors (Visitor_id, Visitor_name, Address, Visit_date, Phone_num, In_time, Out_time) VALUES(9, 'Lee Jordan', '109 Visitor Ln', TO_DATE('2024-09-23',
'YYYY-MM-DD'), '555-3009', TO_TIMESTAMP('2024-09-23 18:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('2024-09-23 20:00:00', 'YYYY-MM-DD HH24:MI:SS'));
INSERT INTO Visitors (Visitor_id, Visitor_name, Address, Visit_date, Phone_num, In_time, Out_time) VALUES(10, 'Taylor Sam', '110 Visitor Ln', TO_DATE('2024-10-24',
'YYYY-MM-DD'), '555-3010', TO_TIMESTAMP('2024-10-24 19:00:00', 'YYYY-MM-DD HH24:MI:SS'), TO_TIMESTAMP('2024-10-24 21:00:00', 'YYYY-MM-DD HH24:MI:SS'));
select * from visitors
```

Results Explain Describe Saved SQL History

VISITOR_ID	VISITOR_NAME	ADDRESS	VISIT_DATE	PHONE_NUM	IN_TIME	OUT_TIME
2	Pat Morgan	102 Visitor Ln	16-FEB-24	555-3002	16-FEB-24 11.00.00.000000 AM	16-FEB-24 01.00.00.000000 PM
3	Jordan Lee	103 Visitor Ln	17-MAR-24	555-3003	17-MAR-24 12.00.00.000000 PM	17-MAR-24 02.00.00.000000 PM
4	Sam Taylor	104 Visitor Ln	18-APR-24	555-3004	18-APR-24 01.00.00.000000 PM	18-APR-24 03.00.00.000000 PM
5	Alex Morgan	105 Visitor Ln	19-MAY-24	555-3005	19-MAY-24 02.00.00.000000 PM	19-MAY-24 04.00.00.000000 PM
6	Drew Lee	106 Visitor Ln	20-JUN-24	555-3006	20-JUN-24 03.00.00.000000 PM	20-JUN-24 05.00.00.000000 PM
7	Taylor Chris	107 Visitor Ln	21-JUL-24	555-3007	21-JUL-24 04.00.00.000000 PM	21-JUL-24 06.00.00.000000 PM
8	Morgan Pat	108 Visitor Ln	22-AUG-24	555-3008	22-AUG-24 05.00.00.000000 PM	22-AUG-24 07.00.00.000000 PM
9	Lee Jordan	109 Visitor Ln	23-SEP-24	555-3009	23-SEP-24 06.00.00.000000 PM	23-SEP-24 08.00.00.000000 PM
10	Taylor Sam	110 Visitor Ln	24-OCT-24	555-3010	24-OCT-24 07.00.00.000000 PM	24-OCT-24 09.00.00.000000 PM
1	Chris Taylor	101 Visitor Ln	15-JAN-24	555-3001	15-JAN-24 10.00.00.000000 AM	15-JAN-24 12.00.00.000000 PM

10 rows returned in 0.03 seconds CSV Export

Furniture:

Autocommit Display 10 Save Run

```
INSERT INTO Furniture (Furniture_id, Furniture_name, Room_id) VALUES(1, 'Bed', 1);
INSERT INTO Furniture (Furniture_id, Furniture_name, Room_id) VALUES(2, 'Table', 2);
INSERT INTO Furniture (Furniture_id, Furniture_name, Room_id) VALUES(3, 'Chair', 3);
INSERT INTO Furniture (Furniture_id, Furniture_name, Room_id) VALUES(4, 'Wardrobe', 4);
INSERT INTO Furniture (Furniture_id, Furniture_name, Room_id) VALUES(5, 'Desk', 5);
INSERT INTO Furniture (Furniture_id, Furniture_name, Room_id) VALUES(6, 'Sofa', 6);
INSERT INTO Furniture (Furniture_id, Furniture_name, Room_id) VALUES(7, 'Lamp', 7);
INSERT INTO Furniture (Furniture_id, Furniture_name, Room_id) VALUES(8, 'Bookshelf', 8);
INSERT INTO Furniture (Furniture_id, Furniture_name, Room_id) VALUES(9, 'Dresser', 9);
INSERT INTO Furniture (Furniture_id, Furniture_name, Room_id) VALUES(10, 'Mirror', 10);
select *from furniture
```

Results Explain Describe Saved SQL History

FURNITURE_ID	FURNITURE_NAME	ROOM_ID
7	Lamp	7
8	Bookshelf	8
9	Dresser	9
10	Mirror	10
1	Bed	1
2	Table	2
3	Chair	3
4	Wardrobe	4
5	Desk	5
6	Sofa	6

10 rows returned in 0.01 seconds CSV Export

Complaints:

```
INSERT INTO Complaints (Complain_id, Complain_date, Details) VALUES(1, TO_DATE('2024-01-10', 'YYYY-MM-DD'), 'Water leakage in room');
INSERT INTO Complaints (Complain_id, Complain_date, Details) VALUES(2, TO_DATE('2024-02-11', 'YYYY-MM-DD'), 'Electricity issue in hall');
INSERT INTO Complaints (Complain_id, Complain_date, Details) VALUES(3, TO_DATE('2024-03-12', 'YYYY-MM-DD'), 'Food quality problem in mess');
INSERT INTO Complaints (Complain_id, Complain_date, Details) VALUES(4, TO_DATE('2024-04-13', 'YYYY-MM-DD'), 'Noise problem during night');
INSERT INTO Complaints (Complain_id, Complain_date, Details) VALUES(5, TO_DATE('2024-05-14', 'YYYY-MM-DD'), 'AC not working in room');
INSERT INTO Complaints (Complain_id, Complain_date, Details) VALUES(6, TO_DATE('2024-06-15', 'YYYY-MM-DD'), 'Internet connectivity issue');
INSERT INTO Complaints (Complain_id, Complain_date, Details) VALUES(7, TO_DATE('2024-07-16', 'YYYY-MM-DD'), 'Broken furniture in common area');
INSERT INTO Complaints (Complain_id, Complain_date, Details) VALUES(8, TO_DATE('2024-08-17', 'YYYY-MM-DD'), 'Unclean washrooms');
INSERT INTO Complaints (Complain_id, Complain_date, Details) VALUES(9, TO_DATE('2024-09-18', 'YYYY-MM-DD'), 'Lighting issue in study room');
INSERT INTO Complaints (Complain_id, Complain_date, Details) VALUES(10, TO_DATE('2024-10-19', 'YYYY-MM-DD'), 'Security concerns in hostel premises');
select *from complaints
```

Results Explain Describe Saved SQL History

COMPLAIN_ID	COMPLAIN_DATE	DETAILS
1	10-JAN-24	Water leakage in room
2	11-FEB-24	Electricity issue in hall
3	12-MAR-24	Food quality problem in mess
4	13-APR-24	Noise problem during night
5	14-MAY-24	AC not working in room
6	15-JUN-24	Internet connectivity issue
7	16-JUL-24	Broken furniture in common area
8	17-AUG-24	Unclean washrooms
9	18-SEP-24	Lighting issue in study room
10	19-OCT-24	Security concerns in hostel premises

10 rows returned in 0.00 seconds CSV Export

Employee_Hostel:

Autocommit Display 10 Save Run

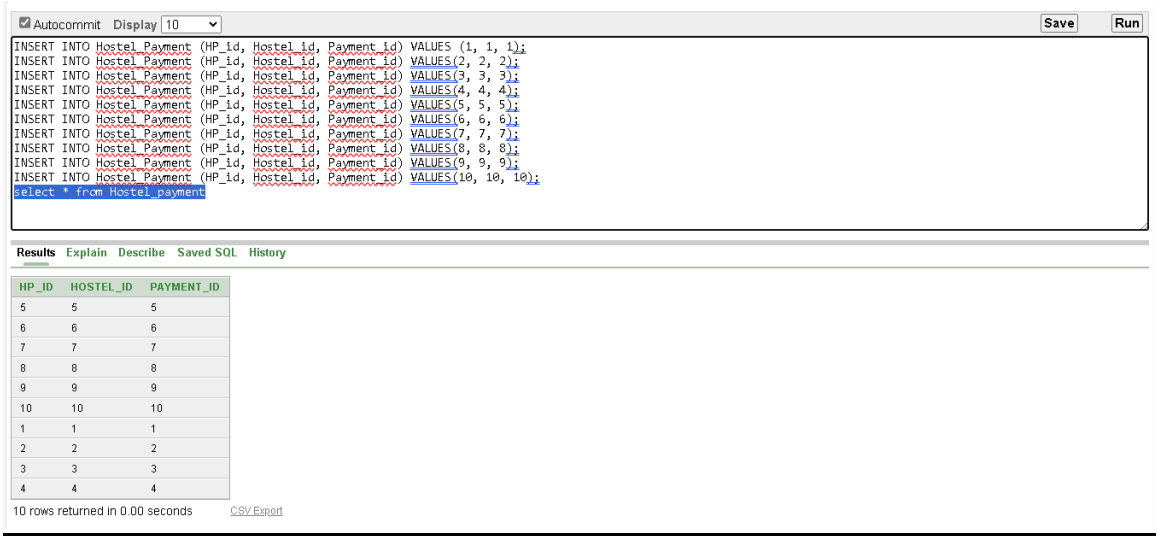
```
INSERT INTO Employee_Hostel (EH_id, Emp_id, Hostel_id) VALUES(1, 1, 1);
INSERT INTO Employee_Hostel (EH_id, Emp_id, Hostel_id) VALUES(2, 2, 2);
INSERT INTO Employee_Hostel (EH_id, Emp_id, Hostel_id) VALUES (3, 3, 3);
INSERT INTO Employee_Hostel (EH_id, Emp_id, Hostel_id) VALUES (4, 4, 4);
INSERT INTO Employee_Hostel (EH_id, Emp_id, Hostel_id) VALUES (5, 5, 5);
INSERT INTO Employee_Hostel (EH_id, Emp_id, Hostel_id) VALUES (6, 6, 6);
INSERT INTO Employee_Hostel (EH_id, Emp_id, Hostel_id) VALUES (7, 7, 7);
INSERT INTO Employee_Hostel (EH_id, Emp_id, Hostel_id) VALUES (8, 8, 8);
INSERT INTO Employee_Hostel (EH_id, Emp_id, Hostel_id) VALUES (9, 9, 9);
INSERT INTO Employee_Hostel (EH_id, Emp_id, Hostel_id) VALUES (10, 10, 10);
select * from Employee_Hostel
```

Results Explain Describe Saved SQL History

EH_ID	EMP_ID	HOSTEL_ID
3	3	3
4	4	4
5	5	5
1	1	1
2	2	2
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10

10 rows returned in 0.00 seconds CSV Export

Hostel_payment:



```
INSERT INTO Hostel_Payment (HP_id, Hostel_id, Payment_id) VALUES (1, 1, 1);
INSERT INTO Hostel_Payment (HP_id, Hostel_id, Payment_id) VALUES(2, 2, 2);
INSERT INTO Hostel_Payment (HP_id, Hostel_id, Payment_id) VALUES(3, 3, 3);
INSERT INTO Hostel_Payment (HP_id, Hostel_id, Payment_id) VALUES(4, 4, 4);
INSERT INTO Hostel_Payment (HP_id, Hostel_id, Payment_id) VALUES(5, 5, 5);
INSERT INTO Hostel_Payment (HP_id, Hostel_id, Payment_id) VALUES(6, 6, 6);
INSERT INTO Hostel_Payment (HP_id, Hostel_id, Payment_id) VALUES(7, 7, 7);
INSERT INTO Hostel_Payment (HP_id, Hostel_id, Payment_id) VALUES(8, 8, 8);
INSERT INTO Hostel_Payment (HP_id, Hostel_id, Payment_id) VALUES(9, 9, 9);
INSERT INTO Hostel_Payment (HP_id, Hostel_id, Payment_id) VALUES(10, 10, 10);
select * from Hostel_payment
```

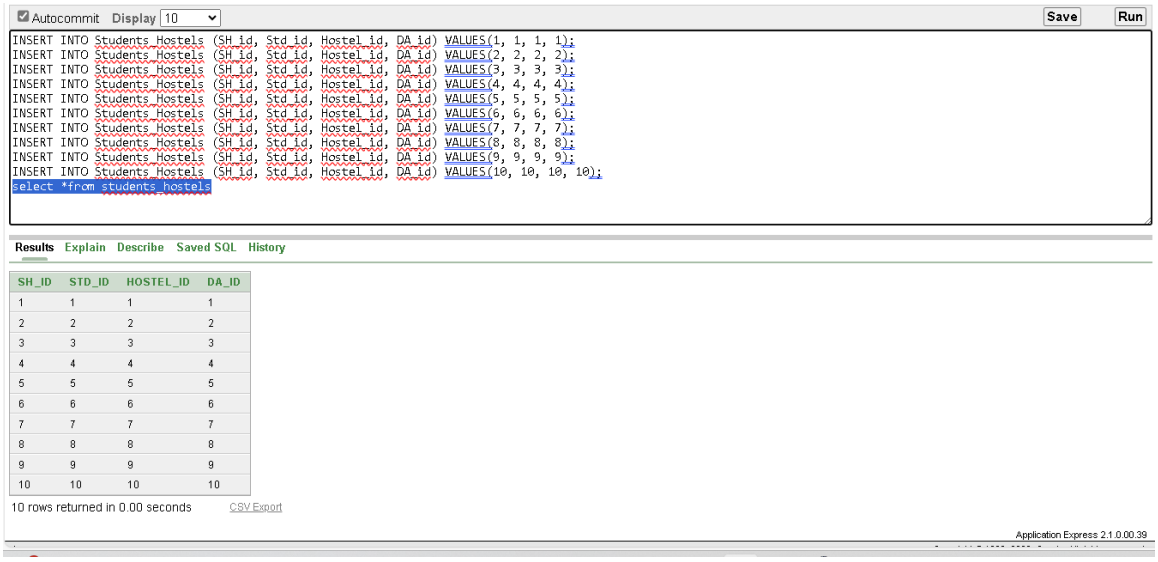
Autocommit Display 10 Save Run

Results Explain Describe Saved SQL History

HP_ID	HOSTEL_ID	PAYMENT_ID
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10
1	1	1
2	2	2
3	3	3
4	4	4

10 rows returned in 0.00 seconds CSV Export

Students_hostels:



```
INSERT INTO Students_Hostels (SH_id, Std_id, Hostel_id, DA_id) VALUES(1, 1, 1, 1);
INSERT INTO Students_Hostels (SH_id, Std_id, Hostel_id, DA_id) VALUES(2, 2, 2, 2);
INSERT INTO Students_Hostels (SH_id, Std_id, Hostel_id, DA_id) VALUES(3, 3, 3, 3);
INSERT INTO Students_Hostels (SH_id, Std_id, Hostel_id, DA_id) VALUES(4, 4, 4, 4);
INSERT INTO Students_Hostels (SH_id, Std_id, Hostel_id, DA_id) VALUES(5, 5, 5, 5);
INSERT INTO Students_Hostels (SH_id, Std_id, Hostel_id, DA_id) VALUES(6, 6, 6, 6);
INSERT INTO Students_Hostels (SH_id, Std_id, Hostel_id, DA_id) VALUES(7, 7, 7, 7);
INSERT INTO Students_Hostels (SH_id, Std_id, Hostel_id, DA_id) VALUES(8, 8, 8, 8);
INSERT INTO Students_Hostels (SH_id, Std_id, Hostel_id, DA_id) VALUES(9, 9, 9, 9);
INSERT INTO Students_Hostels (SH_id, Std_id, Hostel_id, DA_id) VALUES(10, 10, 10, 10);
select *from students_hostels
```

Autocommit Display 10 Save Run

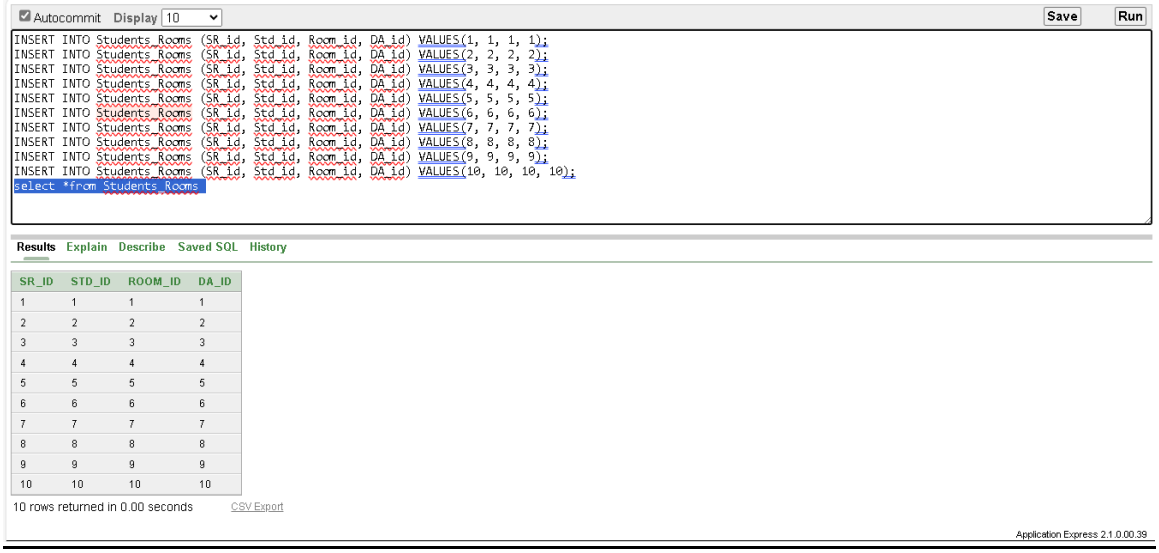
Results Explain Describe Saved SQL History

SH_ID	STD_ID	HOSTEL_ID	DA_ID
1	1	1	1
2	2	2	2
3	3	3	3
4	4	4	4
5	5	5	5
6	6	6	6
7	7	7	7
8	8	8	8
9	9	9	9
10	10	10	10

10 rows returned in 0.00 seconds CSV Export

Application Express 2.1.0.00.39

Students_Rooms:



```
INSERT INTO Students_Rooms (SR_id, Std_id, Room_id, DA_id) VALUES(1, 1, 1, 1);
INSERT INTO Students_Rooms (SR_id, Std_id, Room_id, DA_id) VALUES(2, 2, 2, 2);
INSERT INTO Students_Rooms (SR_id, Std_id, Room_id, DA_id) VALUES(3, 3, 3, 3);
INSERT INTO Students_Rooms (SR_id, Std_id, Room_id, DA_id) VALUES(4, 4, 4, 4);
INSERT INTO Students_Rooms (SR_id, Std_id, Room_id, DA_id) VALUES(5, 5, 5, 5);
INSERT INTO Students_Rooms (SR_id, Std_id, Room_id, DA_id) VALUES(6, 6, 6, 6);
INSERT INTO Students_Rooms (SR_id, Std_id, Room_id, DA_id) VALUES(7, 7, 7, 7);
INSERT INTO Students_Rooms (SR_id, Std_id, Room_id, DA_id) VALUES(8, 8, 8, 8);
INSERT INTO Students_Rooms (SR_id, Std_id, Room_id, DA_id) VALUES(9, 9, 9, 9);
INSERT INTO Students_Rooms (SR_id, Std_id, Room_id, DA_id) VALUES(10, 10, 10, 10);
select *from Students_Rooms
```

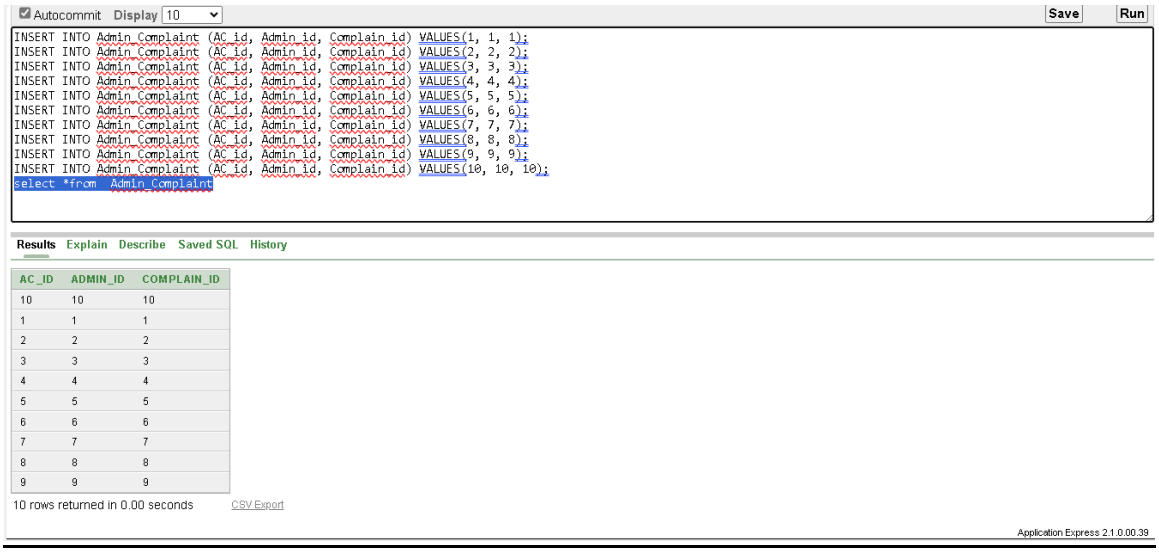
Results | Explain | Describe | Saved SQL | History

SR_ID	STD_ID	ROOM_ID	DA_ID
1	1	1	1
2	2	2	2
3	3	3	3
4	4	4	4
5	5	5	5
6	6	6	6
7	7	7	7
8	8	8	8
9	9	9	9
10	10	10	10

10 rows returned in 0.00 seconds CSV Export

Application Express 2.1.0.00.39

Admin_Complaint:



```
INSERT INTO Admin_Complaint (AC_id, Admin_id, Complain_id) VALUES(1, 1, 1);
INSERT INTO Admin_Complaint (AC_id, Admin_id, Complain_id) VALUES(2, 2, 2);
INSERT INTO Admin_Complaint (AC_id, Admin_id, Complain_id) VALUES(3, 3, 3);
INSERT INTO Admin_Complaint (AC_id, Admin_id, Complain_id) VALUES(4, 4, 4);
INSERT INTO Admin_Complaint (AC_id, Admin_id, Complain_id) VALUES(5, 5, 5);
INSERT INTO Admin_Complaint (AC_id, Admin_id, Complain_id) VALUES(6, 6, 6);
INSERT INTO Admin_Complaint (AC_id, Admin_id, Complain_id) VALUES(7, 7, 7);
INSERT INTO Admin_Complaint (AC_id, Admin_id, Complain_id) VALUES(8, 8, 8);
INSERT INTO Admin_Complaint (AC_id, Admin_id, Complain_id) VALUES(9, 9, 9);
INSERT INTO Admin_Complaint (AC_id, Admin_id, Complain_id) VALUES(10, 10, 10);
select *from  Admin_Complaint
```

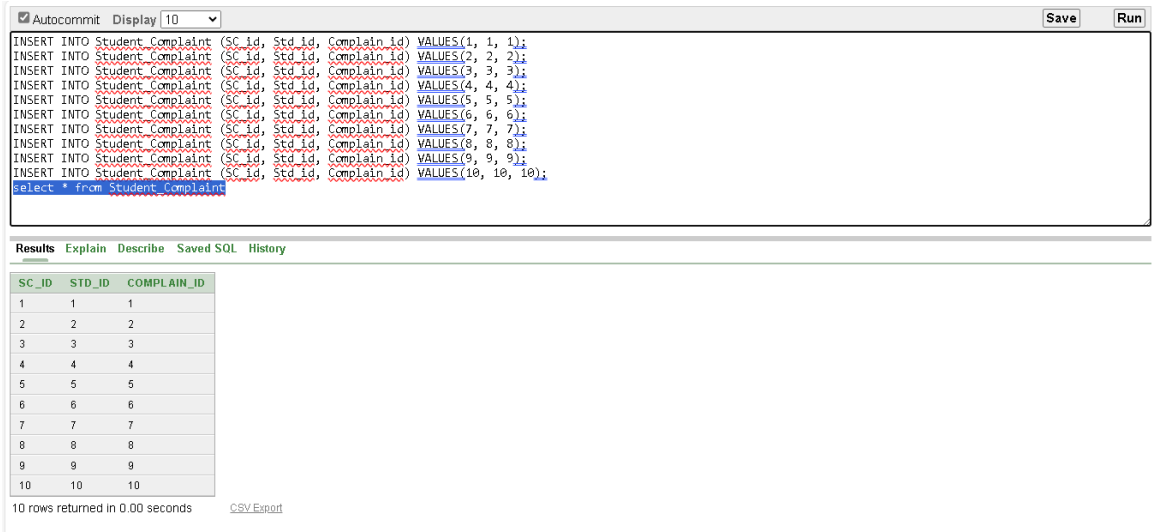
Results | Explain | Describe | Saved SQL | History

AC_ID	ADMIN_ID	COMPLAIN_ID
10	10	10
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9

10 rows returned in 0.00 seconds CSV Export

Application Express 2.1.0.00.39

Student_Complaint:



Autocommit Display 10 Save Run

```
INSERT INTO Student_Complaint (SC_id, Std_id, Complain_id) VALUES(1, 1, 1);
INSERT INTO Student_Complaint (SC_id, Std_id, Complain_id) VALUES(2, 2, 2);
INSERT INTO Student_Complaint (SC_id, Std_id, Complain_id) VALUES(3, 3, 3);
INSERT INTO Student_Complaint (SC_id, Std_id, Complain_id) VALUES(4, 4, 4);
INSERT INTO Student_Complaint (SC_id, Std_id, Complain_id) VALUES(5, 5, 5);
INSERT INTO Student_Complaint (SC_id, Std_id, Complain_id) VALUES(6, 6, 6);
INSERT INTO Student_Complaint (SC_id, Std_id, Complain_id) VALUES(7, 7, 7);
INSERT INTO Student_Complaint (SC_id, Std_id, Complain_id) VALUES(8, 8, 8);
INSERT INTO Student_Complaint (SC_id, Std_id, Complain_id) VALUES(9, 9, 9);
INSERT INTO Student_Complaint (SC_id, Std_id, Complain_id) VALUES(10, 10, 10);
select * from Student_Complaint
```

Results Explain Describe Saved SQL History

SC_ID	STD_ID	COMPLAIN_ID
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10

10 rows returned in 0.00 seconds CSV Export