



AMERICAN INTERNATIONAL UNIVERSITY–BANGLADESH  
(AIUB)

## Introduction to Database

Semester: Fall 2022-23

*“Rent Management System”*

Group  
**01**

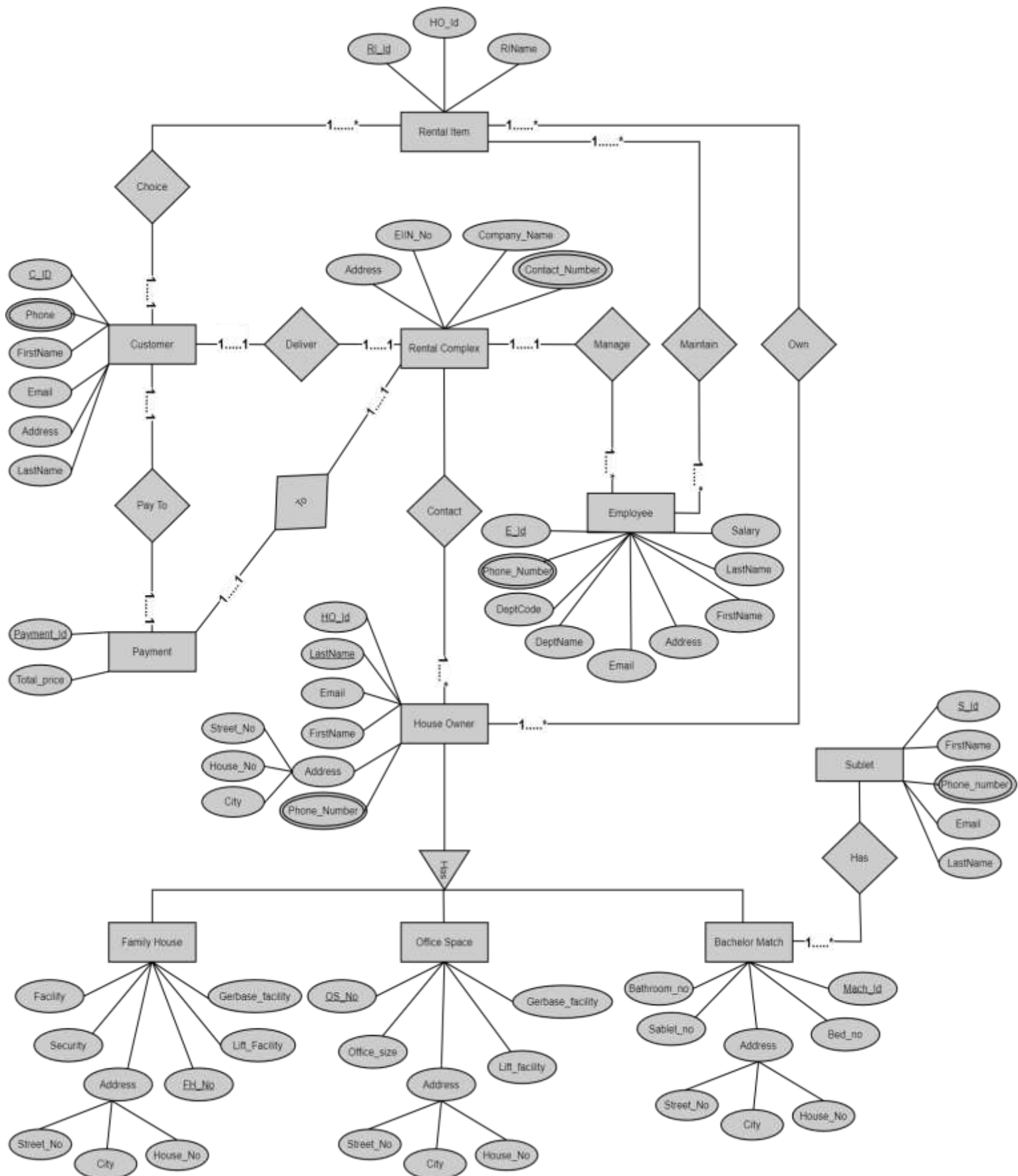
Name	ID
ONIK, MD. SABBIR HOSSAIN	22-46712-1
ISLAM, RAIHANUL	22-46680-1
NINE, MALEK AL JULKAR	22-46702-1
NURSHED, NUR-E HABIB	22-46702-1

## **Scenario:**

In a House Rent Management System Customer will choose their suitable houses that House Owners put in the management system.

In House Rent Management System, a company will be identified by EIIN Number. House Rent Management System Manage Many Employees. Employees can be identified by Eid Number. Employees have Department, Salary, Name. Employee Maintain Rental Items. Rental Item has Unique RI\_id. Customer has CU\_id, Firstname, Lastname, Email, Address. Customer choice Rental Item as they Want. Whatever Rental Item Customer Chose it will deliverd from House Rent Management System. Customer pay Payment. To identify payment individually Payment has Payment\_id. This Payment goes to house Rent Management System. House Owners Contact House Rent Management System to put up their house for rent. House Owner has uniquely identified HO\_id. It also has Address, FirstName, LastName, Email. House Owner has 3 type of houses Family House, Office Space, Bachelor Mech. Each Rental Item has Unique FO\_id, OS\_id, BM\_id. Bachelor Mech has many Sublets. Sublet has S\_id to uniquely identify it.

## ER-Diagram:



## **Normalization:**

### **UNF:** RENTAL COMPLEX MANAGE EMPLOYEE

MANAGE (EIIN\_NO, AUTHORITY, PHONE\_NO, ADDRESS, E\_ID, CITY, EMAIL, FIRST\_NAME, LAST\_NAME, DEPT\_NAME, DEPT\_CODE, SALARY)

#### **1NF:**

- PHONE\_NO IS A MULTIVALUED ATTRIBUTE.

MANAGE (EIIN\_NO, AUTHORITY, ADDRESS, E\_ID, CITY, EMAIL, FIRST\_NAME, LAST\_NAME, DEPT\_NAME, DEPT\_CODE, SALARY)

#### **2NF:**

1. PHONE\_ID, PHONE\_NO.
2. EIIN\_NO, AUTHORITY, ADDRESS, EMAIL, PHONE\_ID.
1. E\_ID, PHONE\_ID, CITY, EMAIL, FIRST\_NAME, LAST\_NAME, DEPT\_CODE, DEPT\_NAME, SALARY.

#### **3NF:**

1. PHONE \_ ID, PHONE,
2. EIIN \_ NO, PHONE ID, AUTHORITY, ADDRESS, EMAIL.
3. NID, FIRST\_NAME, LAST\_NAME.
4. E\_ID, DEPT\_CODE, DEPT\_NAME.
5. E\_ID, PHONE \_ ID, CITY, EMAIL, NID, SALARY.

### **UNF:** RENTAL COMPLEX CONTACTED BY HOUSE OWNER

CONTACT (EIIN\_NO, AUTHORITY\_NAME, FIRST\_NAME, LAST\_NAME, CITY, HOUSE\_NO, STREET\_NO, POST\_NO, EMAIL)

#### **1NF:**

- PHONE IS A MULTI VALUED ATTRIBUTE.

CONTACT (EIIN\_NO, AUTHORITY, FIRST\_NAME, LAST\_NAME, CITY, HOUSE\_NO, STREET\_NO, POST\_NO, EMAIL ADDRESS\_ID, HO-ID)

#### **2NF:**

1. PHONE \_ ID, PHONE\_NO
2. EIIN \_ NO, AUTHORITY\_NAME, ADDRESS\_ID.
3. HOUSE\_ID, O\_FNAME, O\_LNAME, HOUSE – \_NO, STREET-NO, POST-NO,

#### **3NF:**

1. PHONE \_ ID, PHONE\_NO.
2. EIIN \_ NO, PHONE\_ID, AUTHORITY\_NAME, ADDRESS\_ID.
3. NID, FIRST\_NAME, LAST\_NAME.

4.O\_ID, CITY, HOUSE\_NO, STREET\_NO, POST\_NO, EMAIL.

5.HO\_ID, NID, PHONE\_NO, EMAIL.

**UNF:** RENTAL COMPLEX DELIVER TO CUSTOMER

DELIVER(CO\_ID, PHONE, C\_NAME\_EMAIL, ADDRESS, EIIN\_NO, AUTHORITY\_NAME,  
PHONE\_NO, ADDRESS)

**1NF:**

- PHONE\_NO IS A MULTIVALUED ATTRIBUTE.

DELIVER(CO\_ID, C\_NAME\_EMAIL, ADDRESS, EIIN\_NO, AUTHORITY\_NAME, ADDRESS)

**2NF:**

- 1.PHONE\_ID, PHONE
- 2.EIIN\_NO, AUTHORITY, ADDRESS, PHONE ID .
- 3.C\_ID,EMAIL, CITY, CFRONE,CFNAME,CLNAME, PHONE ID .

**3NF:**

1. PHONE\_ID, PHONE,
- 2.EIIN\_NO, AUTHORITY, ADDRESS, PHONE ID .
- 3.N\_ID,C FMAME, C LNAME.
4. CO\_ID, NID, CITY, PHONE\_ID

**UNF:** CUSTOMER PAYS PAYMENT

PAY(CO\_ID, PHONE, C\_NAME\_EMAIL, ADDRESS,PAYMENT\_ID, PRICE)

**1NF:**

- PHONE\_NO IS A MULTIVALUED ATTRIBUTE.

PAY(CO\_ID, C\_NAME\_EMAIL, ADDRESS,PAYMENT\_ID, PRICE)

**2NF:**

- 1.PHONE\_ID, PHONE.
- 2.PAYMENT\_ID, PRICE, PAYMENT METHED.
- 3.C\_ID,EMAIL, CITY, CFRONE, CFNAME, CLNAME, PHONE ID.

**3NF:**

- 1.PHONE\_ID, PHONE.
- 2.PAYMENT\_ID, PRICE, PAYMENT METHED.
- 3.NID, C FMAME, C LNAME.
- 4.CO\_ID, N\_ID, PHONE\_ID, EMAIL,CITY.

**UNF:** PAYMENT PAY TO RENTAL COMPLEX

PAY (PAYMENT\_ID, PRICE, EIIN\_NO, AUTHORITY\_NAME, PHONE\_NO, ADDRESS)

**1NF:**

- PHONE\_NO IS A MULTIVALUED ATTRIBUTE.

PAY(PAYMENT\_ID, PRICE, EIIN\_NO, AUTHORITY\_NAME, ADDRESS)

**2NF:**

1. PHONE\_ID, PHONE.
2. EIIN\_NUM, AUTHORITY, ADDRESS, EMAIL.
3. PAYMENT\_ID, PRICE, PAYMENT\_METHED.

**3NF:**

- NO NEED FOR 3NF

**UNF:** BACHELOR MECH HAS SUBLET

HAS (BATHROOM\_NO, SUBLET\_NO\_BED\_NO\_MECH\_ID,S\_ID, NAME, PHONE\_NO, EMAIL)

**1NF:**

- PHONE\_NO IS A MULTIVALUED ATTRIBUTE.

HAS (BATHROOM\_NO, SUBLET\_NO\_BED\_NO\_MECH\_ID,S\_ID, NAME, EMAIL)

**2NF:**

1. PHONE\_ID, PHONE.
2. MECH\_ID, BACHELOR\_NO, BAD\_NO, SUBLET\_NUM, STREET\_NO, HOUSE\_NO, RENT.
3. SUBLET\_ID, FIRST\_NAME, LAST\_NAME, CITY, EMAIL.

**3NF:**

1. PHONE\_ID, PHONE.
2. MECH\_ID, BACHELOR\_NO, BAD\_NO, SUBLET\_NUM.
3. MECH\_ID, CITY, STREET, HOUSE\_NO.
4. N\_ID, S FMAME, S LNAME.
5. SUBLET\_ID, NID, CITY, EMAIL, PHONE\_ID.

**UNF:** CUSTOMER CHOICE RENTAL ITEM

CHOICE (CO\_ID, PHONE, C\_NAME\_EMAIL, ADDRESS, RI\_ID, HO\_NAME, RI\_NAME)

**1NF:**

- PHONE\_NO IS A MULTIVALUED ATTRIBUTE.

CHOICE (CO\_ID C\_NAME\_EMAIL, ADDRESS, RI\_ID, HO\_NAME, RI\_NAME)

**2NF:**

1. PHONE\_ID, PHONE.
2. CU\_ID, EMAIL, ADDRESS, CUS\_NAME.
3. RI\_ID, HO\_ID, RI\_NAME.

**3NF:**

1. PHONE\_ID, PHONE.
2. N\_ID, FIRST\_NAME, LAST\_NAME.
3. C\_ID, N\_ID, EMAIL, ADDRESS.
4. RI\_ID, HO\_ID, RI\_NAME.

**UNF:** EMPLOYEE MANAGE RENTAL ITEM

MANAGE(E\_ID, PHONE, DEPT\_CODE, DEPT\_NAME, EMAIL, ADDRESS, SALARY, RI\_ID, HO\_NAME, RI\_NAME)

**1NF:**

- PHONE\_NO IS A MULTIVALUED ATTRIBUTE.

MANAGE(E\_ID, DEPT\_CODE, DEPT\_NAME, EMAIL, ADDRESS, SALARY, RI\_ID, HO\_NAME, RI\_NAME)

**2NF:**

1. PHONE\_ID, PHONE.
2. RI\_ID, HO\_ID, RI\_NAME.
3. E\_ID, PHONE\_ID, NID, CITY, EMAIL, DEPT\_CODE.
4. DEPT\_CODE, DEPT\_NAME, SALARY.
5. N\_ID, E FIRST\_NAME, LAST\_NAME.

**3NF:**

- NO NEED FOR 3NF

**UNF:** HOUSE OWNER OWN RENTAL ITEM

OWN(HO\_ID, EMAIL, PHONE, ADDRESS, OWNER\_NAME, RI\_ID, HO\_NAME, RI\_NAME)

**1NF:**

- PHONE\_NO IS A MULTIVALUED ATTRIBUTE.

OWN(HO\_ID, EMAIL, ADDRESS, OWNER\_NAME, RI\_ID, HO\_NAME, RI\_NAME)

**2NF:**

1. PHONE\_ID, PHONE
2. RI\_ID, HO\_ID, RI\_NAME.
3. HO\_ID, FIRST\_NAME, LAST\_NAME, CITY, HOUSE\_NO, EMAIL.

**3NF:**

1. PHONE \_ID, PHONE.
2. RI\_ID, HO\_ID, RI NAME.
3. HO\_ID, EMAIL, N\_ID, PHONE\_ID.
4. FIRST\_NAME, LAST\_NAME, NID.
5. CITY, HOUSE\_NO, STREET\_NO, HO\_ID.

**TABLE SELECTION:**

1. PHONE \_ID, PHONE,
2. EIIN \_ NO, PHONE ID, AUTHORITY, ADDRESS, EMAIL.
3. NID, FIRST\_NAME, LAST\_NAME.
4. E\_ID, DEPT\_CODE, DEPT\_NAME.
5. E\_ID, PHONE \_ ID, CITY, EMAIL, NID, SALARY.
6. PHONE \_ ID, PHONE\_NO.
7. EIIN \_ NO, PHONE\_ID, AUTHORITY\_NAME, ADDRESS\_ID.
8. NID, FIRST\_NAME, LAST\_NAME.
9. O\_ID, CITY, HOUSE\_NO, STREET\_NO, POST\_NO, EMAIL.
10. HO\_ID, NID, PHONE\_NO, EMAIL.
11. PHONE \_ ID, PHONE,
12. EIIN \_ NO, AUTHORITY, ADDRESS, PHONE ID .
13. N\_ID, C FMAME, C LNAME.
14. CO\_ID, NID, CITY, PHONE\_ID
15. PHONE \_ID, PHONE.
16. PAYMENT\_ID, PRICE, PAYMENT METHED.
17. NID, C FMAME, C LNAME.
18. CO\_ID, N\_ID, PHONE \_ID, EMAIL, CITY.
19. PHONE \_ID, PHONE.
20. EIIN \_ NUM, AUTHORITY, ADDRESS, EMAIL .
21. PAYMENT\_ID, PRICE, PAYMENT METHED.
22. PHONE \_ID, PHONE.
23. MECH\_ID, BACHELOR\_NO, BAD\_NO, SUBLET\_NUM.
24. MECH\_ID, CITY, STREET, HOUSE\_NO.
25. N\_ID, S FMAME, S LNAME.
26. SUBLET\_ID, NID , CITY, EMAIL, PHONE \_ID .
27. PHONE \_ID, PHONE.



28. N\_ID, FIRST\_NAME, LAST\_NAME.
29. C\_ID, N\_ID, EMAIL, ADDRESS.
30. RI\_ID, HO\_ID, RI\_NAME.
31. PHONE\_ID, PHONE.
32. RI\_ID, HO\_ID, RI NAME.
33. E\_ID, PHONE\_ID, NID, CITY, EMAIL, DEPT\_CODE.
34. DEPT\_CODE, DEPT\_NAME, SALARY.
35. N\_ID, E FIRST\_NAME, LAST\_NAME.
36. PHONE\_ID, PHONE.
37. RI\_ID, HO\_ID, RI NAME.
38. HO\_ID, EMAIL, N\_ID, PHONE\_ID.
39. FIRST\_NAME, LAST\_NAME, NID.
40. CITY, HOUSE\_NO, STREET\_NO, HO\_ID.

**FINAL TABLE:**

1. PHONE\_ID, PHONE.
2. EIIN\_NO, AUTHORITY\_NAME, PHONE\_ID.
3. NID, FIRST\_NAME, LAST\_NAME.
4. DEPT\_CODE, DEPT\_NAME, SALARY.
5. E\_ID, CITY, EMAIL, NID, SALARY, DEPT\_CODE, PHONE\_ID.
6. ADDRESS\_ID, HOUSE\_NO, STREET\_NO, CITY.
7. HO\_ID, CITY, EMAIL, ADDRESS\_ID, NID, PHONE\_ID.
8. CU\_ID, EMAIL, CITY, N\_ID, PHONE\_ID.
9. PAYMENT\_ID, PRICE, PAYMENT\_METHED.
10. MECH\_ID, BATHROOM\_NO, BED\_NO, SUBLET\_NO.
11. SUBLET\_ID, CITY, EMAIL, NID, MECH\_ID, PHONE\_ID.
12. RI\_ID, RI NAME, HO\_ID.
13. FH\_ID, SECUIRITY, BED\_NO, LIFT\_FACILITY, ADDRESS\_ID.
14. OS\_ID, OFFICE\_SIZE, SECUIRITY, LIFT\_FACILITY, GARAGE\_FACILITY, ADDRESS\_ID.

## Table Creation:

### PHONE TABLE:

```
CREATE TABLE PHONE(  
  PHONE_ID NUMBER(10) PRIMARY KEY,  
  PHONE_NO VARCHAR2(20) );  
DESC PHONE
```

The screenshot shows the Oracle Database Express Edition interface. The SQL Command window contains the following SQL commands:

```
CREATE TABLE PHONE(  
  PHONE_ID NUMBER(10) PRIMARY KEY,  
  PHONE_NO VARCHAR2(20) );  
DESC PHONE
```

The Results tab displays the table structure for the 'PHONE' table:

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
PHONE	PHONE_ID	Number	-	10	0	Y	-	-	-
PHONE	PHONE_NO	Varchar2	20	-	-	-	✓	-	-

The interface also shows the 'Save' and 'Run' buttons, and the 'Language: en-us' setting at the bottom.

### AUTHORITY TABLE:

```
CREATE TABLE AUTHORITY(  
  EIIN_ID NUMBER(10) PRIMARY KEY,  
  AUTHORITY_NAME VARCHAR2(20),  
  PHONE_ID REFERENCES PHONE );  
DESC AUTHORITY
```

The screenshot shows the Oracle Database Express Edition interface. The SQL Command window contains the following SQL commands:

```
CREATE TABLE AUTHORITY(  
  EIIN_ID NUMBER(10) PRIMARY KEY,  
  AUTHORITY_NAME VARCHAR2(20),  
  PHONE_ID REFERENCES PHONE );  
DESC AUTHORITY
```

The Results tab displays the table structure for the 'AUTHORITY' table:

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
AUTHORITY	EIIN_ID	Number	-	10	0	Y	-	-	-
AUTHORITY	AUTHORITY_NAME	Varchar2	20	-	-	-	✓	-	-
AUTHORITY	PHONE_ID	Number	-	10	0	-	✓	-	-

The interface also shows the 'Save' and 'Run' buttons, and the 'Language: en-us' setting at the bottom.

### NAME TABLE:

```
CREATE TABLE NAME(  
NID VARCHAR2(20) PRIMARY KEY,  
FIRST_NAME VARCHAR2(20),  
LAST_NAME VARCHAR2(20) );  
DESC NAME
```

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

```
CREATE TABLE NAME(  
NID VARCHAR2(20) PRIMARY KEY,  
FIRST_NAME VARCHAR2(20),  
LAST_NAME VARCHAR2(20) );  
DESC NAME
```

The Results tab is selected, displaying the table structure for the NAME table:

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
NAME	NID	Varchar2	20			✓	✓		
	FIRST_NAME	Varchar2	20				✓		
	LAST_NAME	Varchar2	20				✓		

At the bottom right, it says "Application Express 2.1.2.00.26 Copyright © 1995-2006, Oracle. All rights reserved."

### DEPARTMENT TABLE:

```
CREATE TABLE DEPARTMENT(  
DEPT_CODE VARCHAR2(20) PRIMARY KEY,  
DEPT_NAME VARCHAR2(20),  
SALARY VARCHAR2(20) );  
DESC DEPARTMENT
```

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

```
CREATE TABLE DEPARTMENT(  
DEPT_CODE VARCHAR2(20) PRIMARY KEY,  
DEPT_NAME VARCHAR2(20),  
SALARY VARCHAR2(20) );  
DESC DEPARTMENT
```

The Results tab is selected, displaying the table structure for the DEPARTMENT table:

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
DEPARTMENT	DEPT_CODE	Varchar2	20			✓	✓		
	DEPT_NAME	Varchar2	20				✓		
	SALARY	Varchar2	20				✓		

At the bottom right, it says "Application Express 2.1.2.00.26 Copyright © 1995-2006, Oracle. All rights reserved."

### EMPLOYEE TABLE:

```
CREATE TABLE EMPLOYEE(  
E_ID NUMBER(10) PRIMARY KEY,  
CITY VARCHAR2(20),  
SALARY VARCHAR2(20),  
DEPT_CODE REFERENCES DEPARTMENT,  
PHONE_ID REFERENCES PHONE,  
NID REFERENCES NAME );  
DESC EMPLOYEE
```

The screenshot shows the SQL Developer interface. The top pane contains the SQL commands to create the EMPLOYEE table and then describe it. The bottom pane shows the results of the DESCRIBE command as a table.

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMPLOYEE	E_ID	Number	-	10	0	Y	N		
	CITY	Varchar2	20	-	-	N	Y		
	SALARY	Varchar2	20	-	-	N	Y		
	DEPT_CODE	Varchar2	20	-	-	N	Y		
	PHONE_ID	Number	-	10	0	N	Y		
	NID	Varchar2	20	-	-	N	Y		

### ADDRESS TABLE:

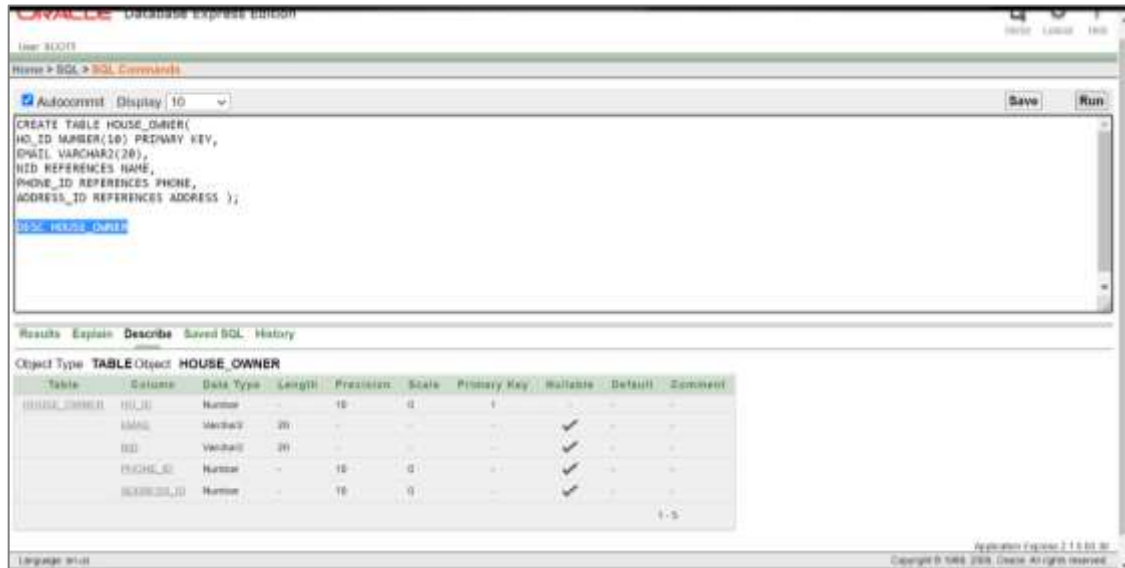
```
CREATE TABLE ADDRESS(  
ADDRESS_ID NUMBER(10) PRIMARY KEY,  
HOUSE_NO VARCHAR2(20),  
STREET_NO VARCHAR2(20),  
CITY VARCHAR2(20) );  
DESC ADDRESS
```

The screenshot shows the SQL Developer interface. The top pane contains the SQL commands to create the ADDRESS table and then describe it. The bottom pane shows the results of the DESCRIBE command as a table.

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ADDRESS	ADDRESS_ID	Number	-	10	0	Y	N		
	HOUSE_NO	Varchar2	20	-	-	N	Y		
	STREET_NO	Varchar2	20	-	-	N	Y		
	CITY	Varchar2	20	-	-	N	Y		

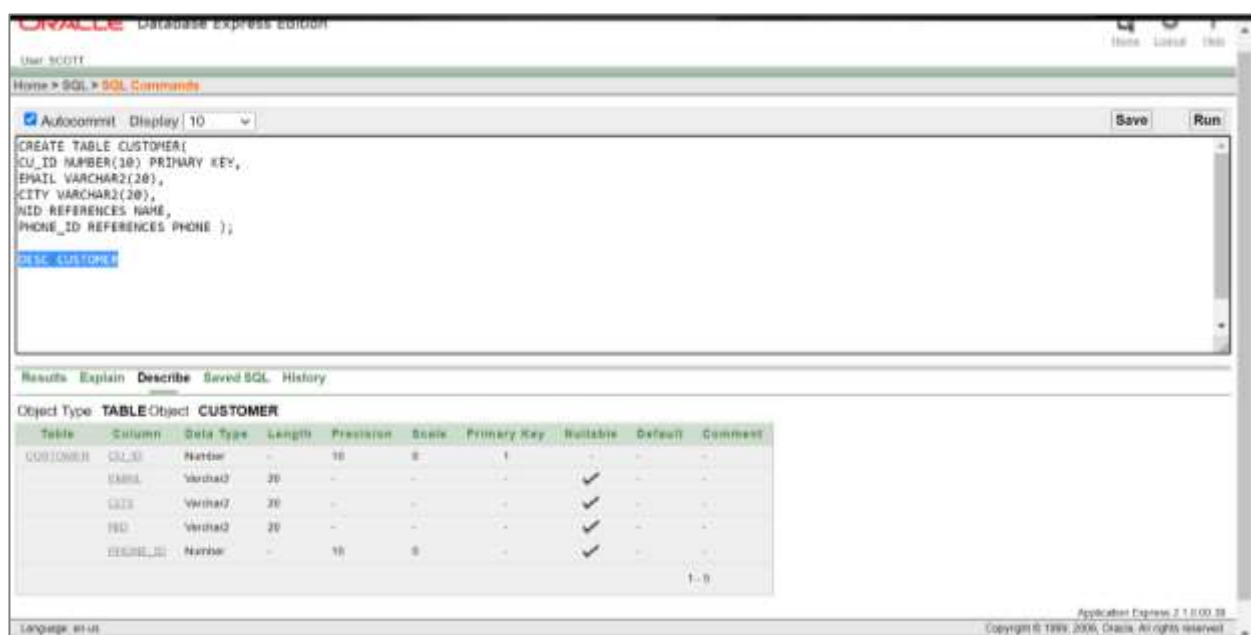
### HOUSE OWNER TABLE:

```
CREATE TABLE HOUSE_OWNER(  
HO_ID NUMBER(10) PRIMARY KEY,  
EMAIL VARCHAR2(20),  
NID REFERENCES NAME,  
PHONE_ID REFERENCES PHONE,  
ADDRESS_ID REFERENCES ADDRESS );  
DESC HOUSE_OWNER
```



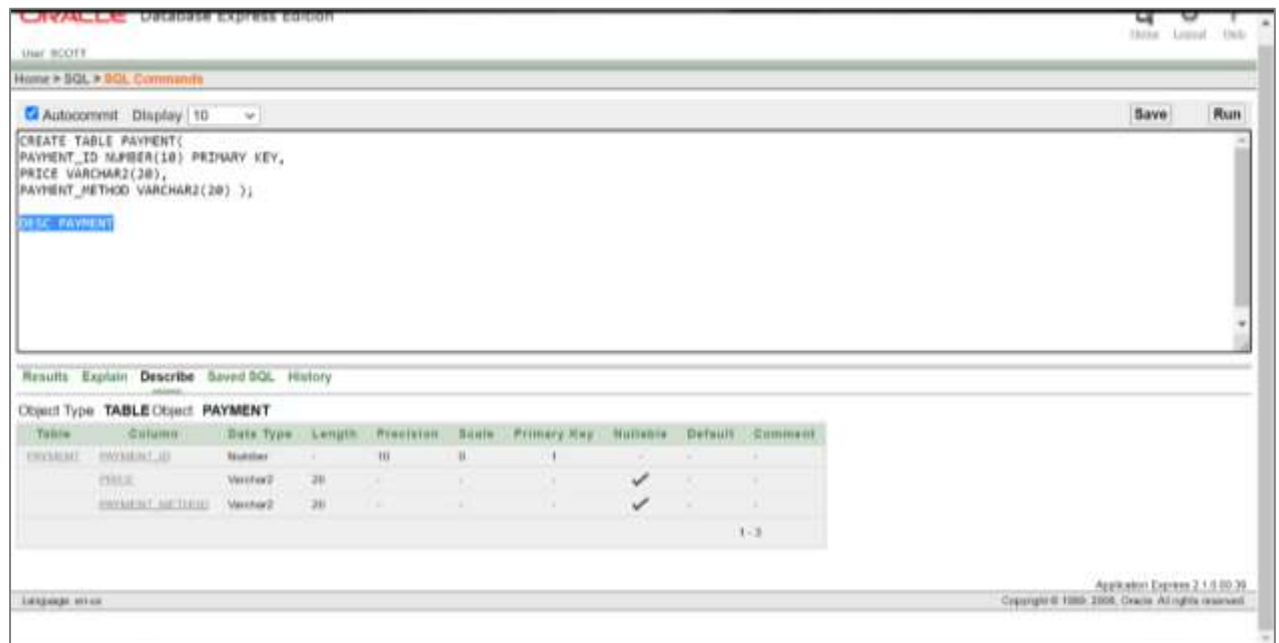
### CUSTOMER TABLE:

```
CREATE TABLE CUSTOMER(  
CU_ID NUMBER(10) PRIMARY KEY,  
EMAIL VARCHAR2(20),  
CITY VARCHAR2(20),  
NID REFERENCES NAME,  
PHONE_ID REFERENCES PHONE );  
DESC CUSTOMER
```



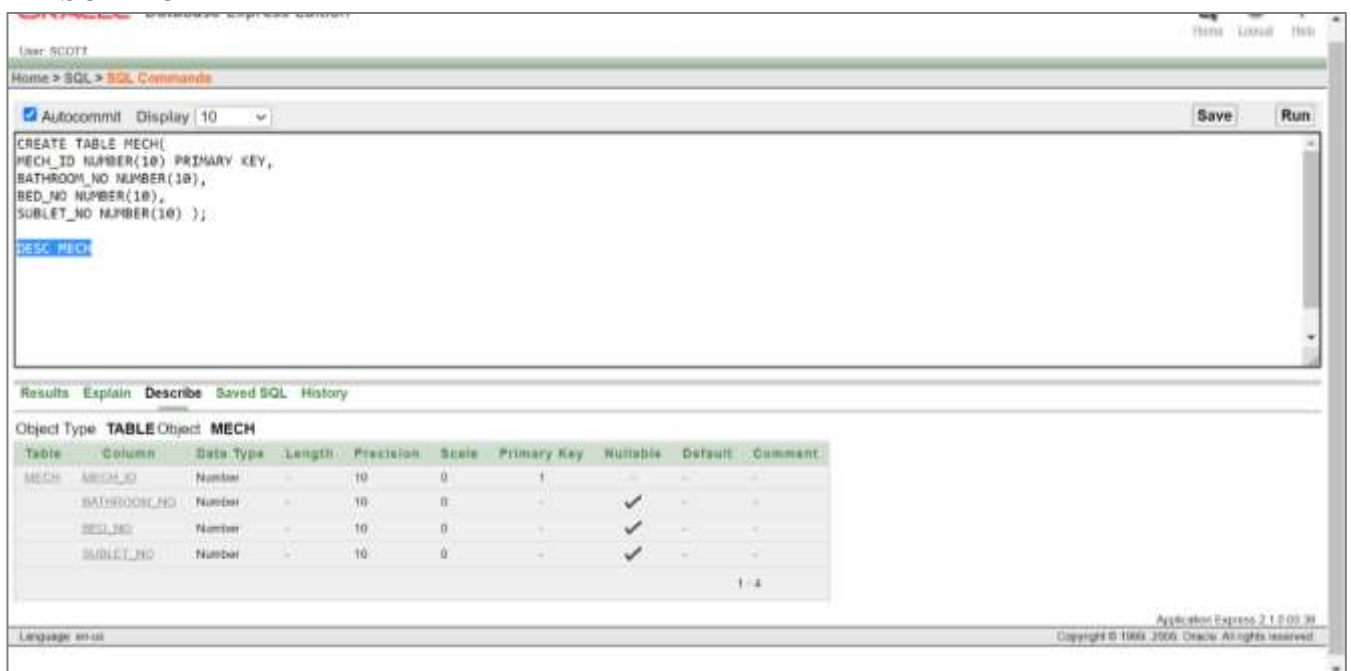
### PAYMENT TABLE:

```
CREATE TABLE PAYMENT(  
  PAYMENT_ID NUMBER(10) PRIMARY KEY,  
  PRICE VARCHAR2(20),  
  PAYMENT_METHOD VARCHAR2(20) );  
DESC PAYMENT
```



### MECH TABLE:

```
CREATE TABLE MECH(  
  MECH_ID NUMBER(10) PRIMARY KEY,  
  BATHROOM_NO NUMBER(10),  
  BED_NO NUMBER(10),  
  SUBLET_NO NUMBER(10) );  
DESC MECH
```



### SUBLET TABLE:

```
CREATE TABLE SUBLET(  
SUBLET_ID NUMBER(10) PRIMARY KEY,  
EMAIL VARCHAR2(20),  
CITY VARCHAR2(20),  
NID REFERENCES NAME,  
MECH_ID REFERENCES MECH,  
PHONE_ID REFERENCES PHONE );  
DESC SUBLET
```

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

```
CREATE TABLE SUBLET(  
SUBLET_ID NUMBER(10) PRIMARY KEY,  
EMAIL VARCHAR2(20),  
CITY VARCHAR2(20),  
NID REFERENCES NAME,  
MECH_ID REFERENCES MECH,  
PHONE_ID REFERENCES PHONE );  
DESC SUBLET
```

The Results tab displays the table structure for SUBLET:

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
SUBLET	SUBLET_ID	Number	--	10	0	1		--	--
	EMAIL	Varchar2	20	--	--		✓	--	--
	CITY	Varchar2	20	--	--		✓	--	--
	NID	Varchar2	20	--	--		✓	--	--
	MECH_ID	Number	--	10	0		✓	--	--
	PHONE_ID	Number	--	10	0		✓	--	--

Application Express 3.1.6.00.30

### RENTAL ITEM TABLE:

```
CREATE TABLE RENTAL_ITEM(  
RI_ID NUMBER(10) PRIMARY KEY,  
RI_NAME VARCHAR2(20),  
HO_ID REFERENCES HOUSE_OWNER );  
DESC RENTAL_ITEM
```

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

```
CREATE TABLE RENTAL_ITEM(  
RI_ID NUMBER(10) PRIMARY KEY,  
RI_NAME VARCHAR2(20),  
HO_ID REFERENCES HOUSE_OWNER );  
DESC RENTAL_ITEM
```

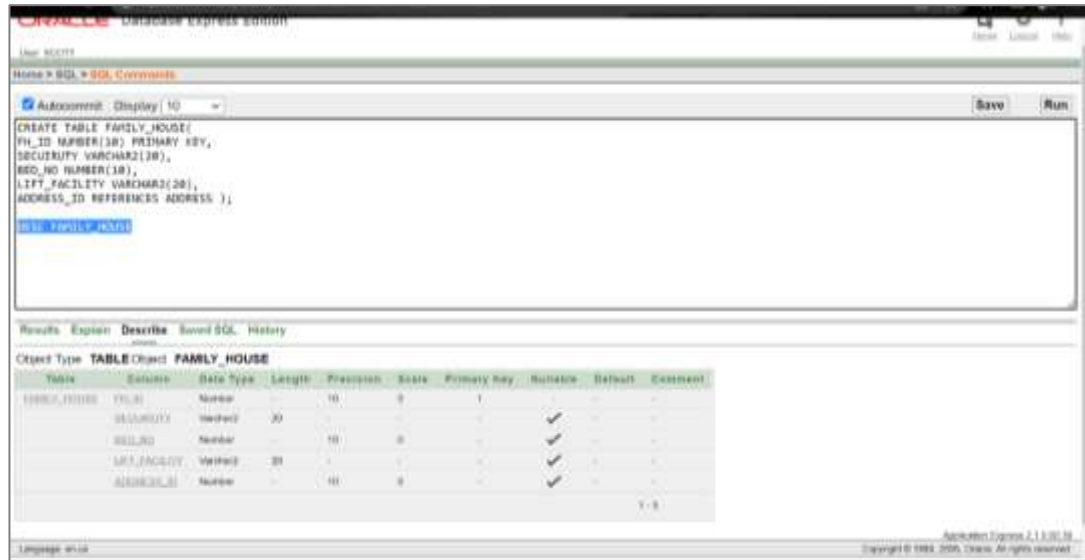
The Results tab displays the table structure for RENTAL\_ITEM:

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
RENTAL_ITEM	RI_ID	Number	--	10	0	1		--	--
	RI_NAME	Varchar2	20	--	--		✓	--	--
	HO_ID	Number	--	10	0		✓	--	--

Application Express 3.1.6.00.30  
Copyright © 1999, 2008, Oracle. All rights reserved.

### **FAMILY HOUSE TABLE:**

```
CREATE TABLE FAMILY_HOUSE(  
FH_ID NUMBER(10) PRIMARY KEY,  
SECURITY VARCHAR2(20),  
BED_NO NUMBER(10),  
LIFT_FACILITY VARCHAR2(20),  
ADDRESS_ID REFERENCES ADDRESS );  
DESC FAMILY_HOUSE
```



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

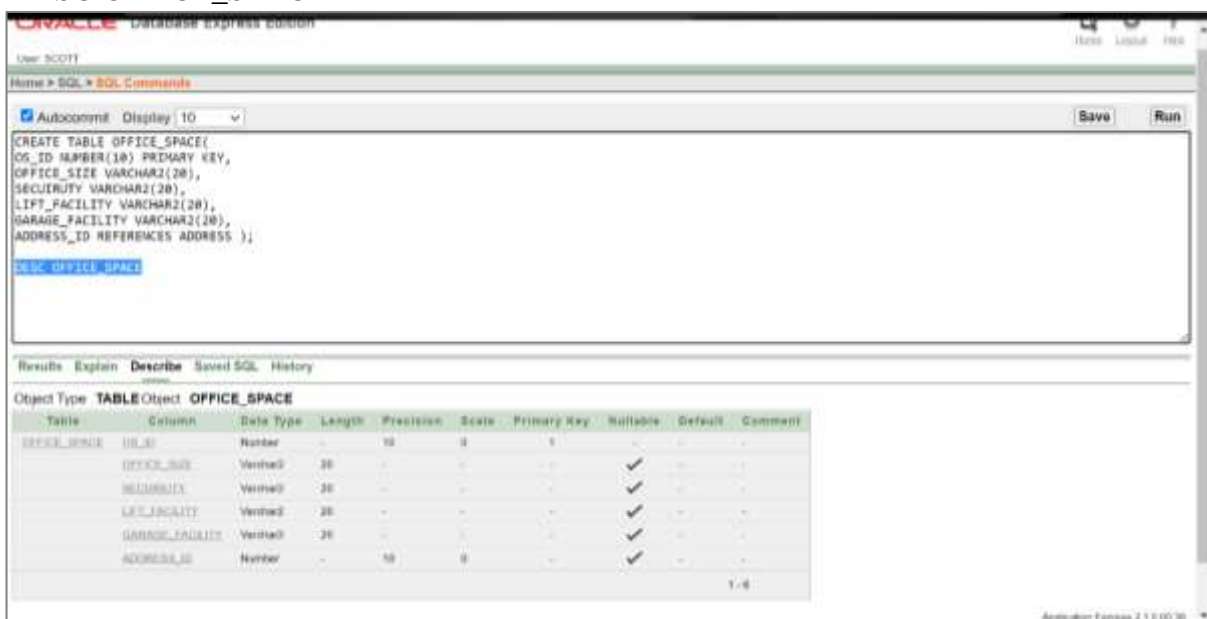
```
CREATE TABLE FAMILY_HOUSE(  
FH_ID NUMBER(10) PRIMARY KEY,  
SECURITY VARCHAR2(20),  
BED_NO NUMBER(10),  
LIFT_FACILITY VARCHAR2(20),  
ADDRESS_ID REFERENCES ADDRESS );  
DESC FAMILY_HOUSE
```

The Results window displays the table structure for FAMILY\_HOUSE:

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
FAMILY_HOUSE	FH_ID	Number	10		0	✓	✓		
	SECURITY	Varchar2	20				✓		
	BED_NO	Number	10		0		✓		
	LIFT_FACILITY	Varchar2	20				✓		
	ADDRESS_ID	Number	10		0		✓		

### **OFFICE SPACE TABLE:**

```
CREATE TABLE OFFICE_SPACE(  
OS_ID NUMBER(10) PRIMARY KEY,  
OFFICE_SIZE VARCHAR2(20),  
SECURITY VARCHAR2(20),  
LIFT_FACILITY VARCHAR2(20),  
GARAGE_FACILITY VARCHAR2(20),  
ADDRESS_ID REFERENCES ADDRESS );  
DESC OFFICE_SPACE
```



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

```
CREATE TABLE OFFICE_SPACE(  
OS_ID NUMBER(10) PRIMARY KEY,  
OFFICE_SIZE VARCHAR2(20),  
SECURITY VARCHAR2(20),  
LIFT_FACILITY VARCHAR2(20),  
GARAGE_FACILITY VARCHAR2(20),  
ADDRESS_ID REFERENCES ADDRESS );  
DESC OFFICE_SPACE
```

The Results window displays the table structure for OFFICE\_SPACE:

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
OFFICE_SPACE	OS_ID	Number	10		0	✓	✓		
	OFFICE_SIZE	Varchar2	20				✓		
	SECURITY	Varchar2	20				✓		
	LIFT_FACILITY	Varchar2	20				✓		
	GARAGE_FACILITY	Varchar2	20				✓		
	ADDRESS_ID	Number	10		0		✓		



# Table Insertion:

## PHONE TABLE:

1.

```
INSERT INTO PHONE VALUES( '01','+8801712345678');
INSERT INTO PHONE VALUES( '02','+8801712345679');
INSERT INTO PHONE VALUES( '03','+8801712345680');
INSERT INTO PHONE VALUES( '04','+8801712345681');
INSERT INTO PHONE VALUES( '05','+8801712345681');
INSERT INTO PHONE VALUES( '06','+8801612345682');
INSERT INTO PHONE VALUES( '07','+8801612345683');
INSERT INTO PHONE VALUES( '08','+8801612345684');
INSERT INTO PHONE VALUES( '09','+8801612345685');
INSERT INTO PHONE VALUES( '10','+8801612345686');
INSERT INTO PHONE VALUES( '11','+8801312345687');
INSERT INTO PHONE VALUES( '12','+8801312345688');
INSERT INTO PHONE VALUES( '13','+8801312345689');
INSERT INTO PHONE VALUES( '14','+8801312345681');
INSERT INTO PHONE VALUES( '15','+8801312345682');
INSERT INTO PHONE VALUES( '16','+8801912345683');
INSERT INTO PHONE VALUES( '17','+8801912345684');
INSERT INTO PHONE VALUES( '18','+8801912345685');
INSERT INTO PHONE VALUES( '19','+8801912345686');
INSERT INTO PHONE VALUES( '20','+8801912345687');
INSERT INTO PHONE VALUES( '21','+8801918345687');
INSERT INTO PHONE VALUES( '22','+8801312345688');
INSERT INTO PHONE VALUES( '23','+8801312345689');
INSERT INTO PHONE VALUES( '24','+8801312345681');
INSERT INTO PHONE VALUES( '25','+8801312345682');
INSERT INTO PHONE VALUES( '26','+8801912345683');
```

```
SELECT *
FROM PHONE
```

The screenshot shows the Oracle Database Express Edition interface. The top pane displays the execution of an INSERT statement, with the results showing 26 rows inserted. The bottom pane shows the contents of the PHONE table, which now contains 26 rows of data.

PHONE_ID	PHONE_NO
1	+8801712345678
2	+8801712345679
3	+8801712345680
4	+8801712345681
5	+8801712345681
6	+8801612345682
7	+8801612345683
8	+8801612345684
9	+8801612345685
10	+8801612345686
11	+8801312345687
12	+8801312345688
13	+8801312345689
14	+8801312345681
15	+8801312345682
16	+8801912345683
17	+8801912345684
18	+8801912345685
19	+8801912345686
20	+8801912345687
21	+8801918345687
22	+8801312345688
23	+8801312345689
24	+8801312345681
25	+8801312345682
26	+8801912345683

## **AUTHORITY TABLE:**

2.

```
INSERT INTO AUTHORITY VALUES('12345','SWEET HOME RENTAL  
AGENCY','01');
```

```
SELECT *  
FROM AUTHORITY
```

The screenshot shows the Oracle Database Express Edition web interface. The top bar indicates the user is 'SCOTT'. The breadcrumb navigation shows 'Home > SQL > SQL Commands'. The main text area contains two SQL statements: an INSERT statement and a SELECT statement. The INSERT statement is: `INSERT INTO AUTHORITY VALUES('12345','SWEET HOME RENTAL AGENCY','01');`. The SELECT statement is: `SELECT *  
FROM AUTHORITY`. Below the text area, there are 'Save' and 'Run' buttons. The 'Results' tab is selected, showing a table with three columns: 'EILN\_NO', 'AUTHORITY\_NAME', and 'PHONE\_ID'. The table contains one row: '12345', 'SWEET HOME RENTAL AGENCY', and '1'. Below the table, it says '1 rows returned in 0.02 seconds' and 'CSV Export'. The bottom status bar shows 'Language: en-us' and 'Application Express 2.1.0.0.10 Copyright © 1999, 2004, Oracle. All rights reserved.'

Oracle Database Express Edition

User: SCOTT

Home > SQL > SQL Commands

☒ Autocommit Display: 10

Save Run

```
INSERT INTO AUTHORITY VALUES('12345','SWEET HOME RENTAL AGENCY','01');
```

```
SELECT *  
FROM AUTHORITY
```

Results Explain Describe Saved SQL History

EILN_NO	AUTHORITY_NAME	PHONE_ID
12345	SWEET HOME RENTAL AGENCY	1

1 rows returned in 0.02 seconds CSV Export

Language: en-us Application Express 2.1.0.0.10 Copyright © 1999, 2004, Oracle. All rights reserved.

## NAME TABLE:

3.

```
INSERT INTO NAME VALUES('101','MD SABBIR','KHAN');
INSERT INTO NAME VALUES('102','MD AKASH','KHAN');
INSERT INTO NAME VALUES('103','NUR-E HABIB','NURSHED');
INSERT INTO NAME VALUES('104','RAIHANUL','ISLAM');
INSERT INTO NAME VALUES('105','JULKER','NINE');
INSERT INTO NAME VALUES('106','HABIUR','RAHAMAN');
INSERT INTO NAME VALUES('107','SAHADAT','HOSSAIN');
INSERT INTO NAME VALUES('108','SABAB','RAHAMAN');
INSERT INTO NAME VALUES('109','RICKI','SEN');
INSERT INTO NAME VALUES('110','DIPON','BISWAS');
INSERT INTO NAME VALUES('111','MAHADI','KABIR');
INSERT INTO NAME VALUES('112','REZAUL','KARIM');
INSERT INTO NAME VALUES('113','RISI','GUPTA');
INSERT INTO NAME VALUES('114','AQIB','HOWLADER');
INSERT INTO NAME VALUES('115','RISHAD','RAHMAN');
INSERT INTO NAME VALUES('116','RASEDUR','RAHAMAN');
INSERT INTO NAME VALUES('117','KHALID','HOSSAIN');
INSERT INTO NAME VALUES('118','RIYANTTI','SEN');
INSERT INTO NAME VALUES('119','SIMA','RAHAMAN');
INSERT INTO NAME VALUES('120','PRIYA','BISWAS');
INSERT INTO NAME VALUES('121','MEHEDI','RAHAMAN');
INSERT INTO NAME VALUES('122','REZAUL','SHARIF');
INSERT INTO NAME VALUES('123','RUMPA','KHAN');
INSERT INTO NAME VALUES('124','ISRAT','EVA');
INSERT INTO NAME VALUES('125','RAJU','RAHMAN');
```

SELECT \*  
FROM NAME



Oracle Database Express Edition

SQL\*Plus

SELECT \*  
FROM NAME

Results

ID	FIRST_NAME	LAST_NAME
101	MD SABBIR	KHAN
102	MD AKASH	KHAN
103	NUR-E HABIB	NURSHED
104	RAIHANUL	ISLAM
105	JULKER	NINE
106	HABIUR	RAHAMAN
107	SAHADAT	HOSSAIN
108	SABAB	RAHAMAN
109	RICKI	SEN
110	DIPON	BISWAS
111	MAHADI	KABIR
112	REZAUL	KARIM
113	RISI	GUPTA
114	AQIB	HOWLADER
115	RISHAD	RAHMAN
116	RASEDUR	RAHAMAN
117	KHALID	HOSSAIN
118	RIYANTTI	SEN
119	SIMA	RAHAMAN
120	PRIYA	BISWAS
121	MEHEDI	RAHAMAN
122	REZAUL	SHARIF
123	RUMPA	KHAN
124	ISRAT	EVA
125	RAJU	RAHMAN

12 rows returned in 0.00 seconds

## DEPARTMENT TABLE:

4.

```
INSERT INTO DEPARTMENT VALUES('8801','FINANCE','10000');
INSERT INTO DEPARTMENT VALUES('8802','ACCOUNTING','15000');
INSERT INTO DEPARTMENT VALUES('8803','MARKETING','20000');
INSERT INTO DEPARTMENT VALUES('8804','OPERATION
MANAGEMENT','25000');
INSERT INTO DEPARTMENT VALUES('8805','SOFTWARE
MANAGEMENT','30000');
```

```
SELECT *
FROM DEPARTMENT
```

The screenshot shows the Oracle Database Express Edition interface. The top bar indicates the user is 'SCOTT'. The main area is titled 'SQL Commands' and contains the following SQL code:

```
INSERT INTO DEPARTMENT VALUES('8801','FINANCE','10000');
INSERT INTO DEPARTMENT VALUES('8802','ACCOUNTING','15000');
INSERT INTO DEPARTMENT VALUES('8803','MARKETING','20000');
INSERT INTO DEPARTMENT VALUES('8804','OPERATION MANAGEMENT','25000');
INSERT INTO DEPARTMENT VALUES('8805','SOFTWARE MANAGEMENT','30000');

SELECT *
FROM DEPARTMENT
```

Below the SQL editor, the 'Results' tab is selected, displaying a table with 5 rows:

DEPT_CODE	DEPT_NAME	SALARY
8801	FINANCE	10000
8802	ACCOUNTING	15000
8803	MARKETING	20000
8804	OPERATION MANAGEMENT	25000
8805	SOFTWARE MANAGEMENT	30000

At the bottom, it states '5 rows returned in 0.00 seconds' and 'CSV Export'.

## EMPLOYEE TABLE:

5.

```
INSERT INTO EMPLOYEE VALUES('2201','DHAKA','10000','8801','07','106');  
INSERT INTO EMPLOYEE VALUES('2202','DHAKA','10000','8802','08','107');  
INSERT INTO EMPLOYEE VALUES('2203','DHAKA','10000','8803','09','108');  
INSERT INTO EMPLOYEE VALUES('2204','DHAKA','10000','8804','10','109');  
INSERT INTO EMPLOYEE VALUES('2205','DHAKA','10000','8805','11','110');
```

```
SELECT *  
FROM EMPLOYEE
```



The screenshot shows the Oracle Database Express Edition interface. The main window displays the following SQL statements:

```
INSERT INTO EMPLOYEE VALUES('2201','DHAKA','10000','8801','07','106');  
INSERT INTO EMPLOYEE VALUES('2202','DHAKA','10000','8802','08','107');  
INSERT INTO EMPLOYEE VALUES('2203','DHAKA','10000','8803','09','108');  
INSERT INTO EMPLOYEE VALUES('2204','DHAKA','10000','8804','10','109');  
INSERT INTO EMPLOYEE VALUES('2205','DHAKA','10000','8805','11','110');
```

Below the SQL editor, the 'Results' tab is selected, showing the output of the SQL statements. The results are displayed in a table with the following columns: ID, CITY, SALARY, DEPT\_CODE, PHONE\_NO, and NO.

ID	CITY	SALARY	DEPT_CODE	PHONE_NO	NO
2201	DHAKA	10000	8801	07	106
2202	DHAKA	10000	8802	08	107
2203	DHAKA	10000	8803	09	108
2204	DHAKA	10000	8804	10	109
2205	DHAKA	10000	8805	11	110

At the bottom of the results window, it states: 5 rows returned in 0.00 seconds.

## ADDRESS TABLE:

6.

```
INSERT INTO ADDRESS VALUES('1000','314','612','CTG');
INSERT INTO ADDRESS VALUES('1001','317','125','DHAKA');
INSERT INTO ADDRESS VALUES('1002','312','516','COXSBAZAR');
INSERT INTO ADDRESS VALUES('1003','316','197','PEKUA');
INSERT INTO ADDRESS VALUES('1004','315','188','KULNA');
INSERT INTO ADDRESS VALUES('1005','344','172','CTG');
INSERT INTO ADDRESS VALUES('1006','327','165','DHAKA');
INSERT INTO ADDRESS VALUES('1007','352','156','COXSBAZAR');
INSERT INTO ADDRESS VALUES('1008','516','147','PEKUA');
INSERT INTO ADDRESS VALUES('1009','815','138','KULNA');
INSERT INTO ADDRESS VALUES('1010','214','122','CTG');
INSERT INTO ADDRESS VALUES('1011','717','418','DHAKA');
INSERT INTO ADDRESS VALUES('1012','212','116','COXSBAZAR');
INSERT INTO ADDRESS VALUES('1013','916','117','PEKUA');
INSERT INTO ADDRESS VALUES('1014','515','138','KULNA');
INSERT INTO ADDRESS VALUES('1015','214','162','CTG');
INSERT INTO ADDRESS VALUES('1016','817','115','DHAKA');
INSERT INTO ADDRESS VALUES('1017','352','116','COXSBAZAR');
INSERT INTO ADDRESS VALUES('1018','326','118','PEKUA');
INSERT INTO ADDRESS VALUES('1019','385','119','KULNA');
INSERT INTO ADDRESS VALUES('1020','395','117','KULNA');
```

```
SELECT *
FROM ADDRESS
```

The screenshot shows the Oracle Database Express Edition interface. The top bar indicates the user is 'SCOTT' and the session is '1000'. The main window displays a list of SQL commands, including 20 INSERT statements for the ADDRESS table. Below the commands, the 'Results' tab is active, showing a table with 20 rows of data. The table has four columns: ADDRESS\_ID, HOUSE\_NO, STREET\_NO, and CITY. The data is sorted by ADDRESS\_ID in ascending order. At the bottom, there is a message indicating that more than 10 rows are available and that the user can click on the 'SQL' link to see more rows.

ADDRESS_ID	HOUSE_NO	STREET_NO	CITY
1000	314	612	CTG
1001	317	125	DHAKA
1002	312	516	COXSBAZAR
1003	316	197	PEKUA
1004	315	188	KULNA
1005	344	172	CTG
1006	327	165	DHAKA
1007	352	156	COXSBAZAR
1008	516	147	PEKUA
1009	815	138	KULNA
1010	214	122	CTG
1011	717	418	DHAKA
1012	212	116	COXSBAZAR
1013	916	117	PEKUA
1014	515	138	KULNA
1015	214	162	CTG
1016	817	115	DHAKA
1017	352	116	COXSBAZAR
1018	326	118	PEKUA
1019	385	119	KULNA
1020	395	117	KULNA

## HOUSE OWNER TABLE:

7.

```
INSERT INTO HOUSE_OWNER
VALUES('884601','sajibul1@gamil.com','111','12','1000');
INSERT INTO HOUSE_OWNER
VALUES('884602','raihanul2@gamil.com','112','13','1001');
INSERT INTO HOUSE_OWNER
VALUES('884603','mdsakib3@gamil.com','113','14','1002');
INSERT INTO HOUSE_OWNER
VALUES('884604','mdabdul4@gamil.com','114','15','1003');
INSERT INTO HOUSE_OWNER
VALUES('884605','oniislam5@gamil.com','115','16','1004');
```

```
SELECT *
FROM HOUSE_OWNER
```

The screenshot displays the Oracle Database Express Edition web interface. The top navigation bar includes the Oracle logo, the text 'Database Express Edition', and user information 'User: SCOTT'. Below the navigation bar, the 'SQL Commands' tab is active, showing a series of INSERT statements for the HOUSE\_OWNER table. The 'Autocommit' checkbox is checked, and the 'Display' dropdown is set to '10'. The 'Save' and 'Run' buttons are visible in the top right of the command area. Below the command area, the 'Results' tab is active, displaying the output of the SELECT \* FROM HOUSE\_OWNER query. The results are shown in a table with 5 rows and 5 columns: HO\_ID, EMAIL, H\_ID, PHONE\_ID, and ADDRESS\_ID. The status bar at the bottom indicates '5 rows returned in 0.00 seconds' and 'Copyright © 1996-2008, Oracle. All rights reserved.'

```
INSERT INTO HOUSE_OWNER VALUES('884601','sajibul1@gamil.com','111','12','1000');
INSERT INTO HOUSE_OWNER VALUES('884602','raihanul2@gamil.com','112','13','1001');
INSERT INTO HOUSE_OWNER VALUES('884603','mdsakib3@gamil.com','113','14','1002');
INSERT INTO HOUSE_OWNER VALUES('884604','mdabdul4@gamil.com','114','15','1003');
INSERT INTO HOUSE_OWNER VALUES('884605','oniislam5@gamil.com','115','16','1004');
```

```
SELECT *
FROM HOUSE_OWNER
```

HO_ID	EMAIL	H_ID	PHONE_ID	ADDRESS_ID
884601	sajibul1@gamil.com	111	12	1000
884602	raihanul2@gamil.com	112	13	1001
884603	mdsakib3@gamil.com	113	14	1002
884604	mdabdul4@gamil.com	114	15	1003
884605	oniislam5@gamil.com	115	16	1004

5 rows returned in 0.00 seconds

Application Express 3.1.0.00.20  
Copyright © 1996-2008, Oracle. All rights reserved.

## CUSTOMER TABLE:

8.

INSERT INTO CUSTOMER

VALUES('884701','abcd1@gamil.com','DHAKA','116','17');

INSERT INTO CUSTOMER

VALUES('884702','efgh2@gamil.com','DHAKA','117','18');

INSERT INTO CUSTOMER

VALUES('884703','ijkl3@gamil.com','RAJSHAI','118','19');

INSERT INTO CUSTOMER

VALUES('884704','mnop4@gamil.com','KHULNA','119','20');

INSERT INTO CUSTOMER

VALUES('884705','qrst@gamil.com','BARISHAL','120','21');

SELECT \*

FROM CUSTOMER

The screenshot displays the Oracle Database Express Edition web interface. The top navigation bar includes 'Home', 'SQL', and 'SQL Commands'. The main text area contains five INSERT statements for the CUSTOMER table, followed by a SELECT \* FROM CUSTOMER query. The 'Results' tab is active, showing a table with 5 rows of data. The table columns are CU\_ID, EMAIL, CITY, NID, and PHONE\_ID. The data rows correspond to the inserted records. The status bar at the bottom indicates '5 rows returned in 0.00 seconds' and 'SQL Exec'd'. The footer shows 'Application Express 2.1.0.00.39' and 'Copyright © 1996-2000, Oracle. All rights reserved.'

```
INSERT INTO CUSTOMER VALUES('884701','abcd1@gamil.com','DHAKA','116','17');
INSERT INTO CUSTOMER VALUES('884702','efgh2@gamil.com','DHAKA','117','18');
INSERT INTO CUSTOMER VALUES('884703','ijkl3@gamil.com','RAJSHAI','118','19');
INSERT INTO CUSTOMER VALUES('884704','mnop4@gamil.com','KHULNA','119','20');
INSERT INTO CUSTOMER VALUES('884705','qrst@gamil.com','BARISHAL','120','21');

SELECT *
FROM CUSTOMER
```

CU_ID	EMAIL	CITY	NID	PHONE_ID
884701	abcd1@gamil.com	DHAKA	116	17
884702	efgh2@gamil.com	DHAKA	117	18
884703	ijkl3@gamil.com	RAJSHAI	118	19
884704	mnop4@gamil.com	KHULNA	119	20
884705	qrst@gamil.com	BARISHAL	120	21

5 rows returned in 0.00 seconds SQL Exec'd

Application Express 2.1.0.00.39  
Copyright © 1996-2000, Oracle. All rights reserved.



## PAYMENT TABLE:

9.

```
INSERT INTO PAYMENT VALUES( '7701','10000','BKASH');  
INSERT INTO PAYMENT VALUES( '7702','12000','NAGAD');  
INSERT INTO PAYMENT VALUES( '7703','14000','ROCKET');  
INSERT INTO PAYMENT VALUES( '7704','16000','UPAY');  
INSERT INTO PAYMENT VALUES( '7705','18000','SURECASH');
```

```
SELECT *  
FROM PAYMENT
```

The screenshot shows the Oracle Database Express Edition interface. The top bar indicates the user is SCOTT. The main window displays the following SQL commands:

```
INSERT INTO PAYMENT VALUES( '7701','10000','BKASH');  
INSERT INTO PAYMENT VALUES( '7702','12000','NAGAD');  
INSERT INTO PAYMENT VALUES( '7703','14000','ROCKET');  
INSERT INTO PAYMENT VALUES( '7704','16000','UPAY');  
INSERT INTO PAYMENT VALUES( '7705','18000','SURECASH');
```

Below the commands, the query `SELECT * FROM PAYMENT` is entered. The results are displayed in a table with the following data:

PAYMENT_ID	PRICE	PAYMENT_METHOD
7701	10000	BKASH
7702	12000	NAGAD
7703	14000	ROCKET
7704	16000	UPAY
7705	18000	SURECASH

5 rows returned in 0.00 seconds. The interface also includes a footer with the text "Application Express 7.1.0.03.20" and "Copyright © 1996, 2009, Oracle. All rights reserved."

## MECH TABLE:

10.

INSERT INTO MECH VALUES('88460011','02','03','02');

INSERT INTO MECH VALUES('88460012','01','02','03');

INSERT INTO MECH VALUES('88460013','02','02','01');

INSERT INTO MECH VALUES('88460014','01','02','04');

INSERT INTO MECH VALUES('88460015','02','03','05');

SELECT \*

FROM MECH

The screenshot shows the Oracle Database Express Edition web interface. The browser address bar displays the URL: 127.0.0.1:8080/apex/f?p=4500:1003:1111960142307644::NO::. The page title is "ORACLE Database Express Edition". The user is logged in as "SCOTT". The "SQL Commands" tab is active, showing a list of five INSERT statements and a SELECT statement. The "Run" button is visible. Below the SQL editor, the "Results" tab is active, displaying a table with 5 rows and 4 columns: MECH\_ID, BATHROOM\_NO, BED\_NO, and SUBLET\_NO. The table contains the following data:

MECH_ID	BATHROOM_NO	BED_NO	SUBLET_NO
88460011	2	3	2
88460012	1	2	3
88460013	2	2	1
88460014	1	2	4
88460015	2	3	5

Below the table, it states "5 rows returned in 0.00 seconds". The footer of the page includes "Language: en-us" and "Application Express 2.1.0.00.30 Copyright © 1996, 2006, Oracle. All rights reserved."

## SUBLET TABLE:

11.

```
INSERT INTO SUBLET
VALUES('8846001001','xyxyxy@gamil.com','CTG','121','88460011','22');
INSERT INTO SUBLET
VALUES('8846001002','ababab01@gamil.com','DHAKA','122','88460012','23');
INSERT INTO SUBLET
VALUES('8846001003','acacac02@gamil.com','KULNA','123','88460013','24');
INSERT INTO SUBLET
VALUES('8846001004','islam03@gamil.com','RAJUK','124','88460014','25');
INSERT INTO SUBLET
VALUES('8846001005','raiha04@gamil.com','COXSBAZAR','125','88460015','26');
```

```
SELECT *
FROM SUBLET
```

The screenshot shows the Oracle Database Express Edition web interface. The top navigation bar includes 'Home', 'SQL', and 'SQL Commands'. The main area contains a text editor with the following SQL commands:

```
INSERT INTO SUBLET VALUES('8846001001','xyxyxy@gamil.com','CTG','121','88460011','22');
INSERT INTO SUBLET VALUES('8846001002','ababab01@gamil.com','DHAKA','122','88460012','23');
INSERT INTO SUBLET VALUES('8846001003','acacac02@gamil.com','KULNA','123','88460013','24');
INSERT INTO SUBLET VALUES('8846001004','islam03@gamil.com','RAJUK','124','88460014','25');
INSERT INTO SUBLET VALUES('8846001005','raiha04@gamil.com','COXSBAZAR','125','88460015','26');

SELECT *
FROM SUBLET
```

Below the editor, the 'Results' tab is active, displaying a table with 6 columns: SUBLET\_ID, EMAIL, CITY, NO, MECH\_ID, and PHONE\_ID. The table contains 5 rows of data. At the bottom, it states '5 rows returned in 0.00 seconds'.

SUBLET_ID	EMAIL	CITY	NO	MECH_ID	PHONE_ID
8846001001	xyxyxy@gamil.com	CTG	121	88460011	22
8846001002	ababab01@gamil.com	DHAKA	122	88460012	23
8846001003	acacac02@gamil.com	KULNA	123	88460013	24
8846001004	islam03@gamil.com	RAJUK	124	88460014	25
8846001005	raiha04@gamil.com	COXSBAZAR	125	88460015	26

Application Express 2.1.0.0.0.0  
Copyright © 1996, 2009, Oracle. All rights reserved.

## RENTAL ITEM TABLE:

12.

```
INSERT INTO RENTAL_ITEM VALUES('88460501','FAMILY HOUSE','884601');
```

```
INSERT INTO RENTAL_ITEM VALUES('88460502','OFFICE SPACE','884602');
```

```
INSERT INTO RENTAL_ITEM VALUES('88460503','BACHELOR MACH','884603');
```

```
SELECT *
```

```
FROM RENTAL_ITEM
```

The screenshot shows the Oracle Database Express Edition web interface. The user is 'SCOTT'. The 'SQL Commands' tab is active, displaying the following SQL script:

```
INSERT INTO RENTAL_ITEM VALUES('88460501','FAMILY HOUSE','884601');
INSERT INTO RENTAL_ITEM VALUES('88460502','OFFICE SPACE','884602');
INSERT INTO RENTAL_ITEM VALUES('88460503','BACHELOR MACH','884603');

SELECT *
FROM RENTAL_ITEM
```

The 'Run' button has been clicked, and the results are displayed in the 'Results' tab. The results show 3 rows returned in 0.00 seconds.

RI_ID	RI_NAME	NO_ID
88460501	FAMILY HOUSE	884601
88460502	OFFICE SPACE	884602
88460503	BACHELOR MACH	884603

At the bottom of the interface, the language is set to 'en-us' and the application version is 'Express 2.1.0.00.20'. Copyright © 1996-2005, Oracle. All rights reserved.

## FAMILY HOUSE TABLE:

13.

```
INSERT INTO FAMILY_HOUSE VALUES('88460111','YES','01','YES','1005');
INSERT INTO FAMILY_HOUSE VALUES('88460112','YES','02','NO','1006');
INSERT INTO FAMILY_HOUSE VALUES('88460113','YES','03','YES','1007');
INSERT INTO FAMILY_HOUSE VALUES('88460114','NO','04','NO','1008');
INSERT INTO FAMILY_HOUSE VALUES('88460115','NO','05','NO','1009');
```

```
SELECT *
FROM FAMILY_HOUSE
```

The screenshot shows the Oracle Database Express Edition web interface. The top bar indicates the user is 'SCOTT'. The main area is titled 'SQL Commands' and contains the following SQL script:

```
INSERT INTO FAMILY_HOUSE VALUES('88460111','YES','01','YES','1005');
INSERT INTO FAMILY_HOUSE VALUES('88460112','YES','02','NO','1006');
INSERT INTO FAMILY_HOUSE VALUES('88460113','YES','03','YES','1007');
INSERT INTO FAMILY_HOUSE VALUES('88460114','NO','04','NO','1008');
INSERT INTO FAMILY_HOUSE VALUES('88460115','NO','05','NO','1009');

SELECT *
FROM FAMILY_HOUSE
```

Below the SQL editor, the 'Results' tab is active, displaying the output of the query. The results are shown in a table with 5 columns: FH\_ID, SECURITY, BED\_NO, LIFT\_FACILITY, and ADDRESS\_ID. There are 5 rows of data.

FH_ID	SECURITY	BED_NO	LIFT_FACILITY	ADDRESS_ID
88460113	YES	3	YES	1007
88460111	YES	1	YES	1005
88460112	YES	2	NO	1006
88460114	NO	4	NO	1008
88460115	NO	5	NO	1009

At the bottom of the results section, it states '5 rows returned in 0.00 seconds' and provides a 'CSV Export' link. The footer of the interface shows 'Application Express 2.1.0.0.0.0' and 'Copyright © 1996, 2005, Oracle. All rights reserved.'

## OFFICE SPACE TABLE:

14.

```
INSERT INTO OFFICE_SPACE
VALUES('88460211','2440*2400','YES','YES','YES','1010');
INSERT INTO OFFICE_SPACE
VALUES('88460212','2440*1600','YES','YES','YES','1011');
INSERT INTO OFFICE_SPACE
VALUES('88460213','2440*1800','NO','YES','NO','1012');
INSERT INTO OFFICE_SPACE
VALUES('88460214','2000*2000','YES','YES','YES','1013');
INSERT INTO OFFICE_SPACE
VALUES('88460215','1600*1600','YES','YES','YES','1014');
```

```
SELECT *
FROM OFFICE_SPACE
```

The screenshot shows the Oracle Database Express Edition web interface. The top bar displays the Oracle logo and 'Database Express Edition'. Below this, the user is logged in as 'SCOTT'. The main area is titled 'SQL Commands' and contains the following SQL script:

```
INSERT INTO OFFICE_SPACE VALUES('88460211','2440*2400','YES','YES','YES','1010');
INSERT INTO OFFICE_SPACE VALUES('88460212','2440*1600','YES','YES','YES','1011');
INSERT INTO OFFICE_SPACE VALUES('88460213','2440*1800','NO','YES','NO','1012');
INSERT INTO OFFICE_SPACE VALUES('88460214','2000*2000','YES','YES','YES','1013');
INSERT INTO OFFICE_SPACE VALUES('88460215','1600*1600','YES','YES','YES','1014');

SELECT *
FROM OFFICE_SPACE
```

Below the SQL editor, the 'Results' tab is active, showing the output of the query. The results are displayed in a table with 6 columns: OS\_ID, OFFICE\_SIZE, SECURITY, LIFT\_FACILITY, GARAGE\_FACILITY, and ADDRESS\_ID. The table contains 5 rows of data. At the bottom, it states '5 rows returned in 0.02 seconds'.

OS_ID	OFFICE_SIZE	SECURITY	LIFT_FACILITY	GARAGE_FACILITY	ADDRESS_ID
88460214	2000*2000	YES	YES	YES	1013
88460215	1600*1600	YES	YES	YES	1014
88460211	2440*2400	YES	YES	YES	1010
88460212	2440*1600	YES	YES	YES	1011
88460213	2440*1800	NO	YES	NO	1012

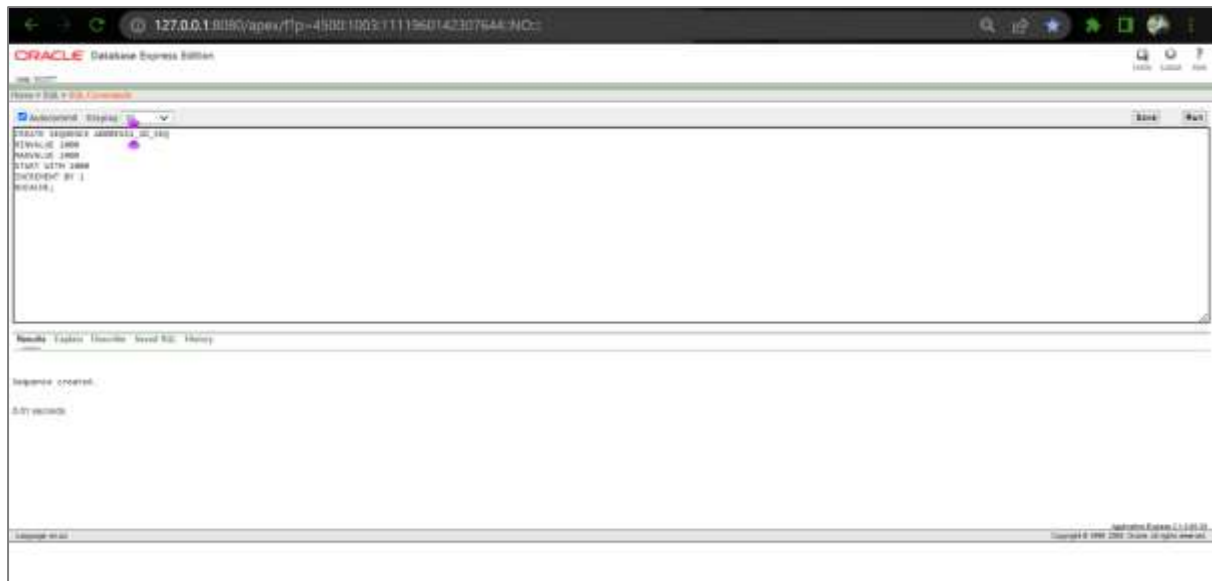
Application Express 2.1.0.00.39  
Copyright © 1996, 2006, Oracle. All rights reserved.

## Sequence:

1.  
CREATE SEQUENCE PHONE\_ID\_SEQ  
MINVALUE 1  
MAXVALUE 100  
START WITH 1  
INCREMENT BY 1  
NOCACHE;



2.  
CREATE SEQUENCE NID1\_SEQ  
MINVALUE 101  
MAXVALUE 200  
START WITH 101  
INCREMENT BY 1  
NOCACHE;



3.  
 CREATE SEQUENCE ADDRESS1\_ID\_SEQ  
 MINVALUE 1000  
 MAXVALUE 2000  
 START WITH 1000  
 INCREMENT BY 1  
 NOCACHE;

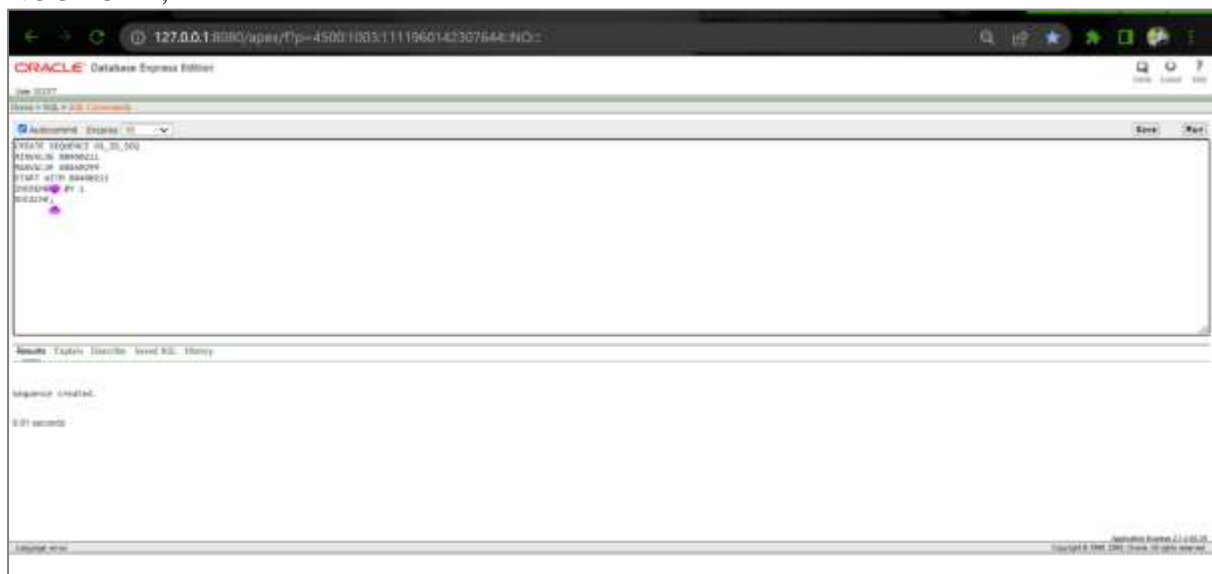


4.  
 CREATE SEQUENCE EID\_SEQ  
 MINVALUE 2201  
 MAXVALUE 2299  
 START WITH 2201  
 INCREMENT BY 1  
 NOCACHE;

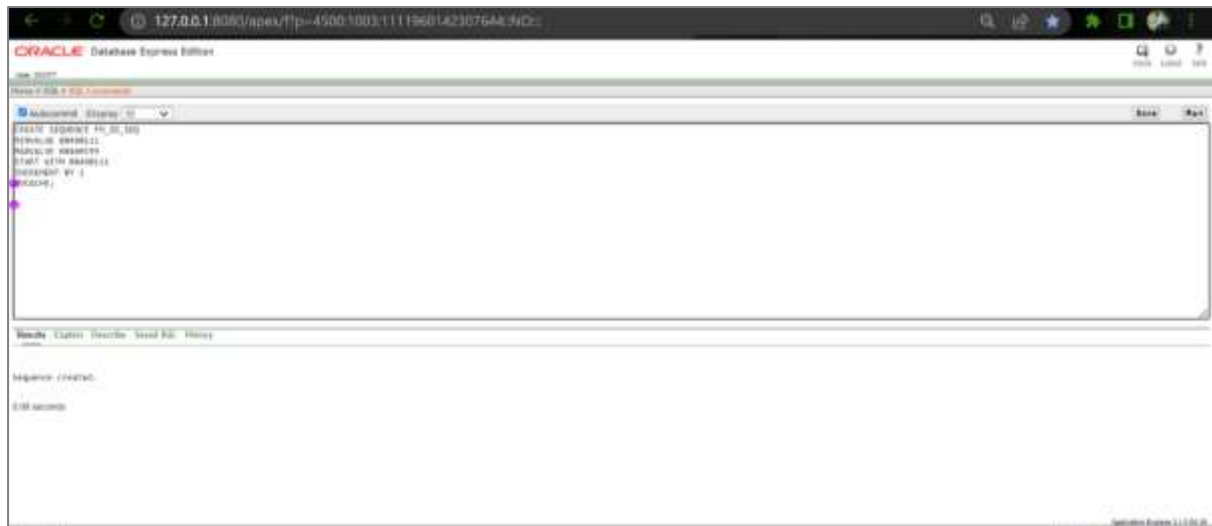




5.  
 CREATE SEQUENCE DEPTCODE\_SEQ  
 MINVALUE 8801  
 MAXVALUE 8820  
 START WITH 8801  
 INCREMENT BY 1  
 NOCACHE;



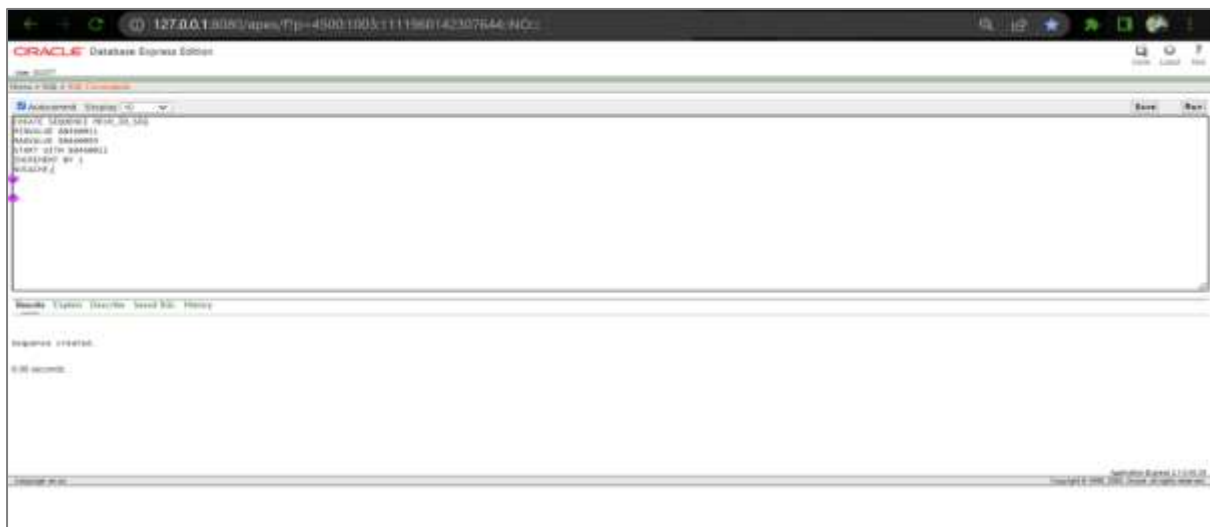
6.  
 CREATE SEQUENCE HO\_ID\_SEQ  
 MINVALUE 884601  
 MAXVALUE 884699  
 START WITH 884601  
 INCREMENT BY 1  
 NOCACHE;



7.  
 CREATE SEQUENCE CU\_ID\_SEQ  
 MINVALUE 884701  
 MAXVALUE 884799  
 START WITH 884701  
 INCREMENT BY 1  
 NOCACHE;



8.  
 CREATE SEQUENCE PAYMENT\_ID\_SEQ  
 MINVALUE 7701  
 MAXVALUE 7799  
 START WITH 7701  
 INCREMENT BY 1  
 NOCACHE;



9.  
 CREATE SEQUENCE MECH\_ID\_SEQ  
 MINVALUE 88460011  
 MAXVALUE 88460099  
 START WITH 88460011  
 INCREMENT BY 1  
 NOCACHE;



10.  
 CREATE SEQUENCE SUBLET\_ID\_SEQ  
 MINVALUE 8846001001  
 MAXVALUE 8846001200

START WITH 8846001001  
INCREMENT BY 1  
NOCACHE;

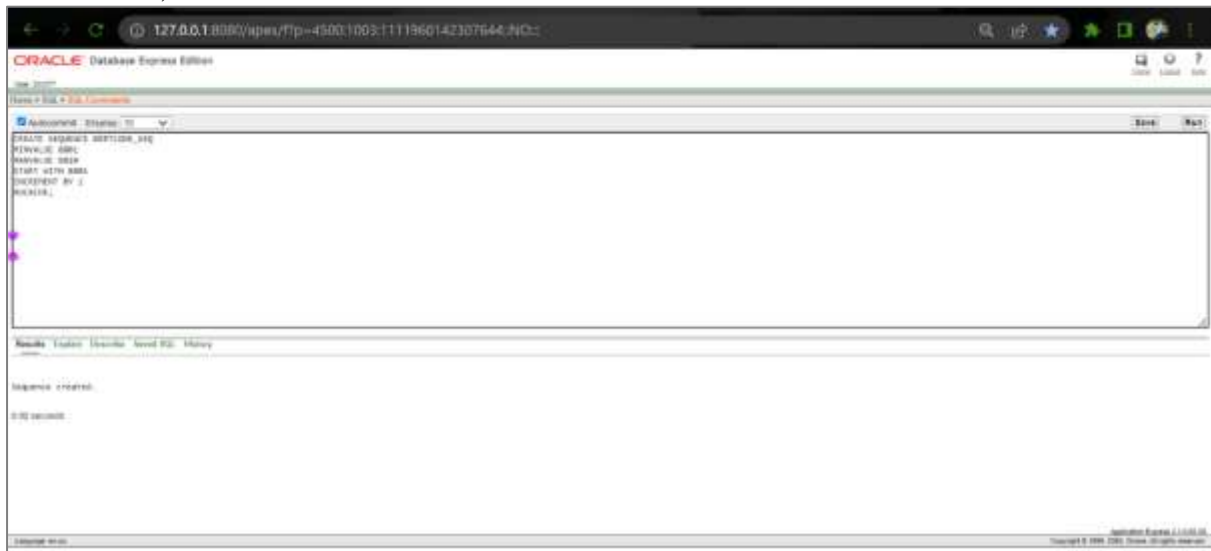


11.  
CREATE SEQUENCE FM\_ID\_SEQ  
MINVALUE 88460111  
MAXVALUE 88460199  
START WITH 88460111  
INCREMENT BY 1  
NOCACHE;

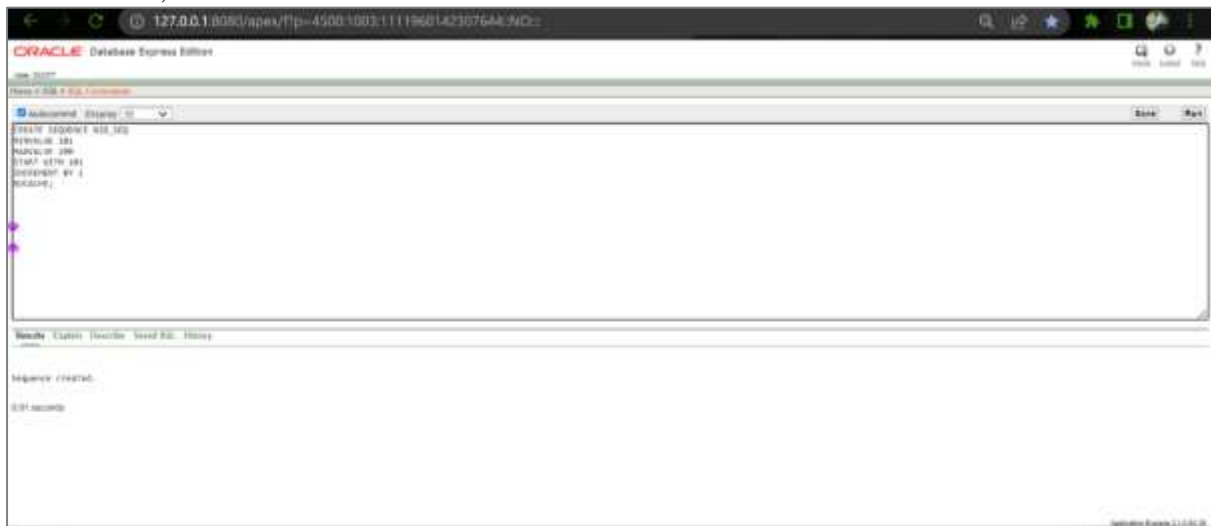


12.  
CREATE SEQUENCE OS\_ID\_SEQ  
MINVALUE 88460211  
MAXVALUE 88460299  
START WITH 88460211

INCREMENT BY 1  
NOCACHE;



13.  
CREATE SEQUENCE RI\_ID\_SEQ  
MINVALUE 88460501  
MAXVALUE 88460999  
START WITH 88460511  
INCREMENT BY 1  
NOCACHE;



## Queries:

1. Show the Employee ID, Salary, First & Last name from Employee and Name Table using Joining

E_ID	SALARY	FIRST_NAME	LAST_NAME
2201	10000	HABIUR	RAHAMAN
2202	10000	SAHADAT	HOSSAIN
2203	10000	SABAB	RAHAMAN
2204	10000	RICKI	SEN
2205	10000	DIPON	BISWAS

```
SELECT EMPLOYEE.E_ID,EMPLOYEE.SALARY,NAME.FIRST_NAME, NAME.LAST_NAME
FROM EMPLOYEE,NAME
WHERE EMPLOYEE.NID=NAME.NID;
```

2. Write a Query to show the Price and Payment-id from Payment Table where the price is greater than the price of Payment\_id = 7702.

```
SELECT PAYMENT_ID,PRICE
FROM PAYMENT
WHERE PRICE>(SELECT PRICE
              FROM PAYMENT
              WHERE PAYMENT_ID = 7702);
```

PAYMENT_ID	PRICE
7703	14000
7704	16000
7705	18000

3. Show the Minimum price, Maximum Price avg Price from Payment Table

```
SELECT MIN(PRICE),MAX(PRICE),AVG(PRICE)
FROM PAYMENT
```

MIN(PRICE)	MAX(PRICE)	AVG(PRICE)
10000	18000	14000

4. Write a query to join Sublet table and Name Table & find the Sublets name where name has S letter at first

```
SELECT S.NID,S.CITY,N.FIRST_NAME,N.NID
FROM SUBLET S,NAME N
WHERE S.NID=N.NID
AND FIRST_NAME LIKE 'S%'
no data found
```

5. Create a view for customer 884701 & 884704 to show their ID and Address

```
CREATE VIEW CUSTOMERVU  
AS SELECT CU_ID ID,CITY ADDRRESS  
FROM CUSTOMER  
WHERE CU_ID = 884701  
OR CU_ID = 884704
```

```
SELECT *  
FROM CUSTOMERVU
```

ID	ADDRRESS
884701	DHAKA
884704	KHULNA

6. Create view for Family house table to show ID, Bed No, Lift Facility

```
CREATE VIEW FAMILY_HOUSEVU  
AS SELECT FH_ID ID,LIFT_FACILITY LIFT,BED_NO BEDROOM  
FROM FAMILY_HOUSE  
SELECT *  
FROM FAMILY_HOUSEVU
```

ID	LIFT	BEDROOM
88460111	YES	1
88460112	NO	2
88460113	YES	3
88460114	NO	4
88460115	NO	5

7. Use Literal Character query to show all the output like “ ##### use Bkash to pay the Price ###”

From Payment table

```
SELECT PAYMENT_ID||' use '|PAYMENT_METHOD||' to pay price '|PRICE  
FROM PAYMENT
```

PAYMENT_ID  'USE'  PAYMENT_METHOD  'TOPAYPRICE'  PRICE
7701 use BKASH to pay price 10000
7702 use NAGAD to pay price 12000
7703 use ROCKET to pay price 14000
7704 use UPAY to pay price 16000
7705 use SURECASH to pay price 18000

8. Increase the Salary by 5000 of the Finance Department AND Marketing Department

```
SELECT DEPT_NAME,SALARY+5000  
FROM DEPARTMENT  
WHERE DEPT_NAME = 'FINANCE'  
OR DEPT_NAME = 'MARKETING'
```

DEPT_NAME	SALARY+5000
FINANCE	15000
MARKETING	25000

9. Show the city,street no, house no, of House Owner from House Owner and Address Table

```
SELECT HOUSE_OWNER.HO_ID,ADDRESS.HOUSE_NO,ADDRESS.STREET_NO,ADDRESS.CITY  
FROM HOUSE_OWNER,ADDRESS  
WHERE HOUSE_OWNER.ADDRESS_ID=ADDRESS.ADDRESS_ID;
```

HO_ID	HOUSE_NO	STREET_NO	CITY
884601	314	612	CTG
884602	317	125	DHAKA
884603	312	516	COXSBAZAR
884604	316	197	PEKUA
884605	315	188	KULNA

10. Find the Salary & the Name of the Department where Salary is greater then 2000

```
SELECT DEPT_CODE,DEPT_NAME,SALARY  
FROM DEPARTMENT  
WHERE SALARY < 20000
```

DEPT_CODE	DEPT_NAME	SALARY
8801	FINANCE	10000
8802	ACCOUNTING	15000