Name: Abrar Khatib Lajim

ID: 2022-3-60-043

Section:7

CSE 302: LAB02 (Exercise- Offline)

Course Instructor: Mahmuda Rawnak Jahan

Lab Task # 01 (Schema Definition):

i) account

account_no	char(5)	primary key
balance	number	Not null and cannot be less than 0

=>

```
--1.1
CREATE TABLE account (
    account_no CHAR(5) PRIMARY KEY,
    balance NUMBER NOT NULL CHECK (balance >= 0)
);

desc account;
```

```
Name        Null?    Type
----------- -------- -------
ACCOUNT_NO  NOT NULL CHAR(5)
BALANCE     NOT NULL NUMBER
```

ii) customer

customer_no	char(5)	primary key
customer_name	varchar2(20)	Not null
customer_city	varchar2(10)	

=>

```
--1.2
CREATE TABLE customer (
    customer_no CHAR(5) PRIMARY KEY,
    customer_name VARCHAR2(20) NOT NULL,
    customer_city VARCHAR2(10)
);
desc customer;
```

```
Table CUSTOMER created.

Name          Null?    Type
------------- -------- ------------
CUSTOMER_NO   NOT NULL CHAR(5)
CUSTOMER_NAME NOT NULL VARCHAR2(20)
CUSTOMER_CITY          VARCHAR2(10)
```

iii) depositor

account_no	char(5)	
customer_no	char(5)	
		primary key (account_no, customer_no)

```
--1.3
CREATE TABLE depositor (
    account_no CHAR(5),
    customer_no CHAR(5),
    PRIMARY KEY (account_no, customer_no),
    FOREIGN KEY (account_no) REFERENCES account(account_no),
    FOREIGN KEY (customer_no) REFERENCES customer(customer_no)
);
desc depositor;
```

```
Name        Null?    Type
----------- -------- -------
ACCOUNT_NO  NOT NULL CHAR(5)
CUSTOMER_NO NOT NULL CHAR(5)
```

Lab Task # 02 (Schema Modification):

i. Write SQL statement to add a new attribute 'date_of_birth' (date type) in customer table.

```
--2.1
ALTER TABLE customer ADD date_of_birth DATE;
desc account;
```

```
Name          Null?    Type
------------- -------- ------------
CUSTOMER_NO   NOT NULL CHAR(5)
CUSTOMER_NAME NOT NULL VARCHAR2(20)
CUSTOMER_CITY          VARCHAR2(10)
DATE_OF_BIRTH          DATE
```

ii. Write SQL statement to drop the attribute 'date_of_birth' from customer table.

```
--2.2
ALTER TABLE customer DROP COLUMN date_of_birth;
desc customer;
```

TABLE CUSTOMER

Column	Null?	Type
CUSTOMER_NO	NOT NULL	CHAR(5)
CUSTOMER_NAME	NOT NULL	VARCHAR2(20)
CUSTOMER_CITY	-	VARCHAR2(10)

iii. Write SQL statement to rename the attribute account_no, customer_no from depositor table to a_no and c_no, respectively.

```
33  --2.3
34  ALTER TABLE depositor RENAME COLUMN account_no TO a_no;
35  ALTER TABLE depositor RENAME COLUMN customer_no TO c_no;
36  desc depositor;
```

TABLE DEPOSITOR

Column	Null?	Type
A_NO	NOT NULL	CHAR(5)
C_NO	NOT NULL	CHAR(5)

iv. Write SQL statements to add two foreign key constraints 'depositor_fk1' and 'depositor_fk2' which identifies a_no and c_no as a foreign key.

```
37  --2.4
38  SELECT constraint_name, constraint_type
39  FROM user_constraints
40  WHERE table_name = 'DEPOSITOR';
41  ALTER TABLE depositor DROP CONSTRAINT SYS_C00171903934;
42  ALTER TABLE depositor DROP CONSTRAINT SYS_C00171903935;
43
44  ALTER TABLE depositor ADD CONSTRAINT depositor_fk1 FOREIGN KEY (a_no) REFERENCES account(account_no);
45  ALTER TABLE depositor ADD CONSTRAINT depositor_fk2 FOREIGN KEY (c_no) REFERENCES customer(customer_no);
```

CONSTRAINT_NAME	CONSTRAINT_TYPE
SYS_C00171903933	P
DEPOSITOR_FK1	R
DEPOSITOR_FK2	R

Lab Task # 03 (Inserting Records into Tables):

Account

ACCOUNT_NO	BALANCE
A-101	12000
A-102	6000
A-103	2500

Customer

CUSTOMER_NO	CUSTOMER_NAME	CUSTOMER_CITY
C-101	Alice	Dhaka
C-102	Annie	Dhaka
C-103	Bob	Chittagong
C-104	Charlie	Khulna

Depositor

A_NO	C_NO
A-101	C-101
A-103	C-102
A-103	C-104
A-102	C-103

```
49  --3.1
50  INSERT INTO account (account_no, balance) VALUES ('A-101', 12000);
51  INSERT INTO account (account_no, balance) VALUES ('A-102', 6000);
52  INSERT INTO account (account_no, balance) VALUES ('A-103', 2500);
53  SELECT*FROM account;
```

ACCOUNT_NO	BALANCE
A-101	12000
A-102	6000
A-103	2500

```
54  --3.2
55  INSERT INTO customer (customer_no, customer_name, customer_city) VALUES ('C-101', 'Alice', 'Dhaka');
56  INSERT INTO customer (customer_no, customer_name, customer_city) VALUES ('C-102', 'Annie', 'Dhaka');
57  INSERT INTO customer (customer_no, customer_name, customer_city) VALUES ('C-103', 'Bob', 'Chittagong');
58  INSERT INTO customer (customer_no, customer_name, customer_city) VALUES ('C-104', 'Charlie', 'Khulna');
59  SELECT*FROM customer;
```

CUSTOMER_NO	CUSTOMER_NAME	CUSTOMER_CITY
C-101	Alice	Dhaka
C-102	Annie	Dhaka
C-103	Bob	Chittagong
C-104	Charlie	Khulna

```
60  --3.3
61  INSERT INTO depositor (a_no, c_no) VALUES ('A-101', 'C-101');
62  INSERT INTO depositor (a_no, c_no) VALUES ('A-103', 'C-102');
63  INSERT INTO depositor (a_no, c_no) VALUES ('A-103', 'C-104');
64  INSERT INTO depositor (a_no, c_no) VALUES ('A-102', 'C-103');
65  SELECT*FROM depositor;
```

A_NO	C_NO
A-101	C-101
A-102	C-103
A-103	C-102
A-103	C-104

Lab Task # 04 (Queries):

i. Display customer name and customer city only.

```
66  --4.1
67  SELECT customer_name, customer_city FROM customer;
```

CUSTOMER_NAME	CUSTOMER_CITY
Alice	Dhaka
Annie	Dhaka
Bob	Chittagong
Charlie	Khulna

ii. Display the unique customer city. No repetitions are allowed.

```
69  --4.2
70  SELECT DISTINCT customer_city FROM customer;
```

CUSTOMER_CITY
Chittagong
Dhaka
Khulna

iii. Find account numbers with balance more than 7000.

```
72  --4.3
73  SELECT account_no FROM account WHERE balance > 7000;
```

ACCOUNT_NO
A-101

-

iv. Find customer number and customer name who live in Khulna.

```
75  --4.4
76  SELECT customer_no, customer_name FROM customer WHERE customer_city = 'Khulna';
```

CUSTOMER_NO	CUSTOMER_NAME
C-104	Charlie

v. Find customer number and customer name who do not live in Dhaka.

```
78  --4.5
79  SELECT customer_no, customer_name FROM customer WHERE customer_city <> 'Dhaka';
```

CUSTOMER_NO	CUSTOMER_NAME
C-103	Bob
C-104	Charlie

vi. Find customer name and customer city who have accounts with balance more than 7000. vii. Find customer name and customer city who have accounts with balance more than 7000 and do not live in Khulna.

```
81  --4.6
82  SELECT customer.customer_name, customer.customer_city
83  FROM customer JOIN depositor ON customer.customer_no = depositor.c_no
84  JOIN account ON depositor.a_no = account.account_no
85  WHERE account.balance > 7000;
```

CUSTOMER_NAME	CUSTOMER_CITY
Alice	Dhaka

viii. Find account number and balance for those accounts which belong to a customer with id 'C-102'.

```
94  --4.8
95  SELECT account.account_no, account.balance
96  FROM account
97  JOIN depositor ON account.account_no = depositor.a_no
98  WHERE depositor.c_no = 'C-102';
99
```

ACCOUNT_NO	BALANCE
A-103	2500

ix. Find all account number and balance for those accounts which belong to customers of Dhaka and Khulna city.

```
100 --4.9
101 SELECT account.account_no, account.balance
102 FROM account
103 JOIN depositor ON account.account_no = depositor.a_no
104 JOIN customer ON depositor.c_no = customer.customer_no
105 WHERE customer.customer_city IN ('Dhaka', 'Khulna');
```

ACCOUNT_NO	BALANCE
A-101	12000
A-103	2500
A-103	2500

x. Find the customer who have no accounts. [Result of this query will be empty for this dataset. However, you must write the correct SQL]

```
107 --4.10
108 SELECT customer.customer_no, customer.customer_name
109 FROM customer LEFT JOIN depositor ON customer.customer_no = depositor.c_no
110 WHERE depositor.a_no IS NULL;
111
```

no data found