

PRACTICE-SET-2

①.

①.

CREATE Customer Table.

```
CREATE TABLE Customer (  
  CustomerID INT PRIMARY KEY AUTO-INCREMENT,  
  FirstName VARCHAR(50),  
  LastName VARCHAR(50),  
  Email VARCHAR(100) UNIQUE,  
  PhoneNumber VARCHAR(15),  
  Address VARCHAR(200)  
);
```

②. Create Accounts Table.

```
CREATE Table Accounts (  
  AccountNumber INT PRIMARY KEY,  
  CustomerID INT,  
  AccountType VARCHAR(20),  
  Balance DECIMAL(10, 2),  
  DateCreated DATE,  
  FOREIGN KEY (CustomerID) REFERENCES  
  Customer (CustomerID)  
);
```

②. Inserting Data

③. Insert Data into Customers Table.

```
INSERT INTO Customers (FirstName, LastName,  
Email, PhoneNumber, Address)
```

```
VALUES
```

```
('John', 'Doe', 'John.doe@gmail.com', '1234567890',  
'123 Main St, Cityville'),
```

```
('Jane', 'Smith', 'Jane.smith@gmail.com',  
'0987654321', '456 Elm St, Townville') ;
```

④. INSERT Data in Account Table.

```
INSERT INTO Accounts (AccountNumber,  
CustomerID, AccountType, Balance, DateCreated)
```

```
VALUES
```

```
(1001, 1, 'Savings', 5000.00, '2023-01-15'),
```

```
(1002, 1, 'Checking', 1500.00, '2023-02-20'),
```

```
(1003, 2, 'Savings', 2000.00, '2023-03-01');
```

③. Update Data

⑤.

```
UPDATE Account  
SET Balance = 5500.000  
WHERE AccountNumber = 1001;
```

⑥.

```
UPDATE Email Customer  
SET email = 'jane.smith@new@gmail.com'  
WHERE CustomerID = 2;
```

⑦.

```
UPDATE Accounts  
SET Balance = Balance * 1.10  
WHERE AccountType = 'Saving';
```

④. SELECT Queries

⑧.

```
SELECT CustomerID, firstName, LastName, Balance  
FOR Saving AccountType = 'Saving';
```

⑨.⑧.

```
SELECT Customer.CustomerID, Customer.firstName,  
Customer.LastName, Accounts.Balance  
FROM Customers  
JOIN Accounts ON Customer.CustomerID =  
Accounts.CustomerID  
WHERE Accounts.AccountType = 'Saving';
```


9.

```
SELECT customers.CustomerID, customers.firstName,  
       customers.lastName, Accounts.Balance  
FROM Customers  
JOIN Accounts ON Customers.CustomerID =  
               Accounts.CustomerID  
WHERE Accounts.Balance > 3000 AND  
       Account.AccountType = 'checking' ;
```

10.

```
SELECT * FROM Accounts  
WHERE Balance < 2000 ;
```

~~11.~~

5.

DELETING DATA

11.

```
DELETE FROM Accounts  
WHERE AccountNumber = 1002 ;
```

12.

```
DELETE FROM Customer  
WHERE phonenumber LIKE '123%.' ;
```

13.

```
DELETE FROM Accounts  
WHERE DateCreated < '2023-02-01'
```

6. Join Queries

(14).
SELECT Customer.firstName, Customer.lastName,
Account.AccountType FROM Customer
JOIN ACCOUNTS ON Customer.CustomerID = Account.
AccountID
WHERE Account.balance > 2000 ;

(15).
SELECT AccountType, SUM(balance) AS TotalBalance
FROM Customer | Account
WHERE AccountType = 'Saving'
GROUP BY AccountType ;

(16).
SELECT Customer.firstName, Customer.lastName,
Account.AccountNumber, Account.Balance
FROM Customer
LEFT JOIN Accounts ON Customer.CustomerID
= Account.customerID ;

7. Constraints and Validation

(17).
ALTER TABLE Accounts
ADD CONSTRAINT Check-balance CHECK (Balance >= 0);

18.

```
ALTER TABLE Customers  
MODIFY COLUMN Email VARCHAR(100) NOT NULL  
UNIQUE;
```

19. ALTER TABLE Accounts

```
ADD CONSTRAINT fk-customer FOREIGN KEY (customerID)  
REFERENCES Customers (customerID);
```

20.

ALTER TABLE Accounts

```
ADD CONSTRAINT Check-account-type CHECK  
(AccountType IN ('saving', 'checking'));
```

21.

Complex Queries

21.

```
SELECT Customers.firstName, Customers.LastName,  
Accounts.AccountNumber, MAX (Accounts.Balance )  
AS HighestBalance  
FROM Customers  
JOIN Accounts ON Customers.customerID =  
Accounts.customerID  
GROUP BY Customers.customerID  
ORDER BY HighestBalance DESC  
LIMIT 1;
```


(22)

BEGIN TRANSACTION;

UPDATE Accounts

SET Balance = Balance - 1000

WHERE AccountNumber = 1003;

UPDATE Accounts

SET Balance = Balance + 1000

WHERE AccountNumber = 1001;

COMMIT;

(23)

SELECT Customers.firstName, Customers.lastName,
SUM (Accounts.Balance) AS Total Balance
FROM Customers

JOIN Accounts ON Customers.CustomerID =
Accounts.customerID

GROUP BY Customers.CustomerID;

(24)

SELECT AVG (Balance) AS AverageBalance
FROM Accounts;

25.

```
SELECT COUNT(*) AS Total Saving Accounts  
FROM Accounts  
WHERE AccountType = 'Saving' ;
```

26.

```
BEGIN TRANSACTION
```

```
UPDATE Accounts
```

```
SET Balance = Balance - 500
```

```
WHERE AccountNumber = 1002 ;
```

```
UPDATE Accounts
```

```
SET Balance = Balance + 500
```

```
WHERE AccountNumber = 1003 ;
```

```
COMMIT ;
```