SQL 실습

1. CREATE TABLE

Orders(OrderID, CustomerID, OrderDate, Amount)

Customer(CustomerID, CustomerName, Country)

--

create table Customer(

 CustomerID text PRIMARY key,

 CustomerName TEXT,

 Country TEXT

);

create table Orders(

 OrderId text,

 CustomerID TEXT,

 OrderDate Date,

 Amount INTEGER

);

--

2. INSERT VALUES

--

INSERT INTO customer VALUES

 ('CutomerID1', 'Jacob','India'),

 ('CutomerID2', 'Michael', 'USA'),

 ('CutomerID3', 'Joshuna', 'India'),

 ('CutomerID4', 'Matthew', 'USA'),

 ('CutomerID5', 'Suhee', 'Korea'),

 ('CutomerID6', 'Youngbok', 'Korea'),

 ('CutomerID7', 'Lucas', 'India'),

 ('CutomerID8', 'Mane', ‘UK’),

 ('CutomerID9', 'Alexander', 'USA'),

 ('CutomerID10', 'James', 'Germany');

INSERT INTO Orders VALUES

 ('OrderID1', 'CutomerID1', CURRENT\_DATE, 1000),

 ('OrderID2', 'CutomerID1', CURRENT\_DATE, 1100),

 ('OrderID3', 'CutomerID1', CURRENT\_DATE, 1200),

 ('OrderID4', 'CutomerID7', CURRENT\_DATE, 1300),

 ('OrderID5', 'CutomerID7', CURRENT\_DATE, 1400),

 ('OrderID6', 'CutomerID3', CURRENT\_DATE, 1500),

 ('OrderID7', 'CutomerID2', CURRENT\_DATE, 1600),

 ('OrderID8', 'CutomerID5', CURRENT\_DATE, 1700),

 ('OrderID9', 'CutomerID5', CURRENT\_DATE, 1800),

 ('OrderID10', 'CutomerID4', CURRENT\_DATE, 1900),

 ('OrderID11', 'CutomerID9', CURRENT\_DATE, 2000);

--

3. 하나 이상의 Order가 있는 Customer의 이름을 출력하여라.

1) option 1

SELECT DISTINCT cus.customername

FROM Customer as cus

WHERE EXISTS ( SELECT cus

 FROM customer, orders

 WHERE cus.customerid=orders.customerid );

Why not ?

SELECT DISTINCT customername

FROM Customer

WHERE EXISTS ( SELECT customer

 FROM customer, orders

 WHERE customer.customerid=orders.customerid );

2) option 2

SELECT C.CustomerName

FROM Customer as C, (SELECT CustomerID

 FROM Orders

 GROUP BY CustomerID

HAVING COUNT(CustomerID)>=1) as O

WHERE C.CustomerID=O.CustomerID

4. 셋 이상의 Order가 있는 Customer의 이름을 출력하여라.

--

SELECT Customer.customername, COUNT(Orders.\*)

FROM Customer, Orders

WHERE Customer.CustomerID = Orders.CustomerID

GROUP BY Customer.customerid

HAVING COUNT(Orders.orderid)>=3 ;

--

5. 세 명 이상의 Customer가 있는 나라를 출력하여라.

--

SELECT country, COUNT(customer.\*)

FROM Customer

GROUP BY country

HAVING COUNT(country)>=3;

--

6. 가장 큰 주문 번호는?

1) 먼저

SELECT max(amount)

FROM orders;

2) Max와 같은 값을 가지고 있는 주문량의 order를 검색

SELECT Orders.orderId, orders.amount

FROM Orders

WHERE Orders.amount >= ( SELECT MAX(amount) FROM Orders);

7. 두 번째로 큰 주문 양은?

SELECT MAX(orders.amount)

FROM Orders

WHERE Orders.amount < ( SELECT max(amount) FROM Orders);

8. 다섯 번째로 큰 주문량은?

SELECT ord1.orderid

FROM orders as ord1

WHERE 5 = (SELECT COUNT(ord2)

 FROM orders as ord2

 Where ord1.amount <= ord2.amount);

or

SELECT orderid

FROM orders

ORDER BY amount DESC

LIMIT 1 OFFSET 4

9. 가장 큰 주문을 한 고객 이름?

1) 먼저 가장 큰 주문을 찾고,

SELECT Max(amount) FROM orders

2) 그 주문량을 한 고객을 찾는다.

SELECT customer.customername

FROM customer, orders

WHERE customer.customerid=orders.customerid AND orders.amount>=

 (SELECT Max(amount) FROM orders);

10. 가장 많은 Order를 하는 나라를 출력하여라.

1) 각 나라별로 주문량을 찾고,

 SELECT COUNT(customer.country)

 FROM Customer, orders

 WHERE customer.customerid=orders.customerid

 GROUP BY customer.country

2) 최대 주문량을 한 나라를 찾는다.

SELECT country, count(country)

FROM customer, orders

WHERE customer.customerid=orders.customerid

GROUP BY country

HAVING COUNT(customer.country) >= ALL (

 SELECT COUNT(customer.country)

 FROM Customer, orders

 WHERE customer.customerid=orders.customerid

 GROUP BY customer.country );

--

11. India에서 총 주문량(Amount)이 가장 많은 Customer의 이름을 출력하여라.

1) India의 각 고객별 총 주문량을 찾고:

SELECT customer.customername, SUM(orders.amount)

 FROM Customer, orders

 WHERE customer.customerid=orders.customerid AND customer.country='India'

 GROUP BY customer.customername;

2) 총 주문량이 가장 많은 고객을 찾는다.

SELECT customer.customername, SUM(orders.amount)

FROM customer, orders

WHERE customer.customerid=orders.customerid AND customer.country='India'

GROUP BY customer.customername

HAVING SUM(orders.amount) >= ALL (

 SELECT SUM(orders.amount)

 FROM Customer, orders

 WHERE customer.customerid=orders.customerid AND customer.country='India'

 group by customer.customername);