

College Of Engineering Trivandrum

Application Software Development Lab



Abhishek Manoharan

S5 CSE Roll No:2

TVE17CS002

Department of Computer Science

August 14, 2019



Cycle 1

Exp No 7

JOIN STATEMENTS, SET OPERATIONS, NESTED QUERIES AND GROUPING

1 Aim

To get introduced to

- -UNION
- -INTERSECTION
- -MINUS
- - JOIN
- - NESTED QUERIES
- - GROUP BY & HAVING

2 Questions

Create Items,Orders,Customers,Delivery tables and populate them with appropriate data.

```
create table items(  
itemid int not null,  
itemname varchar(50) not null,  
category varchar(20) not null,  
price int not null,  
instock int ,  
constraint checkstock check(instock >0),  
constraint pkey primary key(itemid));
```

```
insert into items values(5,'sony z5 premium','electronics',5005,1);  
insert into items values(4,'Samsung Galaxy S4','electronics',5005,1);  
insert into items values(3,'One Plus 7','electronics',6006,2);  
insert into items values(2,'Iphone X','electronics',7007,6);  
insert into items values(1,'Xiomi','electronics',1001,6);
```

0. Display the details items table

```
select * from items;
```

```
asdlab=# select * from items;
 itemid | itemname          | category   | price | instock
-----+-----+-----+-----+-----
      5 | sony z5 premium   | electronics |  5005 |      1
      4 | Samsung Galaxy S4 | electronics |  5005 |      1
      3 | One Plus 7        | electronics |  6006 |      2
      2 | Iphone X          | electronics |  7007 |      6
      1 | Xiomi             | electronics |  1001 |      6
(5 rows)

asdlab=#
```

Figure 1: items Table

```
create table customers(
custid int not null,
custname varchar(20),
address varchar(50) not null,
state varchar(10) not null,
primary key(custid));
```

```
insert into customers values(111,'elvin','202 jai street','delhi');
insert into customers values(113,'soman','puthumana','kerala');
insert into customers values(115,'mickey','juhu','maharashtra');
insert into customers values(112,'patrick','harinagar','tamilnadu');
insert into customers values(114,'jaise','kottarakara','kerala');
```

0. Display the details customers table

```
select * from customers;
```

```
asdlab=# select * from customers;
 custid | custname | address      | state
-----+-----+-----+-----
    111 | elvin    | 202 jai street | delhi
    113 | soman    | puthumana     | kerala
    115 | mickey   | juhu          | maharashtra
    112 | patrick  | harinagar     | tamilnadu
    114 | jaise    | kottarakara   | kerala
(5 rows)

asdlab=#
```

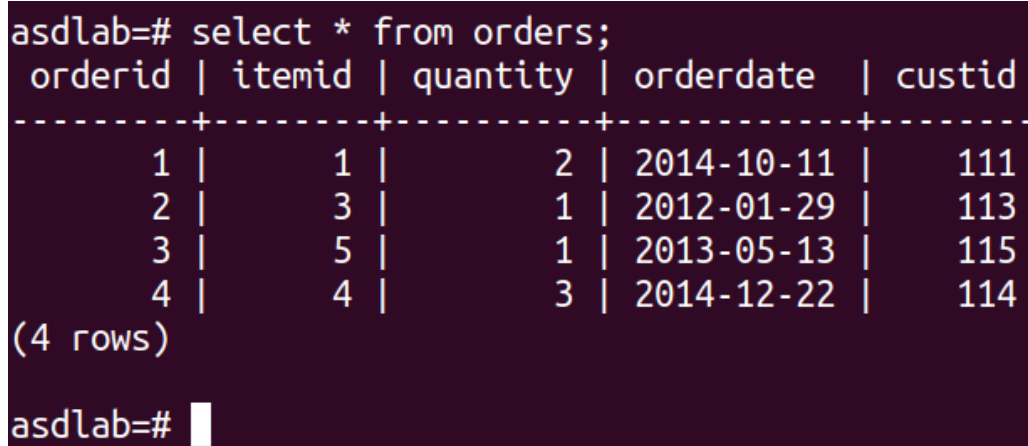
Figure 2: customers Table

```
create table orders(
orderid int not null,
itemid int ,
quantity int not null,
orderdate date ,custid int
primary key(orderid),
foreign key(itemid) references items(itemid) on update cascade on delete cascade ,
foreign key(custid) references customers(custid) on update cascade on delete cascade));
```

```
insert into orders values(1,1,2,'2014-10-11',111);
insert into orders values(2,3,1,'2012-01-29',113);
insert into orders values(3,5,1,'2013-05-13',115);
insert into orders values(4,4,3,'2014-12-22',114);
```

0. Display the details orders table

```
select * from orders;
```



```
asdlab=# select * from orders;
orderid | itemid | quantity | orderdate | custid
-----+-----+-----+-----+-----
      1 |      1 |        2 | 2014-10-11 |    111
      2 |      3 |        1 | 2012-01-29 |    113
      3 |      5 |        1 | 2013-05-13 |    115
      4 |      4 |        3 | 2014-12-22 |    114
(4 rows)

asdlab=#
```

Figure 3: orders Table

```
create table delivery(
deliveryid int not null,
orderid int ,custid int ,
primary key(deliveryid),
foreign key(orderid) references orders(orderid) on update cascade on delete cascade,
foreign key(custid) references customers(custid) on update cascade on delete cascade);
```

```
insert into delivery values(1001,1,111);
insert into delivery values(1002,2,113);
insert into delivery values(1003,3,115);
```

0. Display the details Delivery table

```
select * from delivery;
```

```
asdlab=# select * from delivery;
 deliveryid | orderid | custid
-----+-----+-----
          1001 |          1 |          111
          1002 |          2 |          113
          1003 |          3 |          115
(3 rows)

asdlab=#
```

Figure 4: delivery Table

1. List the details of all customers who have placed an order

```
select customers.custid,custname,address,state from customers , orders
where orders.custid=customers.custid;
```

```
asdlab=# select customers.custid,custname,address,state from customers , orders where orders.custid=customers.custid;
 custid | custname | address | state
-----+-----+-----+-----
      111 | elvin | 202 jai street | delhi
      113 | soman | puthumana | kerala
      115 | mickey | juhu | maharastra
      114 | jaise | kottarakara | kerala
(4 rows)

asdlab=#
```

Figure 5: Customers who placed an order

2. List the details of all customers whose orders have been delivered

```
select customers.custid,custname,address,state from customers , delivery
where delivery.custid=customers.custid;
```

```
asdlab=# select customers.custid,custname,address,state from customers , delivery where delivery.custid=customers.custid;
 custid | custname | address | state
-----+-----+-----+-----
      111 | elvin | 202 jai street | delhi
      113 | soman | puthumana | kerala
      115 | mickey | juhu | maharastra
(3 rows)

asdlab=#
```

Figure 6: Customers whose orders are delivered

3. Find the orderdate for all customers whose name starts in the letter 'J'

```
select orderdate from customers , orders
where orders.custid=customers.custid and custname like 'j%';
```

```
asdlab=# select orderdate from customers , orders where orders.custid=customers.custid and custname like 'j%';
orderdate
-----
2014-12-22
(1 row)

asdlab=#
```

Figure 7: orderdate with customers name starts in J

4.Display the name and price of all items bought by the customer 'Mickey'

```
select itemname,price from items as i ,customers as c,orders as o
where i.itemid=o.itemid and c.custid=o.custid and c.custname like 'mickey';
```

```
asdlab=# select itemname,price from items as i ,customers as c,orders as o
where i.itemid=o.itemid and c.custid=o.custid and c.custname like 'mickey';
 itemname      | price
-----+-----
sony z5 premium |  5005
(1 row)

asdlab=#
```

Figure 8: Mickey's Order

5. List the details of all customers who have placed an order after January 2013 and not received delivery of items.

```
select c.* from customers as c ,orders as o
where o.custid=c.custid and orderdate>='2013-01-01' and c.custid not in
(select custid from delivery );
```

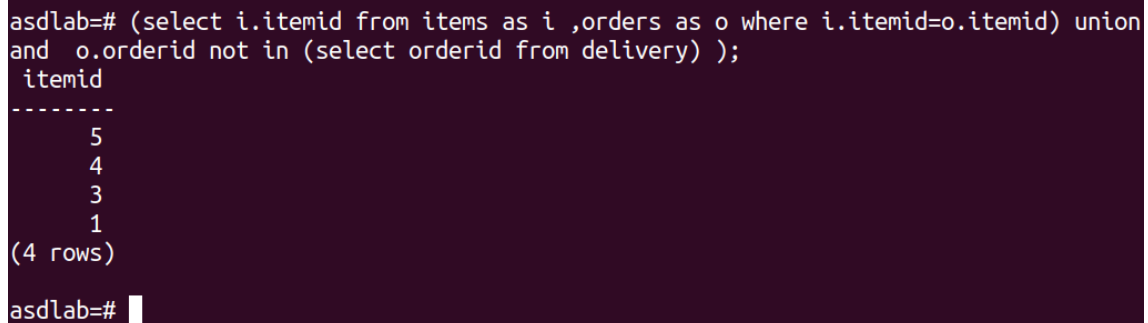
```
asdlab=# select c.* from customers as c ,orders as o where o.custid=c.custid
and orderdate>='2013-01-01' and c.custid not in (select custid from delivery );
custid | custname | address | state
-----+-----+-----+-----
114    | jaise    | kottarakara | kerala
(1 row)

asdlab=#
```

Figure 9: Undelivered Order after 2013 january

6.Find the itemid of items which has either been ordered or not delivered. (Use SET UNION)

```
(select i.itemid from items as i ,orders as o where i.itemid=o.itemid) union
union
(select i.itemid from items as i , orders as o where i.itemid=o.itemid
and o.orderid not in (select orderid from delivery) );
```

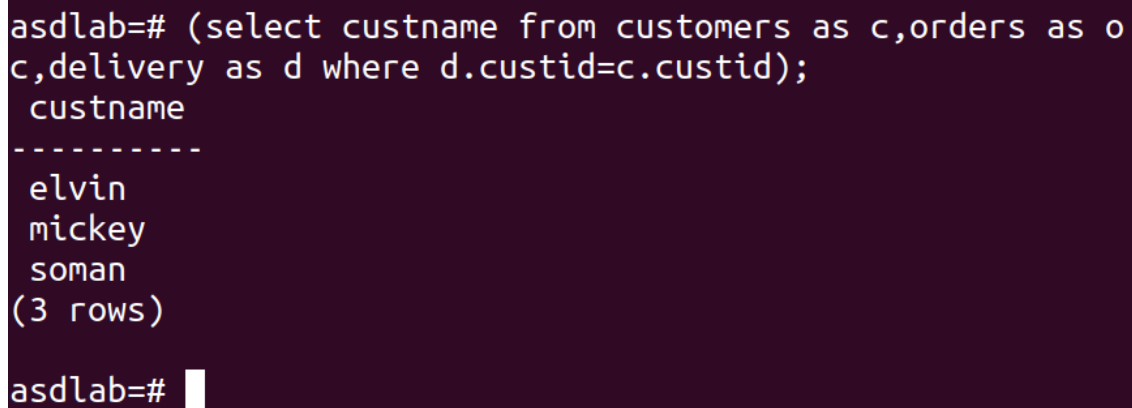


```
asdlab=# (select i.itemid from items as i ,orders as o where i.itemid=o.itemid) union
and o.orderid not in (select orderid from delivery) );
 itemid
-----
      5
      4
      3
      1
(4 rows)
asdlab=#
```

Figure 10: Either ordered or not delivered

7.Find the name of all customers who have placed an order and have their orders delivered.(Use SET INTERSECTION)

```
(select custname from customers as c,orders as o where o.custid=c.custid)
intersect
(select custname from customers as c,delivery as d where d.custid=c.custid);
```



```
asdlab=# (select custname from customers as c,orders as o
c,delivery as d where d.custid=c.custid);
 custname
-----
    elvin
    mickey
    soman
(3 rows)
asdlab=#
```

Figure 11: Ordered and delivered

8. Find the custname of all customers who have placed an order but not having their orders delivered. (Use SET MINUS).

```
(select custname from customers as c, orders as o where o.custid=c.custid)
except
(select custname from customers as c, delivery as d where d.custid=c.custid);
```

```
asdlab=# (select custname from customers as c, orders as o where o.custid=c.custid) except
delivery as d where d.custid=c.custid);
 custname
-----
 jaise
(1 row)

asdlab=#
```

Figure 12: Ordered but not delivered

9. Find the name of the customer who has placed the most number of orders.

```
insert into orders values(5,2,1,'2012-05-25',115);
```

```
select * from customers where custid=(select custid from orders
group by custid order by count(*) desc LIMIT 1);
```

```
asdlab=# select * from customers where custid=(select custid from orders group by custid order by count(*) desc LIMIT 1);
 custid | custname | address | state
-----+-----+-----+-----
    115 | mickey   | juhu    | maharashtra
(1 row)

asdlab=#
```

Figure 13: Customer who placed most number of orders

10. Find the details of all customers who have purchased items exceeding a price of 5000 \$.

```
select c.* from customers as c, items as i, orders as o where o.itemid=i.itemid
and c.custid=o.custid and price>5000;
```

```
asdlab=# select c.* from customers as c, items as i, orders as o where
custid | custname | address | state
-----+-----+-----+-----
    113 | soman    | puthumana | kerala
    115 | mickey   | juhu      | maharashtra
    114 | jaise    | kottarakara | kerala
(3 rows)

asdlab=#
```

Figure 14: More than 5000

11. Find the name and address of customers who has not ordered a 'Samsung Galaxy S4'

```
(select custname,address from customers)
except
(select c.custname,c.address from customers as c ,orders as o, items as i
where o.itemid=i.itemid and c.custid=o.custid
and itemname='Samsung Galaxy S4' );
```

```
asdlab=# (select custname,address from customers)except(select c.custname,c.address fr
i where o.itemid=i.itemid and c.custid=o.custid and itemname='Samsung Galaxy S4' );
custname |      address
-----+-----
elvin    | 202 jai street
mickey   | juhu
soman    | puthumana
patrick  | harinagar
(4 rows)
```

Figure 15: Customers not ordered galaxy s4

12. Perform Left Outer Join and Right Outer Join on Customers & Orders Table.

```
select * from customers left outer join orders on customers.custid=orders.custid;
select * from customers right outer join orders on customers.custid=orders.custid;
```

```
asdlab=# select * from customers left outer join orders on customers.custid=orders.custid;
custid | custname |      address      | state | orderid | itemid | quantity | orderdate | custid
-----+-----+-----+-----+-----+-----+-----+-----+-----
111 | elvin    | 202 jai street | delhi | 1 | 1 | 2 | 2014-10-11 | 111
113 | soman    | puthumana      | kerala | 2 | 3 | 1 | 2012-01-29 | 113
115 | mickey   | juhu           | maharastra | 3 | 5 | 1 | 2013-05-13 | 115
114 | jaise    | kottarakara     | kerala | 4 | 4 | 3 | 2014-12-22 | 114
112 | patrick  | harinagar       | tamilnadu |  |  |  |  |  |
(5 rows)
```

Figure 16: Left outer join

```
asdlab=# select * from customers right outer join orders on customers.custid=orders.custid;
custid | custname |      address      | state | orderid | itemid | quantity | orderdate | custid
-----+-----+-----+-----+-----+-----+-----+-----+-----
111 | elvin    | 202 jai street | delhi | 1 | 1 | 2 | 2014-10-11 | 111
113 | soman    | puthumana      | kerala | 2 | 3 | 1 | 2012-01-29 | 113
115 | mickey   | juhu           | maharastra | 3 | 5 | 1 | 2013-05-13 | 115
114 | jaise    | kottarakara     | kerala | 4 | 4 | 3 | 2014-12-22 | 114
(4 rows)
```

Figure 17: Right outer join

13. Find the details of all customers grouped by state.

```
select count(*),state from customers group by state;
```

```
asdlab=# select count(*),state from customers group by state;
count | state
-----+-----
      1 | maharashtra
      1 | delhi
      2 | kerala
      1 | tamilnadu
(4 rows)

asdlab=#
```

Figure 18: Grouped by state

14.Display the details of all items grouped by category and having a price greater than the average price of all items.

```
select * from items where price>(select avg(price) from items) order by category;
```

```
asdlab=# select * from items where price>(select avg(price) from items) order by category;
itemid | itemname | category | price | instock
-----+-----+-----+-----+-----
      5 | sony z5 premium | electronics | 5005 | 1
      4 | Samsung Galaxy S4 | electronics | 5005 | 1
      3 | One Plus 7 | electronics | 6006 | 2
      2 | Iphone X | electronics | 7007 | 6
(4 rows)

asdlab=#
```

Figure 19: price > Average price

3 Result

The query was executed successfully and output was verified.