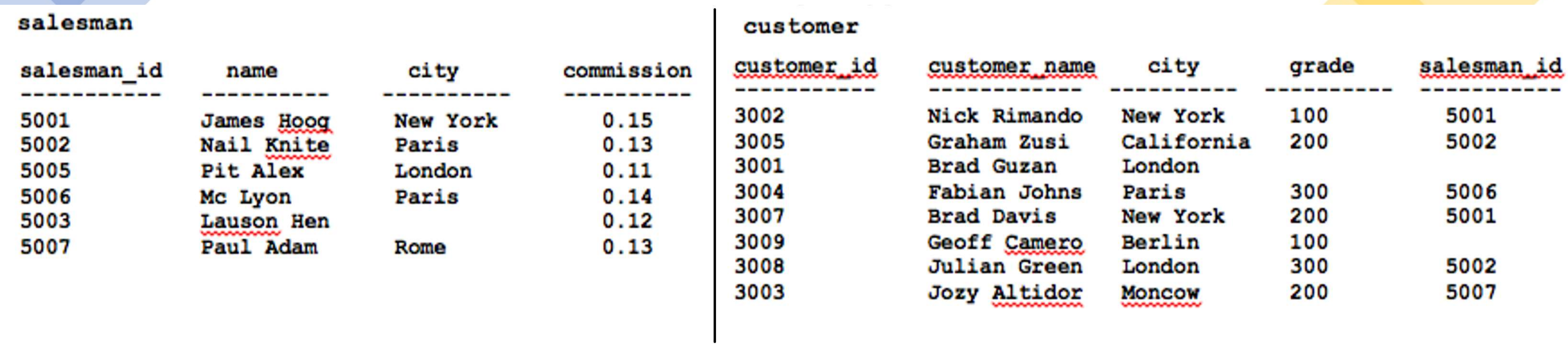
SQL Practice 2

Multiple tables Joins Nested Queries

Link: https://www.w3resource.com/sql-exercises/

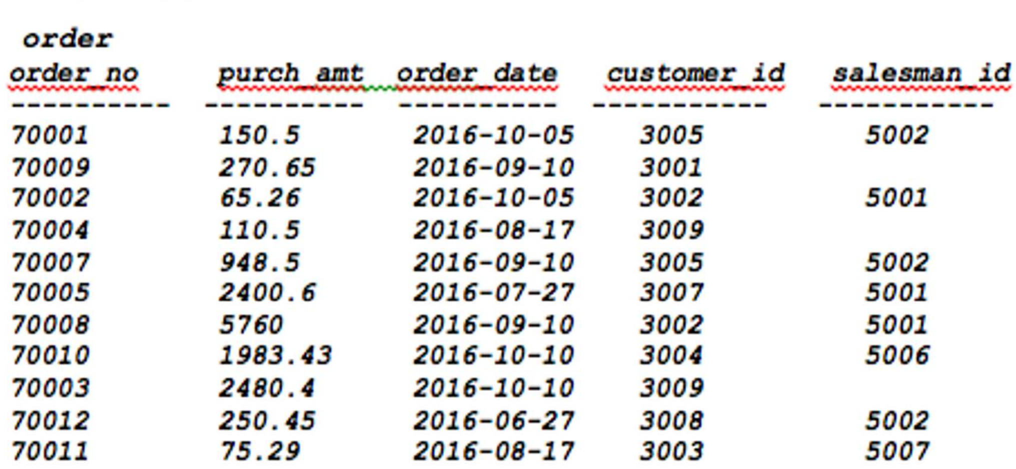


salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5003	Lauson Hen		0.12
5007	Paul Adam	Rome	0.13

customer

customer_id	customer_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3005	Graham Zusi	California	200	5002
3001	Brad Guzan	London		
3004	Fabian Johns	Paris	300	5006
3007	Brad Davis	New York	200	5001
3009	Geoff Camero	Berlin	100	
3008	Julian Green	London	300	5002
3003	Jozy Altidor	Moncow	200	5007



order

order_no	*purch_amt*	*order_date*	*customer_id*	*salesman_id*
70001	*150.5*	*2016-10-05*	*3005*	*5002*
70009	*270.65*	*2016-09-10*	*3001*	
70002	*65.26*	*2016-10-05*	*3002*	*5001*
70004	*110.5*	*2016-08-17*	*3009*	
70007	*948.5*	*2016-09-10*	*3005*	*5002*
70005	*2400.6*	*2016-07-27*	*3007*	*5001*
70008	*5760*	*2016-09-10*	*3002*	*5001*
70010	*1983.43*	*2016-10-10*	*3004*	*5006*
70003	*2480.4*	*2016-10-10*	*3009*	
70012	*250.45*	*2016-06-27*	*3008*	*5002*
70011	*75.29*	*2016-08-17*	*3003*	*5007*

Query 1

Find the name and city of those customers and salesmen who lives in the same city.

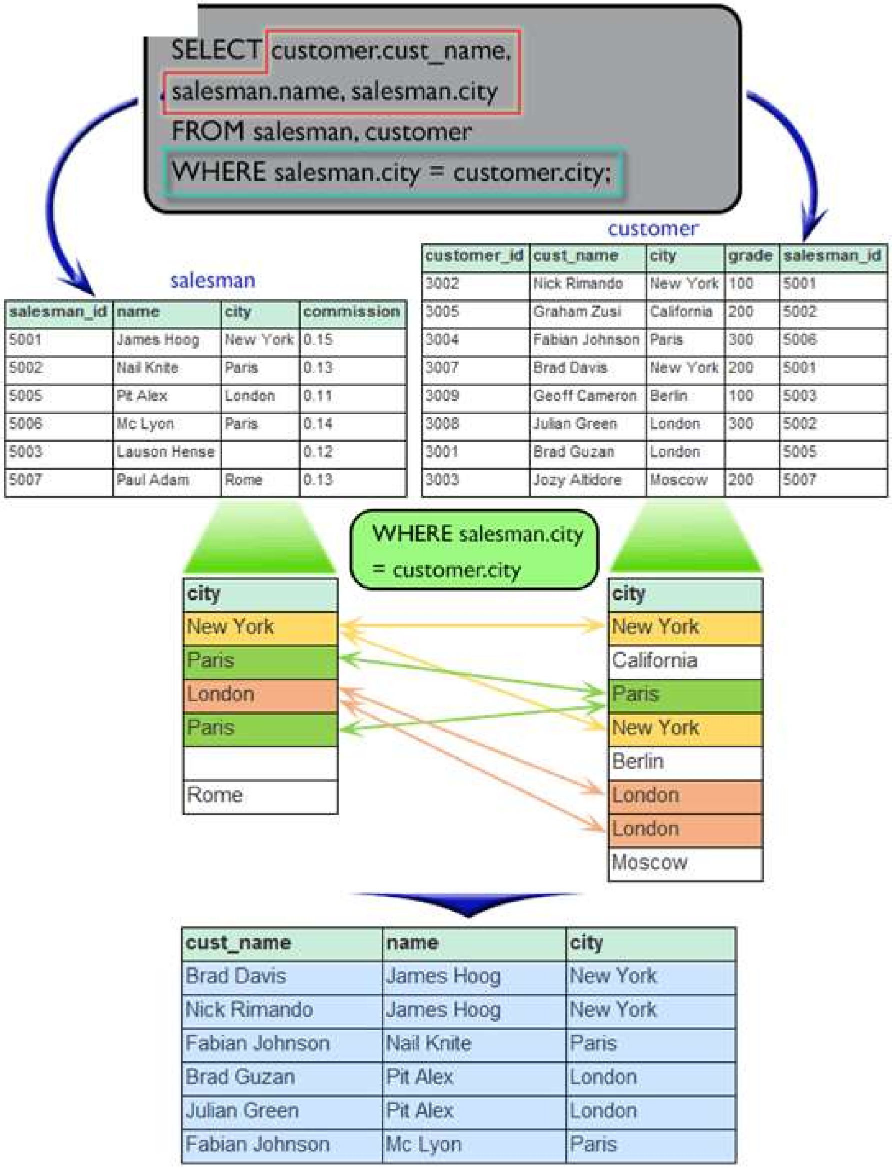
salesman (salesman_id, name, city, commission)

customer(customer_id, cust_name, city, grade, salesman_id)

orders(ord_no, purch_amt, ord_date, customer_id, salesman_id)

cust_name	name	city
Nick Rimando	James Hoog	New York
Brad Davis	James Hoog	New York
Julian Green	Pit Alex	London
Fabian Johnson	Mc Lyon	Paris
Fabian Johnson	Nail Knite	Paris
Brad Guzan	Pit Alex	London

```
SELECT C.cust_name, S.name, S.city
FROM salesman AS S customer AS C
WHERE S.city = C.city
```



```
SELECT customer.cust_name,
salesman.name, salesman.city
FROM salesman, customer
WHERE salesman.city = customer.city;
```

salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5003	Lauson Hense		0.12
5007	Paul Adam	Rome	0.13

customer

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3005	Graham Zusi	California	200	5002
3004	Fabian Johnson	Paris	300	5006
3007	Brad Davis	New York	200	5001
3009	Geoff Cameron	Berlin	100	5003
3008	Julian Green	London	300	5002
3001	Brad Guzan	London		5005
3003	Jozy Altidore	Moscow	200	5007

WHERE salesman.city = customer.city

city
New York
Paris
London
Paris
Rome

city
New York
California
Paris
New York
Berlin
London
London
Moscow

cust_name	name	city
Brad Davis	James Hoog	New York
Nick Rimando	James Hoog	New York
Fabian Johnson	Nail Knite	Paris
Brad Guzan	Pit Alex	London
Julian Green	Pit Alex	London
Fabian Johnson	Mc Lyon	Paris

Query 2

Find the names of all customers along with the salesmen who works for them.

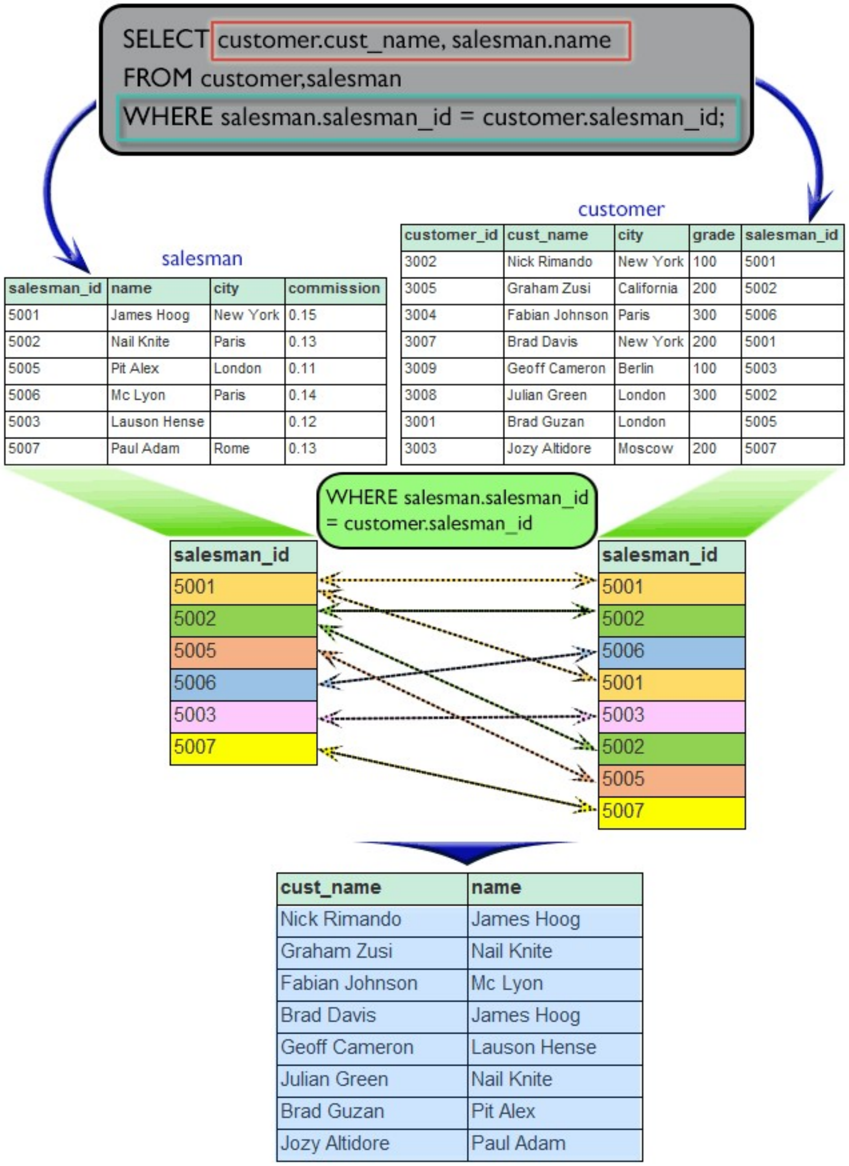
salesman (salesman_id, name, city, commission)

customer(customer_id, cust_name, city, grade, salesman_id)

orders(ord_no, purch_amt, ord_date, customer_id, salesman_id)

cust_name	name
Nick Rimando	James Hoog
Brad Davis	James Hoog
Graham Zusi	Nail Knite
Julian Green	Nail Knite
Fabian Johnson	Mc Lyon
Geoff Cameron	Lauson Hen
Jozy Altidor	Paul Adam
Brad Guzan	Pit Alex

```
SELECT customer.cust_name salesman.name
FROM customer, salesman
WHERE salesman.salesman_id = customer.salesman_id;
```



```
SELECT customer.cust_name, salesman.name
FROM customer,salesman
WHERE salesman.salesman_id = customer.salesman_id;
```

salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5003	Lauson Hense		0.12
5007	Paul Adam	Rome	0.13

customer

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3005	Graham Zusi	California	200	5002
3004	Fabian Johnson	Paris	300	5006
3007	Brad Davis	New York	200	5001
3009	Geoff Cameron	Berlin	100	5003
3008	Julian Green	London	300	5002
3001	Brad Guzan	London		5005
3003	Jozy Altidore	Moscow	200	5007

WHERE salesman.salesman_id = customer.salesman_id

salesman_id
5001
5002
5005
5006
5003
5007

salesman_id
5001
5002
5006
5001
5003
5002
5005
5007

cust_name	name
Nick Rimando	James Hoog
Graham Zusi	Nail Knite
Fabian Johnson	Mc Lyon
Brad Davis	James Hoog
Geoff Cameron	Lauson Hense
Julian Green	Nail Knite
Brad Guzan	Pit Alex
Jozy Altidore	Paul Adam

Query 3

Display all those orders by the customers not located in the same cities where their salesmen live.

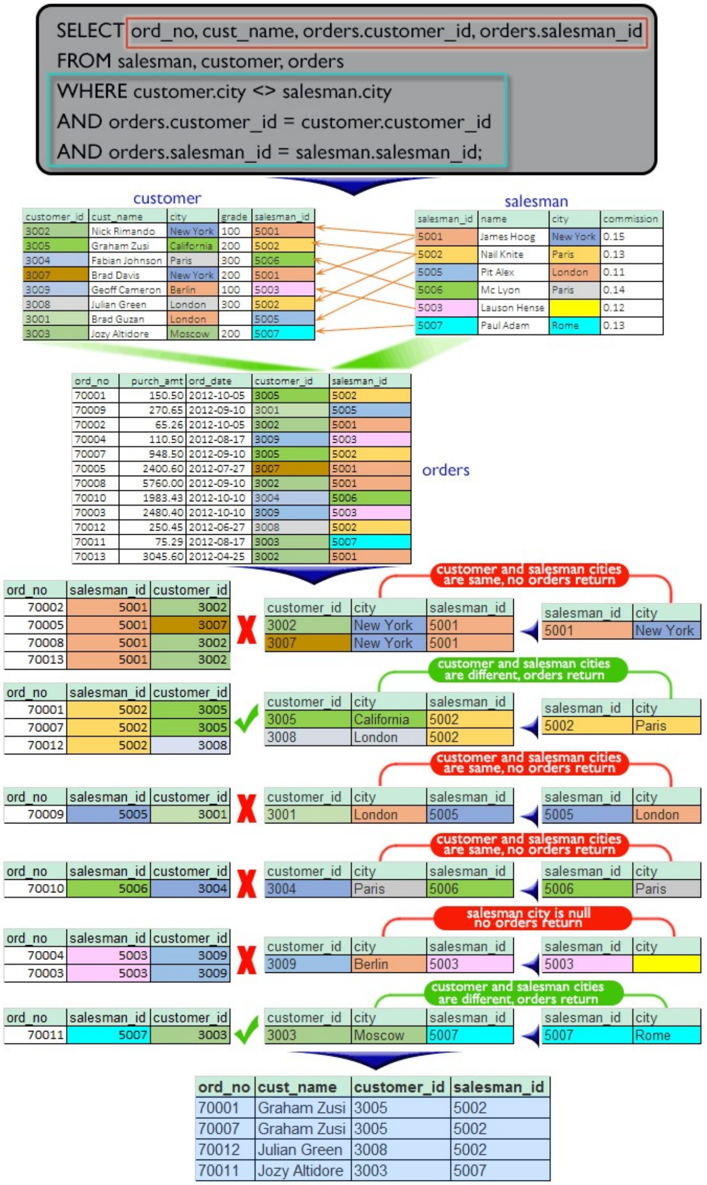
salesman (salesman_id, name, city, commission)

customer(customer_id, cust_name, city, grade, salesman_id)

orders(ord_no, purch_amt, ord_date, customer_id, salesman_id)

ord_no	cust_name	customer_id	salesman_id
70004	Geoff Cameron	3009	5003
70003	Geoff Cameron	3009	5003
70011	Jozy Altidor	3003	5007
70001	Graham Zusi	3005	5002
70007	Graham Zusi	3005	5002
70012	Julian Green	3008	5002

SELECT ord_no cust_name orders.customer_id orders.salesman_id
FROM salesman, customer, orders
WHERE customer.city **<>** salesman.city
AND orders.customer_id = customer.customer_id
AND orders.salesman_id = salesman.salesman_id;



```
SELECT ord_no, cust_name, orders.customer_id, orders.salesman_id
FROM salesman, customer, orders
WHERE customer.city <> salesman.city
AND orders.customer_id = customer.customer_id
AND orders.salesman_id = salesman.salesman_id;
```

customer

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3005	Graham Zusi	California	200	5002
3004	Fabian Johnson	Paris	300	5006
3007	Brad Davis	New York	200	5001
3009	Geoff Cameron	Berlin	100	5003
3008	Julian Green	London	300	5002
3001	Brad Guzan	London		5005
3003	Jozy Altidore	Moscow	200	5007

salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5003	Lauson Hense		0.12
5007	Paul Adam	Rome	0.13

orders

ord_no	purch_amt	ord_date	customer_id	salesman_id
70001	150.50	2012-10-05	3005	5002
70009	270.65	2012-09-10	3001	5005
70002	65.26	2012-10-05	3002	5001
70004	110.50	2012-08-17	3009	5003
70007	948.50	2012-09-10	3005	5002
70005	2400.60	2012-07-27	3007	5001
70008	5760.00	2012-09-10	3002	5001
70010	1983.43	2012-10-10	3004	5006
70003	2480.40	2012-10-10	3009	5003
70012	250.45	2012-06-27	3008	5002
70011	75.29	2012-08-17	3003	5007
70013	3045.60	2012-04-25	3002	5001

customer and salesman cities are same, no orders return

ord_no	salesman_id	customer_id
70002	5001	3002
70005	5001	3007
70008	5001	3002
70013	5001	3002

✗

customer_id	city	salesman_id
3002	New York	5001
3007	New York	5001

salesman_id	city
5001	New York

customer and salesman cities are different, orders return

ord_no	salesman_id	customer_id
70001	5002	3005
70007	5002	3005
70012	5002	3008

✓

customer_id	city	salesman_id
3005	California	5002
3008	London	5002

salesman_id	city
5002	Paris

customer and salesman cities are same, no orders return

ord_no	salesman_id	customer_id
70009	5005	3001

✗

customer_id	city	salesman_id
3001	London	5005

salesman_id	city
5005	London

customer and salesman cities are same, no orders return

ord_no	salesman_id	customer_id
70010	5006	3004

✗

customer_id	city	salesman_id
3004	Paris	5006

salesman_id	city
5006	Paris

salesman city is null no orders return

ord_no	salesman_id	customer_id
70004	5003	3009
70003	5003	3009

✗

customer_id	city	salesman_id
3009	Berlin	5003

salesman_id	city
5003	

customer and salesman cities are different, orders return

ord_no	salesman_id	customer_id
70011	5007	3003

✓

customer_id	city	salesman_id
3003	Moscow	5007

salesman_id	city
5007	Rome

ord_no	cust_name	customer_id	salesman_id
70001	Graham Zusi	3005	5002
70007	Graham Zusi	3005	5002
70012	Julian Green	3008	5002
70011	Jozy Altidore	3003	5007

Query 4 (using subquery)

Display all the orders issued by the salesman 'Paul Adam' from the orders table.

salesman (salesman_id, name, city, commission)

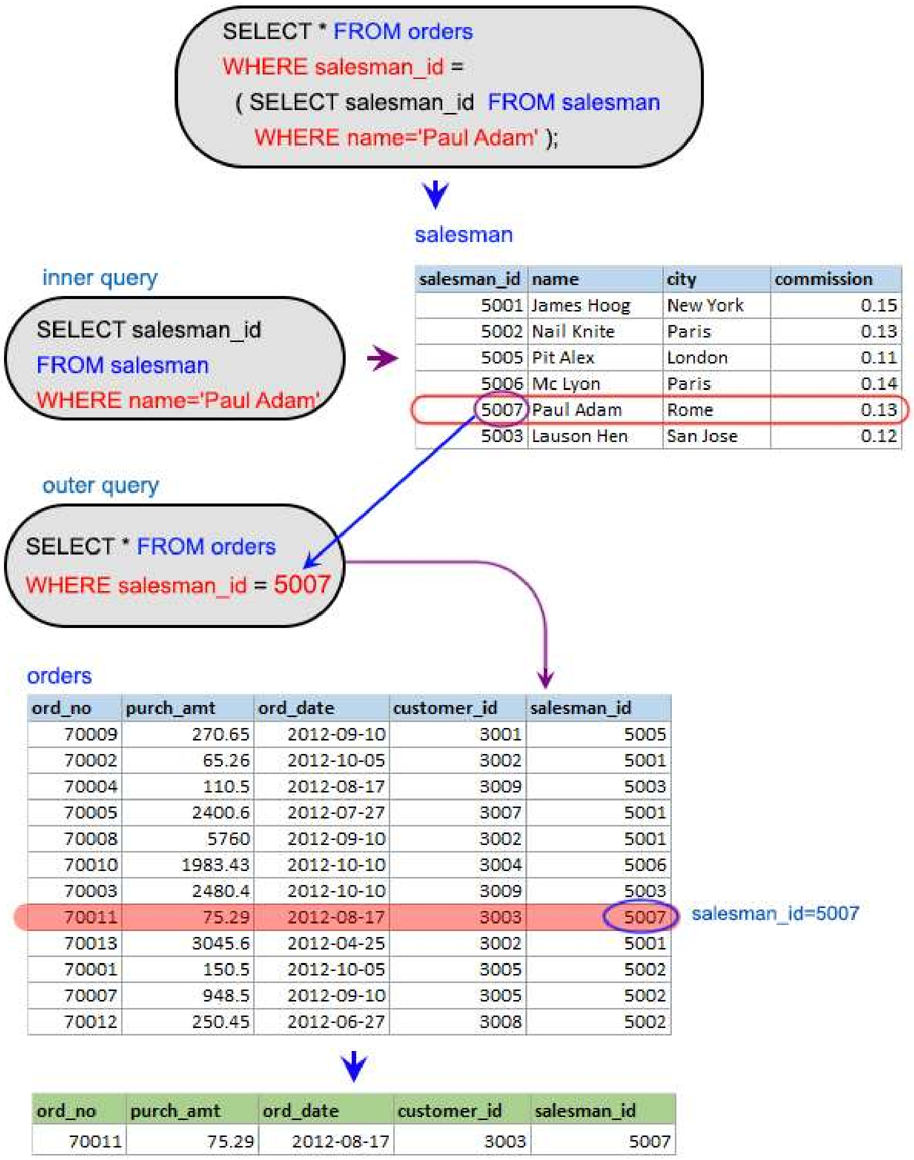
customer(customer_id, cust_name, city, grade, salesman_id)

orders(ord_no, purch_amt, ord_date, customer_id, salesman_id)

ord_no	purch_amt	ord_date	customer_id	salesman_id
70011	75.29	2012-08-17	3003	5007

```
SELECT *
FROM orders
WHERE salesman_id =
   (SELECT salesman_id
    FROM salesman
    WHERE name = 'Paul Adam');
```

- Can we make this query unnested? If yes how?



```
SELECT * FROM orders
WHERE salesman_id =
  ( SELECT salesman_id  FROM salesman
    WHERE name='Paul Adam' );
```

salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5007	Paul Adam	Rome	0.13
5003	Lauson Hen	San Jose	0.12

inner query

```
SELECT salesman_id
FROM salesman
WHERE name='Paul Adam'
```

outer query

```
SELECT * FROM orders
WHERE salesman_id = 5007
```

orders

ord_no	purch_amt	ord_date	customer_id	salesman_id
70009	270.65	2012-09-10	3001	5005
70002	65.26	2012-10-05	3002	5001
70004	110.5	2012-08-17	3009	5003
70005	2400.6	2012-07-27	3007	5001
70008	5760	2012-09-10	3002	5001
70010	1983.43	2012-10-10	3004	5006
70003	2480.4	2012-10-10	3009	5003
70011	75.29	2012-08-17	3003	5007
70013	3045.6	2012-04-25	3002	5001
70001	150.5	2012-10-05	3005	5002
70007	948.5	2012-09-10	3005	5002
70012	250.45	2012-06-27	3008	5002

ord_no	purch_amt	ord_date	customer_id	salesman_id
70011	75.29	2012-08-17	3003	5007

Query 5 (using subquery)

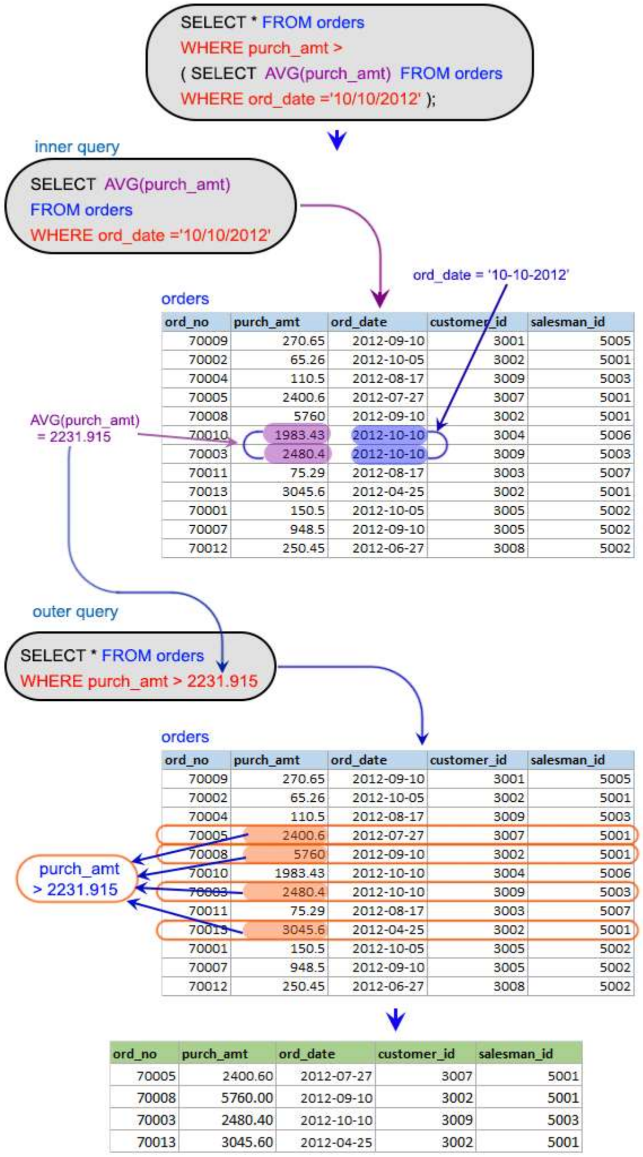
Display all the orders which values are greater than the average order value for 10th October 2012.

salesman (salesman_id, name, city, commission)
customer(customer_id, cust_name, city, grade, salesman_id)
orders(ord_no, purch_amt, ord_date, customer_id, salesman_id)

ord_no	purch_amt	ord_date	customer_id	salesman_id
70005	2400.60	2012-07-27	3007	5001
70008	5760.00	2012-09-10	3002	5001
70003	2480.40	2012-10-10	3009	5003
70013	3045.60	2012-04-25	3002	5001

```
SELECT *
FROM orders
WHERE purch_amt >
   (SELECT  AVG(purch_amt)
    FROM orders
    WHERE ord_date = '2012-10-10');
```

- Can we make this query unnested? If yes how?



```
SELECT * FROM orders
WHERE purch_amt >
( SELECT  AVG(purch_amt)  FROM orders
WHERE ord_date ='10/10/2012' );
```

```
SELECT  AVG(purch_amt)
FROM orders
WHERE ord_date ='10/10/2012'
```

ord_no	purch_amt	ord_date	customer_id	salesman_id
70009	270.65	2012-09-10	3001	5005
70002	65.26	2012-10-05	3002	5001
70004	110.5	2012-08-17	3009	5003
70005	2400.6	2012-07-27	3007	5001
70008	5760	2012-09-10	3002	5001
70010	1983.43	2012-10-10	3004	5006
70003	2480.4	2012-10-10	3009	5003
70011	75.29	2012-08-17	3003	5007
70013	3045.6	2012-04-25	3002	5001
70001	150.5	2012-10-05	3005	5002
70007	948.5	2012-09-10	3005	5002
70012	250.45	2012-06-27	3008	5002

```
SELECT * FROM orders
WHERE purch_amt > 2231.915
```

ord_no	purch_amt	ord_date	customer_id	salesman_id
70009	270.65	2012-09-10	3001	5005
70002	65.26	2012-10-05	3002	5001
70004	110.5	2012-08-17	3009	5003
70005	2400.6	2012-07-27	3007	5001
70008	5760	2012-09-10	3002	5001
70010	1983.43	2012-10-10	3004	5006
70003	2480.4	2012-10-10	3009	5003
70011	75.29	2012-08-17	3003	5007
70013	3045.6	2012-04-25	3002	5001
70001	150.5	2012-10-05	3005	5002
70007	948.5	2012-09-10	3005	5002
70012	250.45	2012-06-27	3008	5002

ord_no	purch_amt	ord_date	customer_id	salesman_id
70005	2400.60	2012-07-27	3007	5001
70008	5760.00	2012-09-10	3002	5001
70003	2480.40	2012-10-10	3009	5003
70013	3045.60	2012-04-25	3002	5001

Query 6 (using subquery)

Find all orders attributed to salesmen in Paris.

salesman (salesman_id, name, city, commission)

customer(customer_id, cust_name, city, grade, salesman_id)

orders(ord_no, purch_amt, ord_date, customer_id, salesman_id)

ord_no	purch_amt	ord_date	customer_id	salesman_id
70001	150.50	2012-10-05	3005	5002
70007	948.50	2012-09-10	3005	5002
70012	250.45	2012-06-27	3008	5002
70010	1983.43	2012-10-10	3004	5006

```
SELECT *
FROM orders
WHERE salesman_id IN
   (SELECT salesman_id
    FROM salesman
    WHERE city ='Paris');
```

- Can we make this query unnested? If yes how?

Query 7 (using subquery)

Extract the data from the orders table for the salesman who earned the maximum commission.

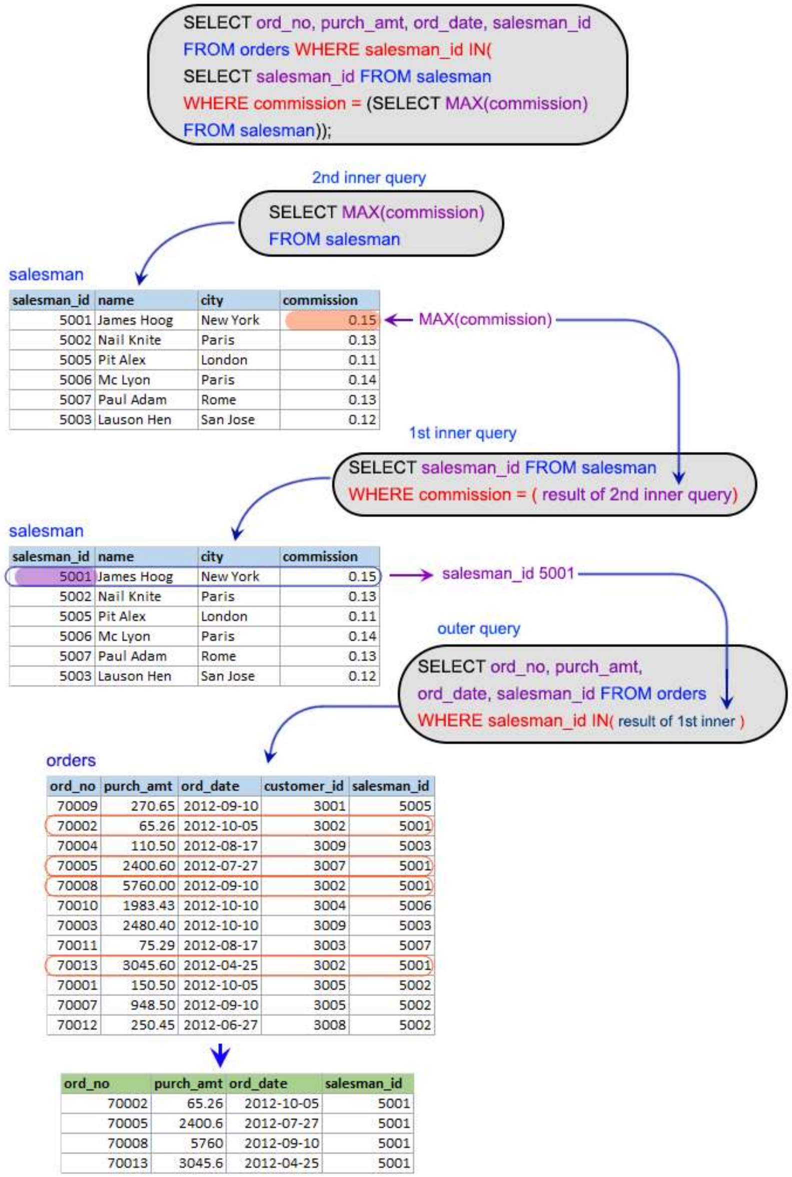
salesman (salesman_id, name, city, commission)

customer(customer_id, cust_name, city, grade, salesman_id)

orders(ord_no, purch_amt, ord_date, customer_id, salesman_id)

ord_no	purch_amt	ord_date	salesman_id
70002	65.26	2012-10-05	5001
70005	2400.60	2012-07-27	5001
70008	5760.00	2012-09-10	5001
70013	3045.60	2012-04-25	5001

```
SELECT ord_no, purch_amt, ord_date, salesman_id
FROM orders
WHERE salesman_id IN (
        SELECT salesman_id
        FROM salesman
        WHERE commission = (
                SELECT MAX(commission)
                FROM salesman)
        );
```



```
SELECT ord_no, purch_amt, ord_date, salesman_id
FROM orders WHERE salesman_id IN(
SELECT salesman_id FROM salesman
WHERE commission = (SELECT MAX(commission)
FROM salesman));
```

2nd inner query

```
SELECT MAX(commission)
FROM salesman
```

salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5007	Paul Adam	Rome	0.13
5003	Lauson Hen	San Jose	0.12

1st inner query

```
SELECT salesman_id FROM salesman
WHERE commission = ( result of 2nd inner query)
```

salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5007	Paul Adam	Rome	0.13
5003	Lauson Hen	San Jose	0.12

outer query

```
SELECT ord_no, purch_amt,
ord_date, salesman_id FROM orders
WHERE salesman_id IN( result of 1st inner )
```

orders

ord_no	purch_amt	ord_date	customer_id	salesman_id
70009	270.65	2012-09-10	3001	5005
70002	65.26	2012-10-05	3002	5001
70004	110.50	2012-08-17	3009	5003
70005	2400.60	2012-07-27	3007	5001
70008	5760.00	2012-09-10	3002	5001
70010	1983.43	2012-10-10	3004	5006
70003	2480.40	2012-10-10	3009	5003
70011	75.29	2012-08-17	3003	5007
70013	3045.60	2012-04-25	3002	5001
70001	150.50	2012-10-05	3005	5002
70007	948.50	2012-09-10	3005	5002
70012	250.45	2012-06-27	3008	5002

ord_no	purch_amt	ord_date	salesman_id
70002	65.26	2012-10-05	5001
70005	2400.6	2012-07-27	5001
70008	5760	2012-09-10	5001
70013	3045.6	2012-04-25	5001

Query 8 (using subquery)

Find the name and ids of all salesmen who had more than one customer.

salesman (salesman_id, name, city, commission)

customer(customer_id, cust_name, city, grade, salesman_id)

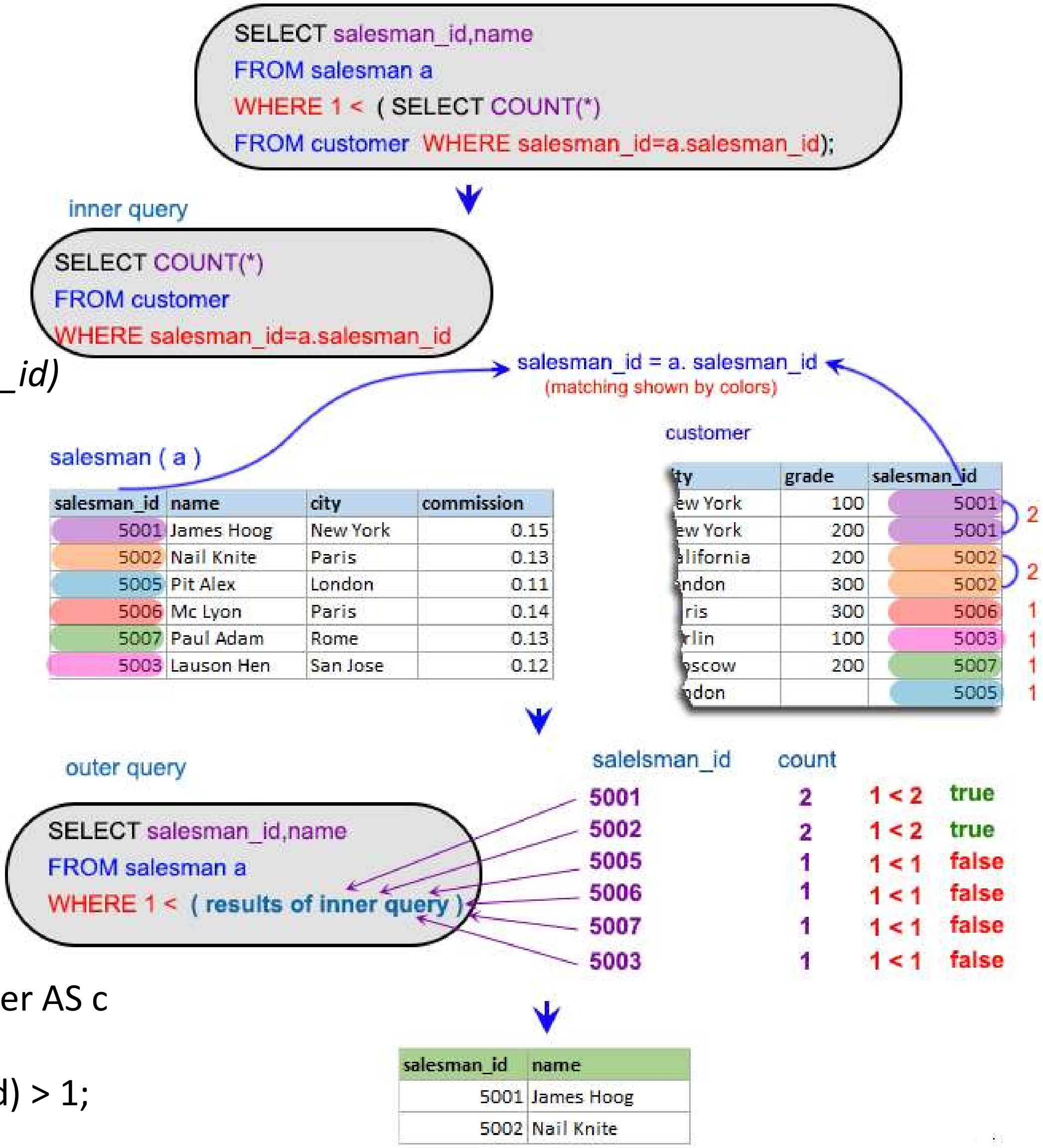
orders(ord_no, purch_amt, ord_date, customer_id, salesman_id)

salesman_id	name
5001	James Hoog
5002	Nail Knite

```
SELECT salesman_id, name
FROM salesman AS a
WHERE 1 <
   (SELECT COUNT(*)
    FROM customer AS c
    WHERE c.salesman_id = a.salesman_id);
```

- Can we make this query unnested? If yes how?

```
SELECT c.salesman_id, s.name  FROM salesman AS s, customer AS c
where s.salesman_id = c.salesman_id
group by c.salesman_id, s.name  Having count(c.salesman_id) > 1;
```



```
SELECT salesman_id,name
FROM salesman a
WHERE 1 < ( SELECT COUNT(*)
FROM customer  WHERE salesman_id=a.salesman_id);
```

inner query

```
SELECT COUNT(*)
FROM customer
WHERE salesman_id=a.salesman_id
```

salesman_id = a. salesman_id
(matching shown by colors)

salesman (a)

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5007	Paul Adam	Rome	0.13
5003	Lauson Hen	San Jose	0.12

customer

ty	grade	salesman_id	
ew York	100	5001	2
ew York	200	5001	
alifornia	200	5002	2
ndon	300	5002	
ris	300	5006	1
rlin	100	5003	1
oscow	200	5007	1
ndon		5005	1

outer query

```
SELECT salesman_id,name
FROM salesman a
WHERE 1 <  ( results of inner query )
```

salelsman_id	count		
5001	2	1 < 2	true
5002	2	1 < 2	true
5005	1	1 < 1	false
5006	1	1 < 1	false
5007	1	1 < 1	false
5003	1	1 < 1	false

salesman_id	name
5001	James Hoog
5002	Nail Knite

Query 9 (using subquery)

Write a query to find all the salesmen who worked for only one customer.

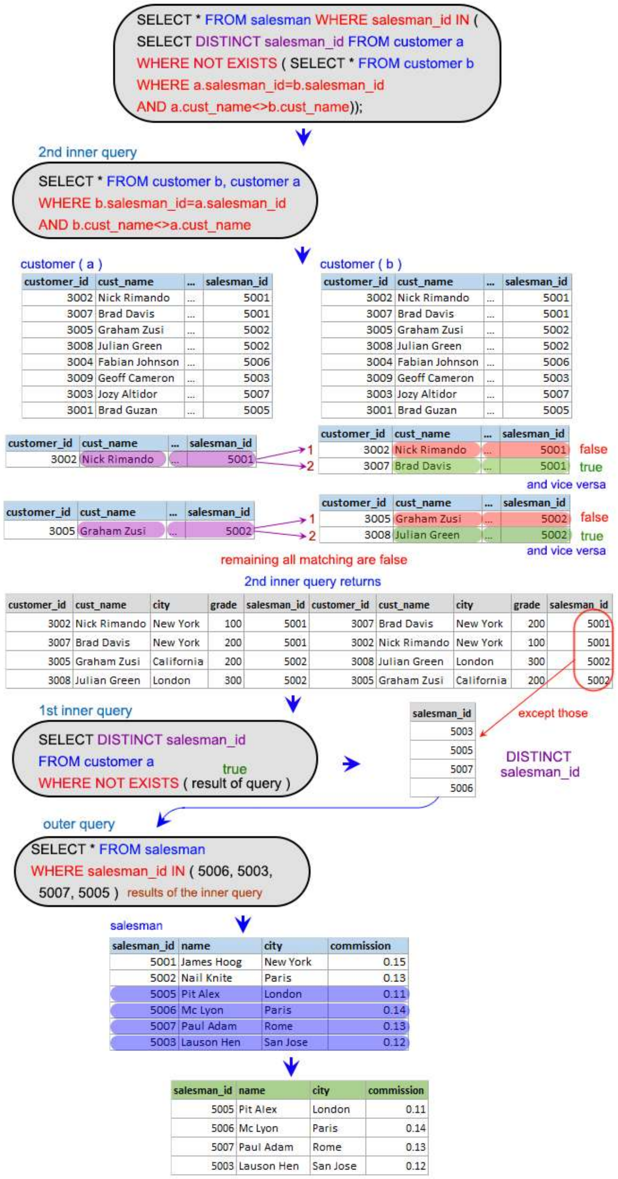
salesman (salesman_id, name, city, commission)

customer(customer_id, cust_name, city, grade, salesman_id)

orders(ord_no, purch_amt, ord_date, customer_id, salesman_id)

salesman_id	name	city	commission
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5007	Paul Adam	Rome	0.13
5003	Lauson Hen	San Jose	0.12

```
SELECT *
FROM salesman
WHERE salesman_id IN (
  SELECT DISTINCT salesman_id
  FROM customer a
  WHERE NOT EXISTS (
    SELECT * FROM customer b
    WHERE a.salesman_id = b.salesman_id
    AND a.cust_name <> b.cust_name));
```



```
SELECT * FROM salesman WHERE salesman_id IN (
SELECT DISTINCT salesman_id FROM customer a
WHERE NOT EXISTS ( SELECT * FROM customer b
WHERE a.salesman_id=b.salesman_id
AND a.cust_name<>b.cust_name));
```

2nd inner query

```
SELECT * FROM customer b, customer a
WHERE b.salesman_id=a.salesman_id
AND b.cust_name<>a.cust_name
```

customer (a)

customer_id	cust_name	...	salesman_id
3002	Nick Rimando	...	5001
3007	Brad Davis	...	5001
3005	Graham Zusi	...	5002
3008	Julian Green	...	5002
3004	Fabian Johnson	...	5006
3009	Geoff Cameron	...	5003
3003	Jozy Altidor	...	5007
3001	Brad Guzan	...	5005

customer (b)

customer_id	cust_name	...	salesman_id
3002	Nick Rimando	...	5001
3007	Brad Davis	...	5001
3005	Graham Zusi	...	5002
3008	Julian Green	...	5002
3004	Fabian Johnson	...	5006
3009	Geoff Cameron	...	5003
3003	Jozy Altidor	...	5007
3001	Brad Guzan	...	5005

customer_id	cust_name	...	salesman_id
3002	Nick Rimando	...	5001

customer_id	cust_name	...	salesman_id	
3002	Nick Rimando	...	5001	false
3007	Brad Davis	...	5001	true

and vice versa

customer_id	cust_name	...	salesman_id
3005	Graham Zusi	...	5002

customer_id	cust_name	...	salesman_id	
3005	Graham Zusi	...	5002	false
3008	Julian Green	...	5002	true

and vice versa

remaining all matching are false

2nd inner query returns

customer_id	cust_name	city	grade	salesman_id	customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001	3007	Brad Davis	New York	200	5001
3007	Brad Davis	New York	200	5001	3002	Nick Rimando	New York	100	5001
3005	Graham Zusi	California	200	5002	3008	Julian Green	London	300	5002
3008	Julian Green	London	300	5002	3005	Graham Zusi	California	200	5002

except those

1st inner query

```
SELECT DISTINCT salesman_id
FROM customer a
WHERE NOT EXISTS ( result of query )
```

true

salesman_id
5003
5005
5007
5006

DISTINCT salesman_id

outer query

```
SELECT * FROM salesman
WHERE salesman_id IN ( 5006, 5003,
5007, 5005 )
```

results of the inner query

salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5007	Paul Adam	Rome	0.13
5003	Lauson Hen	San Jose	0.12

salesman_id	name	city	commission
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5007	Paul Adam	Rome	0.13
5003	Lauson Hen	San Jose	0.12

Query 9: Equivalent Queries

Write a query to find all the salesmen who worked for only one customer.

salesman (salesman_id, name, city, commission)

customer(customer_id, cust_name, city, grade, salesman_id)

orders(ord_no, purch_amt, ord_date, customer_id, salesman_id)

salesman_id	name	city	commission
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5007	Paul Adam	Rome	0.13
5003	Lauson Hen	San Jose	0.12

```
SELECT c.salesman_id, s.name, s.city, s.commission
FROM salesman s, customer c
where s.salesman_id = c.salesman_id
group by c.salesman_id, s.name
Having count(c.salesman_id) = 1;
```

```
SELECT *
FROM salesman
WHERE salesman_id NOT IN (
                          SELECT a.salesman_id
                          FROM customer a, customer b
                          WHERE a.salesman_id = b.salesman_id
                          AND a.cust_name <> b.cust_name);
```

Query 10 (using subquery)

Display all the orders that had amounts that were greater than **at least one of the orders** from September 10th 2012.

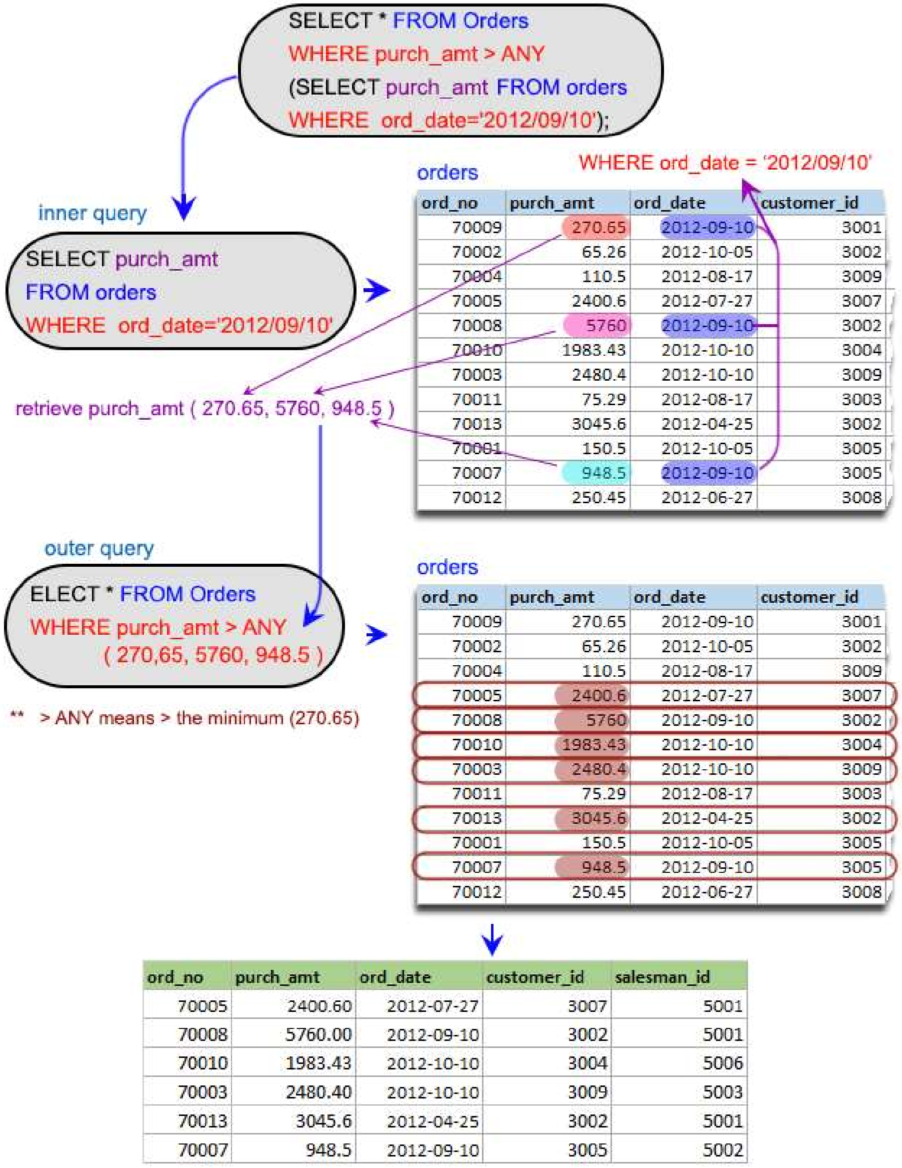
salesman (salesman_id, name, city, commission)

customer(customer_id, cust_name, city, grade, salesman_id)

orders(ord_no, purch_amt, ord_date, customer_id, salesman_id)

ord_no	purch_amt	ord_date	customer_id	salesman_id
70005	2400.60	2012-07-27	3007	5001
70008	5760.00	2012-09-10	3002	5001
70010	1983.43	2012-10-10	3004	5006
70003	2480.40	2012-10-10	3009	5003
70013	3045.60	2012-04-25	3002	5001
70007	948.50	2012-09-10	3005	5002

```
SELECT *
FROM Orders
WHERE purch_amt > ANY
                    (SELECT purch_amt
                     FROM orders
                     WHERE  ord_date = '2012-09-10');
```



```
SELECT * FROM Orders
WHERE purch_amt > ANY
(SELECT purch_amt FROM orders
WHERE  ord_date='2012/09/10');
```

```
SELECT purch_amt
FROM orders
WHERE  ord_date='2012/09/10'
```

ord_no	purch_amt	ord_date	customer_id
70009	270.65	2012-09-10	3001
70002	65.26	2012-10-05	3002
70004	110.5	2012-08-17	3009
70005	2400.6	2012-07-27	3007
70008	5760	2012-09-10	3002
70010	1983.43	2012-10-10	3004
70003	2480.4	2012-10-10	3009
70011	75.29	2012-08-17	3003
70013	3045.6	2012-04-25	3002
70001	150.5	2012-10-05	3005
70007	948.5	2012-09-10	3005
70012	250.45	2012-06-27	3008

retrieve purch_amt (270.65, 5760, 948.5)

```
ELECT * FROM Orders
WHERE purch_amt > ANY
    ( 270,65, 5760, 948.5 )
```

** > ANY means > the minimum (270.65)

ord_no	purch_amt	ord_date	customer_id
70009	270.65	2012-09-10	3001
70002	65.26	2012-10-05	3002
70004	110.5	2012-08-17	3009
70005	2400.6	2012-07-27	3007
70008	5760	2012-09-10	3002
70010	1983.43	2012-10-10	3004
70003	2480.4	2012-10-10	3009
70011	75.29	2012-08-17	3003
70013	3045.6	2012-04-25	3002
70001	150.5	2012-10-05	3005
70007	948.5	2012-09-10	3005
70012	250.45	2012-06-27	3008

ord_no	purch_amt	ord_date	customer_id	salesman_id
70005	2400.60	2012-07-27	3007	5001
70008	5760.00	2012-09-10	3002	5001
70010	1983.43	2012-10-10	3004	5006
70003	2480.40	2012-10-10	3009	5003
70013	3045.6	2012-04-25	3002	5001
70007	948.5	2012-09-10	3005	5002

Query 11 (using subquery)

display only those customers whose grade are, in fact, higher than every customer in New York.

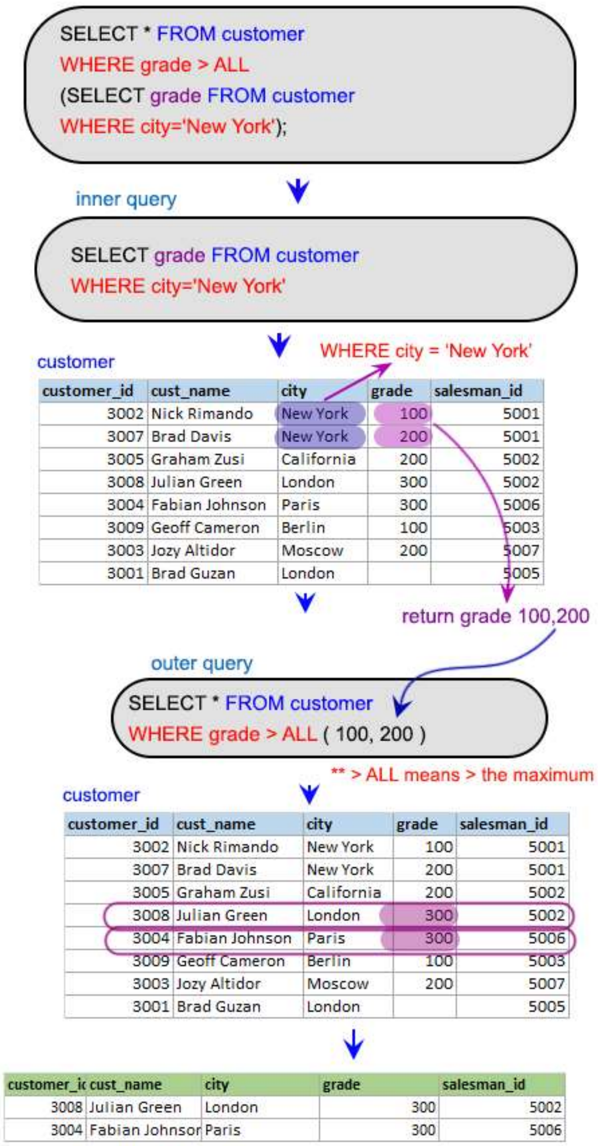
salesman (salesman_id, name, city, commission)

customer(customer_id, cust_name, city, grade, salesman_id)

orders(ord_no, purch_amt, ord_date, customer_id, salesman_id)

customer_id	cust_name	city	grade	salesman_id
3008	Julian Green	London	300	5002
3004	Fabian Johnson	Paris	300	5006

```
SELECT *
FROM customer
WHERE grade > ALL
                    (SELECT grade
                     FROM customer
                     WHERE city = 'NewYork');
```



```
SELECT * FROM customer
WHERE grade > ALL
(SELECT grade FROM customer
WHERE city='New York');
```

```
SELECT grade FROM customer
WHERE city='New York'
```

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3007	Brad Davis	New York	200	5001
3005	Graham Zusi	California	200	5002
3008	Julian Green	London	300	5002
3004	Fabian Johnson	Paris	300	5006
3009	Geoff Cameron	Berlin	100	5003
3003	Jozy Altidor	Moscow	200	5007
3001	Brad Guzan	London		5005

```
SELECT * FROM customer
WHERE grade > ALL ( 100, 200 )
```

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3007	Brad Davis	New York	200	5001
3005	Graham Zusi	California	200	5002
3008	Julian Green	London	300	5002
3004	Fabian Johnson	Paris	300	5006
3009	Geoff Cameron	Berlin	100	5003
3003	Jozy Altidor	Moscow	200	5007
3001	Brad Guzan	London		5005

customer_ic	cust_name	city	grade	salesman_id
3008	Julian Green	London	300	5002
3004	Fabian Johnsor	Paris	300	5006