

```
CREATE DATABASE w3q1_orders;

USE w3q1_orders;

CREATE TABLE Salesman(
    salesman_id int,
    sname varchar(50),
    city varchar(50),
    commission float
);

INSERT INTO
    Salesman(salesman_id, sname, city, commission)
VALUES
    (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);

CREATE TABLE Customer(
    customer_id int,
    cust_name varchar(50),
    city varchar(25),
    grade int,
    salesman_id int
);

INSERT INTO
    Customer(customer_id, cust_name, city, grade, salesman_id)
VALUES
    (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', 300, 5005);
```

```
CREATE TABLE Orders(  
    ord_no int,  
    purch_amt double,  
    ord_date date,  
    salesman_id int,  
    customer_id int  
);
```

```
INSERT INTO
```

```
Orders(  
    ord_no,  
    purch_amt,  
    ord_date,  
    customer_id,  
    salesman_id  
)
```

```
VALUES
```

```
(70001, 150.5 , '2012-10-05', 3005, 5002),  
(70009, 270.65, '2012-09-10', 3001, 5005),  
(70002, 65.26, '2012-10-05', 3002, 5001),  
(70004, 110.5 , '2012-08-17', 3009, 5003),  
(70007, 948.5 , '2012-09-10', 3005, 5002),  
(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),  
(70012, 250.45, '2012-06-27', 3008, 5002),  
(70011, 75.29, '2012-08-17', 3003, 5007),
```

```

(70013, 3045.6 , '2012-04-25', 3002, 5001);

/*SELECT
    *

FROM
    Orders; */

-- 1a , commission > 12%

SELECT
    Customer.cust_name AS "Customer Name",
    Customer.city,
    Salesman.sname AS "Salesman",
    Salesman.commission
FROM
    Customer
    INNER JOIN Salesman ON Customer.salesman_id = Salesman.salesman_id
WHERE
    Salesman.commission > .12;

-- 1b

SELECT
    Customer.cust_name,
    Customer.city,
    Orders.ord_no,
    Orders.ord_date,
    Orders.purch_amt AS "Order Amount"
FROM
    Customer
    LEFT OUTER JOIN Orders ON Customer.customer_id = Orders.customer_id
order by
    Orders.ord_date;

-- 1c

SELECT
    Customer.cust_name,

```

```

Customer.city,
Orders.ord_no,
Orders.ord_date,
Orders.purch_amt AS "Order Amount",
Salesman.sname,
Salesman.commission
FROM
    Customer
    LEFT OUTER JOIN Orders ON Customer.customer_id = Orders.customer_id
    LEFT OUTER JOIN Salesman ON Salesman.salesman_id = Orders.salesman_id;
-- 1d

SELECT
    Customer.cust_name,
    Customer.city,
    Customer.grade,
    Salesman.sname AS "Salesman",
    Salesman.city
FROM
    Customer
    RIGHT OUTER JOIN Salesman ON Salesman.salesman_id = Customer.salesman_id
ORDER BY
    Salesman.salesman_id;
-- 1e

SELECT
    *
FROM
    Salesman
    CROSS JOIN Customer
    -- WHERE Customer.salesman_id = Salesman.salesman_id
;
-- 1f

```

SELECT

*

FROM

Salesman

CROSS JOIN Customer

WHERE

Salesman.city = Customer.city;

-- 1g

SELECT

Customer.cust_name,

Customer.city,

Customer.grade,

Salesman.sname AS "Salesman",

Orders.ord_no,

Orders.ord_date,

Orders.purch_amt

FROM

Customer

RIGHT OUTER JOIN Salesman ON Salesman.salesman_id = Customer.salesman_id

LEFT OUTER JOIN Orders ON Orders.customer_id = Customer.customer_id

WHERE

Orders.purch_amt >= 2000

AND Customer.grade IS NOT NULL;

DELETE FROM Orders;

DELETE FROM Customer;

DELETE FROM Salesman;

DROP TABLE Salesman, Customer, Orders;

DROP DATABASE w3q1_orders;

Inputs:

→ Salesman table

	salesman_id	sname	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 table

	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	300	5005

→ Orders

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

Queries & Outputs:

→ SELECT

Customer.cust_name AS "Customer Name",
Customer.city,
Salesman.sname AS "Salesman",
Salesman.commission

FROM

Customer

INNER JOIN Salesman ON Customer.salesman_id = Salesman.salesman_id
WHERE
Salesman.commission > .12;

	Customer Name	city	Salesman	commission
▶	Nick Rimando	New York	James Hoog	0.15
	Brad Davis	New York	James Hoog	0.15
	Graham Zusi	California	Nail Knite	0.13
	Julian Green	London	Nail Knite	0.13
	Fabian Johnson	Paris	Mc Lyon	0.14
	Jozy Altidor	Moscow	Paul Adam	0.13

→ SELECT
Customer.cust_name,
Customer.city,
Orders.ord_no,
Orders.ord_date,
Orders.purch_amt AS "Order Amount"
FROM
Customer
LEFT OUTER JOIN Orders ON Customer.customer_id = Orders.customer_id
order by
Orders.ord_date;

	cust_name	city	ord_no	ord_date	Order Amount
▶	Nick Rimando	New York	70013	2012-04-25	3045.6
	Julian Green	London	70012	2012-06-27	250.45
	Brad Davis	New York	70005	2012-07-27	2400.6
	Jozy Altidor	Moscow	70011	2012-08-17	75.29
	Geoff Cameron	Berlin	70004	2012-08-17	110.5
	Nick Rimando	New York	70008	2012-09-10	5760
	Graham Zusi	California	70007	2012-09-10	948.5
	Brad Guzan	London	70009	2012-09-10	270.65
	Nick Rimando	New York	70002	2012-10-05	65.26
	Graham Zusi	California	70001	2012-10-05	150.5
	Fabian Johnson	Paris	70010	2012-10-10	1983.43
	Geoff Cameron	Berlin	70003	2012-10-10	2480.4

→ SELECT
Customer.cust_name,
Customer.city,
Orders.ord_no,
Orders.ord_date,
Orders.purch_amt AS "Order Amount",
Salesman.sname,
Salesman.commission
FROM

Customer

LEFT OUTER JOIN Orders ON Customer.customer_id = Orders.customer_id

LEFT OUTER JOIN Salesman ON Salesman.salesman_id = Orders.salesman_id;

	cust_name	city	ord_no	ord_date	Order Amount	sname	commission
▶	Nick Rimando	New York	70002	2012-10-05	65.26	James Hoog	0.15
	Brad Davis	New York	70005	2012-07-27	2400.6	James Hoog	0.15
	Nick Rimando	New York	70008	2012-09-10	5760	James Hoog	0.15
	Nick Rimando	New York	70013	2012-04-25	3045.6	James Hoog	0.15
	Graham Zusi	California	70001	2012-10-05	150.5	Nail Knite	0.13
	Graham Zusi	California	70007	2012-09-10	948.5	Nail Knite	0.13
	Julian Green	London	70012	2012-06-27	250.45	Nail Knite	0.13
	Brad Guzan	London	70009	2012-09-10	270.65	Pit Alex	0.11
	Fabian Johnson	Paris	70010	2012-10-10	1983.43	Mc Lyon	0.14
	Jozy Altidor	Moscow	70011	2012-08-17	75.29	Paul Adam	0.13
	Geoff Cameron	Berlin	70004	2012-08-17	110.5	Lauson Hen	0.12
	Geoff Cameron	Berlin	70003	2012-10-10	2480.4	Lauson Hen	0.12

➔ SELECT

Customer.cust_name,

Customer.city,

Customer.grade,

Salesman.sname AS "Salesman",

Salesman.city

FROM

Customer

RIGHT OUTER JOIN Salesman ON Salesman.salesman_id = Customer.salesman_id

ORDER BY

Salesman.salesman_id;

	cust_name	city	grade	Salesman	city
▶	Brad Davis	New York	200	James Hoog	New York
	Nick Rimando	New York	100	James Hoog	New York
	Julian Green	London	300	Nail Knite	Paris
	Graham Zusi	California	200	Nail Knite	Paris
	Geoff Cameron	Berlin	100	Lauson Hen	San Jose
	Brad Guzan	London	300	Pit Alex	London
	Fabian Johnson	Paris	300	Mc Lyon	Paris
	Jozy Altidor	Moscow	200	Paul Adam	Rome

➔ SELECT *

FROM Salesman

CROSS JOIN Customer

-- WHERE Customer.salesman_id = Salesman.salesman_id;

	salesman_id	sname	city	commission	customer_id	cust_name	city	grade	salesman_id
▶	5001	James Hoog	New York	0.15	3002	Nick Rimando	New York	100	5001
	5002	Nail Knite	Paris	0.13	3002	Nick Rimando	New York	100	5001
	5005	Pit Alex	London	0.11	3002	Nick Rimando	New York	100	5001
	5006	Mc Lyon	Paris	0.14	3002	Nick Rimando	New York	100	5001
	5007	Paul Adam	Rome	0.13	3002	Nick Rimando	New York	100	5001
	5003	Lauson Hen	San Jose	0.12	3002	Nick Rimando	New York	100	5001
	5001	James Hoog	New York	0.15	3007	Brad Davis	New York	200	5001
	5002	Nail Knite	Paris	0.13	3007	Brad Davis	New York	200	5001
	5005	Pit Alex	London	0.11	3007	Brad Davis	New York	200	5001
	5006	Mc Lyon	Paris	0.14	3007	Brad Davis	New York	200	5001
	5007	Paul Adam	Rome	0.13	3007	Brad Davis	New York	200	5001
	5003	Lauson Hen	San Jose	0.12	3007	Brad Davis	New York	200	5001
	5001	James Hoog	New York	0.15	3005	Graham Zusi	California	200	5002
	5002	Nail Knite	Paris	0.13	3005	Graham Zusi	California	200	5002
	5005	Pit Alex	London	0.11	3005	Graham Zusi	California	200	5002
	5006	Mc Lyon	Paris	0.14	3005	Graham Zusi	California	200	5002
	5007	Paul Adam	Rome	0.13	3005	Graham Zusi	California	200	5002
	5003	Lauson Hen	San Jose	0.12	3005	Graham Zusi	California	200	5002
	5001	James Hoog	New York	0.15	3008	Julian Green	London	300	5002
	5002	Nail Knite	Paris	0.13	3008	Julian Green	London	300	5002
	5005	Pit Alex	London	0.11	3008	Julian Green	London	300	5002
	5006	Mc Lyon	Paris	0.14	3008	Julian Green	London	300	5002
	5007	Paul Adam	Rome	0.13	3008	Julian Green	London	300	5002
	5003	Lauson Hen	San Jose	0.12	3008	Julian Green	London	300	5002
	5001	James Hoog	New York	0.15	3004	Fabian John...	Paris	300	5006
	5002	Nail Knite	Paris	0.13	3004	Fabian John...	Paris	300	5006
	5005	Pit Alex	London	0.11	3004	Fabian John...	Paris	300	5006
	5006	Mc Lyon	Paris	0.14	3004	Fabian John...	Paris	300	5006
	5007	Paul Adam	Rome	0.13	3004	Fabian John...	Paris	300	5006
	5003	Lauson Hen	San Jose	0.12	3004	Fabian John...	Paris	300	5006
	5001	James Hoog	New York	0.15	3009	Geoff Came...	Berlin	100	5003
	5002	Nail Knite	Paris	0.13	3009	Geoff Came...	Berlin	100	5003
	5005	Pit Alex	London	0.11	3009	Geoff Came...	Berlin	100	5003
	5006	Mc Lyon	Paris	0.14	3009	Geoff Came...	Berlin	100	5003
	5007	Paul Adam	Rome	0.13	3009	Geoff Came...	Berlin	100	5003
	5003	Lauson Hen	San Jose	0.12	3009	Geoff Came...	Berlin	100	5003
	5001	James Hoog	New York	0.15	3003	Jozy Altidor	Moscow	200	5007
	5002	Nail Knite	Paris	0.13	3003	Jozy Altidor	Moscow	200	5007
	5005	Pit Alex	London	0.11	3003	Jozy Altidor	Moscow	200	5007
	5006	Mc Lyon	Paris	0.14	3003	Jozy Altidor	Moscow	200	5007
	5007	Paul Adam	Rome	0.13	3003	Jozy Altidor	Moscow	200	5007
	5003	Lauson Hen	San Jose	0.12	3003	Jozy Altidor	Moscow	200	5007
	5001	James Hoog	New York	0.15	3001	Brad Guzan	London	300	5005
	5002	Nail Knite	Paris	0.13	3001	Brad Guzan	London	300	5005
	5005	Pit Alex	London	0.11	3001	Brad Guzan	London	300	5005
	5006	Mc Lyon	Paris	0.14	3001	Brad Guzan	London	300	5005
	5007	Paul Adam	Rome	0.13	3001	Brad Guzan	London	300	5005
	5003	Lauson Hen	San Jose	0.12	3001	Brad Guzan	London	300	5005

-> SELECT

*

FROM

Salesman

CROSS JOIN Customer

WHERE

Salesman.city = Customer.city;

	salesman_id	sname	city	commission	customer_id	cust_name	city	grade	salesman_id
▶	5001	James Hoog	New York	0.15	3002	Nick Rimando	New York	100	5001
	5001	James Hoog	New York	0.15	3007	Brad Davis	New York	200	5001
	5005	Pit Alex	London	0.11	3008	Julian Green	London	300	5002
	5002	Nail Knite	Paris	0.13	3004	Fabian Johnson	Paris	300	5006
	5006	Mc Lyon	Paris	0.14	3004	Fabian Johnson	Paris	300	5006
	5005	Pit Alex	London	0.11	3001	Brad Guzan	London	300	5005

➔ SELECT

Customer.cust_name,
Customer.city,
Customer.grade,
Salesman.sname AS "Salesman",
Orders.ord_no,
Orders.ord_date,
Orders.purch_amt

FROM

Customer

RIGHT OUTER JOIN Salesman ON Salesman.salesman_id = Customer.salesman_id

LEFT OUTER JOIN Orders ON Orders.customer_id = Customer.customer_id

WHERE

Orders.purch_amt >= 2000

AND Customer.grade IS NOT NULL;

	cust_name	city	grade	Salesman	ord_no	ord_date	purch_amt
▶	Brad Davis	New York	200	James Hoog	70005	2012-07-27	2400.6
	Nick Rimando	New York	100	James Hoog	70008	2012-09-10	5760
	Geoff Cameron	Berlin	100	Lauson Hen	70003	2012-10-10	2480.4
	Nick Rimando	New York	100	James Hoog	70013	2012-04-25	3045.6

2)

Code:

CREATE DATABASE Empire;

USE Empire;

CREATE TABLE Employee (

employee_id INT PRIMARY KEY,

first_name VARCHAR(25) NOT NULL,

```

last_name VARCHAR(25) NOT NULL,
email VARCHAR(25) NOT NULL,
phone_number VARCHAR(12) NOT NULL,
hire_date DATE NOT NULL,
job_id VARCHAR(13) NOT NULL,
salary FLOAT(7,2) NOT NULL DEFAULT 0.0,
commission_pct FLOAT(7,2) /*NOT NULL*/ DEFAULT 0,
manager_id INT NOT NULL,
department_id INT NOT NULL -- ,
-- CONSTRAINT manager_id CHECK manager_id IN employee_id
);

INSERT INTO Employee
VALUES
(100, "Steven", "King", "SKING", '515.123.4567', '2003-06-17', 'AD_PRES', 24000.00, 0.00,
0, 90),
(101, "Neena", "Kochhar", "NKOCHHAR", '515.123.4568', '2005-09-21', 'AD_VP',
17000.00, 0.00, 100, 40),
(102, "Lex", "De Haan", "LDEHAAN", '515.123.4569', '2001-01-13', 'AD_VP',
17000.00, 0.00, 100, 90),
(103, "Alexander", "Hunold", "AHUNOLD", '590.423.4567', '2006-01-03', 'IT_PROG',
9000.00, 0.00, 102, 60),
(104, "Bruce", "Ernst", "BERNST", '590.423.4568', '2007-05-21', 'IT_PROG', 6000.00,
0.00, 103, 60),
(105, "David", "Austin", "DAUSTIN", '590.423.4569', '2005-06-25', 'IT_PROG', 4800.00,
0.00, 103, 40),
(106, "Valli", "Pataballa", "VPATABAL", '590.423.4560', '2006-02-05', 'IT_PROG',
4800.00, 0.00, 103, 60),
(107, "Diana", "Lorentz", "DLORENTZ", '590.423.5567', '2007-02-07', 'IT_PROG',
4800.00, 0.00, 103, 60),
(108, "Nancy", "Greenberg", "NGREENBE", '515.124.4569', '2002-08-17', 'FI_MGR',
12008.00, 0.00, 103, 100),
(109, "Daniel", "Faviet", "DFAVIET", '515.124.4169', '2002-08-16', 'FI_ACCOUNT',
9000.00, 0.00, 103, 100),
(110, "John", "Chen", "JCHEN", '515.124.4269', '2005-09-28', 'FI_ACCOUNT', 8200.00,
0.00, 103, 40)

```

;

-- 2a

```
SELECT first_name, last_name, department_id
FROM Employee
WHERE employee_id IN (
    SELECT employee_id
    FROM Employee
    WHERE SALARY > 3.7E3
);
```

-- 2a 2nd way

```
/* SELECT first_name, last_name, department_id
FROM Employee
WHERE salary > 3700
; /* why only 3700, not 5k? */
```

-- 2b

```
SELECT department_id, SUM(salary) AS Total_Salary
FROM Employee
WHERE department_id IN (
    SELECT DISTINCT department_id
    FROM Employee
    -- GROUP BY department_id
    -- HAVING COUNT(department_id) >= 1
)
GROUP BY department_id
HAVING COUNT(department_id) >= 1
;
```

-- 2b 2nd way

```
/*SELECT department_id, SUM(salary) AS Total_Salary
FROM Employee
GROUP BY department_id
```

```

-- HAVING COUNT(department_id) >= 1
; /**/
-- SELECT DISTINCT department_id FROM Employee;
-- 2c but no nested query? , Insert meme chimp asks "where nested query?"
SELECT DISTINCT department_id
FROM Employee
WHERE department_id IN (
    SELECT department_id
    FROM Employee
)
GROUP BY department_id
HAVING COUNT(department_id) >= 1
;
-- 2d
SELECT first_name, last_name, department_id
FROM Employee
WHERE salary > (
    SELECT AVG(salary)
    FROM Employee
);
-- 2e
SELECT first_name, last_name, salary, department_id
FROM Employee
WHERE salary > (
    SELECT MAX(salary)
    FROM Employee
    WHERE department_id = 40
);

DELETE FROM Employee;

```

DROP TABLE Employee;

DROP DATABASE Empire;

Inputs:

➔ Employee Table

	employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary	commission_pct	manager_id	department_id
▶	100	Steven	King	SKING	515.123.4567	2003-06-17	AD_PRES	24000.00	0.00	0	90
	101	Neena	Kochhar	NKOCCHAR	515.123.4568	2005-09-21	AD_VP	17000.00	0.00	100	40
	102	Lex	De Haan	LDEHAAN	515.123.4569	2001-01-13	AD_VP	17000.00	0.00	100	90
	103	Alexander	Hunold	AHUNOLD	590.423.4567	2006-01-03	IT_PROG	9000.00	0.00	102	60
	104	Bruce	Ernst	BERNST	590.423.4568	2007-05-21	IT_PROG	6000.00	0.00	103	60
	105	David	Austin	DAUSTIN	590.423.4569	2005-06-25	IT_PROG	4800.00	0.00	103	40
	106	Valli	Pataballa	VPATABAL	590.423.4560	2006-02-05	IT_PROG	4800.00	0.00	103	60
	107	Diana	Lorentz	DLORENTZ	590.423.5567	2007-02-07	IT_PROG	4800.00	0.00	103	60
	108	Nancy	Greenberg	NGREENBE	515.124.4569	2002-08-17	FI_MGR	12008.00	0.00	103	100
	109	Daniel	Faviet	DFAVIET	515.124.4169	2002-08-16	FI_ACCOUNT	9000.00	0.00	103	100
	110	John	Chen	JCHEN	515.124.4269	2005-09-28	FI_ACCOUNT	8200.00	0.00	103	40
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Queries & Outputs:

➔ SELECT first_name, last_name, department_id
FROM Employee
WHERE employee_id IN (
 SELECT employee_id
 FROM Employee
 WHERE SALARY > 3.7E3
);

	first_name	last_name	department_id
▶	Steven	King	90
	Neena	Kochhar	40
	Lex	De Haan	90
	Alexander	Hunold	60
	Bruce	Ernst	60
	David	Austin	40
	Valli	Pataballa	60
	Diana	Lorentz	60
	Nancy	Greenberg	100
	Daniel	Faviet	100
	John	Chen	40

➔ SELECT department_id, SUM(salary) AS Total_Salary
FROM Employee
WHERE department_id IN (
 SELECT DISTINCT department_id
 FROM Employee
 -- GROUP BY department_id
 -- HAVING COUNT(department_id) >= 1
)
GROUP BY department_id

HAVING COUNT(department_id) >= 1

;

	department_id	Total_Salary
▶	40	30000.00
	60	24600.00
	90	41000.00
	100	21008.00

➔ SELECT DISTINCT department_id
FROM Employee
WHERE department_id IN (
 SELECT department_id
 FROM Employee
)
GROUP BY department_id
HAVING COUNT(department_id) >= 1
;

	department_id
▶	40
	60
	90
	100

SELECT first_name, last_name, department_id
FROM Employee
WHERE salary > (
 SELECT AVG(salary)
 FROM Employee
)
ORDER BY salary DESC;

	first_name	last_name	department_id
▶	Steven	King	90
	Neena	Kochhar	40
	Lex	De Haan	90
	Nancy	Greenberg	100

➔ SELECT first_name, last_name, salary, department_id
FROM Employee
WHERE salary > (
 SELECT MAX(salary)
 FROM Employee
 WHERE department_id = 40
)
);

	first_name	last_name	salary	department_id
▶	Steven	King	24000.00	90