Assignment 1

Create a Database name entri_assignment

Create a Table with name `departments`
`Department_id  (pk) Department_name Location_id+`

Create a Table with name `employees`
`Employee_id  (pk)  ,first_name,last_name`
`,email,phone_number,hire_date,`
`job_id, salary, commission_pct, manager_id, department_id (fk reference`

```
## Insert into Departments table
 INSERT INTO departments VALUES ( 20,'Marketing',  180);

 INSERT INTO departments VALUES ( 30,'Purchasing',  1700);

 INSERT INTO departments VALUES ( 40, 'Human Resources',  2400);

 INSERT INTO departments VALUES ( 50, 'Shipping',  1500);

 INSERT INTO departments VALUES ( 60 , 'IT',  1400);

 INSERT INTO departments VALUES ( 70, 'Public Relations',  2700);

 INSERT INTO departments VALUES ( 80 , 'Sales',  2500 );

 INSERT INTO departments VALUES ( 90 , 'Executive',  1700);

 INSERT INTO departments VALUES ( 100 , 'Finance',  1700);

 INSERT INTO departments VALUES ( 110 , 'Accounting',  1700);

 INSERT INTO departments VALUES ( 120 , 'Treasury' ,  1700);

 INSERT INTO departments VALUES ( 130 , 'Corporate Tax' ,  1700 );

 INSERT INTO departments VALUES ( 140, 'Control And Credit'
,  1700);

 INSERT INTO departments VALUES ( 150 , 'Shareholder Services',
1700);

 INSERT INTO departments VALUES ( 160 , 'Benefits', 1700);
```

```
 INSERT INTO departments VALUES ( 170 , 'Payroll' , 1700);
```

employees table

```
INSERT INTO employees VALUES (100, 'Steven', 'King', 'SKING',
'515.123.4567', '1987-06-17' , 'AD_PRES', 24000 , NULL, NULL, 20);
 Insertinto employees VALUES (101, 'Neena' , 'Kochhar' , 'NKOCHHAR'
, '515.123.4568' , '1989-11-21' , 'AD_VP' , 17000 , NULL , 100 ,
20);
INSERT INTO employees VALUES (102 , 'Lex' , 'De Haan' , 'LDEHAAN' ,
'515.123.4569' , '1993-09-12' , 'AD_VP' , 17000 , NULL , 100 , 30);
INSERT INTO employees VALUES (104 , 'Bruce' , 'Ernst' , 'BERNST' ,
'590.423.4568' , '1991-05-21',  'IT_PROG' , 6000 , NULL , 103 , 60);
INSERT INTO employees VALUES (105 , 'David' , 'Austin' , 'DAUSTIN' ,
'590.423.4569' , '1997-06-25',  'IT_PROG' , 4800 , NULL , 103 , 60);
INSERT INTO employees VALUES (106 , 'Valli' , 'Pataballa' ,
'VPATABAL' , '590.423.4560' , '1998-02-05',  'IT_PROG' , 4800 , NULL
, 103 , 40);
INSERT INTO employees VALUES (107 , 'Diana' , 'Lorentz' , 'DLORENTZ'
, '590.423.5567' , '1999-02-09',  'IT_PROG' , 4200 , NULL , 103 ,
40);
INSERT INTO employees VALUES (108 , 'Nancy' , 'Greenberg' ,
'NGREENBE' , '515.124.4569' , '1994-08-17',  'FI_MGR' , 12000 , NULL
, 101 , 100);
INSERT INTO employees VALUES (109 , 'Daniel' , 'Faviet' , 'DFAVIET'
, '515.124.4169' , '1994-08-12',  'FI_ACCOUNT' , 9000 , NULL , 108 ,
170);
INSERT INTO employees VALUES (110 , 'John' , 'Chen' , 'JCHEN' ,
'515.124.4269' , '1997-04-09',  'FI_ACCOUNT' , 8200 , NULL , 108 ,
170);
INSERT INTO employees VALUES (111 , 'Ismael' , 'Sciarra' ,
'ISCIARRA' , '515.124.4369' , '1997-02-01',  'FI_ACCOUNT' , 7700 ,
NULL , 108 , 160);
```

```
INSERT INTO employees VALUES (112 , 'Jose Manuel' , 'Urman' ,
'JMURMAN' , '515.124.4469' , '1998-06-03', 'FI_ACCOUNT' , 7800 ,
NULL 8 , 150);
INSERT INTO employees VALUES (114 , 'Den' , 'Raphaely' , 'DRAPHEAL'
, '515.127.4561' , '1994-11-08',  'PU_MAN' , 11000 , NULL , 100 ,
30);
INSERT INTO employees VALUES (115 , 'Alexander' , 'Khoo' , 'AKHOO' ,
'515.127.4562' , '1995-05-12',  'PU_CLERK' , 3100 , NULL , 114 ,
80);
INSERT INTO employees VALUES (116 , 'Shelli' , 'Baida' , 'SBAIDA' ,
'515.127.4563' ,'1997-12-13', 'PU_CLERK' , 2900 , NULL , 114 , 70);
INSERT INTO employees VALUES (117 , 'Sigal' , 'Tobias' , 'STOBIAS' ,
'515.127.4564' , '1997-09-10', 'PU_CLERK' , 2800 , NULL , 114 , 30);
INSERT INTO employees VALUES (118 , 'Guy' , 'Himuro' , 'GHIMURO' ,
'515.127.4565' , '1998-01-02',  'PU_CLERK' , 2600 , NULL , 114 ,
60);
INSERT INTO employees VALUES (119 , 'Karen' , 'Colmenares' ,
'KCOLMENA' , '515.127.4566' , '1999-04-08',  'PU_CLERK' , 2500 ,
NULL , 114 , 130);
INSERT INTO employees VALUES (120 , 'Matthew' , 'Weiss' , 'MWEISS' ,
'650.123.1234' ,'1996-07-18',  'ST_MAN' , 8000 , NULL , 100 , 50);
INSERT INTO employees VALUES (122 , 'Payam' , 'Kaufling' ,
'PKAUFLIN' , '650.123.3234' ,'1995-05-01',  'ST_MAN' , 7900 , NULL ,
100 , 40);
INSERT INTO employees VALUES (123 , 'Shanta' , 'Vollman' ,
'SVOLLMAN' , '650.123.4234' , '1997-10-12',  'ST_MAN' , 6500 , NULL
, 100 , 50);
INSERT INTO employees VALUES (124, 'Kevin' , 'Mourgos' , 'KMOURGOS'
, '650.123.5234' , '1999-11-12',  'ST_MAN' , 5800 , NULL , 100 ,
80);
INSERT INTO employees VALUES (125, 'Julia' , 'Nayer' , 'JNAYER' ,
'650.124.1214' , '1997-07-02',  'ST_CLERK' , 3200 , NULL , 120 ,
50);
INSERT INTO employees VALUES (126, 'Irene' , 'Mikkilineni' ,
'IMIKKILI' , '650.124.1224' , '1998-11-12', 'ST_CLERK' , 2700 , NULL
, 120 , 50);
INSERT INTO employees VALUES (127, 'James' , 'Landry' , 'JLANDRY' ,
'650.124.1334' , '1999-01-02' , 'ST_CLERK' , 2400 , NULL , 120 ,
90);
INSERT INTO employees VALUES (128, 'Steven' , 'Markle' , 'SMARKLE' ,
'650.124.1434' , '2000-03-04' , 'ST_CLERK' , 2200 , NULL , 120 ,
50);
INSERT INTO employees VALUES (130, 'Mozhe' , 'Atkinson' , 'MATKINSO'
, '650.124.6234' , '1997-10-12' , 'ST_CLERK' , 2800 , NULL , 121 ,
110);
```

Solve SQL Exercises

1. Select employees first name, last name, job_id and salary whose first name starts with alphabet S

```
112
113 •  SELECT first_name ,last_name ,job_id ,salary FROM employees WHERE first_name LIKE 'S%';
114
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

first_name	last_name	job_id	salary
Steven	King	AD_PRES	24000.00
Shelli	Baida	PU_CLERK	2900.00
Sigal	Tobias	PU_CLERK	2800.00
Shanta	Vollman	ST_MAN	6500.00
Steven	Markle	ST_CLERK	2200.00

2. Write a query to select employee with the highest salary (using an inner query)

```
117 •  SELECT * FROM employees WHERE salary = (SELECT MAX(salary) FROM employees);
118
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

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	1987-06-17	AD_PRES	24000.00	NULL	NULL	20
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

3. Select employee with the second highest salary

```
124
125 •  SELECT * FROM employees
126    WHERE salary=(
127    SELECT MAX(salary) FROM employees
128    WHERE salary < (SELECT MAX(salary) FROM employees));
129
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

Employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary	commission_pct	manager_id	department_id
101	Neena	Kochhar	NKOCHHAR	515.123.4568	1989-11-21	AD_VP	17000.00	NULL	100	20
102	Lex	De Haan	LDEHAAN	515.123.4569	1993-09-12	AD_VP	17000.00	NULL	100	30
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

4. Write a query to select employees and their corresponding managers and their salaries

```
mysql>
mysql> SELECT
    ->  e.employee_id AS employee_id,
    ->  e.first_name AS employee_first_name,
    -> e.last_name AS employee_last_name,
    -> e.salary AS employee_salary,
    -> (SELECT m.employee_id FROM employees m WHERE m.employee_id =e.manager_id) AS manager_id,
    -> (SELECT m.first_name FROM employees m WHERE m.employee_id =e.manager_id) AS manager_first_name,
    -> (SELECT m.last_name FROM employees m WHERE m.employee_id =e.manager_id) AS manager_last_name,
    ->  (SELECT m.salary FROM employees m WHERE m.employee_id =e.manager_id) AS manager_salary
    -> FROM employees e ORDER BY manager_id;
```

employee_id	employee_first_name	employee_last_name	employee_salary	manager_id	manager_first_name	manager_last_name	manager_salary
100	Steven	King	24000.00	NULL	NULL	NULL	NULL
101	Neena	Kochhar	17000.00	100	Steven	King	24000.00
102	Lex	De Haan	17000.00	100	Steven	King	24000.00
114	Den	Raphaely	11000.00	100	Steven	King	24000.00
120	Matthew	Weiss	8000.00	100	Steven	King	24000.00
121	Adam	Fripp	8200.00	100	Steven	King	24000.00
122	Payam	Kaufling	7900.00	100	Steven	King	24000.00
123	Shanta	Vollman	6500.00	100	Steven	King	24000.00
124	Kevin	Mourgos	5800.00	100	Steven	King	24000.00
108	Nancy	Greenberg	12000.00	101	Neena	Kochhar	17000.00
103	Alexander	Hunold	9000.00	102	Lex	De Haan	17000.00
104	Bruce	Ernst	6000.00	103	Alexander	Hunold	9000.00
105	David	Austin	4800.00	103	Alexander	Hunold	9000.00
106	Valli	Pataballa	4800.00	103	Alexander	Hunold	9000.00
107	Diana	Lorentz	4200.00	103	Alexander	Hunold	9000.00
109	Daniel	Faviet	9000.00	108	Nancy	Greenberg	12000.00
110	John	Chen	8200.00	108	Nancy	Greenberg	12000.00
111	Ismael	Sciarra	7700.00	108	Nancy	Greenberg	12000.00
112	Jose Manuel	Urman	7800.00	108	Nancy	Greenberg	12000.00
113	Luis	Popp	6900.00	108	Nancy	Greenberg	12000.00
115	Alexander	Khoo	3100.00	114	Den	Raphaely	11000.00
116	Shelli	Baida	2900.00	114	Den	Raphaely	11000.00
117	Sigal	Tobias	2800.00	114	Den	Raphaely	11000.00
118	Guy	Himuro	2600.00	114	Den	Raphaely	11000.00
119	Karen	Colmenares	2500.00	114	Den	Raphaely	11000.00
125	Julia	Nayer	3200.00	120	Matthew	Weiss	8000.00
126	Irene	Mikkilineni	2700.00	120	Matthew	Weiss	8000.00
127	James	Landry	2400.00	120	Matthew	Weiss	8000.00
128	Steven	Markle	2200.00	120	Matthew	Weiss	8000.00
129	Laura	Bissot	3300.00	121	Adam	Fripp	8200.00
130	Mozhe	Atkinson	2800.00	121	Adam	Fripp	8200.00

5. Write a query to select employees and their corresponding managers and their salaries (SELF Join)

```
mysql>
mysql> SELECT
    -> e.employee_id AS employee_id,
    -> e.first_name AS employee_first_name,
    -> e.last_name AS employee_last_name,
    -> e.salary AS employee_salary,
    -> m.employee_id AS manager_id,
    -> m.first_name AS manager_first_name,
    -> m.last_name AS manager_last_name,
    -> m.salary AS manager_salary
    -> FROM employees e
    -> INNER JOIN employees m ON e.manager_id=m.employee_id;
```

employee_id	employee_first_name	employee_last_name	employee_salary	manager_id	manager_first_name	manager_last_name	manager_salary
101	Neena	Kochhar	17000.00	100	Steven	King	24000.00
102	Lex	De Haan	17000.00	100	Steven	King	24000.00
103	Alexander	Hunold	9000.00	102	Lex	De Haan	17000.00
104	Bruce	Ernst	6000.00	103	Alexander	Hunold	9000.00
105	David	Austin	4800.00	103	Alexander	Hunold	9000.00
106	Valli	Pataballa	4800.00	103	Alexander	Hunold	9000.00
107	Diana	Lorentz	4200.00	103	Alexander	Hunold	9000.00
108	Nancy	Greenberg	12000.00	101	Neena	Kochhar	17000.00
109	Daniel	Faviet	9000.00	108	Nancy	Greenberg	12000.00
110	John	Chen	8200.00	108	Nancy	Greenberg	12000.00
111	Ismael	Sciarra	7700.00	108	Nancy	Greenberg	12000.00
112	Jose Manuel	Urman	7800.00	108	Nancy	Greenberg	12000.00
113	Luis	Popp	6900.00	108	Nancy	Greenberg	12000.00
114	Den	Raphaely	11000.00	100	Steven	King	24000.00
115	Alexander	Khoo	3100.00	114	Den	Raphaely	11000.00
116	Shelli	Baida	2900.00	114	Den	Raphaely	11000.00
117	Sigal	Tobias	2800.00	114	Den	Raphaely	11000.00
118	Guy	Himuro	2600.00	114	Den	Raphaely	11000.00
119	Karen	Colmenares	2500.00	114	Den	Raphaely	11000.00
120	Matthew	Weiss	8000.00	100	Steven	King	24000.00
121	Adam	Fripp	8200.00	100	Steven	King	24000.00
122	Payam	Kaufling	7900.00	100	Steven	King	24000.00
123	Shanta	Vollman	6500.00	100	Steven	King	24000.00
124	Kevin	Mourgos	5800.00	100	Steven	King	24000.00
125	Julia	Nayer	3200.00	120	Matthew	Weiss	8000.00
126	Irene	Mikkilineni	2700.00	120	Matthew	Weiss	8000.00
127	James	Landry	2400.00	120	Matthew	Weiss	8000.00
128	Steven	Markle	2200.00	120	Matthew	Weiss	8000.00
129	Laura	Bissot	3300.00	121	Adam	Fripp	8200.00
130	Mozhe	Atkinson	2800.00	121	Adam	Fripp	8200.00

6. Create a view for the above query

```sql
CREATE VIEW Employee_Manager_details AS
SELECT
  e.employee_id AS employee_id,
  e.first_name AS employee_first_name,
  e.last_name AS employee_last_name,
  e.salary AS employee_salary,
  m.employee_id AS manager_id,
  m.first_name AS manager_first_name,
  m.last_name AS manager_last_name,
  m.salary AS manager_salary
FROM employees e
INNER JOIN employees m ON e.manager_id=m.employee_id;
```

```
mysql>
mysql> SELECT * FROM Employee_Manager_details;
+-------------+--------------------+--------------------+-----------------+------------+--------------------+--------------------+----------------+
| employee_id | employee_first_name | employee_last_name | employee_salary | manager_id | manager_first_name | manager_last_name | manager_salary |
+-------------+--------------------+--------------------+-----------------+------------+--------------------+--------------------+----------------+
|         101 | Neena              | Kochhar            |        17000.00 |        100 | Steven             | King               |       24000.00 |
|         102 | Lex                | De Haan            |        17000.00 |        100 | Steven             | King               |       24000.00 |
|         103 | Alexander          | Hunold             |         9000.00 |        102 | Lex                | De Haan            |       17000.00 |
|         104 | Bruce              | Ernst              |         6000.00 |        103 | Alexander          | Hunold             |        9000.00 |
|         105 | David              | Austin             |         4800.00 |        103 | Alexander          | Hunold             |        9000.00 |
|         106 | Valli              | Pataballa          |         4800.00 |        103 | Alexander          | Hunold             |        9000.00 |
|         107 | Diana              | Lorentz            |         4200.00 |        103 | Alexander          | Hunold             |        9000.00 |
|         108 | Nancy              | Greenberg          |        12000.00 |        101 | Neena              | Kochhar            |       17000.00 |
|         109 | Daniel             | Faviet             |         9000.00 |        108 | Nancy              | Greenberg          |       12000.00 |
|         110 | John               | Chen               |         8200.00 |        108 | Nancy              | Greenberg          |       12000.00 |
|         111 | Ismael             | Sciarra            |         7700.00 |        108 | Nancy              | Greenberg          |       12000.00 |
|         112 | Jose Manuel        | Urman              |         7800.00 |        108 | Nancy              | Greenberg          |       12000.00 |
|         113 | Luis               | Popp               |         6900.00 |        108 | Nancy              | Greenberg          |       12000.00 |
|         114 | Den                | Raphaely           |        11000.00 |        100 | Steven             | King               |       24000.00 |
|         115 | Alexander          | Khoo               |         3100.00 |        114 | Den                | Raphaely           |       11000.00 |
|         116 | Shelli             | Baida              |         2900.00 |        114 | Den                | Raphaely           |       11000.00 |
|         117 | Sigal              | Tobias             |         2800.00 |        114 | Den                | Raphaely           |       11000.00 |
|         118 | Guy                | Himuro             |         2600.00 |        114 | Den                | Raphaely           |       11000.00 |
|         119 | Karen              | Colmenares         |         2500.00 |        114 | Den                | Raphaely           |       11000.00 |
|         120 | Matthew            | Weiss              |         8000.00 |        100 | Steven             | King               |       24000.00 |
|         121 | Adam               | Fripp              |         8200.00 |        100 | Steven             | King               |       24000.00 |
|         122 | Payam              | Kaufling           |         7900.00 |        100 | Steven             | King               |       24000.00 |
|         123 | Shanta             | Vollman            |         6500.00 |        100 | Steven             | King               |       24000.00 |
|         124 | Kevin              | Mourgos            |         5800.00 |        100 | Steven             | King               |       24000.00 |
|         125 | Julia              | Nayer              |         3200.00 |        120 | Matthew            | Weiss              |        8000.00 |
|         126 | Irene              | Mikkilineni        |         2700.00 |        120 | Matthew            | Weiss              |        8000.00 |
|         127 | James              | Landry             |         2400.00 |        120 | Matthew            | Weiss              |        8000.00 |
|         128 | Steven             | Markle             |         2200.00 |        120 | Matthew            | Weiss              |        8000.00 |
|         129 | Laura              | Bissot             |         3300.00 |        121 | Adam               | Fripp              |        8200.00 |
|         130 | Mozhe              | Atkinson           |         2800.00 |        121 | Adam               | Fripp              |        8200.00 |
+-------------+--------------------+--------------------+-----------------+------------+--------------------+--------------------+----------------+
30 rows in set (0.00 sec)
```

7. Write a query to show the count of employees under each manager in descending order (from view)

```
174 •  SELECT manager_id,manager_first_name,manager_last_name ,COUNT(employee_id) AS number_of_employees
175    FROM Employee_Manager_details GROUP BY manager_id ORDER BY number_of_employees DESC;
176
177
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

manager_id	manager_first_name	manager_last_name	number_of_employees
100	Steven	King	8
108	Nancy	Greenberg	5
114	Den	Raphaely	5
103	Alexander	Hunold	4
120	Matthew	Weiss	4
121	Adam	Fripp	2
102	Lex	De Haan	1
101	Neena	Kochhar	1

8. Find the count of employees in each department

```
mysql>
mysql> SELECT
    -> d.department_id,d.department_name,COUNT(e.employee_id) AS employee_count
    -> FROM departments d
    -> LEFT JOIN employees e ON d.department_id = e.department_id
    -> GROUP BY d.department_id, d.department_name ORDER BY d.department_id;
+---------------+----------------------+----------------+
| department_id | department_name      | employee_count |
+---------------+----------------------+----------------+
|            20 | Marketing            |              2 |
|            30 | Purchasing           |              3 |
|            40 | Human Resources      |              3 |
|            50 | Shipping             |              7 |
|            60 | IT                   |              4 |
|            70 | Public Relations     |              1 |
|            80 | Sales                |              2 |
|            90 | Executive            |              1 |
|           100 | Finance              |              1 |
|           110 | Accounting           |              1 |
|           120 | Treasury             |              0 |
|           130 | Corporate Tax        |              1 |
|           140 | Control And Credit   |              1 |
|           150 | Shareholder Services |              1 |
|           160 | Benefits             |              1 |
|           170 | Payroll              |              2 |
+---------------+----------------------+----------------+
16 rows in set (0.00 sec)
```

9. Get the count of employees hired year wise

```
mysql>
mysql> SELECT YEAR(hire_date) AS hired_year,COUNT(employee_id)
    -> FROM employees
    -> GROUP BY YEAR(hire_date) ORDER BY YEAR(hire_date);
+------------+--------------------+
| hired_year | COUNT(employee_id) |
+------------+--------------------+
|       1987 |                  1 |
|       1989 |                  1 |
|       1990 |                  1 |
|       1991 |                  1 |
|       1993 |                  1 |
|       1994 |                  3 |
|       1995 |                  2 |
|       1996 |                  1 |
|       1997 |                 10 |
|       1998 |                  4 |
|       1999 |                  5 |
|       2000 |                  1 |
+------------+--------------------+
12 rows in set (0.00 sec)
```

10 . create a stored procedure to get the “ Get the count of employees hired in the input year”(IN year , OUT count)

_

```
261    DELIMITER !!
262    CREATE PROCEDURE Employee_count_by_year(IN input_year INT ,     OUT employee_count INT)
263    BEGIN
264    SELECT  COUNT(*) INTO employee_count FROM employees
265    WHERE YEAR(hire_date) = input_year;
266    END !!
267    DELIMITER ;
268    CALL Employee_count_by_year(1994,@employee_count);
269    SELECT @employee_count AS count_of_emloyess_at_1994;
270
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

count_of_emloyess_at_1994
3

11.Select the employees whose first_name contains “an”

```
215
216    SELECT * FROM employees WHERE first_name LIKE '%an%';
217
218
219
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

Employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary	commission_pct	manager_id	department_id
103	Alexander	Hunold	AHUNOLD	590.423.4567	1990-09-30	IT_PROG	9000.00	NULL	102	60
107	Diana	Lorentz	DLORENTZ	590.423.5567	1999-02-09	IT_PROG	4200.00	NULL	103	40
108	Nancy	Greenberg	NGREENBE	515.124.4569	1994-08-17	FI_MGR	12000.00	NULL	101	100
109	Daniel	Faviet	DFAVIET	515.124.4169	1994-08-12	FI_ACCOUNT	9000.00	NULL	108	170
112	Jose Manuel	Urman	JMURMAN	515.124.4469	1998-06-03	FI_ACCOUNT	7800.00	NULL	108	150
115	Alexander	Khoo	AKHOO	515.127.4562	1995-05-12	PU_CLERK	3100.00	NULL	114	80
123	Shanta	Vollman	SVOLLMAN	650.123.4234	1997-10-12	ST_MAN	6500.00	NULL	100	50
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

12. Select employee first name and the corresponding phone number in the format (_ _ _)-(_ _ _)-(_ _ _ _)

```
mysql> SELECT first_name, phone_number FROM employees;
+-------------+--------------+
| first_name  | phone_number |
+-------------+--------------+
| Steven      | 515.123.4567 |
| Neena       | 515.123.4568 |
| Lex         | 515.123.4569 |
| Alexander   | 590.423.4567 |
| Bruce       | 590.423.4568 |
| David       | 590.423.4569 |
| Valli       | 590.423.4560 |
| Diana       | 590.423.5567 |
| Nancy       | 515.124.4569 |
| Daniel      | 515.124.4169 |
| John        | 515.124.4269 |
| Ismael      | 515.124.4369 |
| Jose Manuel | 515.124.4469 |
| Luis        | 515.124.4567 |
| Den         | 515.127.4561 |
| Alexander   | 515.127.4562 |
| Shelli      | 515.127.4563 |
| Sigal       | 515.127.4564 |
| Guy         | 515.127.4565 |
| Karen       | 515.127.4566 |
| Matthew     | 650.123.1234 |
| Adam        | 650.123.2234 |
| Payam       | 650.123.3234 |
| Shanta      | 650.123.4234 |
| Kevin       | 650.123.5234 |
| Julia       | 650.124.1214 |
| Irene       | 650.124.1224 |
| James       | 650.124.1334 |
| Steven      | 650.124.1434 |
| Laura       | 650.124.5234 |
| Mozhe       | 650.124.6234 |
+-------------+--------------+
31 rows in set (0.00 sec)
```

13. Find the employees who joined in August, 1994.

```
226 •   SELECT * FROM employees
227     WHERE YEAR(hire_date) =1994 AND MONTH(hire_date) =08;
228
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

Employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary	commission_pct	manager_id	department_id
108	Nancy	Greenberg	NGREENBE	515.124.4569	1994-08-17	FI_MGR	12000.00	NULL	101	100
109	Daniel	Faviet	DFAVIET	515.124.4169	1994-08-12	FI_ACCOUNT	9000.00	NULL	108	170
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

14. Find the maximum salary from each department

```
mysql> SELECT d.department_id,d.department_name,MAX(e.salary) AS maximum_salary
    -> FROM departments d
    -> LEFT JOIN employees e ON d.department_id = e.department_id
    -> GROUP BY d.department_id ,d.department_name ORDER BY d.department_id;
+---------------+----------------------+----------------+
| department_id | department_name      | maximum_salary |
+---------------+----------------------+----------------+
|            20 | Marketing            |       24000.00 |
|            30 | Purchasing           |       17000.00 |
|            40 | Human Resources      |        7900.00 |
|            50 | Shipping             |        8200.00 |
|            60 | IT                   |        9000.00 |
|            70 | Public Relations     |        2900.00 |
|            80 | Sales                |        5800.00 |
|            90 | Executive            |        2400.00 |
|           100 | Finance              |       12000.00 |
|           110 | Accounting           |        2800.00 |
|           120 | Treasury             |           NULL |
|           130 | Corporate Tax        |        2500.00 |
|           140 | Control And Credit   |        6900.00 |
|           150 | Shareholder Services |        7800.00 |
|           160 | Benefits             |        7700.00 |
|           170 | Payroll              |        9000.00 |
+---------------+----------------------+----------------+
16 rows in set (0.00 sec)
```

15.Write a SQL query to display the 5 least earning employees

```
246 •   SELECT * FROM employees
247     ORDER BY salary
248     LIMIT 5;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Fetch rows:

Employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary	commission_pct	manager_id	department_id
128	Steven	Markle	SMARKLE	650.124.1434	2000-03-04	ST_CLERK	2200.00	NULL	120	50
127	James	Landry	JLANDRY	650.124.1334	1999-01-02	ST_CLERK	2400.00	NULL	120	90
119	Karen	Colmenares	KCOLMENA	515.127.4566	1999-04-08	PU_CLERK	2500.00	NULL	114	130
118	Guy	Himuro	GHIMURO	515.127.4565	1998-01-02	PU_CLERK	2600.00	NULL	114	60
126	Irene	Mikkilineni	IMIKKILI	650.124.1224	1998-11-12	ST_CLERK	2700.00	NULL	120	50
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

16. Find the employees hired in the 80s

```
250
251 •   SELECT * FROM employees WHERE YEAR(hire_date) >1979 AND YEAR(hire_date)<1990;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

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	1987-06-17	AD_PRES	24000.00	NULL	NULL	20
101	Neena	Kochhar	NKOCHHAR	515.123.4568	1989-11-21	AD_VP	17000.00	NULL	100	20
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

17. Find the employees who joined the company after 15th of the month

```
253
254 •   SELECT * FROM employees WHERE DAY(hire_date) > 15;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

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	1987-06-17	AD_PRES	24000.00	NULL	NULL	20
101	Neena	Kochhar	NKOCHHAR	515.123.4568	1989-11-21	AD_VP	17000.00	NULL	100	20
103	Alexander	Hunold	AHUNOLD	590.423.4567	1990-09-30	IT_PROG	9000.00	NULL	102	60
104	Bruce	Ernst	BERNST	590.423.4568	1991-05-21	IT_PROG	6000.00	NULL	103	60
105	David	Austin	DAUSTIN	590.423.4569	1997-06-25	IT_PROG	4800.00	NULL	103	60
108	Nancy	Greenberg	NGREENBE	515.124.4569	1994-08-17	FI_MGR	12000.00	NULL	101	100
120	Matthew	Weiss	MWEISS	650.123.1234	1996-07-18	ST_MAN	8000.00	NULL	100	50
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL