```
CREATE DATABASE entri_assignment;
USE entri_assignment;
CREATE TABLE departments (
   Department_id INT AUTO_INCREMENT PRIMARY KEY,
   Department_name VARCHAR(255) NOT NULL,
   Location_id INT NOT NULL
);
USE entri_assignment;

CREATE TABLE employees (
   Employee_id INT AUTO_INCREMENT PRIMARY KEY,
   first_name VARCHAR(255) NOT NULL,
   last_name VARCHAR(255) NOT NULL,
   email VARCHAR(255) NOT NULL,
   phone_number VARCHAR(20),
   hire_date DATE NOT NULL,
   job_id VARCHAR(10) NOT NULL,
   salary DECIMAL(10, 2) NOT NULL,
   commission_pct DECIMAL(5, 2),
   manager_id INT,
   department_id INT,
   FOREIGN KEY (department_id) REFERENCES departments(Department_id)
);
USE entri_assignment;

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);


INSERT INTO employees VALUES (100, 'Steven', 'King', 'SKING', '515.123.4567', '1987-06-17' , 'AD_PRES', 24000 , NULL, NULL, 20);

INSERT INTO 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);
```

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

```
SELECT first_name, last_name, job_id, salary

FROM employees

WHERE first_name LIKE 'S%';
```

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

```
SELECT first_name, last_name, job_id, salary
FROM employees
WHERE salary = (SELECT MAX(salary) FROM employees);
```

#3. Select employee with the second highest salary

```
SELECT first_name, last_name, job_id, salary
FROM employees
WHERE salary = (
  SELECT MAX(salary)
  FROM employees
  WHERE salary < (SELECT MAX(salary) FROM employees)
);
```

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

```
SELECT
  e.first_name AS Employee_First_Name,
  e.last_name AS Employee_Last_Name,
  e.salary AS Employee_Salary,
  m.first_name AS Manager_First_Name,
  m.last_name AS Manager_Last_Name,
  m.salary AS Manager_Salary
FROM
  employees e
LEFT JOIN
  employees m
ON
```

```
  e.manager_id = m.employee_id;
```

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

```
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
JOIN
  employees m
ON
  e.manager_id = m.employee_id;
```

#6. Create a view for the above query

```
CREATE VIEW employee_manager_view 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
JOIN
  employees m
ON
  e.manager_id = m.employee_id;

SELECT * FROM employee_manager_view;
```

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

```
SELECT
  Manager_ID,
  Manager_First_Name,
  Manager_Last_Name,
  COUNT(Employee_ID) AS Employee_Count
FROM
  employee_manager_view
GROUP BY
  Manager_ID,
  Manager_First_Name,
  Manager_Last_Name
ORDER BY
  Employee_Count DESC;
```

#8. Find the count of employees in each department

```
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;
```

#9. Get the count of employees hired year wise

```
SELECT
  YEAR(hire_date) AS Hire_Year,
  COUNT(employee_id) AS Employee_Count
FROM
  employees
GROUP BY
  YEAR(hire_date)
ORDER BY
  Hire_Year;
```

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

```
DELIMITER $$

CREATE PROCEDURE GetEmployeeCountByYear(IN input_year INT, OUT employee_count INT)
BEGIN
```

```
   SELECT COUNT(employee_id)

   INTO employee_count

   FROM employees

   WHERE YEAR(hire_date) = input_year;

END $$


DELIMITER ;

CALL GetEmployeeCountByYear(2023, @employee_count);

SELECT @employee_count;
```

#11.Select the employees whose first_name contains “an”

```
SELECT first_name, last_name, job_id, salary

FROM employees

WHERE first_name LIKE '%an%';
```

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

```
SELECT

   first_name,

   CONCAT('(', SUBSTRING(phone_number, 1, 3), ')-(', SUBSTRING(phone_number, 5, 3), ')-(', SUBSTRING(phone_number, 9, 4), ')') AS formatted_phone_number

FROM

   employees;
```

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

```
SELECT first_name, last_name, hire_date

FROM employees

WHERE YEAR(hire_date) = 1994 AND MONTH(hire_date) = 8;
```

#14. Find the maximum salary from each department.

```
SELECT
  department_id,
  MAX(salary) AS max_salary
FROM
  employees
GROUP BY
  department_id;
```

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

```
SELECT
  first_name,
  last_name,
  salary
FROM
  employees
ORDER BY
  salary ASC
LIMIT 5;
```

#16. Find the employees hired in the 80s

```
SELECT
  first_name,
  last_name,
  hire_date
FROM
  employees
WHERE
```

```
  YEAR(hire_date) BETWEEN 1980 AND 1989;
```

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

```
SELECT
  first_name,
  last_name,
  hire_date
FROM
  employees
WHERE
  DAY(hire_date) > 15;
```