1. **Employees**:  
   Contains employee details.

employee\_id | name | department\_id | salary | hire\_date

--------------------------------------------------------------

1 | John Doe | 101 | 80000 | 2015-06-12

2 | Jane Smith | 102 | 95000 | 2017-03-18

3 | Mike Brown | 103 | 75000 | 2019-07-11

4 | Sarah Davis | 101 | 88000 | 2018-01-25

5 | Emma Wilson | 102 | 90000 | 2020-05-03

1. **Departments**:  
   Contains department details.

department\_id | department\_name | location

--------------------------------------------

101 | IT | New York

102 | HR | London

103 | Marketing | Paris

1. **Projects**:  
   Contains project assignments for employees.

project\_id | employee\_id | project\_name

----------------------------------------

201 | 1 | Cloud Migration

202 | 2 | Recruitment Drive

203 | 3 | Marketing Campaign

204 | 4 | Cybersecurity Upgrade

1. **Locations**:  
   Contains office location details.

location\_id | city

--------------------

1 | New York

2 | London

3 | Paris

Single-Row Subquery Questions

1. Find the name of the employee with the highest salary.
2. Get the department name of the employee who has been working the longest.
3. Retrieve the salary of the employee with the earliest hire date.
4. Find the name of the employee assigned to the project "Cybersecurity Upgrade."
5. Display the salary of the employee in the "IT" department with the lowest salary.
6. Find the name of the employee working in the department located in "London."
7. Retrieve the hire date of the employee with the second-highest salary.
8. Find the salary of the employee working on the project "Recruitment Drive."
9. Get the department ID of the employee with the highest salary in "Marketing."
10. Find the name of the employee earning exactly the average salary of the company.
11. Display the location of the department with the least number of employees.
12. Find the name of the employee whose salary is closest to $85,000.
13. Retrieve the project name assigned to the employee with the most recent hire date.
14. Get the hire date of the employee working on the project "Cloud Migration."
15. Find the name of the employee with a salary greater than all employees in the "HR" department.

CREATE TABLE Employee(

employee\_id integer PRIMARY KEY,

name varchar(20),

department\_id integer,

salary integer,

hire\_date date

);

CREATE TABLE Department(

department\_id integer PRIMARY KEY,

department\_name varchar(20),

location varchar(20)

);

CREATE TABLE Projects(

project\_id integer PRIMARY KEY,

employee\_id integer references Employee(employee\_id),

project\_name varchar(30)

);

CREATE TABLE Locations(

location\_id integer PRIMARY KEY,

city varchar(20)

);

INSERT INTO Employee values(1,"John Due",101,80000,"2015-06-12"),(2,"Jane Smith",102,95000,"2017-03-18"),(3,"Mike Brown",103,75000,"2019-07-11"),(4,"Sarah Davis",101,88000,"2018-01-25"),(5,"Emma Wilson",102,90000,"2020-05-03");

INSERT INTO Department values(101,"IT","New York"),(102,"HR","London"),(103,"Marketing","Paris");

INSERT INTO Projects values(201,1,"Cloud Migration"),(202,2,"Recruitment Drive"),(203,3,"Marketing Campaign"),(204,4,"Cybersecurity Upgrade");

INSERT INTO Locations values(1,"New York"),(2,"London"),(3,"Paris");

-- 1rst question

SELECT name FROM Employee WHERE salary=(SELECT max(salary) FROM Employee);

SELECT name FROM Employee ORDER BY salary DESC limit 1;

/\* 2nd \*/

SELECT Department.department\_name

FROM Employee

JOIN Department

ON Employee.department\_id = Department.department\_id

ORDER BY DATEDIFF(CURRENT\_DATE, Employee.hire\_date) DESC

LIMIT 1;

/\* 3rd \*/

SELECT salary FROM Employee ORDER BY hire\_date LIMIT 1;

/\* 4th \*/

SELECT name FROM Employee JOIN Projects ON Employee.employee\_id=Projects.employee\_id

WHERE Projects.project\_name="Cybersecurity Upgrade";

/\* 5th \*/

SELECT salary FROM Employee JOIN Department on Department.department\_id=Employee.department\_id

WHERE Department.department\_name="IT" ORDER BY Employee.salary LIMIT 1;

/\* 6TH \*/

SELECT Employee.name from Employee JOIN Department ON Employee.department\_id=Department.department\_id

where Department.location="London";

/\* 7th\*/

SELECT hire\_date FROM Employee ORDER BY salary LIMIT 1 OFFSET 1;

SELECT hire\_date FROM Employee WHERE salary<(SELECT MAX(salary) FROM Employee) LIMIT 1;

/\* 8TH \*/

SELECT salary from Employee JOIN Projects ON Employee.employee\_id=Projects.employee\_id

WHERE project\_name="Recruitment Drive";

/\* 9TH \*/

SELECT Department.department\_id FROM Department JOIN Employee ON Employee.department\_id=Department.department\_id

WHERE Department.department\_name="Marketing" ORDER BY Employee.salary DESC LIMIT 1;

/\* 10th \*/

(SELECT AVG(salary) FROM Employee);

/\* 11th \*/

SELECT Department.location FROM Department JOIN Employee ON Department.department\_id=Employee.department\_id

GROUP BY Employee.department\_id ORDER BY COUNT(Employee.department\_id) LIMIT 1;

/\* 12TH \*/

SELECT name FROM Employee

ORDER BY ABS(salary-85000)

LIMIT 1;

/\* 13th \*/

SELECT Projects.project\_name FROM Projects JOIN Employee on Employee.employee\_id=Projects.employee\_id

ORDER BY hire\_date DESC limit 1;

/\* 14th \*/

SELECT Employee.hire\_date FROM Employee JOIN Projects ON Projects.employee\_id=Employee.employee\_id

WHERE Projects.project\_name="Cloud Migration";

/\* 15TH \*/

SELECT Employee.name FROM Employee JOIN Department ON Employee.department\_id=Department.department\_id

WHERE Department.department\_name="HR" ORDER BY Employee.salary DESC LIMIT 1;