mysql> show databases;

+--------------------+

| Database |

+--------------------+

| information\_schema |

| leetcode |

| mysql |

| performance\_schema |

| sys |

+--------------------+

5 rows in set (0.00 sec)

mysql> use leecode;

ERROR 1049 (42000): Unknown database 'leecode'

mysql> use leetcode;

Reading table information for completion of table and column names

You can turn off this feature to get a quicker startup with -A

Database changed

mysql> show tables;

+--------------------+

| Tables\_in\_leetcode |

+--------------------+

| Address |

| Customers |

| Department |

| Employee |

| Logs |

| Orders |

| Person |

| Scores |

| Trips |

| Users |

| Weather |

+--------------------+

11 rows in set (0.00 sec)

mysql> select Person.FirstName, Person.LastName, Address.City, Address.State

-> from Person

-> left join Address

-> on Person.PersonId = Address.PersonId;

+-----------+----------+----------+-----------+

| FirstName | LastName | City | State |

+-----------+----------+----------+-----------+

| john | Marsh | New York | New York |

| Yida | Yin | Shanghai | Shanghai |

| Yida | Yin | Madison | Wisconsin |

| john | Campbell | NULL | NULL |

+-----------+----------+----------+-----------+

4 rows in set (0.00 sec)

mysql> select max(Employee.Salary)

-> from Employee

-> where Employee.Salary < (select max(Salary) from Employee);

+----------------------+

| max(Employee.Salary) |

+----------------------+

| 85000 |

+----------------------+

1 row in set (0.00 sec)

mysql> select

-> s.Score,

-> (select count(distinct Score) from Scores where Score >= s.Score) Rank

-> from Scores s

-> order by s.Score Desc;

+-------+------+

| Score | Rank |

+-------+------+

| 4 | 1 |

| 4 | 1 |

| 3.85 | 2 |

| 3.65 | 3 |

| 3.65 | 3 |

| 3.5 | 4 |

+-------+------+

6 rows in set (1.79 sec)

mysql> select distinct l1.Num

-> from Logs l1, Logs l2, Logs l3

-> where l1.Id=l2.Id-1 and l2.Id=l3.Id-1

-> and l1.Num = l2.Num and l2.Num = l3.Num;

+------+

| Num |

+------+

| 1 |

+------+

1 row in set (0.00 sec)

mysql> select e1.Name Employee

-> from Employee e1, Employee e2

-> where e1.ManagerId = e2.Id and e1.Salary > e2.Salary;

+----------+

| Employee |

+----------+

| Joe |

+----------+

1 row in set (0.01 sec)

mysql> SELECT Email FROM Person GROUP BY Email HAVING COUNT(Email) > 1;

+------------------+

| Email |

+------------------+

| john@example.com |

+------------------+

1 row in set (0.00 sec)

mysql> select Customers.Name Customers

-> from Customers

-> left join Orders

-> on Customers.Id = Orders.CustomerId

-> where Orders.Id is null;

+-----------+

| Customers |

+-----------+

| Henry |

| Max |

+-----------+

2 rows in set (0.00 sec)

mysql> select

-> Department.Name Department,

-> Employee.Name Employee,

-> Employee.Salary

-> from

-> Employee,

-> Department,

-> (select max(Salary) mSalary, DepartmentId from Employee group by DepartmentId) temp

-> where

-> Employee.Salary = temp.mSalary and

-> Employee.DepartmentId = temp.DepartmentId and

-> Employee.DepartmentId = Department.Id;

+------------+----------+--------+

| Department | Employee | Salary |

+------------+----------+--------+

| Sales | Henry | 80000 |

| IT | Max | 90000 |

| IT | Cade | 90000 |

+------------+----------+--------+

3 rows in set (0.00 sec)

mysql> select e1.Name, e1.Salary, e1.DepartmentId

-> from Employee as e1

-> where (

-> select count(\*) from Employee as e2

-> where e2.DepartmentId = e1.DepartmentId and e2.Salary >= e1.Salary

-> ) <= 3;

+-------+--------+--------------+

| Name | Salary | DepartmentId |

+-------+--------+--------------+

| Henry | 80000 | 2 |

| Sam | 60000 | 2 |

| Max | 90000 | 1 |

| Randy | 85000 | 1 |

| Cade | 90000 | 1 |

+-------+--------+--------------+

5 rows in set (0.00 sec)

mysql> delete p1

-> from Person p1, Person p2

-> where p1.Email = p2.Email and p1.Id > p2.Id;

Query OK, 1 row affected (0.38 sec)

mysql> select w2.Id

-> from Weather w1, Weather w2

-> where TO\_DAYS(w1.Date) = TO\_DAYS(w2.Date)-1 and w1.Temperature < w2.Temperature;

+----+

| Id |

+----+

| 2 |

| 4 |

+----+

2 rows in set (0.00 sec)

mysql> select total.Request\_at as "Day", round(COALESCE(cancelled.cCount, 0) / total.tCount, 2) as "Cancellation Rate"

-> from

-> ( select count(Trips.Id) tCount, Trips.Request\_at from Trips

-> left join Users on Trips.Client\_Id = Users.Users\_Id

-> where Users.Banned = "No"

-> group by Trips.Request\_at

-> ) total

-> left join

-> ( select count(Trips.Id) cCount, Trips.Request\_at from Trips

-> left join Users on Trips.Client\_Id = Users.Users\_Id

-> where Users.Banned = "No" and Trips.Status <> "completed"

-> group by Trips.Request\_at

-> ) cancelled

-> on total.Request\_at = cancelled.Request\_at

-> where total.Request\_at between '2013-10-01' AND '2013-10-03';

+------------+-------------------+

| Day | Cancellation Rate |

+------------+-------------------+

| 2013-10-01 | 0.33 |

| 2013-10-02 | 0.00 |

| 2013-10-03 | 0.50 |

+------------+-------------------+

3 rows in set (0.00 sec)

mysql>