**Table - EmployeeDetails**

|  |  |  |  |
| --- | --- | --- | --- |
| **EmpId** | **FullName** | **ManagerId** | **DateOfJoining** |
| 121 | John Snow | 321 | 01/31/2014 |
| 321 | Walter White | 986 | 01/30/2015 |
| 421 | Kuldeep Rana | 876 | 27/11/2016 |

**Table - EmployeeSalary**

|  |  |  |
| --- | --- | --- |
| **EmpId** | **Project** | **Salary** |
| 121 | P1 | 8000 |
| 321 | P2 | 1000 |
| 421 | P1 | 12000 |

**Write a SQL query to fetch duplicate records from a table.**

**SELECT** EmpId, Project, Salary, **COUNT**(\*)

**FROM** EmployeeSalary

**GROUP** **BY** EmpId, Project, Salary

**HAVING** **COUNT**(\*) > 1;

**Write a SQL query to remove duplicates from a table without using temporary table.**

**DELETE** **FROM** EmployeeSalary

**WHERE** **EmpId** **IN** (

**SELECT** EmpId

**FROM** EmployeeSalary

**GROUP** **BY** Project, Salary

**HAVING** **COUNT**(\*) > 1));

Using rowId in Oracle-

**DELETE** **FROM** EmployeeSalary

**WHERE** rowid **NOT** **IN**

(**SELECT** **MAX**(rowid) **FROM** EmployeeSalary **GROUP** **BY** EmpId);

**Ques.10. Write a SQL query to fetch only odd rows from table.**  
Ans. This can be achieved by using Row\_number in SQL server-

**SELECT** E.EmpId, E.Project, E.Salary

**FROM** (

**SELECT** \*, Row\_Number() OVER(**ORDER** **BY** EmpId) **AS** RowNumber

**FROM** EmployeeSalary

) E

**WHERE** E.RowNumber % 2 = 1

**Ques.18. Write a SQL query to fetch top n records?**

In Oracle using ROWNUM-

**SELECT** \* **FROM** (**SELECT** \* **FROM** EmployeeSalary **ORDER** **BY** Salary **DESC**)

**WHERE** ROWNUM <= 3;

**Write SQL query to find the 3rd highest salary from table without using TOP/limit keyword.**

**SELECT** Salary

**FROM** EmployeeSalary Emp1

**WHERE** 2 = (

**SELECT** **COUNT**( **DISTINCT** ( Emp2.Salary ) )

**FROM** EmployeeSalary Emp2

**WHERE** Emp2.Salary > Emp1.Salary

)

For nth highest salary-

**SELECT** Salary

**FROM** EmployeeSalary Emp1

**WHERE** N-1 = (

**SELECT** **COUNT**( **DISTINCT** ( Emp2.Salary ) )

**FROM** EmployeeSalary Emp2

**WHERE** Emp2.Salary > Emp1.Salary

)

 Write An SQL Query To Fetch The List Of Employees With The Same Salary.

**Ans.**

The required query is:

Select distinct W.WORKER\_ID, W.FIRST\_NAME, W.Salary

from Worker W, Worker W1

where W.Salary = W1.Salary

and W.WORKER\_ID != W1.WORKER\_ID;

Q-25. Write An SQL Query To Fetch Duplicate Records Having Matching Data In Some Fields Of A Table.

**Ans.**

The required query is:

SELECT WORKER\_TITLE, AFFECTED\_FROM, COUNT(\*)

FROM Title

GROUP BY WORKER\_TITLE, AFFECTED\_FROM

HAVING COUNT(\*) > 1;

Given the table mass\_table:

| **weight** |
| --- |
| 5.67 |
| 34.567 |
| 365.253 |
| 34 |

Write a query that produces the output:

| **weight** | **kg** | **gms** |
| --- | --- | --- |
| 5.67 | 5 | 67 |
| 34.567 | 34 | 567 |

select weight, trunc(weight) as kg, nvl(substr(weight - trunc(weight), 2), 0) as gms

from mass\_table;

Given the following table named A:

x

------

2

-2

4

-4

-3

0

2

Write a single query to calculate the sum of all positive values of x and he sum of all negative values of x

select sum(case when x>0 then x else 0 end)sum\_pos,sum(case when x<0 then x else 0 end)sum\_neg from a;

select \* from test\_a

except

select \* from test\_b;

We can write the above statement in other way like the below

select a.id

from test\_a a

left join test\_b b on a.id = b.id

where b.id is null;

Imagine a single column in a table that is populated with either a single digit (0-9) or a single character (a-z, A-Z). Write a SQL query to print ‘Fizz’ for a numeric value or ‘Buzz’ for alphabetical value for all values in that column.

Example:

['d', 'x', 'T', 8, 'a', 9, 6, 2, 'V']

…should output:

['Buzz', 'Buzz', 'Buzz', 'Fizz', 'Buzz','Fizz', 'Fizz', 'Fizz', 'Buzz']

Hide answer



SELECT col, case when upper(col) = lower(col) then 'Fizz' else 'Buzz' end as FizzBuzz from table;