Week 02: SQL Practice Tasks

Online IDE for practice: <http://www.sqlfiddle.com/>

Practice document: <https://github.com/NYU-DataScienceBootCamp/Week-2-SQL/blob/main/SQL_Practice.pdf>

|  |
| --- |
| **NOTE:** Make sure you answer the queries in the boxes given and paste screenshots in the output box.  **The solution queries will be posted on June 24th before the session** |

Input Data

Use the database which was discussed during the session and feel free to change the attributes of the tables. Make sure that the following conditions are satisfied:

* There are three “tables”. One for storing Employee Details, One for Bonus, and One for Employee Title.
* There are at least 12 employees in the table which stores Employee Details.

NOTE: Make sure that you paste your input data in the box given below

|  |
| --- |
|  |

Tasks

SELECTing data

* Display the entire table containing the details of all the Employees  
    
  **QUERY:**

|  |
| --- |
| select \* from Employee |

**OUTPUT:**

|  |
| --- |
| EMPLOYEE\_ID FIRST\_NAME LAST\_NAME SALARY JOINING\_DATE DEPARTMENT  1 Neville Longbottom 100000 2014-02-20T09:00:00Z HR  2 Ronald Weasley 80000 2014-06-11T09:00:00Z Admin  3 Hermoine Granger 300000 2014-02-20T09:00:00Z HR  4 Harry Potter 500000 2014-02-20T09:00:00Z Admin  5 Severus Snape 500000 2014-06-11T09:00:00Z Admin  6 Luna Lovegood 200000 2014-06-11T09:00:00Z Account  7 Draco Malfoy 75000 2014-01-20T09:00:00Z Account  8 Minerva Mcgonagall 90000 2014-04-11T09:00:00Z Admin |

* Write a query to fetch “FIRST\_NAME” from the Employees table in the UPPER CASE  
    
  **QUERY:**

|  |
| --- |
| select upper(first\_name) AS Name from EMployee |

**OUTPUT:**

|  |
| --- |
| Name  NEVILLE  RONALD  HERMOINE  HARRY  SEVERUS  LUNA  DRACO  MINERVA |

GROUPing them together

* Write a query to fetch the number of Employees for each department in the descending order  
    
  **QUERY:**

|  |
| --- |
| SELECT department, COUNT(\*) FROM employee GROUP BY department ORDER BY COUNT(\*) DESC; |

**OUTPUT:**

|  |
| --- |
| department COUNT(\*)  Admin 4  HR 2  Account 2 |

Using WHERE somewhere

* Write a query to fetch the names of the Employees with salaries >= 90000 and <= 200000  
    
  **QUERY:**

|  |
| --- |
| SELECT CONCAT(first\_NAME, last\_NAME) AS Name FROM employee WHERE 20000< salary and salary < 90000 |

**OUTPUT:**

|  |
| --- |
| Name  RonaldWeasley  DracoMalfoy |

JOINing the tables

* Write a query to print details of Employees who are also “Managers”  
    
  **QUERY:**

|  |
| --- |
| SELECT \* FROM Employee AS E, TITLE AS T  WHERE E.EMPLOYEE\_ID = T.EMPLOYEE\_REF\_ID  AND T.EMPLOYEE\_TITLE = 'Manager' |

**OUTPUT:**

|  |
| --- |
| EMPLOYEE\_ID FIRST\_NAME LAST\_NAME SALARY JOINING\_DATE DEPARTMENT EMPLOYEE\_REF\_ID EMPLOYEE\_TITLE AFFECTED\_FROM  1 Neville Longbottom 100000 2014-02-20T09:00:00Z HR 1 Manager 2016-02-20T00:00:00Z  5 Severus Snape 500000 2014-06-11T09:00:00Z Admin 5 Manager 2016-06-11T00:00:00Z |

COPYing

* Write an SQL query to clone a new table from another table  
    
  **QUERY:**

|  |
| --- |
| CREATE TABLE new\_table SELECT \* FROM employee;  SELECT \* from new\_table |

**OUTPUT:**

|  |
| --- |
| EMPLOYEE\_ID FIRST\_NAME LAST\_NAME SALARY JOINING\_DATE DEPARTMENT  1 Neville Longbottom 100000 2014-02-20T09:00:00Z HR  2 Ronald Weasley 80000 2014-06-11T09:00:00Z Admin  3 Hermoine Granger 300000 2014-02-20T09:00:00Z HR  4 Harry Potter 500000 2014-02-20T09:00:00Z Admin  5 Severus Snape 500000 2014-06-11T09:00:00Z Admin  6 Luna Lovegood 200000 2014-06-11T09:00:00Z Account  7 Draco Malfoy 75000 2014-01-20T09:00:00Z Account  8 Minerva Mcgonagall 90000 2014-04-11T09:00:00Z Admin |

Aliasing

* Find the average salary of employees in each department and name the AVG(SALARY) column as “AverageSalary”  
    
  **QUERY:**

|  |
| --- |
| SELECT AVG(salary) AS Averagesalary from employee GROUP BY department |

**OUTPUT:**

|  |
| --- |
| Averagesalary  137500  292500  200000 |

Some other stuff

* Write an SQL query to show the second-highest salary from a table  
    
  **QUERY:**

|  |
| --- |
| SELECT MAX(salary)  FROM employee where salary < ( SELECT MAX(salary) from employee) |

**OUTPUT:**

|  |
| --- |
| 300000 |

* Write an SQL query to show one row twice in results from a table

**QUERY:**

|  |
| --- |
| select \*  from employee t1 cross join  (select 1 as n union all select 2) n; |

**OUTPUT:**

|  |
| --- |
| EMPLOYEE\_ID FIRST\_NAME LAST\_NAME SALARY JOINING\_DATE DEPARTMENT n  1 Neville Longbottom 100000 2014-02-20T09:00:00Z HR 1  1 Neville Longbottom 100000 2014-02-20T09:00:00Z HR 2  2 Ronald Weasley 80000 2014-06-11T09:00:00Z Admin 1  2 Ronald Weasley 80000 2014-06-11T09:00:00Z Admin 2  3 Hermoine Granger 300000 2014-02-20T09:00:00Z HR 1  3 Hermoine Granger 300000 2014-02-20T09:00:00Z HR 2  4 Harry Potter 500000 2014-02-20T09:00:00Z Admin 1  4 Harry Potter 500000 2014-02-20T09:00:00Z Admin 2  5 Severus Snape 500000 2014-06-11T09:00:00Z Admin 1  5 Severus Snape 500000 2014-06-11T09:00:00Z Admin 2  6 Luna Lovegood 200000 2014-06-11T09:00:00Z Account 1  6 Luna Lovegood 200000 2014-06-11T09:00:00Z Account 2  7 Draco Malfoy 75000 2014-01-20T09:00:00Z Account 1  7 Draco Malfoy 75000 2014-01-20T09:00:00Z Account 2  8 Minerva Mcgonagall 90000 2014-04-11T09:00:00Z Admin 1  8 Minerva Mcgonagall 90000 2014-04-11T09:00:00Z Admin 2 |

* Write an SQL query to fetch the departments that have less than five people in it  
    
  **QUERY:**

|  |
| --- |
|  |

**OUTPUT:**

|  |
| --- |
|  |

* Write an SQL query to fetch the last five records from a table  
    
  **QUERY:**

|  |
| --- |
| (SELECT \* FROM Employee ORDER BY employee\_ID DESC LIMIT 5)  ORDER BY Employee\_id ASC; |

**OUTPUT:**

|  |
| --- |
| EMPLOYEE\_ID FIRST\_NAME LAST\_NAME SALARY JOINING\_DATE DEPARTMENT  4 Harry Potter 500000 2014-02-20 09:00:00 Admin  5 Severus Snape 500000 2014-06-11 09:00:00 Admin  6 Luna Lovegood 200000 2014-06-11 09:00:00 Account  7 Draco Malfoy 75000 2014-01-20 09:00:00 Account  8 Minerva Mcgonagall 90000 2014-04-11 09:00:00 Admin |