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

|  |
| --- |
| CREATE TABLE Employee (  EMPLOYEE\_ID INT NOT NULL PRIMARY KEY AUTO\_INCREMENT,  FIRST\_NAME CHAR(25),  LAST\_NAME CHAR(25),  SALARY INT(15),  JOINING\_DATE DATETIME,  DEPARTMENT CHAR(25)  );  INSERT INTO Employee  (EMPLOYEE\_ID, FIRST\_NAME, LAST\_NAME, SALARY, JOINING\_DATE,  DEPARTMENT) VALUES  (001, "Neville", 'Longbottom', 100000, '14-02-20 09.00.00', 'HR'),  (002, 'Ronald', 'Weasley', 80000, '14-06-11 09.00.00', 'Admin'),  (003, 'Hermione', 'Granger', 300000, '14-02-20 09.00.00', 'HR'),  (004, 'Harry', 'Potter', 500000, '14-02-20 09.00.00', 'Admin'),  (005, 'Severus', 'Snape', 500000, '14-06-11 09.00.00', 'Admin'),  (006, 'Luna', 'Lovegood', 200000, '14-06-11 09.00.00', 'Account'),  (007, 'Draco', 'Malfoy', 75000, '14-01-20 09.00.00', 'Account'),  (008, 'Dean', 'Thomas', 250000, '14-08-05 09.00.00', 'Account'),  (009, 'Remus', 'Lupin', 95000, '14-11-16 09.00.00', 'HR'),  (010, 'Nymphadora', 'Tonks', 130000, '14-07-30 09.00.00', 'Admin'),  (011, 'Albus', 'Dumbledore', 700000, '14-03-13 09.00.00', 'HR'),  (012, 'Minerva', 'Mcgonagall', 90000, '14-04-11 09.00.00', 'Admin');  CREATE TABLE Bonus (  EMPLOYEE\_REF\_ID INT,  BONUS\_AMOUNT INT(10),  BONUS\_DATE DATETIME,  FOREIGN KEY (EMPLOYEE\_REF\_ID)  REFERENCES Employee(EMPLOYEE\_ID)  ON DELETE CASCADE  );  INSERT INTO Bonus  (EMPLOYEE\_REF\_ID, BONUS\_AMOUNT, BONUS\_DATE) VALUES  (001, 5000, '16-02-20'),  (002, 3000, '16-06-11'),  (003, 4000, '16-02-20'),  (001, 4500, '16-02-20'),  (002, 3500, '16-06-11'),  (006, 5000, '16-02-20'),  (007, 3000, '16-06-11'),  (008, 4000, '16-02-20'),  (011, 4500, '16-02-20'),  (012, 3500, '16-06-11');  CREATE TABLE Title (  EMPLOYEE\_REF\_ID INT,  EMPLOYEE\_TITLE CHAR(25),  AFFECTED\_FROM DATETIME,  FOREIGN KEY (EMPLOYEE\_REF\_ID)  REFERENCES Employee(EMPLOYEE\_ID)  ON DELETE CASCADE  );  INSERT INTO Title  (EMPLOYEE\_REF\_ID, EMPLOYEE\_TITLE, AFFECTED\_FROM) VALUES  (001, 'Manager', '2016-02-20 00:00:00'),  (002, 'Executive', '2016-06-11 00:00:00'),  (008, 'Executive', '2016-06-11 00:00:00'),  (005, 'Manager', '2016-06-11 00:00:00'),  (004, 'Assistant Manager', '2016-06-11 00:00:00'),  (007, 'Executive', '2016-06-11 00:00:00'),  (006, 'Lead', '2016-06-11 00:00:00'),  (003, 'Lead', '2016-06-11 00:00:00'),  (009, 'Manager', '2016-06-11 00:00:00'),  (010, 'Assistant Manager', '2016-06-11 00:00:00'),  (011, 'Executive', '2016-06-11 00:00:00'),  (012, 'Lead', '2016-06-11 00:00:00'); |

# 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 | Hermione | 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 | Dean | Thomas | 250000 | 2014-08-05T09:00:00Z | Account |
| 9 | Remus | Lupin | 95000 | 2014-11-16T09:00:00Z | HR |
| 10 | Nymphadora | Tonks | 130000 | 2014-07-30T09:00:00Z | Admin |
| 11 | Albus | Dumbledore | 700000 | 2014-03-13T09:00:00Z | HR |
| 12 | 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 UPCASE\_FIRST\_NAME  from employee  ; |

**OUTPUT:**

|  |
| --- |
| **UPCASE\_FIRST\_NAME** |
| NEVILLE |
| RONALD |
| HERMIONE |
| HARRY |
| SEVERUS |
| LUNA |
| DRACO |
| DEAN |
| REMUS |
| NYMPHADORA |
| ALBUS |
| MINERVA |

## GROUPing them together

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

|  |
| --- |
| select DEPARTMENT, count(employee\_id) as EMPLOYEE\_COUNT  from employee  group by department  order by employee\_count desc  ; |

**OUTPUT:**

|  |  |
| --- | --- |
| **DEPARTMENT** | **EMPLOYEE\_COUNT** |
| Admin | 5 |
| HR | 4 |
| Account | 3 |

## Using WHERE somewhere

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

|  |
| --- |
| select FIRST\_NAME, LAST\_NAME  from employee  where salary >= 90000 and salary <= 200000  ; |

**OUTPUT:**

|  |  |
| --- | --- |
| **FIRST\_NAME** | **LAST\_NAME** |
| Neville | Longbottom |
| Luna | Lovegood |
| Remus | Lupin |
| Nymphadora | Tonks |
| Minerva | Mcgonagall |

## JOINing the tables

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

|  |
| --- |
| select a.\*  from employee a  left join title b  on a.employee\_id = b.employee\_ref\_id  where b.employee\_title = "Manager"  ; |

**OUTPUT:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **EMPLOYEE\_ID** | **FIRST\_NAME** | **LAST\_NAME** | **SALARY** | **JOINING\_DATE** | **DEPARTMENT** |
| 1 | Neville | Longbottom | 100000 | 2014-02-20T09:00:00Z | HR |
| 5 | Severus | Snape | 500000 | 2014-06-11T09:00:00Z | Admin |
| 9 | Remus | Lupin | 95000 | 2014-11-16T09:00:00Z | HR |

## COPYing

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

|  |
| --- |
| CREATE TABLE employees\_clone as  select \*  from employee  ;  select \* from employees\_clone  ; |

**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 | Hermione | 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 | Dean | Thomas | 250000 | 2014-08-05T09:00:00Z | Account |
| 9 | Remus | Lupin | 95000 | 2014-11-16T09:00:00Z | HR |
| 10 | Nymphadora | Tonks | 130000 | 2014-07-30T09:00:00Z | Admin |
| 11 | Albus | Dumbledore | 700000 | 2014-03-13T09:00:00Z | HR |
| 12 | 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 DEPARTMENT, avg(salary) as AverageSalary  from employee  group by department  ; |

**OUTPUT:**

|  |  |
| --- | --- |
| **DEPARTMENT** | **AverageSalary** |
| Account | 175000 |
| Admin | 260000 |
| HR | 298750 |

## Some other stuff

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

|  |
| --- |
| select max(salary) as SECOND\_HIGHEST\_SALARY  from employee  where salary < (select max(salary) as MAX\_SALARY from employee)  ; |

**OUTPUT:**

|  |
| --- |
| **SECOND\_HIGHEST\_SALARY** |
| 500000 |

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

**QUERY:**

|  |
| --- |
| select \*  from employee  where employee\_id = "001"  union all  select \*  from employee  where employee\_id = "001"  ; |

**OUTPUT:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **EMPLOYEE\_ID** | **FIRST\_NAME** | **LAST\_NAME** | **SALARY** | **JOINING\_DATE** | **DEPARTMENT** |
| 1 | Neville | Longbottom | 100000 | 2014-02-20T09:00:00Z | HR |
| 1 | Neville | Longbottom | 100000 | 2014-02-20T09:00:00Z | HR |

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

|  |
| --- |
| select distinct a.DEPARTMENT  from Employee a  inner join (select DEPARTMENT, count(Employee\_ID) as Employee\_Count  from Employee  group by DEPARTMENT  having Employee\_Count < 5) b  on a.DEPARTMENT = b.DEPARTMENT  ; |

**OUTPUT:**

|  |
| --- |
| **DEPARTMENT** |
| HR |
| Account |

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

|  |
| --- |
| select \*  from employee  limit 5 offset 7  ; |

**OUTPUT:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **EMPLOYEE\_ID** | **FIRST\_NAME** | **LAST\_NAME** | **SALARY** | **JOINING\_DATE** | **DEPARTMENT** |
| 8 | Dean | Thomas | 250000 | 2014-08-05T09:00:00Z | Account |
| 9 | Remus | Lupin | 95000 | 2014-11-16T09:00:00Z | HR |
| 10 | Nymphadora | Tonks | 130000 | 2014-07-30T09:00:00Z | Admin |
| 11 | Albus | Dumbledore | 700000 | 2014-03-13T09:00:00Z | HR |
| 12 | Minerva | Mcgonagall | 90000 | 2014-04-11T09:00:00Z | Admin |