DBMS ASSIGNMENT – 4

AGGREGATE FUNCTIONS

***Roll Number: U19CS012***

***Name: BHAGYA VINOD RANA***

**Q1)** Create a table **Employee** with fields

EmpID Number (6) Primary key

Name Character (25)

Department Character (30)

Manager ID Number (6)

JoiningDate Date

Salary Number (8)

Insert 15 Rows in the above created table.

**SQL-Code [**SQLite 3.29.0**]:**

BEGIN TRANSACTION;

CREATE TABLE EMPLOYEE(

    emp\_id integer PRIMARY KEY,

    emp\_name text,

    department text,

    manager\_id integer,

*-- YEAR MONTH DAY [Important Mistake!]*

    joining\_date DATE,

    salary integer

);

*-- Insert 10 Rows in the above created table.*

INSERT INTO EMPLOYEE VALUES(

    4123,

    'Ninja\_Hatori',

    'Production',

    1002,

    '2020-04-01',

    65000

);

INSERT INTO EMPLOYEE VALUES(

    4129,

    'Ajay',

    'Research',

    1027,

    '2018-04-02',

    45000

);

INSERT INTO EMPLOYEE VALUES(

    4230,

    'Mickey',

    'Marketing',

    1022,

    '2016-04-03',

    35000

);

INSERT INTO EMPLOYEE VALUES(

    4428,

    'Kiteretsu',

    'Accounting',

    1012,

    '2019-04-04',

    75000

);

INSERT INTO EMPLOYEE VALUES(

    4073,

    'Shizuka',

    'HR',

    1035,

    '2020-04-05',

    60000

);

INSERT INTO EMPLOYEE VALUES(

    4983,

    'Aditya',

    'HR',

    1035,

    '2017-04-06',

    100000

);

INSERT INTO EMPLOYEE VALUES(

    4009,

    'Nobita',

    'Research',

    1027,

    '2015-04-07',

    50000

);

INSERT INTO EMPLOYEE VALUES(

    4773,

    'Doraemon',

    'Marketing',

    1022,

    '2020-04-08',

    25000

);

INSERT INTO EMPLOYEE VALUES(

    4833,

    'Gian',

    'Accounting',

    1012,

    '2018-04-09',

    95000

);

INSERT INTO EMPLOYEE VALUES(

    4337,

    'Donald',

    'HR',

    1035,

    '2012-04-10',

    55000

);

INSERT INTO EMPLOYEE VALUES(

    4113,

    'Akash',

    'HR',

    1035,

    '2017-04-06',

    110000

);

INSERT INTO EMPLOYEE VALUES(

    4010,

    'Naruto',

    'HR',

    1027,

    '2013-05-09',

    52000

);

INSERT INTO EMPLOYEE VALUES(

    4768,

    'Dishant',

    'Marketing',

    1022,

    '2019-08-02',

    35000

);

INSERT INTO EMPLOYEE VALUES(

    4830,

    'Gopal',

    'Marketing',

    1012,

    '2020-08-01',

    83000

);

INSERT INTO EMPLOYEE VALUES(

    4331,

    'Deepak',

    'Marketing',

    1022,

    '2014-02-14',

    57000

);

*-- Saving the Work*

COMMIT;

*-- For Checking the Inserted Values*

*-- SELECT \* FROM EMPLOYEE*

*-- 1. Display Manager Id of employees whose name starts with ‘A’.*

SELECT manager\_id FROM EMPLOYEE WHERE emp\_name LIKE 'A%'

*-- For Cross Checking*

*-- SELECT emp\_name,manager\_id FROM EMPLOYEE WHERE emp\_name LIKE 'A%'*

*-- 2. Display employees Id and employee name, department wise.*

SELECT emp\_id,emp\_name FROM EMPLOYEE ORDER BY department

*-- For Cross Checking*

*-- SELECT department,emp\_id,emp\_name FROM EMPLOYEE ORDER BY department*

*-- 3. Display employee count department wise.*

SELECT department,COUNT(\*) FROM EMPLOYEE GROUP BY department

*-- 4. Display all columns of employees whose experience is more than 3 years.*

SELECT \* FROM EMPLOYEE WHERE (joining\_date)<(DATE('now','-3 year'))

*-- 5. Display departments with more than 4 employees.*

SELECT department FROM EMPLOYEE GROUP BY department HAVING COUNT(\*)>4

*-- For Cross Checking*

*-- SELECT department,COUNT(\*) FROM EMPLOYEE GROUP BY department HAVING COUNT(\*)>4*

*-- 6. Display employees Id and employee name whose salary is greater than 50000, department wise.*

SELECT emp\_id, emp\_name FROM EMPLOYEE GROUP BY department HAVING salary>50000

*-- For Checking*

*-- SELECT emp\_id, emp\_name, salary FROM EMPLOYEE GROUP BY department HAVING salary>50000*

*-- 7. Display department name and average salary of employees in department wise.*

SELECT department,AVG(salary) FROM EMPLOYEE GROUP BY department

*-- 8. Display Employee Id and Name of employee with highest salary.*

SELECT emp\_id, emp\_name, MAX(salary) FROM EMPLOYEE

*-- 9. Display employees Id and employee name with least salary*

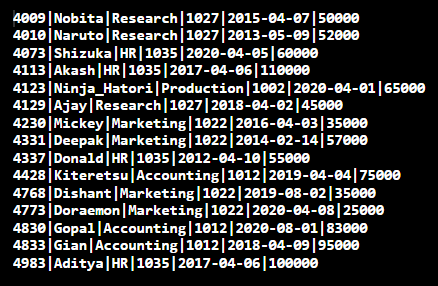
SELECT emp\_id, emp\_name, MIN(salary) FROM EMPLOYEE

*-- 10. Display employees Id and employee name with second highest salary.*

SELECT emp\_id, emp\_name, MAX(salary) FROM EMPLOYEE WHERE salary < (SELECT MAX(salary) FROM EMPLOYEE)

**Use Employee table from Assignment 3**

**Initial Table:**



1. Display Manager Id of employees whose name starts with ‘A’.

**Query:**

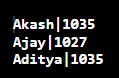
**Q1)**

SELECT manager\_id FROM EMPLOYEE WHERE emp\_name LIKE 'A%'

**Q1) For Checking**

SELECT emp\_name,manager\_id FROM EMPLOYEE WHERE emp\_name LIKE 'A%'

**Output:**



2. Display employees Id and employee name, department wise.

**Query:**

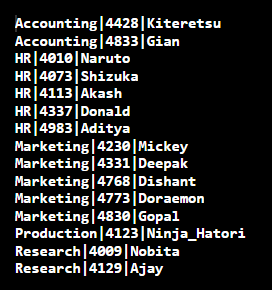
**Q2)**

SELECT emp\_id,emp\_name FROM EMPLOYEE ORDER BY department

**Q2) For Checking**

SELECT department,emp\_id,emp\_name FROM EMPLOYEE ORDER BY department

**Output:**

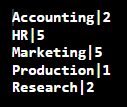


3. Display employee count department wise.

**Query:**

SELECT department,COUNT(\*) FROM EMPLOYEE GROUP BY department

**Output:**

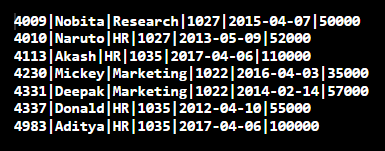


4. Display all columns of employees whose experience is more than 3 years.

**Query:**

SELECT \* FROM EMPLOYEE WHERE (joining\_date)<(DATE('now','-3 year'))

**Output:**



5. Display departments with more than 4 employees.

**Query:**

**Q5)**

SELECT department FROM EMPLOYEE GROUP BY department HAVING COUNT(\*)>4

**Q5) For Checking**

SELECT department,COUNT(\*) FROM EMPLOYEE GROUP BY department HAVING COUNT(\*)>4

**Output:**



6. Display employees Id and employee name whose salary is greater than 50000, Department wise.

**Query:**

**Q6)**

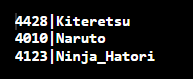
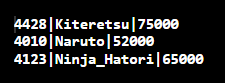
SELECT emp\_id, emp\_name FROM EMPLOYEE GROUP BY department HAVING salary>50000

**Q6) For Checking**

SELECT emp\_id, emp\_name, salary FROM EMPLOYEE

GROUP BY department HAVING salary>50000

**Output:**

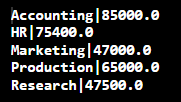
 

7. Display department name and average salary of employees in department wise.

**Query:**

SELECT department,AVG(salary) FROM EMPLOYEE GROUP BY department

**Output:**



8. Display Employee Id and Name of employee with highest salary.

**Query:**

SELECT emp\_id, emp\_name, MAX(salary) FROM EMPLOYEE

**Output:**



9. Display employees Id and employee name with least salary

**Query:**

SELECT emp\_id, emp\_name, MIN(salary) FROM EMPLOYEE

**Output:**



10. Display employees Id and employee name with second highest salary.

**Query:**

SELECT emp\_id, emp\_name, MAX(salary) FROM EMPLOYEE

WHERE salary < (SELECT MAX(salary) FROM EMPLOYEE)

**Output:**



**Use Student table from Assignment 3**

1. Display semester of students whose name has the letter ‘a’.

2. Display count of students semester wise.

3. Display students' names from every department whose roll number is 1.

4. Display student name and semester of students who are not staying in the hostel.

5. Display student count in each semester whose birth month is august.

6. Display roll number and name of the student who was the first one to get admission in

the college.

7. Display the average count of students. ( In any semester)

8. For every month (Jan-Dec) display the count of students who are having birthdays in that

month.

9. Display count of students who have taken admission in the last six months.

10. Display semester with least number of students.

**Submitted By:**

**BHAGYA VINOD RANA**

**U19CS012**