DATABASE MANAGEMENT SYSTEMS SQL – PRACTICE 2

NAME: HARINI.R

REGISTER NO:192324108

COURSE CODE: CSA0564

SELECT MAX(salary)

FROM Employee;

```
SELECT MAX(salary) FROM Employee
```

MAX(SALARY)
75000

Statement processed. 0.00 seconds

SELECT MIN(salary)

FROM Employee;

```
SELECT MIN(salary) FROM Employee
```

MIN(SALARY)
23000

Statement processed. 0.01 seconds

SELECT SUM(salary)

FROM Employee

WHERE emp_id >105;

```
SELECT SUM(salary) FROM Employee WHERE emp_id >105
```

SUM(SALARY)
466000

Statement processed. 0.01 seconds

SELECT AVG(salary)

FROM Employee

WHERE emp_id <105;

```
SELECT AVG(salary) FROM Employee WHERE emp_id <105
```

AVG(SALARY)
49500

Statement processed. 0.01 seconds

SELECT ROUND(VARIANCE(salary),10)

FROM Employee;

```
SELECT ROUND(VARIANCE(salary),10) FROM Employee
```

ROUND(VARIANCE(SALARY),10)
184638095.2380952381

Statement processed. 0.01 seconds

SELECT ROUND(STDDEV(salary),10)

FROM Employee;

```
SELECT ROUND(STDDEV(salary),10) FROM Employee
```

ROUND(STDDEV(SALARY),10)
13588.1601123219

Statement processed. 0.00 seconds

SELECT firstName, MAX(salary)

FROM Employee

WHERE emp_id>105

GROUP BY firstName;

```
SELECT firstName, MAX(salary) FROM Employee WHERE emp_id>105 GROUP BY firstName
```

FIRSTNAME	MAX(SALARY)
Hannah	43000
Ernie	40000
Blaise	62000
Theodore	61000
Cho	25000
Susan	40000
Justin	23000
Pansy	53000
Draco	75000
Dean	44000

10 rows selected. 0.00 seconds

SELECT MIN(salary), MAX(salary)

FROM Employee;

```
SELECT MIN(salary), MAX(salary) FROM Employee
```

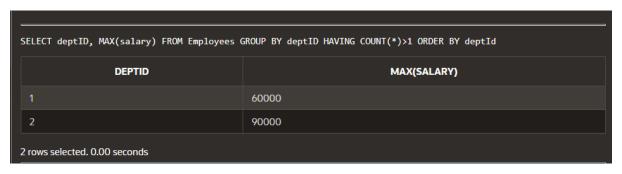| MIN(SALARY)	MAX(SALARY)
23000	75000

Statement processed. 0.00 seconds

SELECT deptID, MAX(salary)

FROM Employees

GROUP BY deptID

HAVING COUNT(*)>1

ORDER BY deptId;

```
SELECT deptID, MAX(salary) FROM Employees GROUP BY deptID HAVING COUNT(*)>1 ORDER BY deptId
```

DEPTID	MAX(SALARY)
1	60000
2	90000

2 rows selected. 0.00 seconds

SELECT COUNT(deptID)

FROM Employees;

```
SELECT COUNT(deptID) FROM Employees
```

COUNT(DEPTID)
5

Statement processed. 0.00 seconds

SELECT COUNT(*)

FROM Employees

WHERE salary>60000;

```
SELECT COUNT(*) FROM Employees WHERE salary>60000
```

COUNT(*)
3

Statement processed. 0.00 seconds

SELECT DISTINCT(salary)

FROM Employee;

```
SELECT DISTINCT(salary) FROM Employee
```

SALARY
55000
75000
45000
62000
53000
61000
43000
40000
44000
25000
23000
50000

12 rows selected. 0.00 seconds

SELECT SUM (DISTINCT salary)

FROM Employee

WHERE emp_id<105;

```
SELECT SUM (DISTINCT salary) FROM Employee WHERE emp_id<105
```

SUM(DISTINCTSALARY)
198000

Statement processed. 0.00 seconds

SELECT AVG(NVL(salary,0))

FROM Employee;

```
SELECT AVG(NVL(salary,0)) FROM Employee
```

AVG(NVL(SALARY,0))
47266.6666666666666666666666666666666667

Statement processed. 0.00 seconds

SELECT lastName, MAX(salary)

FROM Employee

GROUP BY lastName

ORDER BY lastName;

```
SELECT lastName, MAX(salary) FROM Employee GROUP BY lastName ORDER BY lastName
```

LASTNAME	MAX(SALARY)
Abbot	43000
Bones	40000
Chang	25000
Flenchy	23000
Granger	55000
Longbottom	53000
Lovegoog	45000
Malfoy	75000
McMillan	40000
Nott	61000
Parkinson	53000
Potter	50000
Thomas	44000
Weasley	40000
Zabini	62000

15 rows selected. 0.00 seconds

SELECT lastName, COUNT(salary)

FROM Employee

GROUP BY lastName

ORDER BY lastName;

```
SELECT lastName, COUNT(salary) FROM Employee GROUP BY lastName ORDER BY lastName
```

LASTNAME	COUNT(SALARY)
Abbot	1
Bones	1
Chang	1
Flenchy	1
Granger	1
Longbottom	1
Lovegoog	1
Malfoy	1
McMillan	1
Nott	1
Parkinson	1
Potter	1
Thomas	1
Weasley	1
Zabini	1

15 rows selected. 0.01 seconds

SELECT emp_id, COUNT(salary)

FROM Employee

WHERE lastName!= 'Parkinson'

GROUP BY emp_id

ORDER BY emp_id;

```
SELECT emp_id, COUNT(salary) FROM Employee WHERE lastName!= 'Parkinson' GROUP BY emp_id ORDER BY emp_id
```

EMP_ID	COUNT(SALARY)
101	1
102	1
103	1
104	1
105	1
106	1
107	1
108	1
110	1
111	1
112	1
113	1
114	1
115	1

14 rows selected. 0.00 seconds

SELECT firstName, ROUND(AVG (salary)) AS "average salary"

FROM Employee

GROUP BY firstName

ORDER BY firstName;

```
SELECT firstName, ROUND(AVG (salary)) AS "average salary" FROM Employee GROUP BY firstName ORDER BY firstName
```

FIRSTNAME	average salary
Blaise	62000
Cho	25000
Dean	44000
Draco	75000
Ernie	40000
Hannah	43000
Harry	50000
Hermione	55000
Justin	23000
Luna	45000
Neville	53000
Pansy	53000
Ron	40000
Susan	40000
Theodore	61000

15 rows selected. 0.01 seconds

SELECT firstName, emp_id, COUNT(salary) AS "average salary"

FROM Employee

WHERE emp_id>107

GROUP BY firstName, emp_id;

```
SELECT firstName, emp_id, COUNT(salary) AS "average salary" FROM Employee WHERE emp_id>107 GROUP BY firstName, emp_id
```

FIRSTNAME	EMP_ID	average salary
Pansy	109	1
Susan	111	1
Dean	112	1
Theodore	108	1
Ernie	113	1
Hannah	110	1
Justin	115	1
Cho	114	1

8 rows selected. 0.00 seconds

SELECT firstName,

COUNT(salary) AS "average salary"

FROM Employee

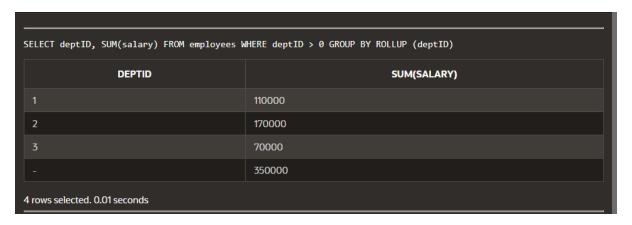GROUP BY firstName

HAVING MIN(salary)>60000

ORDER BY firstName;

```
SELECT firstName, COUNT(salary) AS "average salary" FROM Employee GROUP BY firstName HAVING MIN(salary)>60000
ORDER BY firstName
```

FIRSTNAME	average salary
Blaise	1
Draco	1
Theodore	1

3 rows selected. 0.00 seconds

SELECT deptID, SUM(salary)

FROM employees

WHERE deptID > 0

GROUP BY ROLLUP (deptID);

```
SELECT deptID, SUM(salary) FROM employees WHERE deptID > 0 GROUP BY ROLLUP (deptID)
```

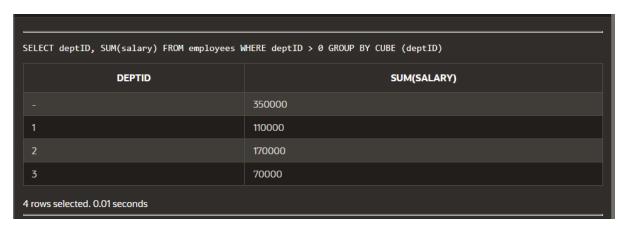| DEPTID	SUM(SALARY)
1	110000
2	170000
3	70000
-	350000

4 rows selected. 0.01 seconds

SELECT deptID, SUM(salary)

FROM employees

WHERE deptID > 0

GROUP BY CUBE (deptID);

```
SELECT deptID, SUM(salary) FROM employees WHERE deptID > 0 GROUP BY CUBE (deptID)
```

DEPTID	SUM(SALARY)
-	350000
1	110000
2	170000
3	70000

4 rows selected. 0.01 seconds

SELECT dept_Id, emp_id, lastName, SUM(salary)

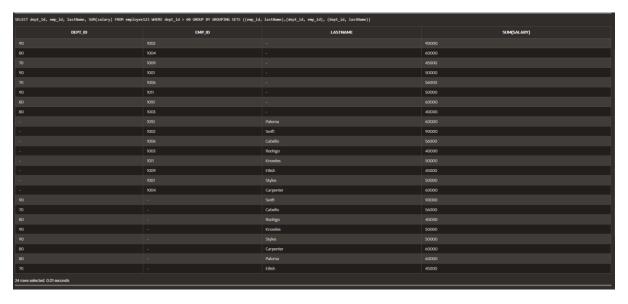FROM employee123

WHERE dept_id > 60

GROUP BY GROUPING SETS

((emp_id, lastName),(dept_id, emp_id),
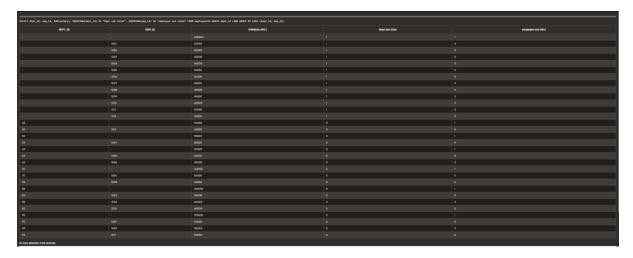
(dept_id, lastName));

```
SELECT dept_Id, emp_id, lastName, SUM(salary) FROM employee123 WHERE dept_id > 60 GROUP BY GROUPING SETS ((emp_id, lastName),(dept_id, emp_id), (dept_id, lastName))
```

DEPT_ID	EMP_ID	LASTNAME	SUM(SALARY)
90	1002	-	90000
80	1004	-	60000
70	1009	-	45000
90	1001	-	50000
70	1006	-	56000
90	1011	-	50000
80	1010	-	60000
80	1003	-	40000
-	1010	Paloma	60000
-	1002	Swift	90000
-	1006	Cabello	56000
-	1003	Rodrigo	40000
-	1011	Knowles	50000
-	1009	Eilish	45000
-	1001	Styles	50000
-	1004	Carpenter	60000
90	-	Swift	90000
70	-	Cabello	56000
80	-	Rodrigo	40000
90	-	Knowles	50000
90	-	Styles	50000
80	-	Carpenter	60000
80	-	Paloma	60000
70	-	Eilish	45000

24 rows selected. 0.01 seconds

SELECT dept_Id, emp_id, SUM(salary),

GROUPING(dept_id) AS "Dept sub total",

GROUPING(emp_id) AS "employee sub total"

FROM employee123

WHERE dept_id <100

GROUP BY CUBE (dept_id, emp_id);

SELECT dept_id, emp_id, SUM(salary), GROUPING(dept_id) AS "Dept sub total", GROUPING(emp_id) AS "employee sub total" FROM employee123 WHERE dept_id <100 GROUP BY CUBE (dept_id, emp_id)

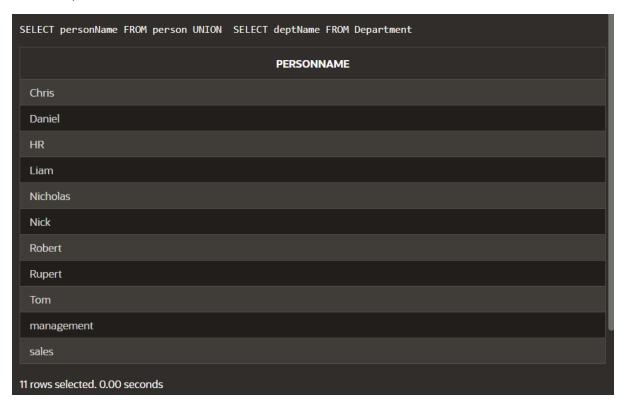| DEPT_ID	EMP_ID	SUM(SALARY)	Dept sub total	employee sub total
-	-	620000	1	1
-	1001	50000	1	0
-	1002	90000	1	0
-	1003	40000	1	0
-	1004	40000	1	0
-	1005	55000	1	0
-	1006	56000	1	0
-	1007	35000	1	0
-	1008	40000	1	0
-	1009	45000	1	0
-	1010	60000	1	0
-	1011	50000	1	0
-	1012	45000	1	0
40	-	45000	0	1
40	1012	45000	0	0
50	-	35000	0	1
50	1007	35000	0	0
60	-	95000	0	1
60	1005	55000	0	0
60	1008	40000	0	0
70	-	101000	0	1
70	1006	56000	0	0
70	1009	45000	0	0
80	-	160000	0	1
80	1003	40000	0	0
80	1004	40000	0	0
80	1010	60000	0	0
90	-	190000	0	1
90	1001	50000	0	0
90	1002	90000	0	0
90	1011	50000	0	0

31 rows selected. 0.00 seconds

SELECT personName

FROM person

UNION

SELECT deptName

FROM Department;

SELECT personName FROM person UNION SELECT deptName FROM Department

PERSONNAME
Chris
Daniel
HR
Liam
Nicholas
Nick
Robert
Rupert
Tom
management
sales

11 rows selected. 0.00 seconds

SELECT personName

FROM person

UNION ALL

SELECT deptName

FROM Department;

```
SELECT personName FROM person UNION ALL SELECT deptName FROM Department
```

PERSONNAME
Liam
Chris
Nick
Nicholas
Robert
Daniel
Rupert
Tom
sales
management
sales
management
HR
management

14 rows selected. 0.01 seconds

SELECT personName

FROM person

MINUS

SELECT deptName

FROM Department;

```
SELECT personName FROM person MINUS SELECT deptName FROM Department
```

PERSONNAME
Chris
Daniel
Liam
Nicholas
Nick
Robert
Rupert
Tom

8 rows selected. 0.01 seconds

SELECT totalAmount,orderID

FROM orders

UNION

SELECT TO_NUMBER(NULL),deptID

FROM Department;

```
SELECT totalAmount,orderID FROM orders UNION SELECT TO_NUMBER(NULL),deptID FROM Department
```

TOTALAMOUNT	ORDERID
100	4
150	2
200	5
250	1
300	3
-	101
-	102
-	103

8 rows selected. 0.00 seconds

SELECT totalAmount,orderID

FROM orders

UNION

SELECT TO_NUMBER(NULL),deptID

FROM Department

ORDER BY orderID;

```
SELECT totalAmount,orderID FROM orders UNION SELECT TO_NUMBER(NULL),deptID FROM Department ORDER BY
orderID
```

TOTALAMOUNT	ORDERID
250	1
150	2
300	3
100	4
200	5
-	101
-	102
-	103

8 rows selected. 0.01 seconds

CREATE TABLE classes(personID numeric,

personName varchar(30));

INSERT INTO classes VALUES(5,'Rahul');

COMMIT;

UPDATE classes SET personName='Abhijit' WHERE personID ='5';

SAVEPOINT A;

INSERT INTO classes VALUES(6,'Chris');

SAVEPOINT B;

INSERT INTO classes VALUES(7,'Bravo');

SAVEPOINT C;

SELECT * FROM classes;

```
CREATE TABLE classes(personID numeric,                    personName varchar(30))

Table created. 0.01 seconds

INSERT INTO classes VALUES(5,'Rahul')

1 row(s) inserted. 0.01 seconds

COMMIT

Statement processed. 0.00 seconds

UPDATE classes SET personName='Abhijit' WHERE personID ='5'

1 row(s) updated. 0.00 seconds

INSERT INTO classes VALUES(6,'Chris')

1 row(s) inserted. 0.01 seconds

INSERT INTO classes VALUES(7,'Bravo')

1 row(s) inserted. 0.00 seconds

SELECT * FROM classes
```

```
SELECT * FROM classes
```

PERSONID	PERSONNAME
5	Abhijit
6	Chris
7	Bravo

3 rows selected. 0.01 seconds