[](https://www.iiuc.ac.bd/)

**Assignment**

**Assignment No. – 04**

**Submission date- 23 February, 2022**

**Course Title- DBMS (Lab)**

**Course Code: CSE-2424**

Submited to-

**Mr. Mohammad Aman Ullah**

**Assistant Professor**

01815641524

ullah047@yahoo.com, aman\_cse@iiuc.ac.bd

Submitted by-

**MD. SOROWAR MAHABUB RABBY**

Matric ID: ***C201032*,** Section: ***4AM* ,** Semester**: *4th***

**Department of CSE (***Computer Science and Engineering***), IIUC**

**Cell:** *01834756433, 01521564157,* [*c201032@ugrad.iiuc.ac.bd*](mailto:c201032@ugrad.iiuc.ac.bd)

1. **Group functions work across many rows to produce one result per group.**

**Answer: True**

1. **Group functions include nulls in calculations.**

**Answer: False**

1. **The WHERE clause restricts rows before inclusion in a group calculation.**

Answer: True

**The HR department needs the following reports:**

1. **Find the highest, lowest, sum, and average salary of all employees. Label the columns Maximum, Minimum, Sum, and Average, respectively. Round your results to the nearest whole number. Place your SQL statement in a text file named lab\_04\_04.sql.**

SELECT ROUND(MAX(salary),0) "Maximum",

ROUND(MIN(salary),0) "Minimum",

ROUND(SUM(salary),0) "Sum",

ROUND(AVG(salary),0) "Average"

FROM employees;

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Maximum** | **Minimum** | | **Sum** | **Average** |
| 24000 | 2100 | | 691400 | 6462 |
| 1 rows returned | |

1. **Modify the query in lab\_04\_04.sql to display the minimum, maximum, sum, and average salary for each job type. Resave lab\_04\_04.sql as lab\_04\_05.sql. Run the statement in lab\_04\_05.sql.**

SELECT job\_id, ROUND(MAX(salary),0) "Maximum",

ROUND(MIN(salary),0) "Minimum",

ROUND(SUM(salary),0) "Sum",

ROUND(AVG(salary),0) "Average"

FROM employees

GROUP BY job\_id;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **JOB\_ID** | **Maximum** | **Minimum** | **Sum** | **Average** |
| IT\_PROG | 9000 | 4200 | 28800 | 5760 |
| AC\_MGR | 12000 | 12000 | 12000 | 12000 |
| AC\_ACCOUNT | 8300 | 8300 | 8300 | 8300 |
| ST\_MAN | 8200 | 5800 | 36400 | 7280 |
| PU\_MAN | 11000 | 11000 | 11000 | 11000 |

19 rows returned

1. **Write a query to display the number of people with the same job.**

SELECT job\_id, COUNT(job\_id)

FROM employees

GROUP BY job\_id;

|  |  |  |
| --- | --- | --- |
| **JOB\_ID** | **COUNT(\*)** | |
| AC\_ACCOUNT | 1 | |
| AC\_MGR | 1 | |
| AD\_ASST | 1 | |
| AD\_PRES | 1 | |
| AD\_VP | 2 | |
| FI\_ACCOUNT | 5 | |
| FI\_MGR | 1 | |
| HR\_REP | 1 | |
| IT\_PROG | 5 | |
| MK\_MAN | 1 | |
| MK\_REP | 1 | |
| PR\_REP | 1 | |
| PU\_CLERK | 5 | |
| PU\_MAN | 1 | |
| SA\_MAN | 5 | |
| SA\_REP | 30 | |
| SH\_CLERK | 20 | |
| ST\_CLERK | 20 | |
| ST\_MAN | 5 | |
| 19 rows returned | |

**Generalize the query so that the user in the HR department is prompted for a job title. Save the script to a file named lab\_04\_06.sql.**

SELECT job\_id, COUNT(job\_id)

FROM employees

WHERE job\_id = '&JOB\_TITTLE'

GROUP BY job\_id;

**Example:**

SELECT job\_id, COUNT(job\_id)

FROM employees

WHERE job\_id = 'IT\_PROG'

GROUP BY job\_id;

|  |  |
| --- | --- |
| **JOB\_ID** | **COUNT(JOB\_ID)** |
| IT\_PROG | 5 |

1. **Determine the number of managers without listing them. Label the column Number of Managers.** *Hint: Use the MANAGER\_ID column to determine the number of managers***.**

SELECT COUNT(DISTINCT manager\_id) AS "Number of Managers"

FROM employees;

|  |
| --- |
| **Number Of Managers** |
| 18 |

1. **Find the difference between the highest and lowest salaries. Label the column DIFFERENCE.**

SELECT MAX(salary) - MIN(salary) AS DIFFERENCE

FROM employees;

|  |
| --- |
| **DIFFERENCE** |
| 21900 |

1. **Create a report to display the manager number and the salary of the lowest-paid employee for that manager. Exclude anyone whose manager is not known. Exclude any groups where the minimum salary is $6,000 or less. Sort the output in descending order of salary.**

SELECT manager\_id, MIN(salary)

FROM employees

WHERE manager\_id IS NOT NULL

GROUP BY manager\_id

HAVING MIN(salary) >= 6000

ORDER BY MIN(salary) DESC;

|  |  |
| --- | --- |
| **MANAGER\_ID** | **MIN(SALARY)** |
| 102 | 9000 |
| 205 | 8300 |
| 146 | 7000 |
| 145 | 7000 |
| 108 | 6900 |
| 149 | 6200 |
| 147 | 6200 |
| 148 | 6100 |
| 201 | 6000 |
| 9 rows returned |

1. **Create a query that will display the total number of employees and, of that total, the number of employees hired in 1995, 1996, 1997, and 1998. Create appropriate column headings.**

SELECT COUNT(\*) AS TOTAL\_EMPLOYEES,

SUM(DECODE(TO\_CHAR(hire\_date, 'YYYY'),1995,1,0)) AS "In\_1995",

SUM(DECODE(TO\_CHAR(hire\_date, 'YYYY'),1996,1,0)) AS "In\_1996",

SUM(DECODE(TO\_CHAR(hire\_date, 'YYYY'),1997,1,0)) AS "In\_1997",

SUM(DECODE(TO\_CHAR(hire\_date, 'YYYY'),1998,1,0)) AS "In\_1998"

FROM employees;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TOTAL\_EMPLOYEES** | **In\_1995** | **In\_1996** | **In\_1997** | **In\_1998** |
| 107 | 4 | 10 | 28 | 23 |
| 1 rows returned |

1. **Create a matrix query to display the job, the salary for that job based on department number, and the total salary for that job, for departments 20, 50, 80, and 90, giving each column an appropriate heading.**

SELECT job\_id "Job",

SUM(DECODE(department\_id , 20, salary)) "Dept 20",

SUM(DECODE(department\_id , 50, salary)) "Dept 50",

SUM(DECODE(department\_id , 80, salary)) "Dept 80",

SUM(DECODE(department\_id , 90, salary)) "Dept 90",

SUM(salary) "Total"

FROM employees

GROUP BY job\_id;

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Job** | **Dept- 20** | | **Dept- 50** | **Dept- 80** | **Dept- 90** | **Total** |
| IT\_PROG | - | | - | - | - | 28800 |
| AC\_MGR | - | | - | - | - | 12000 |
| AC\_ACCOUNT | - | | - | - | - | 8300 |
| ST\_MAN | - | | 36400 | - | - | 36400 |
| PU\_MAN | - | | - | - | - | 11000 |
| AD\_ASST | - | | - | - | - | 4400 |
| AD\_VP | - | | - | - | 34000 | 34000 |
| SH\_CLERK | - | | 64300 | - | - | 64300 |
| FI\_ACCOUNT | - | | - | - | - | 39600 |
| FI\_MGR | - | | - | - | - | 12000 |
| PU\_CLERK | - | | - | - | - | 13900 |
| SA\_MAN | - | | - | 61000 | - | 61000 |
| MK\_MAN | 13000 | | - | - | - | 13000 |
| PR\_REP | - | | - | - | - | 10000 |
| AD\_PRES | - | | - | - | 24000 | 24000 |
| SA\_REP | - | | - | 243500 | - | 250500 |
| MK\_REP | 6000 | | - | - | - | 6000 |
| ST\_CLERK | - | | 55700 | - | - | 55700 |
| HR\_REP | - | | - | - | - | 6500 |
| 19 rows returned | |

Submitted by-

**MD. SOROWAR MAHABUB RABBY**

Matric ID: **C201032**, Section: 4**AM**

Department of CSE (Computer Science and Engineering)