

## NTOMBIZA MPAHLENI

### EXERCISE 2

#### Questions

##### 1. COUNT () Function

Write a SQL query to find the total number of employees in the company.

The screenshot shows a SQL query editor with the following query:

```
1 | Select count (*) as total_employees
2 | from employees_db;
```

Below the query editor, there is a 'Results' tab. The results table has one column, 'TOTAL\_EMPLOYEES', and one row with the value 10.

	# TOTAL_EMPLOYEES
1	10

##### 2. SUM () Function

Write a SQL query to find the total salary paid to all employees in the IT department.

The screenshot shows a SQL query editor with the following query:

```
1 | select SUM (salary) as total_IT_salary
2 | from employees_db
3 | where department= 'IT';
```

Below the query editor, there is a 'Results' tab. The results table has one column, 'TOTAL\_IT\_SALARY', and one row with the value 220000.

	# TOTAL_IT_SALARY
1	220000

##### 3. AVG() Function. Write a SQL query to calculate the average salary of employees in the HR department.

The screenshot shows a SQL query editor with the following query:

```
1 | select avg (salary) AS avg_hr_salary
2 | from employees_db
3 | where department = 'HR';
```

Below the query editor, there is a 'Results' tab. The results table has one column, 'AVG\_HR\_SALARY', and one row with the value 49500.

	# AVG_HR_SALARY
1	49500

#### 4. MIN () and MAX () Functions

Write a SQL query to find the highest and lowest salary in the company.

EMPLOYEES.PUBLIC ▾ Settings ▾

```
1 select
2 min (salary) as lowest_salary,
3 MAX (salary) as highest_salary
4 from employees_db
```

Results Chart

	# LOWEST_SALARY	# HIGHEST_SALARY
1	48000	62000

#### 5. GROUP BY Statement

Write a SQL query to group employees by department and display the total salary paid in each department.

EMPLOYEES.PUBLIC ▾ Settings ▾

```
1 SELECT department,
2 SUM(salary) AS total_salary
3 FROM employees_db
4 GROUP BY department;
```

Results Chart

	A DEPARTMENT	# TOTAL_SALARY
1	IT	220000
2	HR	99000
3	FINANCE	119000
4	MARKETING	52000
5	Marketing	53000

#### 6. GROUP BY and COUNT ()

Write a SQL query to count how many employees work in each city.

EMPLOYEES.PUBLIC ▾ Settings ▾

```
1 SELECT city,
2 COUNT(*) AS total_employees
3 FROM employees_db
4 GROUP BY city;
```

Results Chart

	A CITY	# TOTAL_EMPLOYEES
1	New York	2
2	Chicago	3
3	Los Angeles	2
4	San Francisco	2
5	Houston	1

## 7. GROUP BY and ORDER BY

Write a SQL query to group employees by department, calculate the average salary in each department, and order the results in descending order of average salary.

EMPLOYEES.PUBLIC Settings

```
1 select
2 department,
3 avg (salary) as avg_salary
4 from employees_db
5 group by department
6 order by avg_salary desc;
```

Results Chart

	DEPARTMENT	AVG_SALARY
1	FINANCE	59500.0
2	IT	55000.0
3	Marketing	53000.0
4	MARKETING	52000.0
5	HR	49500.0

## 8. HAVING Clause

Write a SQL query to find departments where the total salary paid exceeds 100,000.

(Use GROUP BY and HAVING).

EMPLOYEES.PUBLIC Settings

```
1 SELECT department,
2 SUM(salary) AS total_salary
3 FROM employees_db
4 GROUP BY department
5 HAVING SUM(salary) > 100000;
```

Results Chart

	DEPARTMENT	TOTAL_SALARY
1	IT	220000
2	FINANCE	119000

## 9. Combining GROUP BY, HAVING, and ORDER BY

Write a SQL query to list cities where more than one employee works, ordered by the number of employees in descending order.

EMPLOYEES.PUBLIC Settings

```
1 SELECT city,
2 COUNT(*) AS employee_count
3 FROM employees_db
4 GROUP BY city
5 HAVING COUNT(*) > 1
6 ORDER BY employee_count DESC;
```

Results Chart

	CITY	EMPLOYEE_COUNT
1	Chicago	3
2	New York	2
3	Los Angeles	2
4	San Francisco	2

## 10. Combining Aggregate Functions

Write a SQL query to find the department with the highest average salary.

EMPLOYEES.PUBLIC ▾ Settings ▾

```
1 SELECT department,  
2 AVG(salary) AS avg_salary  
3 FROM employees_db  
4 GROUP BY department  
5 ORDER BY avg_salary DESC  
6 LIMIT 1;
```

↩ Results Chart

	DEPARTMENT	AVG_SALARY
1	FINANCE	59500.000