SQL Aggregation Functions

Introduction

This handout covers essential SQL aggregation functions and related commands that help in summarizing and analyzing data effectively.

COUNT Function

COUNT is used to count rows in a result set.

```
-- Count all rows in a table
SELECT COUNT(*) FROM employees;

-- Count specific column values (excludes NULL)
SELECT COUNT(salary) FROM employees;

-- Count distinct values
SELECT COUNT(DISTINCT department) FROM employees;

-- Count with conditions
SELECT COUNT(*) FROM employees WHERE salary > 50000;
```

Handling NULL Values

NULL represents missing or unknown values in SQL.

```
-- Check for NULL values
SELECT * FROM employees WHERE salary IS NULL;

-- Replace NULL with a default value using COALESCE
SELECT name, COALESCE(salary, 0) AS salary
FROM employees;

-- IFNULL function (MySQL specific)
SELECT name, IFNULL(salary, 0) AS salary
FROM employees;

-- Count non-NULL values
SELECT COUNT(salary) AS salary_count,
```

```
       COUNT(*) AS total_count
FROM employees;
```

ROUND Command

ROUND is used to round numeric values to a specified number of decimal places.

```
-- Round to nearest integer
SELECT ROUND(salary) FROM employees;

-- Round to 2 decimal places
SELECT ROUND(salary, 2) FROM employees;

-- Round to hundreds (negative decimals)
SELECT ROUND(salary, -2) FROM employees;
```

Arithmetic Commands

SQL supports various arithmetic operations in queries.

```
-- Basic arithmetic
SELECT
    salary,
    salary + 1000 AS salary_with_bonus,
    salary * 1.1 AS salary_with_raise,
    salary / 12 AS monthly_salary,
    salary - 5000 AS salary_after_deduction
FROM employees;

-- Combining arithmetic with ROUND
SELECT
    salary,
    ROUND(salary / 12, 2) AS monthly_salary
FROM employees;

-- Percentage calculations
SELECT
    department,
    ROUND((COUNT(*) * 100.0 / (SELECT COUNT(*) FROM employees)), 2) AS
percentage
```

```
FROM employees
GROUP BY department;
```

GROUP BY

GROUP BY is used to group rows that have the same values into summary rows.

```
-- Basic grouping
SELECT department, COUNT(*) AS employee_count
FROM employees
GROUP BY department;

-- Multiple aggregations
SELECT
    department,
    COUNT(*) AS employee_count,
    ROUND(AVG(salary), 2) AS avg_salary,
    MAX(salary) AS highest_salary,
    MIN(salary) AS lowest_salary
FROM employees
GROUP BY department;

-- Grouping with HAVING clause
SELECT
    department,
    COUNT(*) AS employee_count,
    AVG(salary) AS avg_salary
FROM employees
GROUP BY department
HAVING COUNT(*) > 5;

-- Multiple GROUP BY columns
SELECT
    department,
    job_title,
    COUNT(*) AS employee_count
FROM employees
GROUP BY department, job_title
ORDER BY department;
```

Helpful Links

- PostgreSQL Documentation on Aggregate Functions

- MySQL Documentation on Aggregate Functions
- W3Schools SQL Tutorial

CheatSheets

- Common Aggregation Functions:

```
COUNT()  -- Count rows
SUM()    -- Sum values
AVG()    -- Average of values
MAX()    -- Maximum value
MIN()    -- Minimum value
```

- NULL Handling Functions:

```
COALESCE()  -- Returns first non-NULL value
IFNULL()    -- Returns specified value if NULL
NULLIF()    -- Returns NULL if two expressions are equal
```

- Rounding Functions:

```
ROUND()     -- Round to specified decimals
CEIL()      -- Round up to nearest integer
FLOOR()     -- Round down to nearest integer
TRUNC()     -- Truncate to specified decimals
```