

Experiment 3 – PL/pgSQL Salary and Bonus Calculation

Experiment

Experiment 3: Calculating employee salary and applying a bonus using PL/pgSQL. This experiment demonstrates variable declaration, arithmetic operations, and displaying output using `RAISE NOTICE`.

Aim

To practice writing PL/pgSQL blocks that perform calculations on employee data, calculate bonuses, and display results.

Objective

- To declare variables in PL/pgSQL.
 - To perform arithmetic calculations on employee salary.
 - To calculate a bonus amount based on a percentage of salary.
 - To display employee details and salary information before and after applying the bonus.
-

Software Requirements

- Database: PostgreSQL
 - Tool: pgAdmin or psql
-

Practical / Experiment Steps

1. Declare variables for employee ID, name, and salary.
 2. Calculate a 10% bonus on the employee salary.
 3. Calculate the total salary after adding the bonus.
 4. Display employee details and salary information before and after bonus using `RAISE NOTICE`.
 5. Take screenshots of outputs for documentation.
-

Input / Output Details

Input

- Employee details:
 - `emp_id` INT
 - `emp_name` VARCHAR(50)
 - `emp_salary` INT
- Bonus calculation: 10% of employee salary

Output

- Step 1: Display employee details and original salary.
 - Step 2: Display bonus amount and total salary after applying 10% bonus.
 - Screenshots of outputs (s1 and s2) are attached.
-

SQL / PL/pgSQL Code

`DO`

`$$`

```
DECLARE

    emp_id      INT := 101;
    emp_name    VARCHAR(50) := 'Jay';
    emp_salary   INT := 22222;
    bonus_amount NUMERIC;
    new_salary   NUMERIC;

BEGIN

    -- Calculate 10% bonus

    bonus_amount := emp_salary * 0.10;
    new_salary := emp_salary + bonus_amount;

    -- Display Employee Details

    RAISE NOTICE 'Employee Details';
    RAISE NOTICE '-----';
    RAISE NOTICE 'Employee ID : %', emp_id;
    RAISE NOTICE 'Employee Name : %', emp_name;

    -- Display Salary Details

    RAISE NOTICE '';
    RAISE NOTICE 'Salary Details';
    RAISE NOTICE '-----';
    RAISE NOTICE 'Salary Before Bonus : %', emp_salary;
    RAISE NOTICE '10% Bonus Amount : %', bonus_amount;
    RAISE NOTICE 'Salary After Bonus : %', new_salary;
```

```
END
```

```
$$;
```

Screenshots

- **Screenshot 1:** DO block execution

```
$$;
      new_salary    NUMERIC;
class$# BEGIN
class$#   -- Calculate 10% bonus
class$#   bonus_amount := emp_salary * 0.10;
class$#   new_salary := emp_salary + bonus_amount;
class$#
class$#   -- Display Employee Details
class$#   RAISE NOTICE 'Employee Details';
class$#   RAISE NOTICE '-----';
class$#   RAISE NOTICE 'Employee ID : %', emp_id;
class$#   RAISE NOTICE 'Employee Name : %', emp_name;
class$#
class$#   -- Display Salary Details
class$#   RAISE NOTICE '';
class$#   RAISE NOTICE 'Salary Details';
class$#   RAISE NOTICE '-----';
class$#   RAISE NOTICE 'Salary Before Bonus : %', emp_salary;
class$#   RAISE NOTICE '10% Bonus Amount : %', bonus_amount;
class$#   RAISE NOTICE 'Salary After Bonus : %', new_salary;
class$# END
class$# $$;
```

- **Screenshot 2:** Output showing Employee ID, Name, Salary, Bonus, and Total Salary

```
class$# $$;
NOTICE: Employee Details
NOTICE: -----
NOTICE: Employee ID    : 101
NOTICE: Employee Name  : Sahil
NOTICE:
NOTICE: Salary Details
NOTICE: -----
NOTICE: Salary Before Bonus : 550005
NOTICE: 10% Bonus Amount   : 55000.50
NOTICE: Salary After Bonus : 605005.50
DO
```

Learning Outcome

- Learned to declare variables in PL/pgSQL.
- Learned to perform arithmetic calculations in PostgreSQL.
- Learned to display output using `RAISE NOTICE`.
- Learned to calculate a bonus and total salary dynamically.