set search\_path = hr;

-- 1.   Create pgsql block and to check for all employees using cursor; and update their commission\_pct based on the salary

-- SALARY < 7000  :                    COMM = 0.1

-- 7000 <= SALARY < 10000      COMM = 0.15

-- 10000 <= SALARY < 15000    COMM = 0.2

-- 15000 <= SALARY                      COMM = 0.25

DO $$

DECLARE

    emp\_cursor CURSOR FOR

        SELECT \* FROM employees;

BEGIN

    FOR v\_emp\_record IN emp\_cursor

    LOOP

        IF v\_emp\_record.salary < 7000 THEN

            UPDATE employees

            SET commission\_pct = 0.10

            WHERE employee\_id = v\_emp\_record.employee\_id;

        ELSIF v\_emp\_record.salary >= 7000 AND v\_emp\_record.salary < 10000 THEN

            UPDATE employees

            SET commission\_pct = 0.15

            WHERE employee\_id = v\_emp\_record.employee\_id;

        ELSIF v\_emp\_record.salary >= 10000 AND v\_emp\_record.salary < 15000 THEN

            UPDATE employees

            SET commission\_pct = 0.20

            WHERE employee\_id = v\_emp\_record.employee\_id;

        ELSE -- salary >= 15000

            UPDATE employees

            SET commission\_pct = 0.25

            WHERE employee\_id = v\_emp\_record.employee\_id;

        END IF;

    END LOOP;

END;

$$;

-- 2.   Alter table employees then add column retired\_bonus

-- Create pgsql block to calculate the retired salary for all employees using cursor and update retired\_bonus column

-- Retired bonus = no of working months \* 10 % of his current salary

-- Only for those employees have passed 18 years of their hired date

Alter table employees

add column retired\_bonus  NUMERIC ;

DO $$

DECLARE

    v\_years INT;

    v\_months INT;

    v\_total\_months INT;

    emp\_cursor CURSOR FOR

        SELECT \* FROM employees;

BEGIN

    FOR v\_emp\_record IN emp\_cursor

    LOOP

        v\_years := EXTRACT('year' FROM AGE(NOW(), v\_emp\_record.hire\_date));

        v\_months := EXTRACT('month' FROM AGE(NOW(), v\_emp\_record.hire\_date));

        v\_total\_months := (v\_years \* 12) + v\_months;

        IF v\_years > 18 THEN

            UPDATE employees

            SET  retired\_bonus = v\_total\_months \* (0.1 \*salary)

            WHERE employee\_id = v\_emp\_record.employee\_id;

            raise notice 'v years = %,v months = % , v\_total\_months =%', v\_years,v\_months ,v\_total\_months ;

        END IF;

    END LOOP;

END;

$$;

-- 3.   Create pgsql block that loop over employees table and

-- Increase only those working in ‘IT’ department by 10% of their salary.

DO $$

DECLARE

    emp\_cursor CURSOR FOR

        SELECT e.employee\_id, e.salary

        FROM employees e

        JOIN departments d ON e.department\_id = d.department\_id

        WHERE d.department\_name = 'IT';

BEGIN

    FOR v\_emp\_record IN emp\_cursor

    LOOP

        UPDATE employees

        SET salary = salary \* 1.10

        WHERE employee\_id = v\_emp\_record.employee\_id;

        RAISE NOTICE 'Increased salary for Emp ID = %', v\_emp\_record.employee\_id;

    END LOOP;

END;

$$;

-- 4.   Create a PG/SQL block to award a 500 bonus to all employees who have been working for more than 15 years and have no commission.

DO $$

DECLARE

    v\_years INT;

    emp\_cursor CURSOR FOR

        SELECT \* FROM employees;

BEGIN

    FOR v\_emp\_record IN emp\_cursor

    LOOP

        v\_years := EXTRACT(YEAR FROM AGE(NOW(), v\_emp\_record.hire\_date));

        IF v\_years > 15 AND v\_emp\_record.commission\_pct IS NULL THEN

            UPDATE employees

            SET salary = salary + 500

            WHERE employee\_id = v\_emp\_record.employee\_id;

            RAISE NOTICE 'Bonus added: Emp ID = %, Years = %, New Salary = %',

                         v\_emp\_record.employee\_id, v\_years, v\_emp\_record.salary;

        END IF;

    END LOOP;

END;

$$;

-- 5.   Create a PG/SQL block to give a 5% bonus to all employees in the ‘Sales’ department who have a salary greater than 8000.

DO $$

DECLARE

    emp\_record employees%ROWTYPE;

    emp\_cursor CURSOR FOR

        SELECT e.\*

        FROM employees e

        JOIN departments d ON e.department\_id = d.department\_id

        WHERE d.department\_name = 'Sales'

          AND e.salary > 8000;

BEGIN

    FOR emp\_record IN emp\_cursor

    LOOP

        UPDATE employees

        SET salary = salary \* 1.05

        WHERE employee\_id = emp\_record.employee\_id;

        RAISE NOTICE 'Employee % got 5%% bonus. New Salary = %',

                     emp\_record.employee\_id, emp\_record.salary \* 1.05;

    END LOOP;

END;

$$;

-- 6.

--a- add RETIRED INT  column to employees table using alter

Alter table employees

add column retired  INT ;

-- b- Create and call

-- CHECK\_RETIRED FUNCTION(V\_EMP\_ID INT, V\_MAX\_HIRE\_YEAR INT) RETURN INT;

-- that will return 1 if employee has passed no of years >=  V\_MAX\_HIRE\_YEAR, return 0 for otherwise

CREATE OR REPLACE FUNCTION check\_retired(v\_emp\_id INT, v\_max\_hire\_year INT)

RETURNS INT AS $$

DECLARE

    v\_hire\_date DATE;

    v\_years     INT;

BEGIN

    SELECT hire\_date

    INTO v\_hire\_date

    FROM employees

    WHERE employee\_id = v\_emp\_id;

    v\_years := EXTRACT(YEAR FROM AGE(NOW(), v\_hire\_date));

    IF v\_years >= v\_max\_hire\_year THEN

        RETURN 1;

    ELSE

        RETURN 0;

    END IF;

END;

$$ LANGUAGE plpgsql;

SELECT check\_retired(101, 18);

SELECT check\_retired(101, 50);

-- c- create block to call and update the emp with set retired = 1  if this employee will retired

DO $$

DECLARE

    v\_emp\_record employees%ROWTYPE;

    emp\_cursor CURSOR FOR SELECT \* FROM employees;

BEGIN

    FOR v\_emp\_record IN emp\_cursor

    LOOP

        IF check\_retired(v\_emp\_record.employee\_id, 30) = 1 THEN

            UPDATE employees

            SET retired = 1

            WHERE employee\_id = v\_emp\_record.employee\_id;

            RAISE NOTICE 'Employee id % , name % has retired', v\_emp\_record.employee\_id,v\_emp\_record.first\_name;

        END IF;

    END LOOP;

END;

$$;