

דו"ח שלב ג :

נעם טויטו : 215044884

רפאל ריאחי : 327839171

מערכת המטפלת בצרכי בית חולים. היחידה המטופלת : כוח אדם.

תוכן עניינים :

: function 1

פונקציה זו מחשבת את השכר הממוצע של עובדים במחלקה מסוימת באמצעות שימוש בקורסור מפורש כדי לאסוף נתונים מטבלת המשרות. אם אין עובדים במחלקה, הפונקציה מחזירה 0, ובמקרה של שגיאה מחזירה -1.

```
-- Function 1: Get Average Salary Using Explicit Cursor
CREATE OR REPLACE FUNCTION GetAverageSalary(dept_id INT)
RETURN NUMBER
IS
    avg_salary NUMBER := 0;
    total_salary NUMBER := 0;
    employee_count NUMBER := 0;
    salary NUMBER; -- Variable to fetch individual salary
    CURSOR salary_cursor IS
        SELECT Salary FROM Position WHERE DepartmentID = dept_id;
BEGIN
    OPEN salary_cursor;
    LOOP
        FETCH salary_cursor INTO salary;
        EXIT WHEN salary_cursor%NOTFOUND;
        total_salary := total_salary + salary; -- Accumulate salaries
        employee_count := employee_count + 1;
    END LOOP;
    CLOSE salary_cursor;

    IF employee_count > 0 THEN
        avg_salary := total_salary / employee_count; -- Calculate average salary
    ELSE
        avg_salary := 0;
    END IF;

    RETURN avg_salary;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RETURN 0;
    WHEN OTHERS THEN
        RETURN -1; -- Indicates an error
END GetAverageSalary;
/
-- End of Function 1
```

: function 2

פונקציה זו מחזירה קורסור שמכיל את פרטי המחלקה של עובד מסוים לפי מזהה העובד. אם אין נתונים או במקרה של שגיאה, הפונקציה מחזירה NULL.

-- Function 2: Get Department by Employee ID Returning REF CURSOR

CREATE OR REPLACE FUNCTION GetDepartmentByEmployee(employee_id INT)

RETURN SYS_REFCURSOR

IS

dept_cursor SYS_REFCURSOR;

BEGIN

OPEN dept_cursor FOR

SELECT d.DepartmentID, d.DepartmentName, d.DateEstablished

FROM Department d

JOIN Position p ON d.DepartmentID = p.DepartmentID

JOIN Employee e ON p.PositionID = e.PositionID

WHERE e.EmployeeID = employee_id;

RETURN dept_cursor;

EXCEPTION

WHEN NO_DATA_FOUND THEN

RETURN NULL;

WHEN OTHERS THEN

RETURN NULL;

END GetDepartmentByEmployee;

/

-- End of Function 2

: procedure1

פרוצדורה זו מעדכנת את שכרו של עובד מסוים לפי מזהה העובד והשכר החדש. אם אין עדכון, היא מעלה שגיאה ומבצעת רולבק במקרה של שגיאה.

-- Procedure 1: Update Employee Salary with Multiple DML Commands and Records

```
CREATE OR REPLACE PROCEDURE UpdateEmployeeSalary(  
    emp_id INT,  
    new_salary DECIMAL  
)  
IS  
    old_salary DECIMAL(10, 2);  
    pos_id INT;  
BEGIN  
    SELECT Salary, PositionID INTO old_salary, pos_id  
    FROM Position  
    WHERE PositionID = (SELECT PositionID FROM Employee WHERE EmployeeID  
= emp_id);  
  
    UPDATE Position  
    SET Salary = new_salary  
    WHERE PositionID = pos_id;  
  
    IF SQL%ROWCOUNT = 0 THEN  
        RAISE_APPLICATION_ERROR(-20001, 'No rows updated');  
    END IF;  
  
    COMMIT;  
EXCEPTION  
    WHEN OTHERS THEN  
        ROLLBACK;  
        DBMS_OUTPUT.PUT_LINE('Error updating salary: ' || SQLERRM);  
END UpdateEmployeeSalary;  
/  
-- End of Procedure 1
```

: procedure 2

פרוצדורה זו מעניקה חופשה לעובד אם יש לו דירוג ביצועים גבוה מעל 6. אם לא, היא לא מעניקה חופשה. במקרה של שגיאה היא מבצעת רולבק ומדפיסה הודעת שגיאה.

-- Procedure2: Grant Leave to Employee with IF-ELSE and Loop

```
CREATE OR REPLACE PROCEDURE GrantLeave(
    emp_id INT,
    leave_type VARCHAR2,
    return_date DATE
)
IS
    has_high_rating BOOLEAN := FALSE;
BEGIN
    -- Check if the employee has any performance reviews with rating > 6
    FOR review_rec IN (
        SELECT Rating
        FROM PerformanceReview
        WHERE EmployeeID = emp_id
    ) LOOP
        IF review_rec.Rating > 6 THEN
            has_high_rating := TRUE;
            EXIT; -- Exit the loop if a high rating is found
        END IF;
    END LOOP;
    -- If employee has a high rating, grant leave; otherwise, do not grant leave
    IF has_high_rating THEN
        -- Outer loop to simulate multiple conditions
        FOR i IN 1..3 LOOP
            DBMS_OUTPUT.PUT_LINE('Processing condition set ' || i || ':');

            -- Inner loop to simulate processing within each condition set
            FOR j IN 1..2 LOOP
                DBMS_OUTPUT.PUT_LINE(' Processing condition ' || j || ' within set ' || i || ':');
            END LOOP;
        END LOOP;
        -- Grant leave if all conditions are met
        INSERT INTO Leave (LeaveID, EmployeeID, LeaveType, ReturnDate)
        VALUES (Leave_SEQ.NEXTVAL, emp_id, leave_type, return_date);

        COMMIT;
        DBMS_OUTPUT.PUT_LINE('Leave granted successfully.');
```

ELSE

```
        DBMS_OUTPUT.PUT_LINE('Employee does not meet leave criteria based on
performance review.');
```

```

END IF;
EXCEPTION
    WHEN OTHERS THEN
        ROLLBACK;
        DBMS_OUTPUT.PUT_LINE('Error granting leave: ' || SQLERRM);
END GrantLeave;
/
-- End of Procedure2

```

: main 1

תוכנית זו מחשבת ומציגה את השכר הממוצע למספר מחלקות ומעדכנת את שכרם של מספר עובדים. היא משלבת בין פונקציה 1 ופרוצדורה 1.

-- Main Program 1: Calculate and Display Average Salary for Multiple Employees and Departments

```

DECLARE
    dept_id INT;
    avg_salary NUMBER;
    emp_id INT;
    new_salary DECIMAL(10, 2) := 75000.00;
BEGIN
    FOR dept_id IN 190..193 LOOP
        avg_salary := GetAverageSalary(dept_id);
        DBMS_OUTPUT.PUT_LINE('Department ID: ' || dept_id || ' - Average Salary: ' ||
avg_salary);
    END LOOP;
    FOR emp_id IN 101..103 LOOP
        UpdateEmployeeSalary(emp_id, new_salary);
        DBMS_OUTPUT.PUT_LINE('Employee ID: ' || emp_id || ' salary updated to ' ||
new_salary || '.');
    END LOOP;
END;
/
-- End of Main Program 1

```

: main 2

תוכנית זו שולפת את פרטי המחלקה של עובד מסוים ומעניקה לו חופשה. אם התהליך מצליח,
מדפיסה הודעה מתאימה. היא משלבת בין פוקציה 2 ופרוצדורה 2.

-- Main Program 2: Get Department and Grant Leave

DECLARE

emp_id INT := 101;

dept_cursor SYS_REFCURSOR;

dept_rec Department%ROWTYPE;

leave_type VARCHAR2(50) := 'Medical';

return_date DATE := SYSDATE + 30;

BEGIN

dept_cursor := GetDepartmentByEmployee(emp_id);

LOOP

FETCH dept_cursor INTO dept_rec;

EXIT WHEN dept_cursor%NOTFOUND;

DBMS_OUTPUT.PUT_LINE('Department ID: ' || dept_rec.DepartmentID);

DBMS_OUTPUT.PUT_LINE('Department Name: ' ||
dept_rec.DepartmentName);

DBMS_OUTPUT.PUT_LINE('Date Established: ' || dept_rec.DateEstablished);

END LOOP;

CLOSE dept_cursor;

GrantLeave(emp_id, leave_type, return_date);

DBMS_OUTPUT.PUT_LINE('Leave granted.');

END;

/

-- End of Main Program 2

תוצאת ריצה של התוכנית הראשית 1 :

```
Department ID: 190 - Average Salary: 28986.0166666666666666666666666667  
Department ID: 191 - Average Salary: 0  
Department ID: 192 - Average Salary: 44382.07  
Department ID: 193 - Average Salary: 25581.12  
Employee ID: 101 salary updated to 75000.  
Employee ID: 102 salary updated to 75000.  
Employee ID: 103 salary updated to 75000.
```

	EMPLOYEEID	FIRSTNAME	LASTNAME	POSITIONID	
49	101	Embeth	Hurt		271
50	102	Cameron	Parish		381
51	103	Spencer	Love		201

	POSITIONID	POSITIONNAME	DEPARTMENTID	SPECIALTYID	SALARY
381	381	Physician381	248	393	75000.00
201	201	Medical Assistant201	222	348	75000.00
271	271	Occupational Therapist271	184	357	75000.00

תוצאת ריצה של התוכנית הראשית 2 :

```
Department ID: 184
Department Name: Dermatology85
Date Established: 21-OCT-12
Employee does not meet leave criteria based on performance review.
Leave granted.
```