Edmundas Riškus
PRIf-15/2

Laboratorinis darbas nr. 6

---

1. Kodo turinys:

```
DECLARE
    v_deptno    NUMBER      := 20;
    CURSOR      c_emp_cursor IS
        SELECT last_name, salary, manager_id FROM employees
        WHERE department_id = v_deptno;
BEGIN
    FOR emp_record IN c_emp_cursor LOOP
        IF (emp_record.salary < 5000)
            AND ( (emp_record.manager_id = 101) OR (emp_record.manager_id = 124) )
        THEN
            DBMS_OUTPUT.PUT_LINE(emp_record.last_name || ' Due for a raise');
        ELSE
            DBMS_OUTPUT.PUT_LINE(emp_record.last_name || ' Not due for a raise');
        END IF;
    END LOOP;
END;
```

2. Kodo turinys:

```
DECLARE
    CURSOR      dept_cursor IS
        SELECT department_id, department_name FROM departments
        WHERE department_id < 100
        ORDER BY department_name DESC;
    CURSOR      emp_cursor (deptno NUMBER) IS
        SELECT last_name, job_id, hire_date, salary FROM employees
        WHERE employee_id < 120 AND department_id = deptno;
    record_dept_cursor      dept_cursor%ROWTYPE;
    record_emp_cursor       emp_cursor%ROWTYPE;
BEGIN
    OPEN dept_cursor;
    LOOP
        FETCH dept_cursor INTO record_dept_cursor;
        EXIT WHEN dept_cursor%NOTFOUND;
        DBMS_OUTPUT.PUT_LINE('Department number '
            || record_dept_cursor.department_id
            || ' Department name ' || record_dept_cursor.department_name);
        IF NOT emp_cursor%ISOPEN THEN
            OPEN emp_cursor (record_dept_cursor.department_id);
        END IF;
        LOOP
            FETCH emp_cursor INTO record_emp_cursor;
            EXIT WHEN emp_cursor%NOTFOUND;
            DBMS_OUTPUT.PUT_LINE(record_emp_cursor.last_name
                || ' ' || record_emp_cursor.job_id
                || ' ' || record_emp_cursor.hire_date
                || ' ' || record_emp_cursor.salary);
        END LOOP;
        CLOSE emp_cursor;

DBMS_OUTPUT.PUT_LINE('--------------------------------------------------------');
    END LOOP;
    CLOSE dept_cursor;
END;
```

3. Kodo turinys:

```
DROP TABLE top_salaries;
/
CREATE TABLE top_salaries (salary NUMBER(8,2));
/
DECLARE
    v_num       NUMBER      := 5;
    sal         employees.salary%TYPE;
    CURSOR c_emp_cursor IS
        SELECT salary FROM employees
        ORDER BY salary DESC;
BEGIN
    OPEN c_emp_cursor;
    LOOP
        FETCH c_emp_cursor INTO sal;
        EXIT WHEN NOT c_emp_cursor%FOUND
            OR c_emp_cursor%ROWCOUNT > v_num;
        INSERT INTO top_salaries (salary) VALUES (sal);
    END LOOP;
    CLOSE c_emp_cursor;
END;
/

SELECT * FROM top_salaries;
```