DBMS Assignment 4

Devam Desai IIT2022035

Q1

```
DECLARE
   v_employee_id employee.employee_id%TYPE;
   v_salary      employee.salary%TYPE;
BEGIN
   FOR employee_rec IN (SELECT employee_id, salary FROM employee WHERE salary > 80000 AND department_id IN ('CS', 'ECE')) LOOP
      v_employee_id := employee_rec.employee_id;
      DBMS_OUTPUT.PUT_LINE('Employee ID: ' || v_employee_id);
   END LOOP;
END;
/
```

Q2.

```
SET SERVEROUTPUT ON;
```

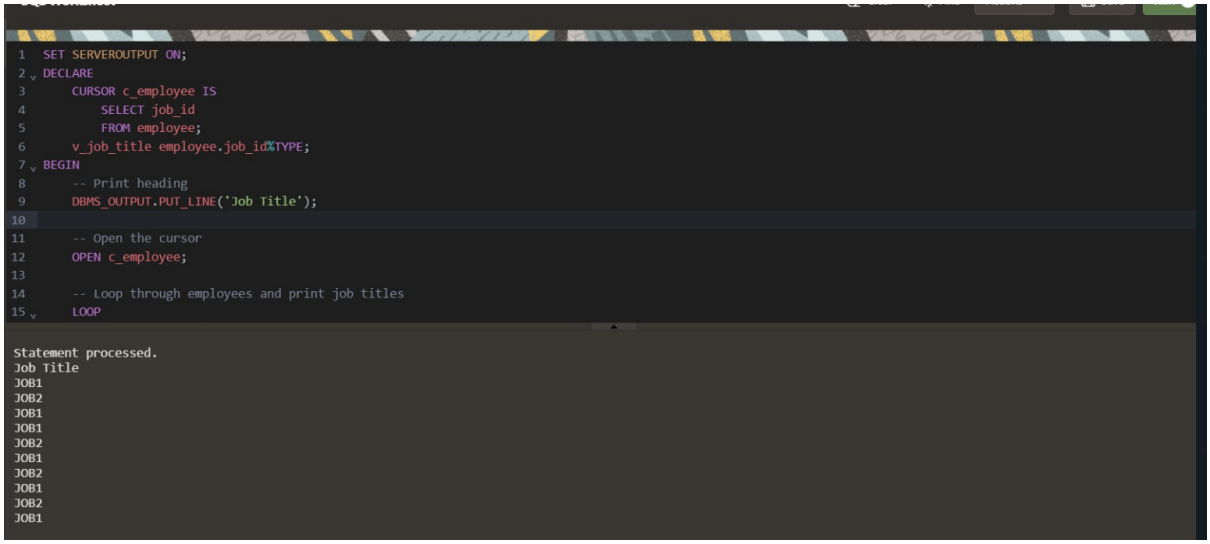
DECLARED

```
    v_employee_id   employee.employee_id%TYPE;
    v_first_name    employee.first_name%TYPE;
    v_hire_year     NUMBER;
BEGIN
    FOR employee_rec IN (SELECT employee_id, first_name, EXTRACT(YEAR FROM hire_date) AS hire_year
                         FROM employee
                         WHERE dept_id = 'CS' AND EXTRACT(YEAR FROM hire_date) = 2021) LOOP
        v_employee_id := employee_rec.employee_id;
        v_first_name := employee_rec.first_name;
        v_hire_year := employee_rec.hire_year;
        DBMS_OUTPUT.PUT_LINE('Employee ID: ' || v_employee_id || ', First Name: ' || v_first_name || ', Hire Year: ' || v_hire_year);
    END LOOP;
END;
/
```

Q3.

```
SET SERVEROUTPUT ON;
DECLARE
```

```
    CURSOR c_employee IS
        SELECT job_id
        FROM employee;
    v_job_title employee.job_id%TYPE;
BEGIN
    -- Print heading
    DBMS_OUTPUT.PUT_LINE('Job Title');

    -- Open the cursor
    OPEN c_employee;

    -- Loop through employees and print job titles
    LOOP
        FETCH c_employee INTO v_job_title;
        EXIT WHEN c_employee%NOTFOUND;

        -- Print job title
        DBMS_OUTPUT.PUT_LINE(v_job_title);
    END LOOP;

    -- Close the cursor
    CLOSE c_employee;
END;
/
```

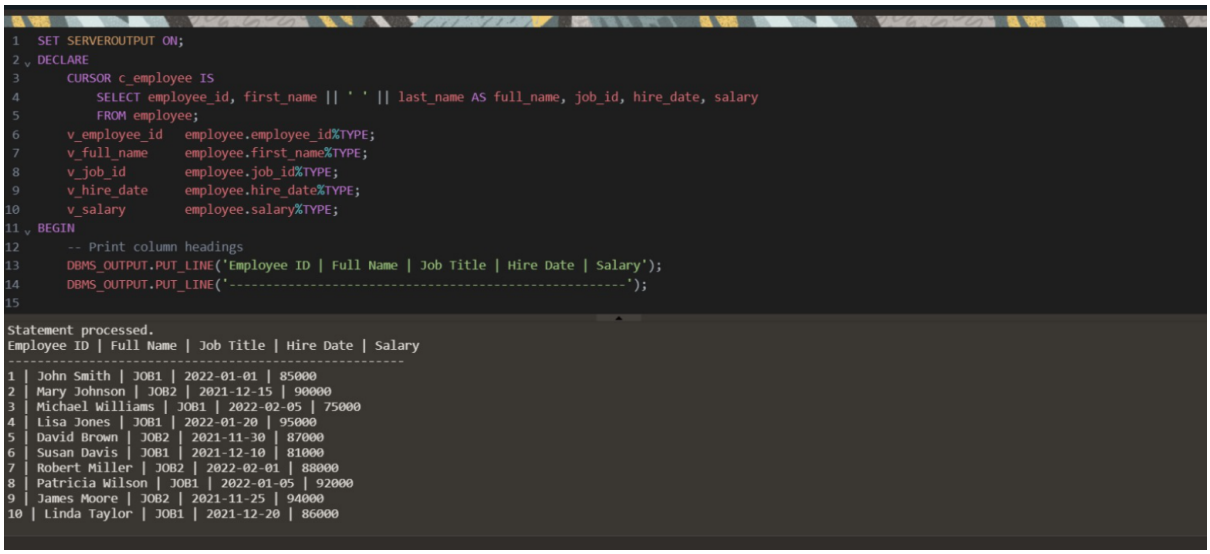


```
SET SERVEROUTPUT ON;
DECLARE
    CURSOR c_employee IS
        SELECT job_id
        FROM employee;
    v_job_title employee.job_id%TYPE;
BEGIN
    -- Print heading
    DBMS_OUTPUT.PUT_LINE('Job Title');

    -- Open the cursor
    OPEN c_employee;

    -- Loop through employees and print job titles
    LOOP

Statement processed.
Job Title
JOB1
JOB2
JOB1
JOB1
JOB2
JOB1
JOB2
JOB1
JOB2
JOB1
```

Q4

```
DECLARE
   CURSOR c_employee IS
      SELECT employee_id, first_name || ' ' || last_name AS full_name, job_id, hire_date, salary
      FROM employee;
   v_employee_id   employee.employee_id%TYPE;
   v_full_name     employee.first_name%TYPE;
   v_job_id        employee.job_id%TYPE;
   v_hire_date     employee.hire_date%TYPE;
   v_salary        employee.salary%TYPE;
BEGIN
   -- Print column headings
   DBMS_OUTPUT.PUT_LINE('Employee ID | Full Name | Job Title | Hire Date | Salary');
   DBMS_OUTPUT.PUT_LINE('-------------------------------------------------');


   -- Open the cursor
```

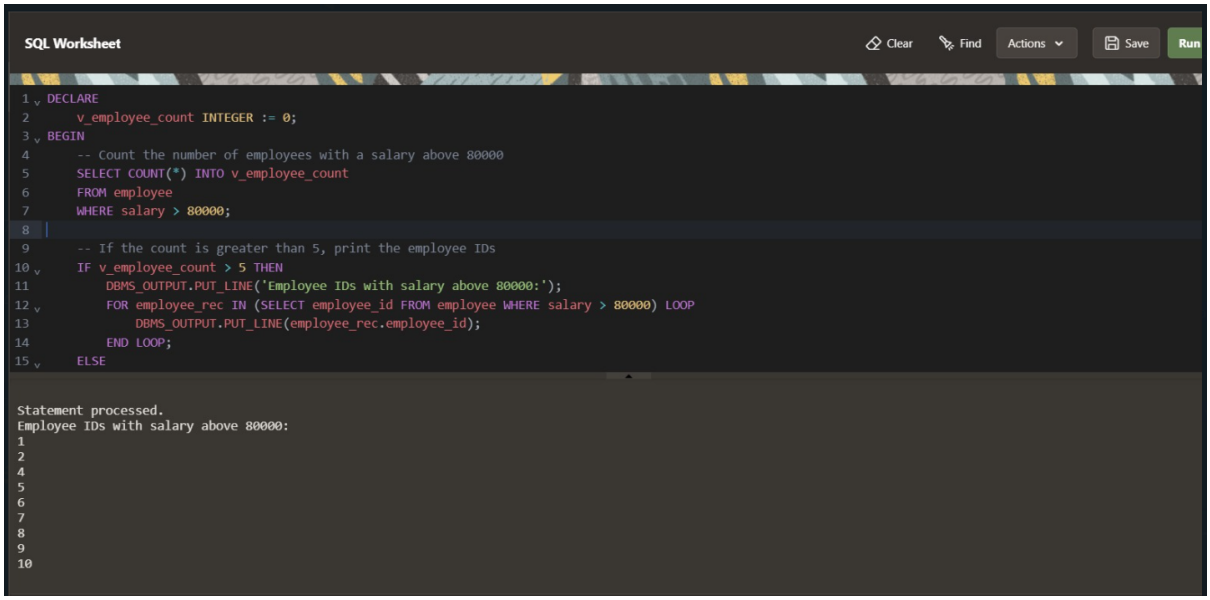
```
    OPEN c_employee;

    -- Loop through employees and print details
    LOOP
        FETCH c_employee INTO v_employee_id, v_full_name, v_job_id, v_hire_date, v_salary;
        EXIT WHEN c_employee%NOTFOUND;

        -- Print employee details
        DBMS_OUTPUT.PUT_LINE(v_employee_id || ' | ' || v_full_name || ' | ' || v_job_id || ' | ' || TO_CHAR(v_hire_date, 'YYYY-MM-DD') || ' | ' || v_salary);
    END LOOP;

    -- Close the cursor
    CLOSE c_employee;
END;
/
```



```
SET SERVEROUTPUT ON;
DECLARE
    CURSOR c_employee IS
        SELECT employee_id, first_name || ' ' || last_name AS full_name, job_id, hire_date, salary
        FROM employee;
    v_employee_id   employee.employee_id%TYPE;
    v_full_name     employee.first_name%TYPE;
    v_job_id        employee.job_id%TYPE;
    v_hire_date     employee.hire_date%TYPE;
    v_salary        employee.salary%TYPE;
BEGIN
    -- Print column headings
    DBMS_OUTPUT.PUT_LINE('Employee ID | Full Name | Job Title | Hire Date | Salary');
    DBMS_OUTPUT.PUT_LINE('-------------------------------------------------------');

Statement processed.
Employee ID | Full Name | Job Title | Hire Date | Salary
-------------------------------------------------------
1 | John Smith | JOB1 | 2022-01-01 | 85000
2 | Mary Johnson | JOB2 | 2021-12-15 | 90000
3 | Michael Williams | JOB1 | 2022-02-05 | 75000
4 | Lisa Jones | JOB1 | 2022-01-20 | 95000
5 | David Brown | JOB2 | 2021-11-30 | 87000
6 | Susan Davis | JOB1 | 2021-12-10 | 81000
7 | Robert Miller | JOB2 | 2022-02-01 | 88000
8 | Patricia Wilson | JOB1 | 2022-01-05 | 92000
9 | James Moore | JOB2 | 2021-11-25 | 94000
10 | Linda Taylor | JOB1 | 2021-12-20 | 86000
```

Q5

```
DECLARE
   v_employee_count INTEGER := 0;
BEGIN
   -- Count the number of employees with a salary above 80000
   SELECT COUNT(*) INTO v_employee_count
   FROM employee
   WHERE salary > 80000;


   -- If the count is greater than 5, print the employee IDs
   IF v_employee_count > 5 THEN
      DBMS_OUTPUT.PUT_LINE('Employee IDs with salary above 80000:');
      FOR employee_rec IN (SELECT employee_id FROM employee WHERE
salary > 80000) LOOP
         DBMS_OUTPUT.PUT_LINE(employee_rec.employee_id);
      END LOOP;
   ELSE
      DBMS_OUTPUT.PUT_LINE('There are less than or equal to 5 employees
with salary above 80000.');
   END IF;
END;
/
```



Q6

```
DECLARE
   v_employee_id  employee.employee_id%TYPE := 123; -- Specify the employee ID here
   v_count        INTEGER;
BEGIN
   -- Check if a record exists for the given employee ID
   SELECT COUNT(*) INTO v_count
   FROM employee
   WHERE employee_id = v_employee_id;

   -- If no record exists, insert a new record
   IF v_count = 0 THEN
      INSERT INTO employee (employee_id) -- Assuming employee_id is the only column required
      VALUES (v_employee_id);
      COMMIT; -- Commit the transaction
```

```
        DBMS_OUTPUT.PUT_LINE('Record inserted for Employee ID ' ||
v_employee_id);
    ELSE
        DBMS_OUTPUT.PUT_LINE('Record already exists for Employee ID ' ||
v_employee_id);
    END IF;
EXCEPTION
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('Error occurred: ' || SQLERRM);
END;
/
```