

Name: Kshitij Chandrakar

Batch: 49

SAP: 500124827

1. PL/SQL Block to Check Even or Odd Number

```
DECLARE
    num NUMBER;
BEGIN
    num := :num;

    IF MOD(num, 2) = 0 THEN
        DBMS_OUTPUT.PUT_LINE('Even');
    ELSE
        DBMS_OUTPUT.PUT_LINE('Odd');
    END IF;
END;
```

Output with Input 10



The screenshot shows the Oracle APEX SQL Editor interface. At the top, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is active, displaying the output of the PL/SQL block. The output consists of three lines: 'Even', 'Statement processed.', and '0.00 seconds'. At the bottom of the interface, there is a footer with user information, session details, and copyright information.

Results	Explain	Describe	Saved SQL	History
Even				
Statement processed.				
0.00 seconds				

Footer: kshitij.124827@stu.upes.ac.in | 500124827_kshitij | en | Copyright © 1999, 2024, Oracle and/or its affiliates. | Oracle APEX 24.1.5

Output with Input 9

Enter Bind Variables — Mozilla Firefox

https://apex.oracle.com/pls/apex/f?p=4500:138:108852708084059:::

Submit

Bind Variable	Value
:INPUT_NUMBER	9

Results Explain Describe Saved SQL History

Odd

Statement processed.

0.01 seconds

kshiti124827@stu.upes.ac.in 500124827_kshiti en Copyright © 1999, 2024, Oracle and/or its affiliates. Oracle APEX 24.1.5

2. PL/SQL Block Using a WHILE Loop to Display First 10 Fibonacci Numbers

```
DECLARE
  a NUMBER := 0;
  b NUMBER := 1;
  next NUMBER;
  count1 NUMBER := 1; -- Initialize counter
```

```

BEGIN
    DBMS_OUTPUT.PUT_LINE('First 10 Fibonacci Numbers:');

    WHILE count1 <= 10 LOOP
        DBMS_OUTPUT.PUT_LINE(a);
        next := a + b;
        a := b;
        b := next;
        count1 := count1 + 1;
    END LOOP;
END;

```

The screenshot shows the Oracle APEX interface with the 'Results' tab selected. The output displays the first 10 Fibonacci numbers: 0, 1, 1, 2, 3, 5, 8, 13, 21, and 34. The interface includes tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. At the bottom, there is a footer with the user 'kshtij.124827@stu.upes.ac.in', a session ID '500124827_kshtij', and the Oracle APEX version '24.1.5'.

3. PL/SQL Block to Display Employee Names by Department ID

```

DECLARE
    dept_id NUMBER;
BEGIN
    dept_id := :dept_id;

    DBMS_OUTPUT.PUT_LINE('Employees in Department ' || dept_id || ':');

    FOR emp_rec IN (SELECT employee_name FROM employees WHERE department_id =
dept_id) LOOP
        DBMS_OUTPUT.PUT_LINE(emp_rec.employee_name);
    END LOOP;
END;
/

```

Results			
Employees in Department 1:			
Frodo Baggins			
Gollum			
Samwise Gamgee			
Statement processed.			
0.01 seconds			
<div> <div>kshtij124827@stu.upes.ac.in</div> <div>500124827_kshtij</div> <div>en</div> </div> <div>Copyright © 1999, 2024, Oracle and/or its affiliates.</div> <div>Oracle APEX 241.5</div>			

Results			
EMPLOYEE_ID EMPLOYEE_NAME SALARY DEPARTMENT_ID			
26	Gimli	52000	3
24	Galadriel	62000	2
21	Frodo Baggins	50000	1
25	Boromir	58000	4
<div> <div>kshtij124827@stu.upes.ac.in</div> <div>500124827_kshtij</div> <div>en</div> </div> <div>Copyright © 1999, 2024, Oracle and/or its affiliates.</div> <div>Oracle APEX 241.5</div>			

4. PL/SQL Block to Display Employee's Salary and Department

```
DECLARE
    emp_id NUMBER;
    emp_salary NUMBER;
    emp_dept NUMBER;
BEGIN
    emp_id := :emp_id;

    SELECT salary, department_id INTO emp_salary, emp_dept
    FROM employees
    WHERE employee_id = emp_id;

    DBMS_OUTPUT.PUT_LINE('Employee Salary: ' || emp_salary);
    DBMS_OUTPUT.PUT_LINE('Department ID: ' || emp_dept);
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.PUT_LINE('Employee not found.');
```

```
END;
```

```
/
```

Results			
Employee Salary: 30000			
Department ID: 1			
Statement processed.			
0.01 seconds			
<div> <div>kshtij124827@stu.upes.ac.in</div> <div>500124827_kshtij</div> <div>en</div> </div> <div>Copyright © 1999, 2024, Oracle and/or its affiliates.</div> <div>Oracle APEX 241.5</div>			

5. Procedure to Update Employee's Salary

```
CREATE OR REPLACE PROCEDURE update_employee_salary (
    p_emp_id IN NUMBER,
```

```

        p_increase_percentage IN NUMBER
    ) AS
        v_new_salary NUMBER;
BEGIN
    UPDATE employees
    SET salary = salary + (salary * p_increase_percentage / 100)
    WHERE employee_id = p_emp_id
    RETURNING salary INTO v_new_salary;

    DBMS_OUTPUT.PUT_LINE('Updated Salary: ' || v_new_salary);
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.PUT_LINE('Employee not found.');
```

END;

update_employee_salary(2, 10);

The first screenshot shows the execution of the SQL block. The output is 'Updated Salary: 33000' and 'Statement processed.' The execution time is 0.00 seconds.

The second screenshot shows the execution of the SQL block. The output is 'Employee Salary: 33000' and 'Department ID: 1' and 'Statement processed.' The execution time is 0.00 seconds.

6. Nested PL/SQL Block to Calculate Average, Highest, and Lowest Salaries

```

    DECLARE
        avg_salary NUMBER;
        highest_salary NUMBER := 0;
        lowest_salary NUMBER := NULL;
BEGIN
    SELECT AVG(salary) INTO avg_salary
    FROM employees;

    FOR emp_rec IN (SELECT salary FROM employees) LOOP
        IF lowest_salary IS NULL OR emp_rec.salary < lowest_salary THEN
            lowest_salary := emp_rec.salary;
        END IF;
    END LOOP;
END;
```

```
IF emp_rec.salary > highest_salary THEN
    highest_salary := emp_rec.salary;
END IF;
END LOOP;


DBMS_OUTPUT.PUT_LINE('Average Salary: ' || avg_salary);
DBMS_OUTPUT.PUT_LINE('Highest Salary: ' || highest_salary);
DBMS_OUTPUT.PUT_LINE('Lowest Salary: ' || lowest_salary);
END;
```

Results Explain Describe Saved SQL History

Average Salary: 59800
Highest Salary: 90000
Lowest Salary: 33000

1 row(s) updated.

0.01 seconds

 kshiti124827@stu.upes.ac.in  500124827_kshiti  en

Copyright © 1999, 2024, Oracle and/or its affiliates. Oracle APEX 2415