**IMAMA KAINAT**

**22F3661**

**SE-4A**

**QUESTION1**

**Question 1**

DECLARE

V\_EMPNO INT:= 7369;

V\_ENAME VARCHAR2(50):= 'SAREA';

V\_EMP\_NO EMP.EMPNO%TYPE; -- Corrected variable name

V\_EMP\_NAME EMP.ENAME%TYPE; -- Corrected variable name

V\_EMP\_JOB EMP.JOB%TYPE;

V\_EMP\_MGR EMP.MGR%TYPE;

V\_EMP\_HIREDATE EMP.HIREDATE%TYPE;

V\_EMP\_SAL EMP.SAL%TYPE;

V\_EMP\_COMM EMP.COMM%TYPE;

V\_EMP\_ROW EMP%ROWTYPE;

V\_DEPT\_ROW DEPT%ROWTYPE;

BEGIN

SELECT EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM

INTO V\_EMP\_NO, V\_EMP\_NAME, V\_EMP\_JOB, V\_EMP\_MGR, V\_EMP\_HIREDATE, V\_EMP\_SAL, V\_EMP\_COMM

FROM EMP

WHERE EMPNO = 7788;

DBMS\_OUTPUT.PUT\_LINE('EMPNO: ' || V\_EMP\_NO);

DBMS\_OUTPUT.PUT\_LINE('ENAME: ' || V\_EMP\_NAME);

DBMS\_OUTPUT.PUT\_LINE('JOB: ' || V\_EMP\_JOB);

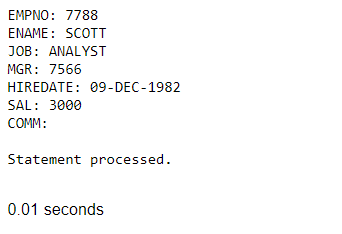
DBMS\_OUTPUT.PUT\_LINE('MGR: ' || V\_EMP\_MGR);

DBMS\_OUTPUT.PUT\_LINE('HIREDATE: ' || TO\_CHAR(V\_EMP\_HIREDATE, 'DD-MON-YYYY'));

DBMS\_OUTPUT.PUT\_LINE('SAL: ' || V\_EMP\_SAL);

DBMS\_OUTPUT.PUT\_LINE('COMM: ' || V\_EMP\_COMM);

END;



**QUESTION 2**

**DECLARE**

**EMP\_INCENTIVE INT := 5;**

**V\_EMPNO EMP.EMPNO%TYPE;**

**V\_EMPNAME EMP.ENAME%TYPE;**

**V\_EMPSAL EMP.SAL%TYPE;**

**V\_EMP\_UPDATESALARY EMP.SAL%TYPE; -- Added for the updated salary**

**BEGIN**

**SELECT EMPNO, ENAME, SAL, SAL \* EMP\_INCENTIVE**

**INTO V\_EMPNO, V\_EMPNAME, V\_EMPSAL, V\_EMP\_UPDATESALARY**

**FROM EMP**

**WHERE EMPNO = 7788;**

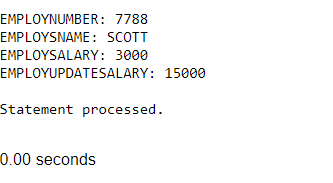
**DBMS\_OUTPUT.PUT\_LINE('EMPLOYNUMBER: ' || V\_EMPNO);**

**DBMS\_OUTPUT.PUT\_LINE('EMPLOYSNAME: ' || V\_EMPNAME);**

**DBMS\_OUTPUT.PUT\_LINE('EMPLOYSALARY: ' || V\_EMPSAL);**

**DBMS\_OUTPUT.PUT\_LINE('EMPLOYUPDATESALARY: ' || V\_EMP\_UPDATESALARY);**

**END;**

****

**QUESTION 3**

**DECLARE**

**N NUMBER := 1432;**

**R NUMBER;**

**BEGIN**

**R := MOD(n, 2);**

**IF R = 0 THEN**

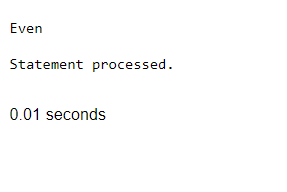
**dbms\_output.Put\_line('Even');**

**ELSE**

**dbms\_output.Put\_line('Odd');**

**END IF;**

**END;**

****

**DECLARE**

**N NUMBER := 1431;**

**R NUMBER;**

**BEGIN**

**R := MOD(n, 2);**

**IF R = 0 THEN**

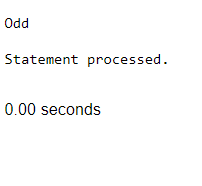
**dbms\_output.Put\_line('Even');**

**ELSE**

**dbms\_output.Put\_line('Odd');**

**END IF;**

**END;**

****

**QUESTION 4**

**DECLARE**

**num1 NUMBER := 14;**

**BEGIN**

**IF num1 < 0 THEN**

**DBMS\_OUTPUT.PUT\_LINE ('The number '||num1||' is a negative number');**

**ELSIF num1 = 0 THEN**

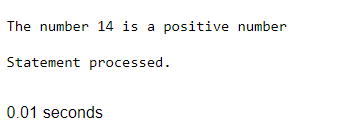
**DBMS\_OUTPUT.PUT\_LINE ('The number '||num1||' is equal to zero');**

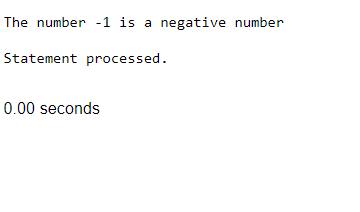
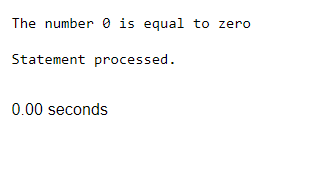
**ELSE**

**DBMS\_OUTPUT.PUT\_LINE ('The number '||num1||' is a positive number');**

**END IF;**

**END;**

****

****

**QUESTION 5**

**DECLARE**

**V\_TOTAL\_EMPLOYEES INT;**

**V\_VACANCIES INT := 45;**

**BEGIN**

**SELECT COUNT(EMPNO) INTO V\_TOTAL\_EMPLOYEES**

**FROM EMP**

**WHERE DEPTNO = 50;**

**DBMS\_OUTPUT.PUT\_LINE('TOTAL NUMBER OF EMPLOYEES IN DEPARTMENT 50: ' || V\_TOTAL\_EMPLOYEES);**

**IF V\_TOTAL\_EMPLOYEES < V\_VACANCIES THEN**

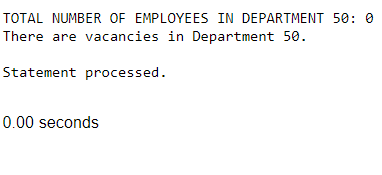
**DBMS\_OUTPUT.PUT\_LINE('There are vacancies in Department 50.');**

**ELSE**

**DBMS\_OUTPUT.PUT\_LINE('No vacancies in Department 50.');**

**END IF;**

**END;**

****

**QUESTION 6**

**DECLARE**

**A NUMBER := 10;**

**B NUMBER := 5;**

**OPERATOR CHAR(1) := '/'; -- Change the operator to perform different operations**

**RESULT NUMBER;**

**BEGIN**

**CASE OPERATOR**

**WHEN '+' THEN**

**RESULT := A + B;**

**DBMS\_OUTPUT.PUT\_LINE('Addition Result: ' || RESULT);**

**WHEN '-' THEN**

**RESULT := A - B;**

**DBMS\_OUTPUT.PUT\_LINE('Subtraction Result: ' || RESULT);**

**WHEN '\*' THEN**

**RESULT := A \* B;**

**DBMS\_OUTPUT.PUT\_LINE('Multiplication Result: ' || RESULT);**

**WHEN '/' THEN**

**IF B != 0 THEN**

**RESULT := A / B;**

**DBMS\_OUTPUT.PUT\_LINE('Division Result: ' || RESULT);**

**ELSE**

**DBMS\_OUTPUT.PUT\_LINE('Error: Division by zero');**

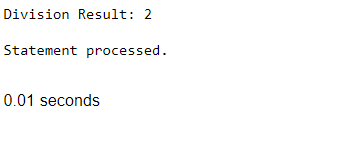
**END IF;**

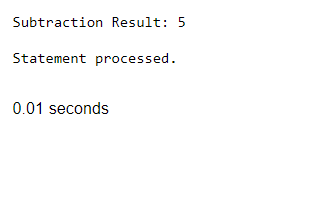
**ELSE**

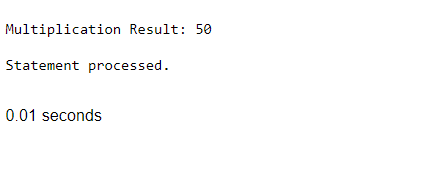
**DBMS\_OUTPUT.PUT\_LINE('Error: Invalid Operator');**

**END CASE;**

**END;**

****

****

****

**QUESTION 7**

**DECLARE**

**num INT := 5; -- Change the number to calculate its factorial**

**factorial INT := 1;**

**BEGIN**

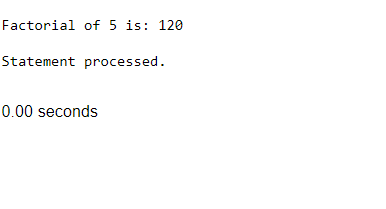
**FOR i IN 1..num LOOP**

**factorial := factorial \* i;**

**END LOOP;**

**DBMS\_OUTPUT.PUT\_LINE('Factorial of ' || num || ' is: ' || factorial);**

**END;**

****

**QUESTION 8**

**DECLARE**

**v\_empno EMP.EMPNO%TYPE := 7788;**

**v\_new\_salary EMP.SAL%TYPE;**

**v\_mid\_range\_salary NUMBER;**

**v\_old\_salary EMP.SAL%TYPE;**

**v\_job EMP.JOB%TYPE;**

**BEGIN**

**SELECT SAL, JOB INTO v\_old\_salary, v\_job**

**FROM EMP**

**WHERE EMPNO = v\_empno;**

**SELECT AVG(SAL) INTO v\_mid\_range\_salary**

**FROM EMP**

**WHERE JOB = v\_job;**

**IF v\_old\_salary > v\_mid\_range\_salary THEN**

**v\_new\_salary := v\_old\_salary \* 1.08; -- Increase salary by 8%**

**ELSE**

**v\_new\_salary := v\_mid\_range\_salary;**

**END IF;**

**UPDATE EMP**

**SET SAL = v\_new\_salary**

**WHERE EMPNO = v\_empno;**

**IF v\_old\_salary > v\_mid\_range\_salary THEN**

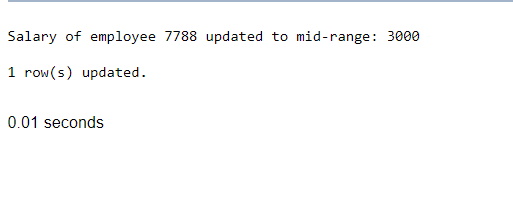
**DBMS\_OUTPUT.PUT\_LINE('Salary of employee ' || v\_empno || ' increased by 8% to ' || v\_new\_salary);**

**ELSE**

**DBMS\_OUTPUT.PUT\_LINE('Salary of employee ' || v\_empno || ' updated to mid-range: ' || v\_new\_salary);**

**END IF;**

**END;**

****

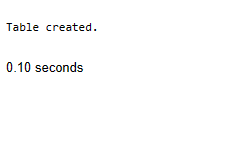
**QUESTION 9**

**CREATE TABLE emp\_TEMPORARY(**

**emp\_id NUMBER,**

**emp\_email VARCHAR2(100)**

**);**

****

**DECLARE**

**NUMBER\_OF\_EMP NUMBER;**

**BEGIN**

**SELECT COUNT(EMPLOYEE\_ID) INTO NUMBER\_OF\_EMP**

**FROM EMP;**

**FOR EMP\_REC IN (SELECT EMPLOYEE\_ID FROM EMP) LOOP**

**INSERT INTO EMP\_TEMP (EMP\_ID, EMP\_EMAIL)**

**VALUES(EMP\_REC.EMPLOYEE\_ID, 'NOT AVAILABLE NOW');**

**END LOOP;**

**DBMS\_OUTPUT.PUT\_LINE('INSERTION COMPLETED SUCCESSFULLY.');**

**EXCEPTION**

**WHEN OTHERS THEN**

**DBMS\_OUTPUT.PUT\_LINE('AN ERROR OCCURRED: ' || SQLERRM);**

**END;**