**Class Test 12**

**PL/SQL**

**Part 01:**

1. Write a query that can multiply two numbers taking input from user.
2. Write a query that can add two numbers if the numbers are equal. Use CASE Statement.
3. Write a query that can check if two strings are equal or not. Use IF-THEN-ELSIF Statement.
4. Write a query that can multiply two numbers. If the result obtained is less than 100, **Hi** is displayed, if the result obtained is more than 100, **Bye** is displayed and if the result obtained is equal to 100, **ADBMS** is displayed. Use IF-THEN-ELSIF Statement
5. Write a query that can check if two numbers are equal or not. Use CASE Statement.

**Part 02:**

*To solve the following use the scott schema*

1. Write a query that can display the salary of employee ALLEN. If ALLEN’s salary is greater than 2000 display ‘SALARY GREATER THAN 2000’ and If not then display ‘SALARY LESS THAN 2000’.
2. Write a query that can ask user to input the EMPNO of employee WARD and display his salary.
3. Write a query that can ask user to input the EMPNO of employee BLAKE,CLARK and TURNER and display their respective salary.
4. Write a query that can ask user to input the EMPNO of employee BLAKE,CLARK and TURNER and display their respective salary, add the salaries and display the total.
5. Write a query that displays the commission of employee SMITH. If SMITH’s commission is NULL. Display ‘NOT APPLICABLE FOR COMMISSION’

**Part 03:**

*Find the output*

DECLARE

a number:=10;

b number;

BEGIN

b:=(+a); --identity

dbms\_output.put\_line(b);

b:=(-a); --negation

dbms\_output.put\_line(b);

END;

/

**Answer:**

*--server on*

**SET** SERVEROUTPUT **ON**

*-----Part 01:*

1. **DECLARE**

a NUMBER := &a;

b NUMBER := &b;

**c** NUMBER;

**BEGIN**

**c** := a \* b;

dbms\_output.put\_line('The result of multiplication is: ' || **c**);

**END**;

/

2. **DECLARE**

num1 NUMBER;

num2 NUMBER;

**result** NUMBER;

**BEGIN**

num1 := &input\_number1;

num2 := &input\_number2;

**CASE**

**WHEN** num1 = num2 **THEN**

**result** := num1 + num2;

**ELSE**

**result** := 0;

**END** **CASE**;

dbms\_output.put\_line('The result of addition is: ' || **result**);

**END**;

/

3. **DECLARE**

str1 VARCHAR2(100);

str2 VARCHAR2(100);

**BEGIN**

str1 := '&input\_string1';

str2 := '&input\_string2';

IF str1 = str2 **THEN**

dbms\_output.put\_line('The strings are equal.');

**ELSE**

dbms\_output.put\_line('The strings are not equal.');

**END** IF;

**END**;

/

4. **DECLARE**

num1 NUMBER;

num2 NUMBER;

**result** NUMBER;

**BEGIN**

num1 := &input\_number1;

num2 := &input\_number2;

**result** := num1 \* num2;

IF **result** < 100 **THEN**

dbms\_output.put\_line('Hi');

ELSIF **result** > 100 **THEN**

dbms\_output.put\_line('Bye');

**ELSE**

dbms\_output.put\_line('ADBMS');

**END** IF;

**END**;

/

5. **DECLARE**

num1 NUMBER;

num2 NUMBER;

**result** VARCHAR2(20);

**BEGIN**

num1 := &input\_number1;

num2 := &input\_number2;

**result** := **CASE**

**WHEN** num1 = num2 **THEN** 'Equal'

**ELSE** 'Not Equal'

**END**;

dbms\_output.put\_line('The numbers are ' || **result**);

**END**;

/

*------Part 02:*

6. **DECLARE**

emp\_salary NUMBER;

**BEGIN**

**SELECT** sal **INTO** emp\_salary **FROM** emp **WHERE** ename = 'ALLEN';

IF emp\_salary > 2000 **THEN**

dbms\_output.put\_line('SALARY GREATER THAN 2000');

**ELSE**

dbms\_output.put\_line('SALARY LESS THAN 2000');

**END** IF;

**END**;

/

7. **DECLARE**

empno\_input NUMBER;

emp\_salary NUMBER;

**BEGIN**

empno\_input := &input\_empno;

**BEGIN**

**SELECT** sal **INTO** emp\_salary **FROM** emp **WHERE** empno = empno\_input;

dbms\_output.put\_line('The salary of the employee with EMPNO ' || empno\_input || ' is: ' || emp\_salary);

EXCEPTION

**WHEN** NO\_DATA\_FOUND **THEN**

dbms\_output.put\_line('Employee with EMPNO ' || empno\_input || ' not found.');

**END**;

**END**;

/

8. **DECLARE**

empno\_blake NUMBER;

empno\_clark NUMBER;

empno\_turner NUMBER;

salary\_blake NUMBER;

salary\_clark NUMBER;

salary\_turner NUMBER;

**BEGIN**

empno\_blake := &input\_empno\_blake;

empno\_clark := &input\_empno\_clark;

empno\_turner := &input\_empno\_turner;

**SELECT** sal **INTO** salary\_blake **FROM** emp **WHERE** empno = empno\_blake;

**SELECT** sal **INTO** salary\_clark **FROM** emp **WHERE** empno = empno\_clark;

**SELECT** sal **INTO** salary\_turner **FROM** emp **WHERE** empno = empno\_turner;

dbms\_output.put\_line('Salary of BLAKE: ' || salary\_blake);

dbms\_output.put\_line('Salary of CLARK: ' || salary\_clark);

dbms\_output.put\_line('Salary of TURNER: ' || salary\_turner);

**END**;

/

9. **DECLARE**

empno\_blake NUMBER;

empno\_clark NUMBER;

empno\_turner NUMBER;

salary\_blake NUMBER;

salary\_clark NUMBER;

salary\_turner NUMBER;

total\_salary NUMBER;

**BEGIN**

empno\_blake := &input\_empno\_blake;

empno\_clark := &input\_empno\_clark;

empno\_turner := &input\_empno\_turner;

**SELECT** sal **INTO** salary\_blake **FROM** emp **WHERE** empno = empno\_blake;

**SELECT** sal **INTO** salary\_clark **FROM** emp **WHERE** empno = empno\_clark;

**SELECT** sal **INTO** salary\_turner **FROM** emp **WHERE** empno = empno\_turner;

total\_salary := salary\_blake + salary\_clark + salary\_turner;

dbms\_output.put\_line('Salary of BLAKE: ' || salary\_blake);

dbms\_output.put\_line('Salary of CLARK: ' || salary\_clark);

dbms\_output.put\_line('Salary of TURNER: ' || salary\_turner);

dbms\_output.put\_line('Total Salary: ' || total\_salary);

**END**;

/

10. **DECLARE**

emp\_commission NUMBER;

**BEGIN**

**SELECT** comm **INTO** emp\_commission **FROM** emp **WHERE** ename = 'SMITH';

IF emp\_commission **IS** **NULL** **THEN**

dbms\_output.put\_line('NOT APPLICABLE FOR COMMISSION');

**ELSE**

dbms\_output.put\_line('Commission of SMITH: ' || emp\_commission);

**END** IF;

**END**;

/

*-----Part 03:*

11. **Output**:

10

-10