

## **QUIZ – 5 ANSWERS**

### **1 Answer:-**

#### **Table creation:**

```
CREATE TABLE Employee_Salaries (Ssn CHAR(9), Salary VARCHAR(10), Log_Date DATE);
```

#### **Trigger Creation:**

```
CREATE OR REPLACE TRIGGER trg_emp
AFTER INSERT OR UPDATE OF Salary
ON Employee_Salaries
FOR EACH ROW
DECLARE thisEMP Employee_Salaries%ROWTYPE;
BEGIN
IF (thisEMP.Salary > 60000) THEN
INSERT INTO Employee_Salaries values (thisEMP.Ssn, 'HIGH', thisEMP.Log_Date);
ELSIF ((thisEMP.Salary >= 40000) and (thisEMP.Salary <=60000)) THEN
INSERT INTO Employee_Salaries values (thisEMP.Ssn, 'MEDIUM', thisEMP.Log_Date);
ELSE INSERT INTO Employee_Salaries values (thisEMP.Ssn, 'LOW', thisEMP.Log_Date);
END;
```

### **2 Answer:-**

```
CREATE OR REPLACE PROCEDURE overdue AS
BEGIN
DECLARE
CURSOR book_due IS
SELECT BOOK.Book_id, BOOK.Title, BORROWER.Name, BOOK_LOANS.Due_date FROM BOOK,
BORROWER, BOOK_LOANS WHERE BOOK.Book_id = BOOK_LOANS.Book_id AND
BORROWER.Card_no = BOOK_LOANS.Card_no AND SYSDATE > BOOK_LOANS.Due_date AND
BOOK_LOANS.Return_date IS NULL;
BookName BOOK.TITLE%TYPE;
BorrowName BORROWER.NAME%TYPE;
BEGIN
OPEN book_due;
LOOP
```

```
FETCH book_due INTO BookName, BorrowName;
EXIT WHEN (book_due%NOTFOUND);
DBMS_OUTPUT.PUT_LINE ('Book Title: ' || BookName || ' Book Borrower: ' || BorrowName);
END LOOP;
CLOSE book_due;
END;
END overdue;
```