

Data & Information Management
8.2D: PL/SQL (Trigger, Stored Procedures and Functions)

Task 01

Write a trigger named TRG_UPDATE_CUST_BALANCE to update the CUST_BALANCE in the CUSTOMER table when a new invoice is added in the INVOICE table. The value of the INV_AMOUNT column of the new invoice should be added to the respective customer's balance in the CUSTOMER table.

```
CREATE OR REPLACE TRIGGER TRG_UPDATE_CUST-BALANCE
AFTER INSERT ON INVOICE
FOR EACH ROW
BEGIN
    UPDATE CUSTOMER
    SET CUST_BALANCE = CUST_BALANCE +: NEW.INV_AMOUNT
    WHERE CUST-_NUM =: NEW.CUST-NUM;
END;
```

TO VARIFY,

```
SELECT * FROM CUSTOMER;
INSERT INTO INVOICE VALUES ('8005','1001','27-Apr-2016','225.40');
SELECT * FROM CUSTOMER;
```

2. Task 02

Write a procedure named PRC_ADD_CUSTOMER to add a new customer to the CUSTOMER table. The procedure will take four IN arguments which are values of the four columns of the CUSTOMER table for the new customer to be added.

```
CREATE OR REPLACE PROCEDURE PRC_ADD_CUSTOMER (
    CUST_NUM IN CUSTOMER.CUST_NUM%TYPE,
    CUST_LNAME IN CUSTOMER.CUST_LNAME%TYPE,
    CUST_FNAME IN CUSTOMER.CUST_FNAME%TYPE,
    CUST_BALANCE IN CUSTOMER.CUST_BALANCE%TYPE
)
AS
BEGIN
    INSERT INTO CUSTOMER (CUST_NUM, CUST_LNAME, CUST_FNAME, CUST_BALANCE)
    VALUES (CUST_NUM, CUST_LNAME, CUST_FNAME, CUST_BALANCE);
    COMMIT;
    DBMS_OUTPUT.PUT_LINE ('Customer added successfully');
END;
```

TO VERIFY,

```
SELECT * FROM CUSTOMER;
EXEC PRC_ADD_CUSTOMER ('1002','Rauthor','Peter',0.0');
SELECT * FROM CUSTOMER;
```

Task 03

Write a procedure named PRC_ADD_INVOICE to add a new invoice record to the INVOICE table. The procedure will take four IN arguments which are values of the four columns of the INVOICE table for the new invoice to be added.

```
CREATE OR REPLACE PROCEDURE PRC_ADD_INVOICE (
    INV_NUM IN INVOICE.INV_NUM%TYPE,
    CUST_NUM IN INVOICE.CUST_NUM%TYPE,
    INV_DATE IN INVOICE.INV_DATE%TYPE,
    INV_AMOUNT IN INVOICE.INV_AMOUNT%TYPE
)
AS
BEGIN
    INSERT INTO INVOICE (INV_NUM, CUST_NUM, INV-DATE, INV_AMOUNT)
    VALUES (INV_NUM, CUST_NUM, TO_DATE (INV_DATE, 'DD-MM-YYYY') INV-AMOUNT);
    COMMIT;
    DBMS_OUTPUT.PUT_LINE ('Invoice added successfully');
END;
```

TO VARIFY,

```
SELECT * FROM CUSTOMER;
SELECT * FROM INVOICE;
EXEC PRC_ADD_INVOICE ('8006','1002','29-Apr-2016','175.85');
SELECT * FROM INVOICE;
SELECT * FROM CUSTOMER;
```

Task 04

Write a function named `get_invoice_count` to retrieve the number of invoices for a given customer. Your function will take `CUST_NUM` as argument and returns the number of invoices for the customer.

```
CREATE OR REPLACE FUNCTION get_invoice_count (  
  cust_num_param IN INVOICE.CUST_NUM%TYPE  
) RETURN NUMBER  
IS  
  Invoice_count NUMBER;  
BEGIN  
  SELECT COUNT (*) INTO invoice_count  
  FROM INVOICE  
  WHERE CUST_NUM = cust_num_param;  
  RETURN invoice_count;  
END;
```

TO VARIFY,

```
SELECT get_invoice_count (0) FROM DUAL;  
SELECT get_invoice_count (1000) FROM DUAL;  
SELECT get_invoice_count (1005) FROM DUAL;
```