

Exercise # 6

- 1) Write a procedure to fetch data from table SALES for a given parameter orderid and display the data(use ROWTYPE to capture the data)
- 2) Modify the above procedure to return the row you have stored in the ROWTYPE variable using an OUT parameter.
- 3) Write a procedure to call the above procedure and display the data.
- 4) Perform the following steps.
 - Create a table SALES_COPY which is similar to SALES table.
 - Write a procedure to call the procedure you have created in #2 and insert the data in the SALES_COPY table.
- 5) Write a procedure to call the procedure you have created in #2 and update the column TOTAL_AMOUNT to SALES_AMOUNT + TAX_AMOUNT in the SALES table.
- 6) Write a procedure to fetch SALES_DATE, ORDER_ID, PRODUCT_ID, CUSTOMER_ID and QUANTITY from SALES table and display the data. (Use a User defined record type).

Answers

1)

```
CREATE OR REPLACE PROCEDURE FETCH_SALES (S_ORDERID NUMBER)
AS
  SALE_REC SALES%ROWTYPE;
BEGIN

  SELECT SALES_DATE, ORDER_ID, PRODUCT_ID, CUSTOMER_ID, SALESPERSON_ID, QUANTITY,
  UNIT_PRICE, SALES_AMOUNT, TAX_AMOUNT, TOTAL_AMOUNT
  INTO
  SALE_REC
  FROM SALES
  WHERE ORDER_ID = S_ORDERID;

  DBMS_OUTPUT.PUT_LINE (SALE_REC.SALES_DATE);
  DBMS_OUTPUT.PUT_LINE (SALE_REC.ORDER_ID);
  DBMS_OUTPUT.PUT_LINE (SALE_REC.PRODUCT_ID);
  DBMS_OUTPUT.PUT_LINE (SALE_REC.CUSTOMER_ID);
  DBMS_OUTPUT.PUT_LINE (SALE_REC.SALESPERSON_ID);
  DBMS_OUTPUT.PUT_LINE (SALE_REC.QUANTITY);
  DBMS_OUTPUT.PUT_LINE (SALE_REC.UNIT_PRICE);
  DBMS_OUTPUT.PUT_LINE (SALE_REC.SALES_AMOUNT);
  DBMS_OUTPUT.PUT_LINE (SALE_REC.TAX_AMOUNT);
  DBMS_OUTPUT.PUT_LINE (SALE_REC.TOTAL_AMOUNT);

EXCEPTION
  WHEN NO_DATA_FOUND THEN
    DBMS_OUTPUT.PUT_LINE('No such Order!');
  WHEN TOO_MANY_ROWS THEN
    DBMS_OUTPUT.PUT_LINE('You got more than 1 row!');
  WHEN OTHERS THEN
    DBMS_OUTPUT.PUT_LINE('Error!');
END;
```

2)

```
CREATE OR REPLACE PROCEDURE FETCH_SALES (S_ORDERID NUMBER, S_REC OUT
SALES%ROWTYPE)
AS
SALE_REC SALES%ROWTYPE;
BEGIN

SELECT SALES_DATE, ORDER_ID, PRODUCT_ID, CUSTOMER_ID, SALESPERSON_ID, QUANTITY,
UNIT_PRICE, SALES_AMOUNT, TAX_AMOUNT, TOTAL_AMOUNT
INTO
SALE_REC
FROM SALES
WHERE ORDER_ID = S_ORDERID;

S_REC := SALE_REC;

EXCEPTION
WHEN NO_DATA_FOUND THEN
    DBMS_OUTPUT.PUT_LINE('No such Order!');
WHEN TOO_MANY_ROWS THEN
    DBMS_OUTPUT.PUT_LINE('You got more than 1 row!');
WHEN OTHERS THEN
    DBMS_OUTPUT.PUT_LINE('Error!');
END;
```

3)

```
CREATE OR REPLACE PROCEDURE GET_SALES (S_ORD NUMBER)
AS
SLE_REC SALES%ROWTYPE;
BEGIN
FETCH_SALES (S_ORD, SLE_REC);
    DBMS_OUTPUT.PUT_LINE (SLE_REC.SALES_DATE);
    DBMS_OUTPUT.PUT_LINE (SLE_REC.ORDER_ID);
    DBMS_OUTPUT.PUT_LINE (SLE_REC.PRODUCT_ID);
    DBMS_OUTPUT.PUT_LINE (SLE_REC.CUSTOMER_ID);
    DBMS_OUTPUT.PUT_LINE (SLE_REC.SALESPERSON_ID);
    DBMS_OUTPUT.PUT_LINE (SLE_REC.QUANTITY);
    DBMS_OUTPUT.PUT_LINE (SLE_REC.UNIT_PRICE);
    DBMS_OUTPUT.PUT_LINE (SLE_REC.SALES_AMOUNT);
    DBMS_OUTPUT.PUT_LINE (SLE_REC.TAX_AMOUNT);
    DBMS_OUTPUT.PUT_LINE (SLE_REC.TOTAL_AMOUNT);
END;

EXEC GET_SALES (1269);
```

4)

```
CREATE TABLE SALES_COPY AS  
SELECT * FROM SALES WHERE 1 = 2;
```

```
CREATE OR REPLACE PROCEDURE INSERT_SALES (S_ORD NUMBER)  
AS  
SLE_REC SALES%ROWTYPE;  
BEGIN  
FETCH_SALES (S_ORD, SLE_REC);
```

```
INSERT INTO SALES_COPY VALUES (  
    SLE_REC.SALES_DATE,  
    SLE_REC.ORDER_ID,  
    SLE_REC.PRODUCT_ID,  
    SLE_REC.CUSTOMER_ID,  
    SLE_REC.SALESPERSON_ID,  
    SLE_REC.QUANTITY,  
    SLE_REC.UNIT_PRICE,  
    SLE_REC.SALES_AMOUNT,  
    SLE_REC.TAX_AMOUNT,  
    SLE_REC.TOTAL_AMOUNT  
);  
COMMIT;  
END;
```

```
EXEC INSERT_SALES (1268);
```

5)

```
CREATE OR REPLACE PROCEDURE UPDATE_SALES (S_ORD NUMBER)  
AS  
SLE_REC SALES%ROWTYPE;  
BEGIN  
FETCH_SALES (S_ORD, SLE_REC);
```

```
UPDATE SALES SET TOTAL_AMOUNT = SLE_REC.SALES_AMOUNT + SLE_REC.TAX_AMOUNT  
WHERE ORDER_ID = S_ORD;
```

```
COMMIT;  
END;
```

```
EXEC UPDATE_SALES (1268);
```

6)

```
CREATE OR REPLACE PROCEDURE FETCH_SALES_USER (S_ORDERID NUMBER)
AS
  TYPE SALES_INFO_REC IS RECORD
  (
    SALES_DATE SALES.SALES_DATE%TYPE,
    ORDER_ID   SALES.ORDER_ID%TYPE,
    PRODUCT_ID SALES.PRODUCT_ID%TYPE,
    CUSTOMER_ID SALES.CUSTOMER_ID%TYPE,
    QUANTITY   SALES.QUANTITY%TYPE
  );

  SALE_REC SALES_INFO_REC;

BEGIN

  SELECT SALES_DATE, ORDER_ID, PRODUCT_ID, CUSTOMER_ID, QUANTITY
  INTO
  SALE_REC
  FROM SALES
  WHERE ORDER_ID = S_ORDERID;

  DBMS_OUTPUT.PUT_LINE (SALE_REC.SALES_DATE);
  DBMS_OUTPUT.PUT_LINE (SALE_REC.ORDER_ID);
  DBMS_OUTPUT.PUT_LINE (SALE_REC.PRODUCT_ID);
  DBMS_OUTPUT.PUT_LINE (SALE_REC.CUSTOMER_ID);
  DBMS_OUTPUT.PUT_LINE (SALE_REC.QUANTITY);

EXCEPTION
  WHEN NO_DATA_FOUND THEN
    DBMS_OUTPUT.PUT_LINE('No such Order!');
  WHEN TOO_MANY_ROWS THEN
    DBMS_OUTPUT.PUT_LINE('You got more than 1 row!');
  WHEN OTHERS THEN
    DBMS_OUTPUT.PUT_LINE('Error!');
END;

EXEC FETCH_SALES_USER(1268);
```