|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY | | | | | | | | | | | | | | |
| **grey_logo_portrait\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Signature** | **Development Software 2B**  DSO23BT | | | | | | | | | | | | | |
| Class Test  **Time: 50 min**  **Full Marks: 45** | | | | | | **No on Classlist:** | | | | | | | |
| **Student number:** | | | | | | | | | | | | | |
|  |  |  |  |  | | |  |  | |  | |  |
| Surname: | | | | | Initials: | | | | **/40** | | **%** | | |

**Instructions:**

1. Answer the questions in the space provided.
2. Use the Kaleidoscope Database on p.18.

## Question 1 [15]

Create a PL/SQL block that declares a record based on the following fields from the BK\_ORDERITEMS, BK\_BOOKS and BK\_COSTS tables:

QUANTITY (from BK\_ORDERITEMS),

TITLE, AUTH (from BK\_BOOKS) and

RETAIL (from BK\_COSTS)

LINETOTAL (calculated: QUANTITY \* RETAIL)

Use a SELECT query to retrieve values from the three tables into the record based on the **order number** (ORDER#) and **item number** (ITEM#) entered at the prompt. Join the tables using the ISBN number from these three tables. Print the fields in the record. Use the proper predefined Oracle exception, when no data is returned.

i.e.

Enter value for o\_no: 1003

Enter value for i\_no: 1

old 19: I.ORDER# = &O\_NO AND I.ITEM# = &I\_NO;

new 19: I.ORDER# = 1003 AND I.ITEM# = 1;

1 of DATABASE IMPLEMENTATION by P105 sold @ 55.95 = 55.95

PL/SQL procedure successfully completed.

**SET SERVEROUTPUT ON**

**DECLARE**

**TYPE ORDER\_TYPE IS RECORD ✓✓**

**(O\_QUANTITY NUMBER(2),**

**O\_TITLE VARCHAR2(30), ✓**

**O\_AUTHORID VARCHAR2(4),**

**O\_RETAIL NUMBER(5,2), ✓**

**O\_LINETOTAL NUMBER(5,2));**

**O\_REC ORDER\_TYPE; ✓**

**BEGIN**

**SELECT QUANTITY, TITLE, AUTHORID, RETAIL, ✓**

**(QUANTITY \* RETAIL) AS LINETOTAL ✓✓**

**INTO O\_REC ✓**

**FROM BK\_ORDERITEMS I, BK\_BOOKS B , BK\_COSTS C ✓**

**WHERE I.ISBN = B.ISBN**

**AND I.ISBN = C.ISBN**

**AND I.ORDER# = &O\_NO**

**AND I.ITEM# = &I\_NO; ✓✓**

**DBMS\_OUTPUT.PUT\_LINE(O\_REC.O\_QUANTITY ||' of '|| O\_REC.O\_TITLE**

**||' by '|| O\_REC.O\_AUTHORID || ' sold @ '**

**|| O\_REC.O\_RETAIL ||' = '|| O\_REC.O\_LINETOTAL); ✓**

**EXCEPTION**

**WHEN NO\_DATA\_FOUND THEN ✓**

**DBMS\_OUTPUT.PUT\_LINE('NO SUCH ORDERITEM'); ✓**

**END; [15]**

## Question 2 [20]

Kaleidoscope book shop from time to time need to know how many copies of a certain book has been sold.

Declare a table of type NUMBER(4).

Declare a cursor that receives one parameter (the ISBN number as supplied by the user). The cursor must retrieve all records from the BK\_ORDERITEMS table with this ISBN number.

Using a basic loop, read through the cursor, storing the QUANTITY sold in consecutive elements of the table. Use a counter to determine the number of items stored in the table. Then use a for loop to read through the table, finding the sum of all the elements. Display the sum.

i.e.

## Enter value for i\_isbn: 8843172113

## old 12: OPEN C\_ITEMS(&I\_ISBN);

## new 12: OPEN C\_ITEMS(8843172113);

## We sold: 7 of the book

## PL/SQL procedure successfully completed

**SET SERVEROUTPUT ON**

**DECLARE**

**TYPE I\_TABLE\_TYPE IS TABLE OF NUMBER(3) ✓✓**

**INDEX BY BINARY\_INTEGER; ✓**

**I\_TABLE I\_TABLE\_TYPE; ✓**

**CURSOR C\_ITEMS (p\_ISBNno BK\_ORDERITEMS.ISBN%TYPE) IS ✓✓**

**SELECT \***

**FROM BK\_ORDERITEMS ✓**

**WHERE ISBN =p\_ ISBNno; ✓**

**I\_REC BK\_ORDERITEMS%ROWTYPE; ✓**

**c BINARY\_INTEGER := 0; ✓**

**TOT NUMBER(4) := 0; ✓**

**BEGIN**

**OPEN C\_ITEMS(&I\_ISBN); ✓**

**LOOP**

**FETCH C\_ITEMS INTO I\_REC; ✓**

**EXIT WHEN C\_ITEMS%NOTFOUND; ✓**

**C := C + 1; ✓**

**I\_TABLE(C) := I\_REC.QUANTITY; ✓**

**END LOOP;**

**FOR CNT IN 1..C LOOP ✓**

**TOT := TOT + I\_TABLE(CNT); ✓✓**

**END LOOP;**

**DBMS\_OUTPUT.PUT\_LINE('We sold: ' || TOT || ' of the book'); ✓**

**END; [20]**

Question 3 [10]

We want to accomplish the same result as in Question 2 with the following difference.

Declare a cursor that receives one parameter (the ISBN number as supplied by the user). The cursor must retrieve all records from the BK\_ORDERITEMS table with this ISBN number.

Now declare a table of the same type as that of the cursor.

Using a basic loop, read through the cursor, storing the rows from the cursor in consecutive elements of the table. Use a counter to determine the number of items stored in the table. Then use a for loop to read through the table, finding the sum of all the QUANTITY sold attributes. Display the sum.

i.e.

## Enter value for i\_isbn: 8843172113

## old 12: OPEN C\_ITEMS(&I\_ISBN);

## new 12: OPEN C\_ITEMS(8843172113);

## We sold: 7 of the book

## PL/SQL procedure successfully completed

**SET SERVEROUTPUT ON**

**DECLARE**

**CURSOR C\_ITEMS ( ISBNno BK\_ORDERITEMS.ISBN%TYPE) IS**

**SELECT \***

**FROM BK\_ORDERITEMS**

**WHERE ISBN = ISBNno;**

**TYPE I\_TABLE\_TYPE IS TABLE OF C\_ITEMS%ROWTYPE ✓✓**

**INDEX BY BINARY\_INTEGER; ✓**

**I\_TABLE I\_TABLE\_TYPE; ✓**

**c BINARY\_INTEGER := 1; ✓**

**TOT NUMBER(4) := 0;**

**BEGIN**

**OPEN C\_ITEMS(&I\_ISBN);**

**LOOP**

**FETCH C\_ITEMS INTO I\_TABLE(C); ✓✓**

**EXIT WHEN C\_ITEMS%NOTFOUND;**

**C := C + 1;**

**END LOOP;**

**C := C - 1; ✓**

**FOR CNT IN 1..C LOOP**

**TOT := TOT + I\_TABLE(CNT).QUANTITY; ✓✓**

**END LOOP;**

**DBMS\_OUTPUT.PUT\_LINE('We sold: ' || TOT || ' of the book');**

**END; [10]**

--MARK ONLY WHERE DIFFERENT FROM Question 2