**Lab objectives:**

**- Embed SQL statements in two RPGLE programs**

**- Provide related RPGLE code for the embedded SQL statements**

**Requirements to pass the lab:**

**Successfully run the PhoneSQL program after copying supplied code and then adding the embedded SQL code and related RPGLE code that was given out in class. Be able to set the overflow line at 30 before running your program.**

**Successfully run the OrdersSQL SQL cursor program after copying supplied code and then adding the embedded SQL code and related RPGLE code that was given out in class.**

PhoneSQL

Input: CUSTSU22 (accessed by RPGLE read statement)

Graphical user interface, text, application, email

Description automatically generated

CONTACTS22 (accessed through embedded SQL statement)

Graphical user interface, text, application, email

Description automatically generated

Output:

Table

Description automatically generated

CONTACTS22

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Lentch | Description |
| CONTACTN | Packed Dec | 5,0 | Contact Number |
| CNUMBER | Packed Dec | 6,0 | Customer Number |
| LCALLD | Date | 10 | Last Date Called |
| CALLBINT | Zoned Deci | 3,0 | Callback Interval in Days |
| CNTNAME | Char | 30 | Contact Name |
| PHONE | Packed Dec | 10,0 | Telephone Number |
| SLSNUMBER | Zoned Dec | 3,0 | Sales Person Number |
| PHNCOM | Char | 25 | Comment |

PHONESQL (code available in BCI433LIB/LAB9SU22)

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CustSu22 READ BY NATIVE LANGUAGE (ILE RPG)

// ContactS22 ROW RETRIEVED WITH EMBEDDED SQL

// DETAIL REPORT LINE INCLUDES INFORMATION FROM CustSu22 AND CONTACTW

// SUMMARRY REPORT INFORMATION RETRIEVED WITH EMBEDDED SQL STATEMENTS

DCL-F PHNREPORT PRINTER OFLIND(\*IN01) ;

DCL-F CustSu22 DISK(\*EXT) KEYED USAGE(\*INPUT)

RENAME(CustSu22:CUSTR);

// data structure for host variables from CONTACTS22

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

// Standalone fields for indicator variables

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

DCL-S Dummy Zoned(1);

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//\* \*\*\* M A I N R O U T I N E \*\*\*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

EXSR SummaryInfo;

WRITE NEWPAGE;

READ CustSu22;

DOW NOT %EOF;

EXSR SQLSelect;

IF \*IN01 = \*ON;

Write NEWPAGE;

\*IN01 = \*OFF;

ENDIF;

Write RPTLINE;

READ CustSu22;

ENDDO;

Write SUMMARY;

\*INLR = \*ON;

RETURN;

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// S Q L S E L E C T S U B R O U T I N E

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

BEGSR SQLSelect ;

// A row from the Contacts table that has the same customer number as the

// record read from the CustSu22 file is retrieved to find out the last date

// called, phone number, comments and the salesperson number.

// The call back interval is added to the last date called to determine the

// next date to call. Since null values may be stored in the last date called

// indicator variables are used.

ErrorMsg = ' ';

ENDSR ;

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// S U M M A R Y I N F O S U B R O U T I N E

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

BEGSR SummaryInfo;

// D E T E R M I N E T O T A L S F O R ContactS22 & CustSu22

// D E T E R M I N E N U M B E R O F U N K N O W N LAST DATE CALLED

// D E T E R M I N E O L D E S T & M O S T R E C E N T LAST DATE CALLED

// D E T E R M I N E T H E U S E R S E R V E R & T I M E S T A M P

// SYSIBM/SYSDUMMY1 from System registers,Don't need check SQLCODE

ENDSR;

PHNREPORT (code available in BCI433LIB/LAB9SU22)

A R NEWPAGE

A 2 6'User:'

A 1 1'Timestamp:'

A 3 3'Server:'

A 1105'Page:'

A 1112PAGNBR

A TIMESTAMP Z 1 12

A USER 10 2 12

A SERVER 10 3 12

A 2 48'P H O N E L O G R E P O R T'

A 5 12'Sales'

A 6 13'ID'

A 5 2'Customer'

A 5 24'Customer Name'

A 6 3'Number'

A 6 70'Called'

A 5 54'Telephone'

A 5 69'Last Date'

A 5 83'Oldest Call'

A 6 86'Date'

A 5 97'Next Call'

A 6 99'Date'

A 7131'\*'

A R RPTLINE SPACEB(1)

A CSTNUM R 2REFFLD(CUSTSU22/CSTNUM BCI433LIB/CU-

A STSU22)

A SLSNUMBER R 11REFFLD(CONTACTS22/SLSNUMBER +

A BCI433LIB/CONTACTS22)

A CSTNAM R 18REFFLD(CUSTSU22/CSTNAM BCI433LIB/CU-

A STSU22)

A PHONE R 52REFFLD(CONTACTS22/PHONE BCI433LIB/C-

A ONTACTS22)

A EDTWRD('0( )& - ')

A LCALLD R 69REFFLD(CONTACTS22/LCALLD +

A BCI433LIB/CONTACTS22)

A HIGHLIGHT 8 85

A NEXTCDATE L 98

A ERRORMSG 15 112

A R SUMMARY SPACEB(3)

A 12'S U M M A R Y I N F O R M A T I -

A O N'

A 2'CONTACTS22 Total Records:'

A SPACEB(2)

A CONTACTT 5 0 27EDTCDE(1)

A 44'Oldest Last Callback Date:'

A OLDESTDATE L 71

A 2'CUSTSU22 Total Records:'

A SPACEB(1)

A CUSTOMERT 5 0 27EDTCDE(1)

A 39'Most Recent Last Callback Date:'

A MOSTRECENT L 71

A 1'Total Records with an Unknown Last-

A Callback Date:'

A SPACEB(2)

A UNKNOWNT 5 0 52EDTCDE(1)

OrdersSQL (code available in BCI433LIB/LAB9SU22)

Provide the code for an RPGLE application that produces a report based on three files using embedded SQL.

INPUT: SQL table and Column names are shown. All tables are located in the BCI433LIB library.

|  |  |
| --- | --- |
| ORDER1 | ORDERLINE  Table  Description automatically generated |

CUSTOMER2

Table

Description automatically generated

==> CALL ORDERSSQL 100.00

OUTPUT (CUSTREPORT)

Graphical user interface, text, application, table

Description automatically generated

==> CALL ORDERSSQL 10.00

Graphical user interface, text

Description automatically generated

Processing

Handled by embedded SQL – join of the tables, determining the extended price based on quoted price and number ordered and determining the customer full name. Determining the number of order lines ignored and not included in the printed detail lines with the number showing at the bottom of the report.

Handled by RPGLE – printing report lines, page overflow handling, control break on order number (when it changes print a line with the order date, the order number and the customer full name.) Under the order number line print all the purchases for that order where the extended price is greater than the number passed to the program. Keep totals for each order to print when the order changes and keep a final total to print at the end of the report.

CustReport (code available in BCI433LIB/LAB9SU22)

A R REPORTITLE

A 2 28'C u s t o m e r O r d e r s Whe-

A n an Extended Price Is >'

A SPENDLIMIT 6 2 2 87EDTCDE(1)

A R COLHEADING

A 3'Order Date'

A SPACEB(2)

A 15'Order'

A 25' Full Name'

A 59'Part'

A 67'Quantity'

A 81'Price'

A 92'Extended'

A 15'Number'

A SPACEB(1)

A 58'Number'

A 94'Price'

A 118'\*'

A SPACEB(1)

A R NEXTCUST SPACEB(1)

A ORDERDATE R 3

A REFFLD(ORDER1/ORDDAT PANGBORN/ORDER-

A 1)

A ORDERNUM R 15REFFLD(ORDER1/ORDNUM PANGBORN/ORDER-

A 1)

A CUSTNAME 31 23

A R ORDDETAIL

A PARTNUM R 59REFFLD(ORDERLINE/OLPNUM PANGBORN/OR-

A DERLINE)

A SPACEB(1)

A QUANTITY R 70REFFLD(ORDERLINE/OLNORD PANGBORN/OR-

A DERLINE)

A EDTCDE(1)

A QTPRICE R 80REFFLD(ORDERLINE/OLQPRC PANGBORN/OR-

A DERLINE)

A EDTCDE(1)

A EXTPRICE 7 2 92EDTCDE(1)

A R ORDTOTAL SPACEB(1)

A 78'Order Total:'

A TOTALORD 7 2 91EDTCDE(1 $)

A SPACEA(1)

A R ALLORDERS SPACEB(1)

A 68'Total For All Orders:'

A TOTALALL 8 2 90

A EDTCDE(1 $)

A R NOTINCLUDE SPACEB(1)

A 3'Number of order lines not included-

A in report:'

A UNDERLIMIT 4 0 49EDTCDE(1)

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//\* THIS PROGRAM USES A CURSOR TO LOAD A TEMPORARY RESULT TABLE FROM 3

//\* SEPARATE TABLES, CUSTOMER2, ORDER1 AND ORDERLINE. A NUMBER IS PASSED

//\* TO THE PROGRAM TO DETERMINE WHICH RECORDS ARE INLCUDED

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

DCL-F CUSTREPORT PRINTER OFLIND(\*IN01) ;

DCL-S OrderNumH PACKED(5:0);

// Declare an indicator to handle end of result set (file)

// All Host Variables available under a single name

// LimitIn is passed to the program

// M A I N R O U T I N E

SpendLimit = LimitIn;

EXSR PrepareFiles;

Write ReporTitle;

Write ColHeading;

EXSR GetRow;

Write NextCust;

OrderNumH = OrderNum;

DOW NOT EndOfFile;

IF \*IN01 = \*ON;

Write ReporTitle;

Write ColHeading;

\*IN01 = \*OFF;

ENDIF;

// Handle Control Breaks on Order Number

TotalOrd = TotalOrd + ExtPrice;

EXSR GetRow;

ENDDO;

Write OrdTotal;

TotalAll = TotalAll + TotalOrd;

Write AllOrders;

EXSR WRAPUP;

Write NotInclude;

\*INLR = \*ON;

RETURN;

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// O P E N F I L E S S U B R O U T I N E

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

BEGSR PrepareFiles;

// S E T U P T H E T E M P O R A R Y R E S U L T S T R U C T U R E

// A T E M P O R A R Y R E S U L T T A B L E I S C R E A T E D

EXEC SQL

ENDSR;

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// G E T R O W S U B R O U T I N E

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

BEGSR GETROW;

ENDSR;

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// W R A P U P S U B R O U T I N E

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

BEGSR WRAPUP;

ENDSR;