Advanced DB2 Application Programming – Part 2
(5 Day)

Benefits

The student learns advanced programming techniques for accessing DB2 data, and acquires skills in assessing the performance of SQL statements.

Audience

Application programmers who need to be able to employ advanced tools to access DB2 data.

Prerequisites

The student should be skilled in coding basic SQL statements, and in embedding these statements in application programs

Major Topics Include:
- External stored procedures
- User-defined data types
- User-defined functions (scalar and table)
- Dynamic SQL
- Call Attachment and Recoverable Resource Services Attachment Facilities
- The ODBC Interface
- REXX and DB2
- Native SQL stored procedures and user defined functions
- XML and DB2
- Locking and performance considerations
- Extended indicator variables

There are 11 machine exercises.

Course Outline

DB2 Stored Procedures - Introduction
Stored Procedure Address Spaces
Defining Stored Procedures
    CREATE / ALTER / DROP PROCEDURE
    Language Environment Run-time Parameters
Writing Stored Procedures
    Parameters
    Null Values
Invoking Stored Procedures
    Computer Exercise: Writing, Preparing, and
    Executing a Stored Procedure

DB2 Stored Procedures - Result Sets
    From the Stored Procedure's Perspective
    From the Caller's Perspective
Testing Stored Procedures
Stored Procedure Catalog Tables
    Computer Exercise: Stored procedures with
    result sets

User-defined data types
User-defined functions
    Sourced functions
    External scalar functions
    External table functions
    SQL scalar functions
    Modifying and deleting function definitions
    Functions and security
        Computer Exercise: User-defined
        Functions

Dynamic SQL - 1
    Concepts
    EXECUTE IMMEDIATE
    PREPARE
    Parameter Markers
    EXECUTE
        Computer Exercise: Dynamic SQL -
        Parameter Markers

Dynamic SQL - 2
    Data Retrieval
    Fixed List SELECT statements
    DECLARE CURSOR
    OPEN, FETCH, CLOSE
    Dynamic SQL Bind Options
    Special Registers and Dynamic SQL
        Computer Exercise: Dynamic SQL - Fixed
        List SELECT

Dynamic SQL - 3
    Variable List SELECT statements
    SQLDA
    Dynamic memory management
    DESCRIBE INTO
    SQLTYPE
        Computer Exercise: Dynamic SQL -
        Variable List SELECT

Alternatives to the TSO Attachment Facility
    Call Attachment Facility
        CONNECT, OPEN, CLOSE, and
        DISCONNECT
        Program preparation
        SQL statements
        Tracing
    Recoverable Resource Services Attachment
    Facility
        IDENTIFY, SIGNON, CREATE THREAD,
        TERMINATE THREAD,
           TERMINATE IDENTIFY, TRANSLATE
        Program preparation
        SQL statements
    Computer Exercise: Alternative Attachment
    Facilities

The Open Database Connectivity (ODBC) interface
Connections and handles
Coding ODBC Programs
    Preparing SQL statements
    Binding parameter markers
    Executing SQL statements
    Working with result sets
Calling stored procedures
Establishing the Environment for ODBC
    Computer Exercise: Exploring ODBC

REXX and DB2
    Connecting to DB2
    The REXX SQLCA
    Embedding SQL statements
    Using cursors for data access
    Indicator variables
    Using SQLDAs
REXX Stored Procedures
    Calling Stored Procedures
    Writing Stored Procedures
    Handling Null Values
    Working with Result Sets
        Computer Exercise: DB2 Data Access from a REXX program

Native SQL procedures
CREATE / ALTER PROCEDURE
Versioning
Commands for Native SQL procedures
    Computer Exercise: Native SQL Stored Procedures

XML overview
XML and DB2
XML and SQL
XML in application programs
XMLMODIFY function and partial updating
    Computer Exercise: Using XML in DB2

Version 10 features
    Native SQL Scalar Functions
    SQL Table Functions
    Extended Indicator Variables
    DSNULI: A Universal Interface
    LOAD / UNLOAD Utilities with Spanned Records
    Computer Exercise: Exploring Version 10 Features

Locking in DB2
Performance Considerations
Cursors in Online Systems
Batch Performance Guidelines
    "Smart" COMMIT
    Restart strategies