

DB2 SQL PL Stored Procedure Language Course Summary

Description

This course introduces the student to how to code, debug, and call DB2 SQL PL Stored Procedures.

Topics

- Introduction
- Basic SQL Procedure Structure
- SQL PL Language Elements
- Using Flow of Control Statements
- Understanding and Using Cursors and Result Sets
- Condition Handling
- Working With Dynamic SQL
- Nested SQL Procedures
- User-Defined Functions (UDFs) and Triggers
- Leveraging DB2 Application Development Features
- Deploying SQL Procedures, Functions, and Triggers
- Performance Tuning
- Best Practices

Audience

This course is targeted at programmers and power users, analysts, team leaders and project managers who need to understand how to code, debug, and call DB2 SQL PL Stored Procedures.

Prerequisites

Students should have an understanding of DB2 SQL.

Duration

Three days

DB2 SQL PL Stored Procedure Language

Course Outline

- I. Introduction**
 - A. History of Stored Procedures
 - B. Introduction to Stored Procedures, Triggers, and Functions
- II. Basic SQL Procedure Structure**
 - A. The CREATE PROCEDURE Statement
 - B. The SQL Procedure Body Structure
 - C. A First Example
 - D. DB2 UDB for iSeries Considerations
 - E. DB2 UDB for zSeries Considerations
- III. SQL PL Language Elements**
 - A. DB2 Data Types
 - B. Large Objects
 - C. Choosing Proper Data Types
 - D. Working with User-Defined Distinct Types
 - E. Data Manipulation
 - F. Working with Generated Columns
 - G. Working with Identity Columns and Sequence Objects
 - H. Platform Portability Considerations
- IV. Using Flow of Control Statements**
 - A. Compound Statements (ATOMIC, NOT ATOMIC, and LABELS)
 - B. Conditional Statements (IF and CASE)
 - C. Looping Statements (FOR, WHILE, REPEAT, and LOOP)
 - D. Transfer of Control Statements (GOTO, LEAVE, ITERATE, RETURN, COMMIT, and ROLLBACK)
- V. Understanding and Using Cursors and Result Sets**
 - A. Using Cursors in SQL Procedures
 - B. Positioned Delete
 - C. Positioned Update
 - D. Selecting Data from UPDATE, INSERT, and DELETE Statements
 - E. Cursor Behavior with COMMIT and ROLLBACK
 - F. Using Cursors to Return Result Sets
 - G. Returning Multiple Result Sets
 - H. Cursors and Locking
- VI. Condition Handling**
 - A. Basic Error Checking: SQLCODE and SQLSTATE
 - B. Condition Handlers
 - C. Custom Errors and Error Messages
 - D. Processing Results from Called Procedures
 - E. DB2 UDB for iSeries Considerations
 - F. DB2 UDB for zSeries Considerations
- VII. Working With Dynamic SQL**
 - A. PREPARE and EXECUTE: The Two Phases of Any SQL Statement
 - B. Dynamic SQL Versus Static SQL
 - C. Restrictions and Considerations
 - D. Using Dynamic SQL with EXECUTE IMMEDIATE
 - E. Using the Escape Character (Apostrophe)
 - F. Reusing Dynamic SQL Statements with PREPARE and EXECUTE
 - G. Using Dynamic SQL in Cursors
 - H. Dynamic CALL Statements
 - I. Authorization Considerations
- VIII. Nested SQL Procedures**
 - A. Basic Nested SQL Procedures
 - B. Passing Parameters Between Nested SQL Procedures
 - C. Returning Values from Nested SQL Procedures
 - D. Returning Result Sets from Nested SQL Procedures
 - E. Receiving Results from a Procedure in a Trigger
 - F. Levels of Nesting
- IX. User-Defined Functions (UDFs) and Triggers**
 - A. The CREATE FUNCTION Statement
 - B. User-Defined Functions By Example
 - C. The CREATE TRIGGER Statement
 - D. Triggers By Example
 - E.
- I. DB2 for iSeries Considerations**

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically

DB2 SQL PL Stored Procedure Language

Course Outline (cont'd)

- F. Considerations for Invoking SQL Procedures from UDFs and Triggers
- G. DB2 for iSeries Considerations
- H. DB2 for zSeries Considerations

X. Leveraging DB2 Application Development Features

- A. Leveraging Advanced SQL
- B. Declared Global Temporary Tables
- C. Working with Save Points
- D. Sequence Objects

XI. Deploying SQL Procedures, Functions, and Triggers

- A. Deploying on Linux, UNIX, and Windows (LUW)
- B. Deployment Considerations for DB2 UDB for zSeries
- C. Common Deployment Considerations for LUW and zSeries
- D. Deploying to DB2 UDB for iSeries

XII. Performance Tuning

- A. Performance Considerations for Linux, UNIX, and Windows (LUW)
- B. DB2 UDB for iSeries Considerations
- C. DB2 UDB for zSeries Considerations

XIII. Best Practices

- A. Table and Index Best Practices
- B. Best Practices for Easier Code Maintenance
- C. Best Practices for Performance
- D. Working with Result Sets
- E. Pre-compile Options for iSeries