DB2 Application Programming

CDT620

This course makes extensive use of hands-on examples and exercises to familiarize the student with the fundamentals of relational database concepts in general and DB2 in particular. First, the student is introduced to relational database design concepts, including the creation of entity relationship diagrams (ERDs) and how to reduce ERDs to relations in third normal form (3NF). Then, the student will use DB2's data definition language (DDL) to create the actual tables. Next, the student will use DB2's data manipulation language (DML) to query the tables. Finally, the student will learn how to code embedded SQL within COBOL programs. *In this class we do it all, from analysis to data modeling to creating the tables to querying the tables to accessing the tables through programs!*

Audience

- Programmers who need a better understanding of relational database concepts.
- Programmers who need to be able to perform ad hoc queries of relational databases.
- Programmers who need to be able to access DB2 tables through their programs.

Prerequisites

- Experience with TSO/ISPF is required.
- Experience with COBOL is required.

Course Length

- 5 days (Day 1 is equilvalent to CDT600, Day 2 is equilvalent to CDT610)

Learning Objectives

- Given data, create the ERDs
- Reduce ERDs to relations in 3NF
- Define DB2 tables
- Run ad hoc queries against one table
- Run ad hoc queries against multiple tables
- Access DB2 tables thru embedded SQL
- Create reports through QMF

Teaching Methods

- Lecture
- Hands-on workshops
- Supplemental hands-on exercises

Course Outline

QC2

R/DBMS Design Concepts

- Logical data modeling using ERDs
- Data normalization: from ERD to 3NF
- Referential integrity
- Introduction to SQL's DDL and DML

Creating DB2 tables

- Using SPUFI
- CREATE
- Field types (CHAR, DECIMAL, DATE, etc.)
- INSERT
- Indexes
- Primary keys
- Foreign keys
- CREATE INDEX
- Entity integrity
- Domain integrity
- Referential integrity

Queries of a single table

- SELECT
- FROM
- WHERE
- GROUP BY
- HAVING
- ORDER BY

Queries of multiple tables

- UNION
- Subqueries vs. joins
- Cartesian product

Maintaining tables

- UPDATE
- DELETE
- DROP

Programming

- DCLGEN
- EXEC SQL
- SQLCA
- Precompile, Compile, and Bind

Sequential Access

- DECLARE CURSOR
- OPEN
- FETCH
- CLOSE
- SQLCODE

Random Access

- SELECT...INTO
- SQLCODE

Views

- Why views
- How to create views

Query Management Facility

- Queries in QMF
- QMF reporting