SQL 2014 Developer, Part 04 of 13: Transactions

Meet the expert: Don Kiely is a featured instructor on many of our SQL Server and Visual Studio courses. He is a nationally recognized author, instructor, and consultant specializing in Microsoft technologies. Don has many years of teaching experience, is the author or co-author of several programming books, and has spoken at many industry conferences and user groups. In addition, Don is a consultant for a variety of companies that develop distributed applications for public and private organizations.

Prerequisites: This course assumes that you have a basic familiarity with the concept of relational databases and a basic understanding of what SQL Server is and the high-level tools in it, as well as how to create and manage objects using Management Studio. You should also have a basic understanding of how SQL Server implements security, including its authentication and authorization schemes, and how to assign permissions on securable objects to principals. You should know the fundamentals of Transact-SQL to write queries to retrieve and update data, and how to execute scripts using the query editor in Management Studio. It will be very helpful, but not absolutely necessary, to have experience with .NET development using Visual Studio 2010 or 2012 for the portions of the course that deal with SQLCLR development. The .NET code included in this course is written in C#, but if you prefer Visual Basic it should be fairly easy to convert the code.

Runtime: 01:37:10

Course description: In this course you’ll learn about transaction concepts and the ACID test that SQL Server uses to protect the integrity of data. You’ll also learn about locking and blocking, and how to avoid these inevitable effects of heavily used relational databases. You’ll see how to use transactions in applications, as well as how to handle compile and run time errors caused by your code. Then you’ll learn about how to take control of transactions and defined them explicitly in your code. Then we’ll wrap with a look at a relatively new feature of T-SQL, structured exception handling that has long been a feature of general purpose programming languages.

Course outline:

Transaction Concepts
- Introduction
- Passing the ACID Test
- Transactions Types
- Transaction Details
- Isolation Levels
- Locking
- Viewing Lock Information
- Avoiding Blocked Transactions
- Demo: Blocking
- Demo: Read Uncommitted
- Working Around DeadLocks
- Deadlocks
- Preventing Deadlocks
- Summary

Applications and Transactions
- Introduction
- Designing Transaction Support
- Efficiency Guidelines
- Compile and Runtime Errors
- Demo: Compiler Errors
- Demo: Runtime Errors
- Demo:Invalid Object Name Error
- Summary

Creating Explicit Transactions
- Introduction
- Explicit Transaction Syntax
- Nested Transactions
- Demo: @@TRANCOUNT
- Error Handling
- Using @@ERROR
- Demo: Error Handling
- Manage with Stored Procedures
- Demo: Proactive Checks
- Demo:Errors In Transaction
- Demo: Testing Stored Procedure
- Using RAISERROR
- Demo: Using RAISERROR
- Demo: User-Defined RAISERROR
- Summary

Structured Error Handling
- Introduction
- TRY/CATCH Overview
- Demo: TRY/CATCH
- Error Chaining
- Errors Not Handled
- Error Handling Functions
- Demo: Error Handling Functions
- Using RAISERROR With TRY/CATCH
- Demo: RAISERROR With TRY/CATCH
- The THROW Statement
- Demo: Using THROW Statement
- Demo: Within Stored Procedure
- Uncommittable Transactions
- XACT_STATE Function
- Demo: XACT_STATE
- Summary