SQL 2012 Developer, Part 05 of 13: Common Language Runtime

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: 01:26:19

Course description: In this course you'll learn how to write various kinds of code modules using C# code. You'll learn the details of creating SQLCLR code manually as well as with Visual Studio. You'll learn about the Sql data types and how using them can avoid data conversion issues. Most of the code you write will use basic ADO.NET for data access, but there is a special context connection you need to make to the database where the code is executing. I'll take a close look at creating triggers and user-defined functions, then will explore the security issues of running SQLCLR code in the SQL Server process, and how to make it work. The course wraps up with some recommendations about how to decide whether to use T-SQL or SQLCLR code to create different kinds of code modules.

Course outline:

Introduction to SQL CLR
- Introduction
- Introduction to SQLCLR
- Role of SQL Server
- Role of CLR
- Ways to Use SQLCLR
- Demo: Enabling SQL CLR
- Summary

Writing SQL CLR Code
- Introduction
- Demo: Writing SQLCLR Code
- Demo: Compiling SQLCLR Code
- Demo: Loading SQLCLR Assembly
- Demo: Creating a Function
- SQLCLR Using Visual Studio
- Demo: Database Project Template
- Demo: Creating SQLCLR File
- Demo: Building and Publishing
- Demo: Solution Explorer Files
- Demo: Server Explorer
- Summary

Using SQL CLR Code
- Introduction
- SQL Data Types
- Data Type Mapping
- Accessing Local Data
- Connection String
- Returning Results
- Demo: Stored Procedure
- Demo: Now Using SQLCLR Code
- Demo: Executing SQLCLR Code
- Demo: Realistic SQLCLR Example
- Summary

SQL CLR Code Modules
- Introduction
- Triggers
- Demo: SQLCLR Trigger
- Demo: Testing the Trigger
- User-Defined Functions
- Demo: User-Defined Function
- Summary

SQL CLR Security
- Introduction
- Demo: External Resources
- Code Access Security
- How Access Security Works
- SQLCLR Permission Sets
- Demo: Changing Permission Set
- Demo: Executing External Code
- T-SQL vs .NET Code
- Deciding: SQLCLR or T-SQL?
- Summary