1.

Record Nr.	UNINA9910483475803321
Autore	Leonard Andy
Titolo	Building custom tasks for SQL server integration services : the power of .NET for ETL for SQL Server 2019 and beyond / / Andy Leonard
Pubbl/distr/stampa	[Place of publication not identified] : , : Apress, , [2021] ©2021
ISBN	1-4842-6482-7
Edizione	[Second edition.]
Descrizione fisica	1 online resource (XXII, 692 p. 721 illus.)
Disciplina	005.7565
Soggetti	Database management
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Nota di contenuto	1. The Story of This Book -- 2. Preparing the Environment -- 3. Creating the Assembly Project -- 4. Check-In the Project Code -- 5. Signing the Assembly -- 6. Preparing to Build -- 7. Coding the Task -- 8. Coding a Simple Task Editor -- 9. Signing and Binding -- 10. Expanding Editor Functionality -- 11. Minimal Coding for the Complex Editor -- 12. Editor Integration -- 13. Implement Views and Properties -- 14. Implement New Connection -- 15. Implement Use32bit, Synchronized, and LoggingLevel SettingsView Properties -- 16. Refactoring SourceConnection -- 17. Refactoring the SSIS Package Hierarchy -- 18. Instrumentation and Validation -- 19. Crushing Bugs -- 20. Adding Synchronous Execution Properties -- 21. Testing the Task -- 22. Building the Setup Project -- 23. Using the Execute Catalog Package Task in an SSIS Framework -- 24. Deploying to Azure-SSIS -- 25. Test the Task in Azure Data Factory -- 26. Notes from my Experience.
Sommario/riassunto	Build custom SQL Server Integration Services (SSIS) tasks using Visual Studio Community Edition and C#. Bring all the power of Microsoft .NET to bear on your data integration and ETL processes, and for no added cost over what you’ve already spent on licensing SQL Server. New in this edition is a demonstration deploying a custom SSIS task to the Azure Data Factory (ADF) Azure-SSIS Integration Runtime (IR). All examples in this new edition are implemented in C#. Custom task developers are shown how to implement custom tasks using the widely

accepted and default language for .NET development. Why are custom components necessary? Because even though the SSIS catalog of built-in tasks and components is a marvel of engineering, gaps remain in the available functionality. One such gap is a constraint of the built-in SSIS Execute Package Task, which does not allow SSIS developers to select SSIS packages from other projects in the SSIS Catalog. Examples in this book show how to create a custom Execute Catalog Package task that allows SSIS developers to execute tasks from other projects in the SSIS Catalog. Building on the examples and patterns in this book, SSIS developers may create any task to which they aspire, custom tailored to their specific data integration and ETL needs. You will: Configure and execute Visual Studio in the way that best supports SSIS task development Create a class library as the basis for an SSIS task, and reference the needed SSIS assemblies Properly sign assemblies that you create in order to invoke them from your task Implement source code control via Azure DevOps, or your own favorite tool set Troubleshoot and execute custom tasks as part of your own projects Create deployment projects (MSIs) for distributing code-complete tasks Deploy custom tasks to Azure Data Factory Azure-SSIS IRs in the cloud Create advanced editors for custom task parameters.