1.

Record Nr.	UNINA990001299300403321
Autore	Harish-Chandra
Titolo	Automorphic forms on semisimple lie groups / by HARISH-CHANDRA
Pubbl/distr/stampa	Berlin : Springer-Verlag, 1968
Collana	Lecture Notes in Mathematics
Locazione	MA1
Collocazione	C-20-(62
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	VOLUME 62

2.

Record Nr.	UNINA9910300461503321
Autore	Mukherjee Sudipta
Titolo	Thinking in LINQ : Harnessing the Power of Functional Programming in . NET Applications / / by Sudipta Mukherjee
Pubbl/distr/stampa	Berkeley, CA : , : Apress : , : Imprint : Apress, , 2014
ISBN	9781430268444 1430268441
Edizione	[1st ed. 2014.]
Descrizione fisica	1 online resource (259 p.)
Collana	Expert's Voice In Networking
Disciplina	006.7882
Soggetti	Microsoft software Microsoft .NET Framework Software engineering Microsoft and .NET Software Engineering/Programming and Operating Systems
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Includes index.

Nota di contenuto

Contents at a Glance; Contents; About the Author; About the Technical Reviewer; Acknowledgments; Introduction; Chapter 1: Thinking Functionally; 1-1. Understanding Functional Programming; 1-2. Using Func in C# to Represent Functions; 1-3. Using Various Types of Functions; Generator Functions; Statistical Functions; Projector Functions; Filters; 1-4. Understanding the Benefits of Functional Programming; Composability; Lazy Evaluation; Immutability; Parallelizable; Declarative; 1-5. Getting LINQPad; Chapter 2: Series Generation; 2-1. Math and Statistics: Finding the Dot Product of Two Vectors

ProblemSolution; How It Works; 2-2. Math and Statistics: Generating Pythagorean Triples; Problem; Solution; How It Works; 2-3. Math and Statistics: Finding a Weighted Sum; Problem; Solution; How It Works; 2-4. Math and Statistics: Finding the Percentile for Each Element in an Array of Numbers; Problem; Solution; How It Works; 2-5. Math and Statistics: Finding the Dominator in an Array; Problem; Solution; How It Works; 2-6. Math and Statistics: Finding the Minimum Number of Currency Bills Required for a Given Amount; Problem; Solution; How It Works

2-7. Math and Statistics: Finding Moving AveragesProblem; Solution; How It Works; 2-8. Math and Statistics: Finding a Cumulative Sum; Problem; Solution; How It Works; 2-9. Recursive Series and Patterns: Generating Recursive Structures by Using L-System Grammar; Problem; Solution; How It Works; 2-10. Recursive Series and Patterns Step-by-Step Growth of Algae; Problem; Solution; How It Works; 2-11. Recursive Series and Patterns: Generating Logo Commands to Draw a Koch Curve; Problem; Solution; How It Works

2-17. Collections: Finding the Larger or Smaller of Several Sequences at Each IndexProblem; Solution; How It Works; 2-18. Number Theory: Generating Armstrong Numbers and Similar Number Sequences; Problem; Solution; How It Works; 2-19. Number Theory: Generating Pascal's Triangle Nonrecursively; Problem; Solution; How It Works; 2-20. Game Design: Finding All Winning Paths in an Arbitrary Tic-Tac-Toe Board; Problem; Solution; How It Works; 2-21. Series in Game Design: Solving Go Figure; Problem; Solution; How It Works

2-22. Miscellaneous Series: Finding Matching Pairs from Two Unsorted Collections

Sommario/riassunto

LINQ represents a paradigm shift for developers used to an imperative/object oriented programming style, because LINQ draws on functional programming principles. Thinking in LINQ addresses the differences between these two by providing a set of succinct recipes arranged in several groups, including: Basic and extended LINQ operators Text processing Loop refactoring Monitoring code health Reactive Extensions (Rx.NET) Building domain-specific languages Using the familiar "recipes" approach, Thinking in LINQ shows you how to approach building LINQ-based solutions, how such solutions are different from what you already know, and why they’re better. The recipes cover a wide range of real-world problems, from using LINQ to replace existing loops, to writing your own Swype-like keyboard entry routines, to finding duplicate files on your hard drive. The goal of these recipes is to get you "thinking in LINQ," so you can use the techniques in your own code to write more efficient and concise data-intensive applications.