

1.	Record Nr.	UNISALENTO991002762029707536
	Autore	Furtwängler, Andreas E.
	Titolo	Monnaies grecques en Gaule : le trésor et le monnayage de Massilia 525/520-460 av. J.C. / Andreas E. Furtwängler
	Pubbl/distr/stampa	Fribourg : Office du livre, 1978
	Descrizione fisica	1 v. ; 25 cm
	Collana	Typos : monographien zur antiken numismatik ; 3
	Disciplina	737.4
	Soggetti	Monete greche - Francia
	Lingua di pubblicazione	Francese
	Formato	Materiale a stampa
	Livello bibliografico	Monografia
2.	Record Nr.	UNINA9910798228603321
	Autore	Wali Akhil
	Titolo	Mastering Clojure : understand the philosophy of the Clojure language and dive into its inner workings to unlock its advanced features, methodologies, and constructs // Akhil Wali
	Pubbl/distr/stampa	Birmingham : , : Packt Publishing, , 2016
	ISBN	1-78588-205-8
	Descrizione fisica	1 online resource (267 p.)
	Collana	Community experience distilled
	Soggetti	Clojure (Computer program language) Social networks - Mathematical models Geographic information systems
	Lingua di pubblicazione	Inglese
	Formato	Materiale a stampa
	Livello bibliografico	Monografia
	Note generali	Includes index.
	Nota di bibliografia	Includes bibliographical references and index.
	Nota di contenuto	Cover ; Copyright; Credits; About the Author; About the Reviewer; www. PacktPub.com; Table of Contents; Preface; Chapter 1: Working with

Sequences and Patterns; Defining recursive functions; Thinking in sequences; Using the seq library; Creating sequences; Transforming sequences; Filtering sequences; Lazy sequences; Using zippers; Working with pattern matching; Summary; Chapter 2: Orchestrating Concurrency and Parallelism; Managing concurrent tasks; Using delays; Using futures and promises; Managing state; Using vars; Using refs; Using atoms; Using agents; Executing tasks in parallel Controlling parallelism with thread poolsSummary; Chapter 3: Parallelization Using Reducers; Using reduce to transform collections; What's wrong with sequences?; Introducing reducers; Using fold to parallelize collections; Processing data with reducers; Summary; Chapter 4: Metaprogramming with Macros; Understanding the reader; Reading and evaluating code; Quoting and unquoting code; Transforming code; Expanding macros; Creating macros; Encapsulating patterns in macros; Using reader conditionals; Avoiding macros; Summary; Chapter 5: Composing Transducers; Understanding transducers Producing results from transducersComparing transducers and reducers; Transducers in action; Managing volatile references; Creating transducers; Summary; Chapter 6: Exploring Category Theory; Demystifying category theory; Using monoids; Using functors; Using applicative functors; Using monads; Summary; Chapter 7: Programming with Logic; Diving into logic programming; Solving logical relations; Combining logical relations; Thinking in logical relations; Solving the n-queens problem; Solving a Sudoku puzzle; Summary; Chapter 8: Leveraging Asynchronous Tasks; Using channels; Customizing channels Connecting channelsRevisiting the dining philosophers problem; Using actors; Creating actors; Passing messages between actors; Handling errors with actors; Managing state with actors; Comparing processes and actors; Summary; Chapter 9: Reactive Programming; Reactive programming with fibers and dataflow variables; Using Reactive Extensions; Using functional reactive programming; Building reactive user interfaces; Introducing Om; Summary; Chapter 10: Testing Your Code; Writing tests; Defining unit tests; Using top-down testing; Testing with specs; Generative testing; Testing with types; Summary Chapter 11: Troubleshooting and Best PracticesDebugging your code; Using tracing; Using Spyscope; Logging errors in your application; Thinking in Clojure; Summary; Appendix: References; Index
