

1. Record Nr.	UNINA9910463318003321
Autore	Kainulainen Petri
Titolo	Spring data [[electronic resource] /] / Petri Kainulainen
Pubbl/distr/stampa	Birmingham, : Packt Publishing, 2012
ISBN	1-62198-909-7 1-283-83676-9 1-84951-905-6
Edizione	[1st edition]
Descrizione fisica	1 online resource (161 p.)
Disciplina	006.312
Soggetti	Java (Computer program language) Application software - Development Electronic books.
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Description based upon print version of record.
Nota di contenuto	Cover; Copyright; Credits; About the Author; About the Reviewers; www.PacktPub.com; Table of Contents; Preface; Chapter 1:Getting Started; Java Persistence API; Key concepts; Creating database queries; Native SQL queries; Java Persistence Query Language; The Criteria API; Redis; Supported data types; Persistence; Replication; Publish/subscribe messaging pattern; Summary; Chapter 2:Getting Started with Spring Data JPA; Downloading dependencies with Maven; Configuring the Spring application context; Creating the properties file; Creating the application context configuration class Creating the application context configuration skeletonConfiguring the data source bean; Configuring the entity manager factory bean; Configuring the transaction manager bean; Loading the application context configuration; Implementing CRUD functionality for an entity; Domain model; Contact; Address; Creating a custom repository; Creating a custom repository in the old school way; Creating a custom repository with Spring Data JPA; CRUD; Create; Read; Update; Delete; Summary; Chapter 3:Building Queries with Spring Data JPA; Building queries; Query methods; Query generation from method name Named queries@Query annotation; JPA Criteria API; Adding the JPA Criteria API support to a repository; Creating the criteria query;

Creating the service method; Pros and cons; Querydsl; Configuring Querydsl-Maven integration; Generating Querydsl query types; Adding Querydsl support to a repository; Creating the executed query; Executing the created query; Pros and cons; What technique should we use?; Sorting query results; Sorting with method name; Creating the query method; Modifying the service method; Sorting with query strings; JPQL queries; SQL queries; Sorting with the Sort class JpaRepositoryQuery generation from the method name; @Query annotation; JPA Criteria API; Sorting with Querydsl; What technique should we use?; Paginating query results; Changing the service layer; Creating a class for pagination parameters; Changing the service interface; Creating PageRequest objects; Implementing pagination; JpaRepository; Query generation from the method name; Named queries; @Query annotation; JPA Criteria API; Querydsl; Summary; Chapter 4: Adding Custom Functionality to JPA Repositories; Adding custom functionality to a single repository; Creating the custom interface
Implementing the created interface
Configuring the repository class;
Implementing the custom methods; Creating the repository interface;
Creating the service implementation; What did we just do?; Adding custom functionality to all repositories; Creating the base repository interface; Implementing the base repository interface; Creating the repository factory bean; Creating the skeleton of the repository factory bean class; Creating the repository factory inner class; Creating the builder method for the repository factory; Configuring Spring Data JPA;
Creating the repository interface
Implementing the service layer

Sommario/riassunto

Implement JPA repositories and harness the performance of Redis in your applications.
