

The Little Redis Book

by Karl Seguin



About This Book

License

The Little Redis Book is licensed under the Attribution-NonCommercial 3.0 Unported license. You should not have paid for this book.

You are free to copy, distribute, modify or display the book. However, I ask that you always attribute the book to me, Karl Seguin, and do not use it for commercial purposes.

You can see the *full text* of **the license at**:

<http://creativecommons.org/licenses/by-nc/3.0/legalcode>

About The Author

Karl Seguin is a developer with experience across various fields and technologies. He's an active contributor to Open-Source Software projects, a technical writer and an occasional speaker. He's written various articles, as well as a few tools, about Redis. Redis powers the ranking and statistics of his free service for casual game developers: mogade.com.

Karl wrote [The Little MongoDB Book](#), the free and popular book about MongoDB.

His blog can be found at <http://openmymind.net> and he tweets via [\[@karlseguin\]\(http://twitter.com/karlseguin\)](http://twitter.com/karlseguin)

With Thanks To

A special thanks to [Perry Neal](#) for lending me his eyes, mind and passion. You provided me with invaluable help. Thank you.

Latest Version

The latest source of this book is available at: <http://github.com/karlseguin/the-little-redis-book>

Introduction

Over the last couple years, the techniques and tools used for persisting and querying data have grown at an incredible pace. While it's safe to say that relational databases aren't going anywhere, we can also say that the ecosystem around data is never going to be the same.

Of all the new tools and solutions, for me, Redis has been the most exciting. Why? First because it's unbelievably easy to learn. Hours is the right unit to use when talking about length of time it takes to get comfortable with Redis. Secondly, it solves a specific set of problems while at the same time being quite generic. What exactly does that mean? Redis doesn't try to be all things to all data. As you get to know Redis, it'll become increasingly evident what does and what does not belong in it. And when it does, as a developer, it's a great experience.

While you can build a complete system using Redis only, I think most people will find that it supplements their more generic data solution - whether that be a traditional relational database, a document-oriented system, or something else. It's the kind of solution you use to implement specific features. In that way, it's similar to an indexing engine. You wouldn't build your entire application on Lucene. But when you need good search, it's a much better experience - for both you and your users. Of course, the similarities between Redis and indexing engines end there.

The goal of this book is to build the foundation you'll need to master Redis. We'll focus on learning Redis' five data structures and look at various data modeling approaches. We'll also touch on some key administrative details and debugging techniques.

Getting Started

We all learn differently: some like to get their hands dirty, some like to watch videos, and some like to read. Nothing will help you understand Redis more than actually experiencing it. Redis is easy to install and comes with a simple shell that'll give us everything we need. Let's take a couple minutes and get it up and running on our machine.

On Windows

Redis itself doesn't officially support Windows, but there are options available. You wouldn't run these in production, but I've never experienced any limitations while doing development.

A port by Microsoft Open Technologies, Inc. can be found at <https://github.com/MicrosoftOpenTech/redis>. As of this writing the solution is not ready for use in production systems.

Another solution, which has been available for some time, can be found at <https://github.com/dmajkic/redis/downloads>. You can download the most up to date version (which should be at the top of the list). Extract the zip file and, based on your architecture, open either the 64bit or 32bit folder.

On *nix and MacOSX

For *nix and Mac users, building it from source is your best option. The instructions, along with the latest version number, are available at <http://redis.io/download>. At the time of this writing the latest version is 3.0.3; to install this version we would execute:

```
wget http://download.redis.io/releases/redis-3.0.3.tar.gz
tar xzf redis-3.0.3.tar.gz
cd redis-3.0.3
make
```

(Alternatively, Redis is available via various package managers. For example, MacOSX users with Homebrew installed can simply type `brew install redis`.)

If you built it from source, the binary outputs have been placed in the `src` directory. Navigate to the `src` directory by executing `cd src`.

Running and Connecting to Redis

If everything worked, the Redis binaries should be available at your fingertips. Redis has a handful of executables. We'll focus on the Redis server and the Redis command line interface (a DOS-like client). Let's start the server. In Windows, double click `redis-server`. On *nix/MacOSX run `./redis-server`.

If you read the start up message you'll see a warning that the `redis.conf` file couldn't be found. Redis will instead use built-in defaults, which is fine for what we'll be doing.

Next start the Redis console by either double clicking `redis-cli` (Windows) or running `./redis-cli` (*nix/MacOSX). This will connect to the locally-running server on the default port (6379).

You can test that everything is working by entering `info` into the command line interface. You'll hopefully see a bunch of key-value pairs which provide a great deal of insight into the server's status.

If you are having problems with the above setup I suggest you seek help in the [official Redis support group](#).

[Click here to download full PDF material](#)