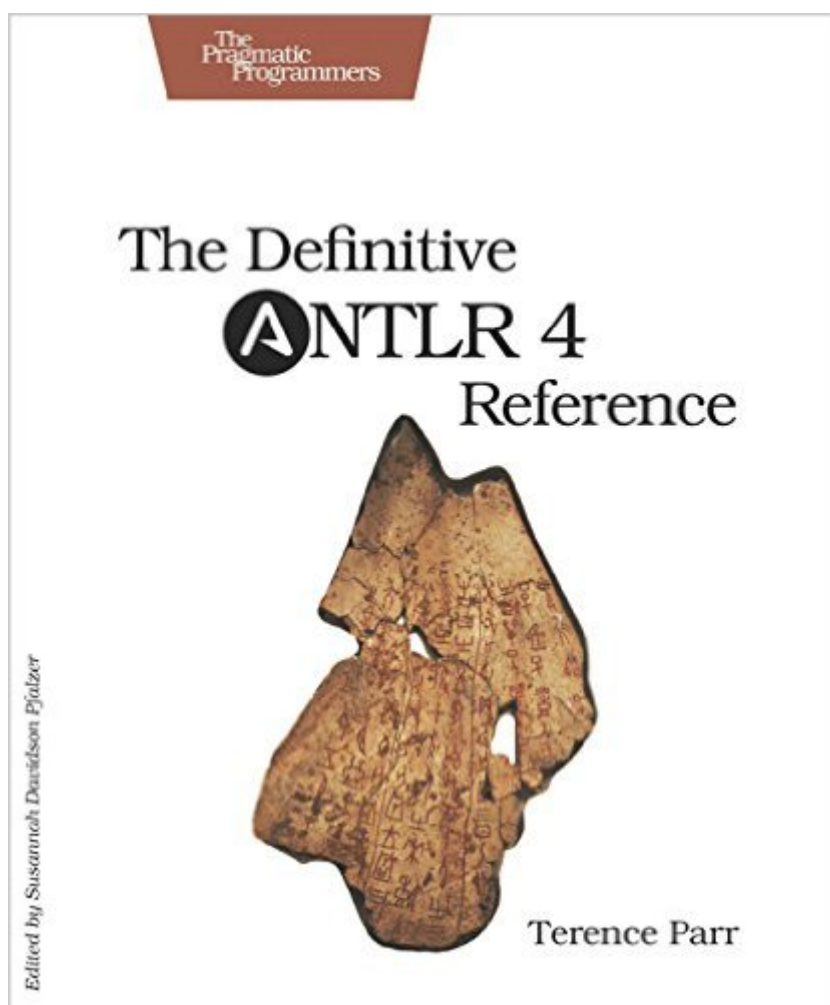


The book was found

The Definitive ANTLR 4 Reference



Synopsis

Programmers run into parsing problems all the time. Whether it's a data format like JSON, a network protocol like SMTP, a server configuration file for Apache, a PostScript/PDF file, or a simple spreadsheet macro language--ANTLR v4 and this book will demystify the process. ANTLR v4 has been rewritten from scratch to make it easier than ever to build parsers and the language applications built on top. This completely rewritten new edition of the bestselling Definitive ANTLR Reference shows you how to take advantage of these new features. Build your own languages with ANTLR v4, using ANTLR's new advanced parsing technology. In this book, you'll learn how ANTLR automatically builds a data structure representing the input (parse tree) and generates code that can walk the tree (visitor). You can use that combination to implement data readers, language interpreters, and translators. You'll start by learning how to identify grammar patterns in language reference manuals and then slowly start building increasingly complex grammars. Next, you'll build applications based upon those grammars by walking the automatically generated parse trees. Then you'll tackle some nasty language problems by parsing files containing more than one language (such as XML, Java, and Javadoc). You'll also see how to take absolute control over parsing by embedding Java actions into the grammar. You'll learn directly from well-known parsing expert Terence Parr, the ANTLR creator and project lead. You'll master ANTLR grammar construction and learn how to build language tools using the built-in parse tree visitor mechanism. The book teaches using real-world examples and shows you how to use ANTLR to build such things as a data file reader, a JSON to XML translator, an R parser, and a Java class->interface extractor. This book is your ticket to becoming a parsing guru! What You Need: ANTLR 4.0 and above. Java development tools. Ant build system optional (needed for building ANTLR from source)

Book Information

Paperback: 328 pages

Publisher: Pragmatic Bookshelf; 2 edition (January 25, 2013)

Language: English

ISBN-10: 1934356999

ISBN-13: 978-1934356999

Product Dimensions: 7.5 x 0.8 x 9.2 inches

Shipping Weight: 1.4 pounds (View shipping rates and policies)

Average Customer Review: 4.9 out of 5 stars Â Â See all reviews Â (23 customer reviews)

Best Sellers Rank: #266,277 in Books (See Top 100 in Books) #22 in Â Books > Computers &

Technology > Programming > Languages & Tools > Compiler Design #45 in Books > Computers & Technology > Programming > Languages & Tools > Compilers #312 in Books > Computers & Technology > Programming > Languages & Tools > Java

Customer Reviews

As an author of a "competing" parser generator (Yacc++), I must first say that ANLTR in its various versions is arguably the most popular parser generator for good reasons. (Yacc and its derivatives are the only competitor for that title.) It is free. It is high quality. It is easy to use. (Recursive descent technology outputs parsers the people can actually understand without advanced theoretical knowledge.) It has advanced concepts that make parsing easy. (Predicates were Terence's invention and opened up a whole world of parsing innovations. The ability to deal with left-recursion in LL that v4 introduces looks similarly inspired.) It helps people write maintainable (and theoretically sound) parsers. What's not to like? If you aren't using ANTLR, one has to ask "Why not?". Now, if you are [considering] using ANTLR, you should get this book. Not only is it the DEFINITIVE guide, but it is an easy to read, easy to understand book. The same characteristics that make ANTLR itself popular. In fact, if you didn't love ANTLR before this book, reading this book will probably convince you to love it. Writing good, maintainable, correct parsers shouldn't be hard, with this book and ANTLR in your toolkit, it isn't.

Yesterday I finished my first ANTLR-based grammar using ANTLR4. I've always had an interest in small custom languages, going back to the mid 90s with lex & yacc. And I've been circling around writing my own language parser for a year now, after having read the ANTLR3 book, this ANTLR4 book, and Parr's Language Implementation Patterns. ANTLR3 is powerful in its own right, but you have to do more work to get an abstract syntax tree from your parsed input. That, and target language constructs appear in an ANTLR3 grammar. In ANTLR4, you no longer need to build an AST grammar from the input grammar because the listener/visitor implementations ANTLR4 emits relieve you of that. And, an ANTL4 grammar is generally target language agnostic, which means target language constructs no longer show up in the grammar (unless you really need them to). Which means at some point, you'll be able to specify the target language as option to the grammar compilation. Currently, Java is the default target language. So if you're looking to get started with ANTLR, start with version 4. As for the material in the book: it's superb. It's extremely well written, completely lacks pith, and the concepts are paid out in exactly the right order at exactly the right pace. And there are plenty of examples. There aren't many books written this well. ANTLR4 is highly

evolved from where it started years ago. You'll enjoy working with it. I certainly am.

If you are going to implement a domain-specific language of any sort, ANTLR 4 is the only real choice. Anything else is just a waste of time. And this book is absolutely required reading! Parr is very entertaining to read and the subject is skillfully presented with a practical approach and lots of useful examples. With ANTLR 4 and this book we were in one week's time able to implement a powerful new language custom-tailored to our needs in C#. ANTLR 4's new listener and visitor patterns (fully described in the book) even allowed us to implement a complete IL bytecode compiler within the same timeframe, so our language runs directly on the .NET CLR.

ANTLR v4 is one of the most powerful language parser tools available today. Terrance Parr, being the primary developer of ANTLR has a wealth of knowledge about this tool and uses a clever writing style and hands on approach to explaining the capabilities of ANTLR v4 and languages in general. It is an excellent reference and I highly recommend it to those who are interested in learning about compilers as well as those who are in the middle of developing their own parsers. LL(*) with Left-Recursive Descent parsing is a very powerful tool which ANTLR v4 delivers!

The title says it all: "The Definitive ANTLR 4 Reference". If you are using or plan to use ANTLR 4, you need to get this book. This is your source of wisdom. Besides explaining all features of ANTLR 4 it also gives you insight about the basics of parsing and building languages. Highly recommended.

I have no formal training in computer science but have been mucking with them since core memory times. This book is a joy to read (I am slow and had to read it a couple times.) The language is immensely powerful but very easy to use. I was able to translate the book java to C# easily and write a solution that I needed by sundown without knowing anything about lexing or parsing till I got to page 13 early one morning. On page 14 I learned that "you can't put too much water into a nuclear reactor." The co-author Sam Harwell has done the C# translation and the Visual Studio bits. Also look at antlrworks2 tool. It is amazing that something so powerful is free. After reading this, I bought the author's other popular book: "Language implementation patterns" and learned a lot more. There is a large public domain library of ANTLR4 grammars for a host of situations so you may not need to start from scratch. If you are trying to write code to parse stuff, stop playing with RegEx's and learn this.

[Download to continue reading...](#)

The Definitive ANTLR 4 Reference HTML & XHTML: The Definitive Guide: The Definitive Guide (Definitive Guides) Oracle SQL*Plus: The Definitive Guide (Definitive Guides) 802.11 Wireless Networks: The Definitive Guide: The Definitive Guide The Definitive Guide to GCC (Definitive Guides (Paperback)) Vertical Gardening: The Definitive Guide To Vertical Gardening For Beginners. (The Definitive Gardening Guides) The World Encyclopedia of Coins & Coin Collecting: The definitive illustrated reference to the world's greatest coins and a professional guide to ... collection, featuring over 3000 colour images C# 6.0 in a Nutshell: The Definitive Reference Crystal Reports XI: The Complete Reference (Osborne Complete Reference Series) MySQL Pocket Reference: SQL Statements, Functions and Utilities and more (Pocket Reference (O'Reilly)) Python Pocket Reference (Pocket Reference (O'Reilly)) UNIX: The Complete Reference, Second Edition (Complete Reference Series) Regular Expression Pocket Reference: Regular Expressions for Perl, Ruby, PHP, Python, C, Java and .NET (Pocket Reference (O'Reilly)) GDB Pocket Reference (Pocket Reference (O'Reilly)) Java Beginners Reference: Quick Reference The Ferns of Florida: A Reference and Field Guide (Reference and Field Guides) LINQ Pocket Reference (Pocket Reference (O'Reilly)) Linux iptables Pocket Reference (Pocket Reference (O'Reilly)) XML Pocket Reference (Pocket Reference (O'Reilly)) Chronic Myeloid Leukemia - A Reference Guide (BONUS DOWNLOADS) (The Hill Resource and Reference Guide Book 113)

[Dmca](#)