SCRIPTING WITH OBJECTS

A COMPARATIVE PRESENTATION OF OBJECT-ORIENTED SCRIPTING WITH **PERL** AND **PYTHON**

AVINASH C. KAK

Scripting with Objects

Scripting with Objects

A Comparative Presentation of Object-Oriented Scripting With Perl and Python

Avinash C. Kak
Purdue University

WILEY-
INTERSCIENCE

A John Wiley & Sons, Inc., Publication

Copyright © 2008 by John Wiley & Sons, Inc. All rights reserved.

Published by John Wiley & Sons, Inc., Hoboken, New Jersey.
Published simultaneously in Canada.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning, or otherwise, except as permitted under Section 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 750-4470, or on the web at www.copyright.com. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at http://www.wiley.com/go/permission.

Limit of Liability/Disclaimer of Warranty: While the publisher and author have used their best efforts in preparing this book, they make no representations or warranties with respect to the accuracy or completeness of the contents of this book and specifically disclaim any implied warranties of merchantability or fitness for a particular purpose. No warranty may be created or extended by sales representatives or written sales materials. The advice and strategies contained herein may not be suitable for your situation. You should consult with a professional where appropriate. Neither the publisher nor author shall be liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential, or other damages.

For general information on our other products and services or for technical support, please contact our Customer Care Department within the United States at (800) 762-2974, outside the United States at (317) 572-3993 or fax (317) 572-4002.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic format. For information about Wiley products, visit our web site at www.wiley.com.

Library of Congress Cataloging-in-Publication Data:

Kak, Avinash C.
Scripting with objects : a comparative presentation of object-oriented scripting with Perl and Python / Avinash C. Kak.
p. cm.
ISBN 978-0-470-17923-9
1. Object-oriented programming (Computer science) 2. Scripting languages (Computer science) 3. Perl (Computer program language) 4. Python (Computer program language) I. Title.
QA76.64.K3555 2008
005.1'17—dc22 2007035480

Printed in the United States of America.

10 9 8 7 6 5 4 3 2 1

To

my daughter

Carina

Contents in Brief

1 Multilanguage View of Application Development and OO Scripting 1

2 Perl — A Review of the Basics 13

3 Python — A Review of the Basics 185

4 Regular Expressions for String Processing 325

5 References in Perl 423

6 The Notion of a Class in Perl 465

7 The Notion of a Class in Python 523

8 Inheritance and Polymorphism in Perl 583

9 Inheritance and Polymorphism in Python 631

10 Handling Exceptions 687

11 Abstract Classes and Methods 719

12 Weak References for Memory Management 739

13 Scripting for Graphical User Interfaces 771

14 Multithreaded Scripting 861

15 Scripting for Network Programming 915

16 Interacting with Databases 1005

17 Processing XML with Perl and Python 1067

References 1237

Contents

Preface xxvii

Acknowledgments xxxiii

1 Multilanguage View of Application Development and OO Scripting 1

1.1 Scripting Languages Versus Systems Programming Languages 3

1.2 Organization of this Book 9

1.3 Credits and Suggestions for Further Reading 11

2 Perl — A Review of the Basics 13

2.1 Scalar Values in Perl 14

2.1.1 Numbers 15

2.1.2 Strings 21

2.2 Perl's Variables: Scalars, Arrays, and Hashes 33

2.2.1 Scalar 33

2.2.2 Array 34

2.2.3 Hash 43

2.3 Lexical Scope, Lexical Variables, and Global Variables 48

2.3.1 Lexical Variables 49

2.3.2 Package Variables 51

2.4 Displaying Arrays 58

2.5 Displaying Hashes 63

2.6 Terminal and File I/O 67

2.6.1 Terminal I/O 68

2.6.2 File I/O 70

2.6.2.1 I/O for Text Files 72

2.6.2.2 I/O for Binary Files 74

2.7 Functions, Subroutines, and Functions Used as Operators 83

2.7.1 Using a Function as an Operator 83

2.7.2 User-Defined Functions 86

2.7.3 Passing Arguments to Functions 87

2.7.4 Functions Can be Called with Keyword Arguments 90

2.7.5 Default Values for Function Arguments 91

2.8 What Is Returned by Evaluation Depends on Context 93

2.9 Conditional Evaluation and Loop Control Structures 100

2.9.1 Controlling an Outer Loop from an Inner Loop 107

2.9.2 When Is a Conditional True or False? 109

2.9.3 Complex Conditionals 110

2.10 Functions Supplied with Here-Doc Arguments 113

2.11 Modules and Packages in Perl 118

2.11.1 Creating Your Own Module 120
2.11.2 Importing Names from a Module 124
2.11.3 "Nesting" of Modules 127
2.11.4 The Autoloading Feature 127
2.11.5 Package Constructors and Destructors 129

2.12 Temporarily Localizing a Global Variable 131

2.13 Typeglobs for Global Names 134

2.13.1 Creating Global Variables by Direct Assignments to Typeglob Slots 138

2.14 The eval Operator 139

2.15 grep( ) and map( ) Functions 142

2.16 Interacting with the Directory Structure 147

2.16.1 Directory Handles 150
2.16.2 File Tests 150
2.16.3 Taking Advantage of Shell's Globbing 151
2.16.4 Scanning a Directory Tree 152

2.17 Launching Processes 156

2.17.1 Launching a Child Process with system() 159
2.17.2 Launching a Child Process with Backticks 164
2.17.3 exec( ) for Transferring Control to a Process 166
2.17.4 Launching a Child Process with fork() 167
2.17.5 open( ) for Interprocess Communications 172

2.18 Sending and Trapping Signals 176

2.19 Credits and Suggestions for Further Reading 180

2.20 Homework 180

3 **Python — A Review of the Basics** **185**

3.1 Language Model: Perl versus Python 190

3.2 Numbers 192

3.3 Python Containers: Sequences 200

3.3.1 Strings 204

3.3.2 Tuples 217

3.3.3 Lists 220

3.3.4 Xrange Sequences 226

3.4 Python Containers: Dictionaries 227

3.5 Built-in Types as Classes 233

3.5.1 String Type as a Class 233

3.5.2 Numeric Types as Classes 234

3.6 Subclassing the Built-in Types 235

3.6.1 Subclassing the String Type 235

3.6.2 Subclassing the Integer Type 238

3.7 Terminal and File I/O 241

3.7.1 Terminal I/O 241

3.7.2 File I/O 244

3.7.2.1 I/O for Text Files 246

3.7.2.2 I/O for Binary Files 248

3.8 User-defined Functions 252

3.8.1 A Function Is an Object 253

3.8.2 The Object Returned by a Function Call 254

3.8.3 Default Arguments for Function Parameters 256

3.8.4 Functions Can Be Called with Arbitrary Number of Arguments 257

3.8.5 Functions Can Be Called with Keyword Arguments 258

3.8.6 Anonymous Functions with Lambda Expressions 260

3.8.7 Closures 261

3.9 Control Structures 262

3.9.1 When Is a Conditional True or False? 265

3.9.2 Complex Conditionals 266

3.10 Modules in Python 267

3.10.1 Importing a Module 271

3.10.2 Importing Specific Names from a Module 272

3.11 Scoping Rules, Namespaces, and Name Resolution 274

3.11.1 Nested Namespaces 276

3.11.2 Name Resolution for Imported Modules 280

3.11.3 What about the Names Imported with from...import Syntax? 284

3.11.4 Deeply Nested Namespaces and the global Declaration 286

3.11.5 Python Is Lexically Scoped 287

3.12 The eval( ) Function 288

3.13 map( ) and filter( ) Functions 289

3.14 Interacting with the Directory Structure 292

3.14.1 File Tests 297

3.14.2 Taking Advantage of Shell's Globbing 298

3.14.3 Scanning a Directory Tree 300

3.15 Launching Processes 302

3.15.1 Launching a Child Process with os.system( ) 302

3.15.2 os.exec Functions for Launching External Commands 306

3.15.3 Launching a Child Process with os.fork( ) 310

3.15.4 os.popen( ) for Interprocess Communication 315

3.16 Sending and Trapping Signals 319

3.17 Credits and Suggestions for Further Reading 321

3.18 Homework 321

4 Regular Expressions for String Processing 325

4.1 What is an Input String? 326

4.2 Simple Substring Search 326

4.3 What is Meant by a Match between a Regex and an Input String? 332

4.4 Regex Matching at Line and Word Boundaries 333

4.5 Character Classes for Regex Matching 338

4.6 Specifying Alternatives in a Regex 345

4.7 Subexpression of a Regex 346

4.8 Extracting Substrings from an Input String 347

4.8.1 Other Uses of Parentheses in Regular Expressions 352

4.9 Abbreviated Notation for Character Classes 354

4.10 Quantifier Metacharacters 359

4.10.1 Greediness of Quantifiers 363

4.10.2 The Sometimes Unintended Consequence of Greedy Quantifiers 364

4.10.3 Nongreedy Quantifiers 367

4.10.4 Perl and Python Example Scripts with Quantifiers 369

4.11 Match Modifiers 377

4.11.1 Case-Insensitive Matching 377

4.11.2 Going Global 380

4.11.3 Input Strings Consisting of Multiple Lines 386

4.11.4 Multiline Regular Expressions 393

4.11.5 Other Match Modifiers 396

4.12 Splitting Strings 397

4.12.1 Joining Strings 404

4.13 Regexes for Search and Replace Operations 406

4.14 Credits and Suggestions for Further Reading 411

4.15 Homework 411

5 References in Perl 423

5.1 Referencing and Dereferencing Operators (Summary) 424

5.2 Referencing and Dereferencing a Scalar 424

5.3 Referencing and Dereferencing a Named Array 427

5.4 Referencing and Dereferencing an Anonymous Array 429

5.5 Referencing and Dereferencing a Named Hash 434

5.6 Referencing and Dereferencing an Anonymous Hash 437

5.7 Referencing and Dereferencing A Named Subroutine 440

5.8 Referencing and Dereferencing An Anonymous Subroutine 443

5.9 Subroutines Returning References to Subroutines 445

5.10 Closures 446

5.11 Enforcing Privacy in Modules 449

5.12 References to Typeglobs 451

5.13 The ref( ) Function 452

5.14 Symbolic References 454

5.14.1 Symbolic References to Subroutines 460

5.15 Credits and Suggestions for Further Reading 461