

Intermediate to Advanced Perl Programming

Length:

5 days

Description

Intermediate to Advanced Perl Programming is a unique introduction to the language and its culture and provides a unique perspective on the evolution of Perl and its future direction. It is a comprehensive guide to all the nooks and crannies of the language. This course has been expanded to cover version 5.6 of this maturing language.

Course Objectives:

Upon completion of this course you will:

- Be able to manipulate number and text files and directories, computers, networks and programs
- Be able to easily run external programs and scan their output for interesting tidbits
- Learn an easy way to develop, modify and debug your own programs
- Know how to compile and run your programs portably and on any modern operating system
- Understand typeglobs, pseudohashes and closures and how they really work
- Learn about threading, the compiler, Unicode, and other new features that have been added since the previous edition

Audience:

UNIX, Linux and/or Windows administrators, software engineers, programmers and power users. Previous experience with a procedural or object-oriented programming language is assumed. Knowledge of UNIX scripting tools and regular expressions is very helpful.

Prerequisite:

A course in Beginning Perl Programming or equivalent work experience is required.

Course Content:**Chapter 1 - An Overview of Perl**

Getting Started

Natural and Artificial Languages

An Average Example

Filehandles

Operators

Control Structures

Regular Expressions

List Processing

What You Don't Know Won't Hurt You (Much)

Chapter 2 - Bits and Pieces

- Atoms
- Molecules
- Built-in Data Types
- Variables
- Names
- Scalar Values
- Context
- List Values and Arrays
- Hashes
- Typeglobs and Filehandles
- Input Operators

Chapter 3 - Unary and Binary Operators

- Terms and List Operators (Leftward)
- The Arrow Operator
- Autoincrement and Autodecrement
- Exponentiation
- Ideographic Unary Operators
- Binding Operators
- Multiplicative Operators
- Additive Operators
- Shift Operators
- Named Unary and File Test Operators
- Relational Operators
- Equality Operators
- Bitwise Operators
- C-Style Logical (Short-Circuit) Operators
- Range Operator
- Conditional Operator
- Assignment Operators
- Comma Operators
- List Operators (Rightward)
- Logical and, or, not, and xor
- C Operators Missing from Perl

Chapter 4 - Statements and Declarations

- Simple Statements
- Compound Statements
- if and unless Statements
- Loop Statements

- Bare Blocks
- goto
- Global Declarations
- Scoped Declarations
- Pragmas

Chapter 5 - Pattern Matching

- The Regular Expression Bestiary
- Pattern-Matching Operators
- Metacharacters and Metasymbols
- Character Classes
- Quantifiers
- Positions
- Capturing and Clustering
- Alternation
- Staying in Control
- Fancy Patterns

Chapter 6 - Subroutines

- Syntax
- Semantics
- Passing References
- Prototypes
- Subroutine Attributes

Chapter 7 - Formats

- Format Variables
- Footers

Chapter 8 - References

- What Is a Reference?
- Creating References
- Using Hard References
- Symbolic References
- Braces, Brackets, and Quoting

Chapter 9 - Data Structures

- Arrays of Arrays
- Hashes of Arrays
- Arrays of Hashes
- Hashes of Hashes

Hashes of Functions
More Elaborate Records
Saving Data Structures

Chapter 10 - Packages

Symbol Tables
Autoloading

Chapter 11 - Modules

Using Modules
Creating Modules
Overriding Built-in Functions

Chapter 12 - Objects

Brief Refresher on Object-Oriented Lingo
Perl's Object System
Method Invocation
Object Construction
Class Inheritance
Instance Destructors
Managing Instance Data
Managing Class Data
Summary

Chapter 13 - Overloading

The overload Pragma
Overload Handlers
Overloadable Operators
The Copy Constructor (=)
When an Overload Handler Is Missing (nomethod and fallback)
Overloading Constants
Public Overload Functions
Inheritance and Overloading
Run-Time Overloading
Overloading Diagnostics

Chapter 14 - Tied Variables

Tying Scalars
Tying Arrays
Tying Hashes
Tying Filehandles

A Subtle Untying Trap
Tie Modules on CPAN

Chapter 15 - Unicode

Building Character
Effects of Character Semantics
Caution, `\[ren2bold]` Working

Chapter 16 - Interprocess Communication

Signals
Files
Pipes
System V IPC
Sockets

Chapter 17 - Threads

The Process Model
The Thread Model

Chapter 18 - Compiling

The Life Cycle of a Perl Program
Compiling Your Code
Executing Your Code
Compiler Backends
Code Generators
Code Development Tools
Avant-Garde Compiler, Retro Interpreter

Chapter 19 - The Command-Line Interface

Command Processing
Environment Variables

Chapter 20 - The Perl Debugger

Using the Debugger
Debugger Commands
Debugger Customization
Unattended Execution
Debugger Support
The Perl Profiler

Chapter 21 - Internals and Externals

- How Perl Works
- Internal Data Types
- Extending Perl (Using C from Perl)
- Embedding Perl (Using Perl from C)
- The Moral of the Story

Chapter 22 - CPAN

- The CPAN modules Directory
- Using CPAN Modules
- Creating CPAN Modules

Chapter 23 - Security

- Handling Insecure Data
- Handling Timing Glitches
- Handling Insecure Code

Chapter 24 - Common Practices

- Common Goofs for Novices
- Efficiency
- Programming with Style
- Fluent Perl
- Program Generation

Chapter 25 - Portable Perl

- Newlines
- Endianness and Number Width
- Files and Filesystems
- System Interaction
- Interprocess Communication (IPC)
- External Subroutines (XS)
- Standard Modules
- Dates and Times
- Internationalization
- Style

Chapter 26 - Plain Old Documentation

- Pod in a Nutshell
- Pod Translators and Modules
- Writing Your Own Pod Tools
- Pod Pitfalls
- Documenting Your Perl Programs

Chapter 27 - Perl Culture

History Made Practical

Perl Poetry

Chapter 28 - Special Names

Special Names Grouped by Type

Special Variables in Alphabetical Order

Chapter 29 - Functions

Perl Functions by Category

Perl Functions in Alphabetical Order

Chapter 30 - The Standard Perl Library

Library Science

A Tour of the Perl Library

Chapter 31 - Pragmatic Modules

Chapter 32 - Standard Modules

Listings by Type

Benchmark

Chapter 33 - Diagnostic Messages