

# PROFESSIONAL LINUX CONSULTING & TRAINING

## CURSO PERL INTERMEDIO

### OBJETIVO

Que el estudiante entienda y aplique de manera efectiva los conceptos de módulos, paquetes, referencias, programación orientada a objetos, así como prueba y distribución de sistemas desarrollados con Perl.

### TEMARIO

- Intermediate Foundations
  - List Operators
  - Trapping Errors with eval
  - Dynamic Code with eval
  - Exercises
- Using Modules
  - The Standard Distribution
  - Using Modules
  - Functional Interfaces
  - Selecting What to Import
  - Object-Oriented Interfaces
  - A More Typical Object-Oriented Module: Math::BigInt
  - The Comprehensive Perl Archive Network
  - Installing Modules from CPAN
  - Setting the Path at the Right Time
  - Exercises
- Introduction to References
  - Performing the Same Task on Many Arrays
  - Taking a Reference to an Array
  - Dereferencing the Array Reference
  - Getting Our Braces Off
  - Modifying the Array
  - Nested Data Structures
  - Simplifying Nested Element References with Arrows
  - References to Hashes
  - Exercises
- References and Scoping
  - More Than One Reference to Data
  - What If That Was the Name?
  - Reference Counting and Nested Data Structures



- When Reference Counting Goes Bad
- Creating an Anonymous Array Directly
- Creating an Anonymous Hash
- Autovivification
- Autovivification and Hashes
- Exercises
- Manipulating Complex Data Structures
  - Using the Debugger to View Complex Data
  - Viewing Complex Data with Data::Dumper
  - YAML
  - Storing Complex Data with Storable
  - Using the map and grep Operators
  - Applying a Bit of Indirection
  - Selecting and Altering Complex Data
  - Exercises
- Subroutine References
  - Referencing a Named Subroutine
  - Anonymous Subroutines
  - Callbacks
  - Closures
  - Returning a Subroutine from a Subroutine
  - Closure Variables as Inputs
  - Closure Variables as Static Local Variables
  - Exercises
- Filehandle References
  - The Old Way
  - The Improved Way
  - The Even Better Way
  - IO::Handle
  - Directory Handle References
- Exercises
- Practical Reference Tricks
  - Review of Sorting
  - Sorting with Indices
  - Sorting Efficiently
  - The Schwartzian Transform
  - Multi-Level Sort with the Schwartzian Transform
  - Recursively Defined Data
  - Building Recursively Defined Data
  - Displaying Recursively Defined Data
  - Exercises
- Building Larger Programs
  - The Cure for the Common Code
  - Inserting Code with eval
  - Using do
  - Using require
  - require and @INC
  - The Problem of Namespace Collisions
  - Packages as Namespace Separators
  - Scope of a Package Directive
  - Packages and Lexicals
  - Exercises
- Introduction to Objects
  - If We Could Talk to the Animals...
  - Introducing the Method Invocation Arrow
  - The Extra Parameter of Method Invocation
  - Calling a Second Method to Simplify Things
  - A Few Notes About @ISA
  - Overriding the Methods
  - Starting the Search from a Different Place



- The SUPER Way of Doing Things
- What to Do with @\_
- Where We Are So Far...
- Exercises
- Objects with Data
  - A Horse Is a Horse, of Course of Courseor Is It?
  - Invoking an Instance Method
  - Accessing the Instance Data
  - How to Build a Horse
  - Inheriting the Constructor
  - Making a Method Work with Either Classes or Instances
  - Adding Parameters to a Method
  - More Interesting Instances
  - A Horse of a Different Color
  - Getting Our Deposit Back
  - Don't Look Inside the Box
  - Faster Getters and Setters
  - Getters That Double as Setters
  - Restricting a Method to Class-Only or Instance-Only
  - Exercise
- Object Destruction
  - Cleaning Up After Yourself
  - Nested Object Destruction
  - Beating a Dead Horse
  - Indirect Object Notation
  - Additional Instance Variables in Subclasses
  - Using Class Variables
  - Weakening the Argument
  - Exercise
- Some Advanced Object Topics
  - UNIVERSAL Methods
  - Testing Our Objects for Good Behavior
  - AUTOLOAD as a Last Resort
  - Using AUTOLOAD for
- Accessors
  - Creating Getters and Setters More Easily
  - Multiple Inheritance
  - Exercises
- Exporter
  - What use Is Doing
  - Importing with Exporter
  - @EXPORT and @EXPORT\_OK
  - %EXPORT\_TAGS
  - Exporting in a Primarily OO Module
  - Custom Import Routines
  - Exercises
- Writing a Distribution
  - There's More Than One Way To Do It
  - Using h2xs
  - Embedded Documentation
  - Controlling the Distribution with Makefile.PL
  - Alternate Installation Locations (PREFIX=...)
  - Trivial make test
  - Trivial make install
  - Trivial make dist
  - Using the Alternate Library Location
  - Exercise
  - Essential Testing
  - More Tests Mean Better Code
  - A Simple Test Script
  - The Art of Testing
  - The Test Harness
  - Writing Tests with Test::More
  - Testing Object-Oriented Features
  - A Testing To-Do List
  - Skipping Tests
  - More Complex Tests (Multiple Test Scripts)
  - Exercise



- Advanced Testing

- Testing Large Strings
- Testing Files
- Testing STDOUT or STDERR
- Using Mock Objects
- Testing POD

- Coverage Testing
- Writing Your Own Test::\*  
Modules
- Exercises

Duración: 30 horas  
Material: DVD de Fedora Core Linux y CD de programas y utilerías de Perl  
Documentación: [Intermediate Perl](#)

