CP150 - Advanced Perl Programming

Course Description:

Perl has evolved from its beginnings as an eclectic scripting tool for UNIX administrators into one of the most popular, influential, and widely used computer languages in history. In this course, you will learn how to fully utilize the Perl programming language.

Audience:

Application programmers, system administrators, web-site authors, webmasters, and UNIX/NT power users.

Prerequisites:

Perl Programming and Perl application development experience. Full comprehension of the extending and embedding material will require some C or C++ programming experience.

Course Contents

- Warnings
- Diagnostic Messages
- Carping, Confessing, and Croaking
- Strict Checks
- Compiler Pragmas
- Debugging Flags
- Your Perl Configuration
- The Devel::Peek Module
- The Data::Dumper Module

Expert List Manipulation

- The grep Operator
- Lists, Arrays, and List Operators
- Context
- Context and Subroutines
- Initializing Arrays and Hashes
- Reference Syntax
- Auto-vivification
- Defined Values
- Other List Operators
- Usage of map, grep, and foreach

Blocks and Code References

- Blocks
- Subroutines
- Subroutine Prototypes
- Code Refs and Anonymous Subroutines
- Typeglobbing for the Non-Squeamish
- Local (Dynamic) Variables
- Lexical Variables
- Persistent Private Subroutine Variables
- Closures
- The eval Operator
- The Block Form of eval
- The String Form of eval
- Block Form of eval for Exception Handling

Packages

- Review of Packages
- BEGIN and END Blocks
- Symbol Tables
- Package Variables
- Calling Package Subroutines
- Importing Package Symbols
- Exporting Package Symbols
- Using the Exporter Package
- The use Function
- AUTOLOAD and @ISA
- AutoLoader and SelfLoader

Objects and Classes

- Object-Oriented Stuff
- Making Perl Object-Oriented
- References
- The bless Function
- So, What’s a Blessed Thing Good For?
- Calling Class and Object Methods
- Object Methods
- Writing Classes
- Constructors
- Inheritance
- What Perl Doesn’t Do

Tied Variables

- Why Use tie?
- Tying a Scalar
- Inside Tied Variables
- untie
- Another Tied Scalar Example
- Tying an Array
- A Tied Array Example
- Tying Hashes
- Tie::Hash and Tie::Array
- Tying Filehandles
- What Are DBM, NDBM, GDBM, SDBM, etc?
- Using the DBM Modules

Installing and Using Perl Modules

- Laziness, Impatience, and Hubris
- CPAN
- Using Modules
- Installing a Perl Module
- Unpacking the Module Source
- The Configuration Step
- The Build Step
- The Test Step
- The Install Step
- Using CPAN.pm
- Using Module Documentation

Introduction to DBI/DBD

- The Old Way - DBPerls
- A Better Way - DBI/DBD
- Database Programming
- Handles
- Connecting to the Database
- Creating a SQL Query
- Getting the Results
- Updating Database Data
- Transaction Management
- Finishing Up

DBI/DBD SQL Programming

- Error Checking in DBI
- Getting Connected
- Drivers
- Using Parameterized Statements
- Statement Handle Attributes
- Other Handle Attributes
- Column Binding
- The do Method
- BLOBs and LONGs and Such
- Installing DBI Drivers

CP150 - Advanced Perl Programming

Introduction to Perl/Tk

- Tcl, Tk, Tcl/Tk, Tkperl, Perl/Tk, etc.
- Perl/Tk
- Creating a Perl/Tk Application
- GUI Programming Overview
- Adding Widgets
- Scrolled Widgets
- Configuring Widgets
- Menus
- More Fun with Menus
- Using FileSelect

Perl/Tk Programming

- Tk::Error and Tk::ErrorDialog
- Configuring Widgets
- Geometry Management
- Geometry Management with grid()
- The Frame Widget
- Defining Widget Callbacks
- Bindings
- Nonblocking I/O with fileevent()
- Tags
- Other Widgets
- Other Tk Commands
- Getting Tk

Extending Perl with C/C++

- Extending the Perl Interpreter
- Overview of Perl5 XSUBs
- Get Started with h2xs
- Set up the Perl Wrapper Class
- Write the XS Code
- The XS File
- Write Some Test Code
- What Do You Want?
- Returning Values on the Stack
- A Walk Through an XSUB
- Arguments to XSUBs
- Other h2xs Options

Embedding the Perl Interpreter

- Why Embed Perl?
- Embedding Perl in a C Program
- Compiling the Program
- perlmain.c
- Perl Data Types
- Macros and Functions
- Manipulating Scalars
- Memory Management
- Script Space
- Evaluating Perl Expressions
- Dynamic Loading
- Multiple Perl Interpreters

Module Development and Distribution

- Distributing Modules
- Get Started with h2xs
- Files Created by h2xs
- The Build Library (blib) Directory
- Unit Testing and test.pl
- Versions
- Using blib
- POD
- POD Translators
- Cutting a Distribution
- Other Niceties
- Makefile.PL

Design and Implementation

- Think First
- Object-Oriented Design
- Object-Oriented Development
- Library Modules
- Utility Programs
- Filters
- Performance
- Timing with Benchmark