

Course duration

- 4 days

Course Benefits

- Learn to debug Perl applications
- Learn to efficiently manipulate lists in Perl
- Learn to work more effectively with Perl packages
- Learn to work with Tied Variables in Perl
- Learn to manipulate databases with Perl
- Learn to effectively use Perl Modules
- Learn to work with Perl/Tk
- Learn to extend Perl with C/C++
- Learn to embed the Perl Interpreter into a program
- Learn to create your own Perl modules

Course Outline

1. Debugging Perl
 1. Warnings
 2. Diagnostic Messages
 3. Carping, Confessing, and Croaking
 4. Strict Checks
 5. Compiler Pragmas
 6. Debugging Flags
 7. Your Perl Configuration
 8. The Devel::Peek Module
 9. The Data::Dumper Module
2. Expert List Manipulation in Perl
 1. The grep Operator
 2. Lists, Arrays, and List Operators
 3. Context
 4. Context and Subroutines
 5. Initializing Arrays and Hashes
 6. Reference Syntax
 7. Auto-vivification
 8. Defined Values
 9. Other List Operators
 10. Usage of map, grep, and foreach
3. Blocks and Code References in Perl
 1. Blocks

2. Subroutines
3. Subroutine Prototypes
4. Code Refs and Anonymous Subroutines
5. Typglobbing for the Non-Squeamish
6. Local (Dynamic) Variables
7. Lexical Variables
8. Persistent Private Subroutine Variables
9. Closures
10. The eval Operator
11. The Block Form of eval
12. The String Form of eval
13. Block Form of eval for Exception Handling
4. Perl Packages
 1. Review of Packages
 2. BEGIN and END Blocks
 3. Symbol Tables
 4. Package Variables
 5. Calling Package Subroutines
 6. Importing Package Symbols
 7. Exporting Package Symbols
 8. Using the Exporter Package
 9. The use Function
 10. AUTOLOAD and @ISA
 11. AutoLoader and SelfLoader
5. Objects and Classes in Perl
 1. Object-Oriented Stuff
 2. Making Perl Object-Oriented
 3. References
 4. The bless Function
 5. So, What's a Blessed Thing Good For?
 6. Calling Class and Object Methods
 7. Object Methods
 8. Writing Classes
 9. Constructors
 10. Inheritance
 11. What Perl Doesn't Do
6. Tied Variables in Perl
 1. Why Use tie?
 2. Tying a Scalar
 3. Inside Tied Variables
 4. untie
 5. Another Tied Scalar Example
 6. Tying an Array
 7. A Tied Array Example
 8. Tying Hashes
 9. Tie::Hash and Tie::Array
 10. Tying Filehandles

11. What Are DBM, NDBM, GDBM, SDBM, etc?
12. Using the DBM Modules
7. Installing and Using Perl Modules
 1. Laziness, Impatience, and Hubris
 2. CPAN
 3. Using Modules
 4. Installing a Perl Module
 5. Unpacking the Module Source
 6. The Configuration Step
 7. The Build Step
 8. The Test Step
 9. The Install Step
10. Using CPAN.pm
11. Using Module Documentation
8. Introduction to DBI/DBD in Perl
 1. The Old Way - DBPerls
 2. A Better Way - DBI/DBD
 3. Database Programming
 4. Handles
 5. Connecting to the Database
 6. Creating a SQL Query
 7. Getting the Results
 8. Updating Database Data
 9. Transaction Management
10. Finishing Up
9. DBI/DBD SQL Programming
 1. Error Checking in DBI
 2. Getting Connected
 3. Drivers
 4. Using Parameterized Statements
 5. Statement Handle Attributes
 6. Other Handle Attributes
 7. Column Binding
 8. The do Method
 9. BLOBs and LONGs and Such
10. Installing DBI Drivers
10. Introduction to Perl/Tk
 1. Tcl, Tk, Tcl/Tk, Tkperl, Perl/Tk, etc.
 2. Perl/Tk
 3. Creating a Perl/Tk Application
 4. GUI Programming Overview
 5. Adding Widgets
 6. Scrolled Widgets
 7. Configuring Widgets
 8. Menus
 9. More Fun with Menus
10. Using FileSelect

- 11. Perl/Tk Programming
 - 1. Tk::Error and Tk::ErrorDialog
 - 2. Configuring Widgets
 - 3. Geometry Management
 - 4. Geometry Management with grid()
 - 5. The Frame Widget
 - 6. Defining Widget Callbacks
 - 7. Bindings
 - 8. Nonblocking I/O with fileevent()
 - 9. Tags
 - 10. Other Widgets
 - 11. Other Tk Commands
 - 12. Getting Tk
- 12. Extending Perl with C/C++
 - 1. Extending the Perl Interpreter
 - 2. Overview of Perl5 XSUBS
 - 3. Get Started with h2xs
 - 4. Set up the Perl Wrapper Class
 - 5. Write the XS Code
 - 6. The XS File
 - 7. Write Some Test Code
 - 8. What Do You Want?
 - 9. Returning Values on the Stack
 - 10. A Walk Through an XSUB
 - 11. Arguments to XSUBS
 - 12. Other h2xs Options
- 13. Embedding the Perl Interpreter
 - 1. Why Embed Perl?
 - 2. Embedding Perl in a C Program
 - 3. Compiling the Program
 - 4. perlmain.c
 - 5. Perl Data Types
 - 6. Macros and Functions
 - 7. Manipulating Scalars
 - 8. Memory Management
 - 9. Script Space
 - 10. Evaluating Perl Expressions
 - 11. Dynamic Loading
 - 12. Multiple Perl Interpreters
- 14. Module Development and Distribution in Perl
 - 1. Distributing Modules
 - 2. Get Started with h2xs
 - 3. Files Created by h2xs
 - 4. The Build Library (blib) Directory
 - 5. Unit Testing and test.pl
 - 6. Versions
 - 7. Using blib

8. POD
9. POD Translators
10. Cutting a Distribution
11. Other Niceties
12. Makefile.PL
15. Design and Implementation in Perl
 1. Think First
 2. Object-Oriented Design
 3. Object-Oriented Development
 4. Library Modules
 5. Utility Programs
 6. Filters
 7. Performance
 8. Timing with Benchmark

Class Materials

Each student will receive a comprehensive set of materials, including course notes and all the class examples.

Class Prerequisites

Experience in the following *is required* for this Perl class:

- Perl Programming and Perl application development experience

Experience in the following *would be useful* for this Perl class:

- C or C++ programming