

# COURSE CONTENTS FOR PERL

## COURSE LENGTH:

4 Days

## COURSE DESCRIPTION:

Perl is a scripting language which allows for rapid prototyping of projects formerly done with a programming language or a shell. It incorporates all the functionality of C (including a UNIX system interface), the Shells, grep, sed, and awk. The topics in the course will aid all computer users - from end user to programmer to administrator alike. Many in-class labs support the course material.

## WHO SHOULD ATTEND:

This course is for programmers, end users, system administrators, network administrators, CGI script writers, or anybody who is interested in automating tasks but doesn't want to learn all the details of a full blown programming language.

## BENEFITS OF ATTENDANCE:

Upon completion of this course, students will be able to:

- Manipulate files and directories
- Use arrays and array functions to solve a variety of problems
- Use the powerful regular expression capabilities of Perl
- Generate reports
- Use hashes to solve commonly encountered problems
- Take advantage of Perl's powerful system interface
- Write programs that solve many system administrator problems
- Use Perl to write CGI applications
- Use Modules from the standard Perl distribution
- Use Perl references

## PREREQUISITES:

Students should have some experience with either a programming language (preferably C), or any of the UNIX shells.

## COURSE OUTLINE:

- **CHAPTER 1: GETTING STARTED WITH PERL**
  1. What is Perl?
  2. Where Can I Get Perl?

3. A Simple Perl Program
4. Simple I/O
5. Perl Variables
6. Control Flow - Decisions
7. Control Flow - Loops
8. Altering Loop Control Flow
9. Statement Modifiers
10. What Is True And What Is False?
11. The Special Variable `$_`

- **CHAPTER 2: PERL OPERATORS**

1. Introduction
2. Table Of Perl Operators
3. Arithmetic Operators
4. String Operators
5. Relational Operators
6. Logical Operators
7. Bitwise Operators
8. Assignment Operators
9. The Conditional Operator
10. Range Operator
11. String Functions
12. The `eval` Function

- **CHAPTER 3: I/O**

1. Introduction
2. String Literals
3. The `print` Function
4. Here Documents
5. The `printf` Function
6. The `sprintf` Function
7. Filehandles
8. Opening Disk Files
9. File Open Errors
10. The `die` and `warn` Functions
11. File Operators

- **CHAPTER 4: ARRAYS**

1. Basic Concepts

2. Assigning Values To An Array
3. Accessing Array Elements
4. Array Functions
5. `push` and `pop`
6. `shift`
7. `sort`, `reverse`, and `chop`
8. `split` and `join`
9. `grep`
10. `splice`
11. Command Line Arguments

- **CHAPTER 5: ASSOCIATIVE ARRAYS**

1. Basic Concepts
2. Associative Array Functions
3. Updating Associative Arrays
4. Accessing Environment Variables

- **CHAPTER 6: SUBROUTINES**

1. Calling Subroutines
2. Passing Arguments to Subroutines
3. Returning Values from Subroutines
4. The `require` Function
5. Packages and Modules
6. The `@INC` Array
7. Predefined Subroutines
8. Comparison Subroutines for Sorting

- **CHAPTER 7: PATTERN MATCHING AND REGULAR EXPRESSIONS**

1. Introduction
2. Regular Expression Syntax
3. The Match Operator
4. Regular Expression Meta-Characters
5. Anchors
6. Single Character Matches
7. Some Special Issues
8. Character Classes
9. Multiple Character Matches
10. Alternation
11. The Substitution Operator

## 12. The Translation Operator

- **CHAPTER 8: ACCESSING SYSTEM RESOURCES**

1. Introduction
2. File and Directory System Calls
3. The `stat` Function
4. The `utime` Function
5. The `fork` Function
6. The `exec` and `wait` Functions
7. Handling Signals
8. The `system` Function
9. Command Substitution
10. Opening Pipe Files

- **CHAPTER 9: GENERATING REPORTS WITH PERL**

1. Formats
2. Formatting Examples
3. Multi-Line Values
4. Multi-Line Text Blocks
5. Sending a Report to a File
6. The `select` Function
7. The Special Variable `$~`
8. Top-of-Page Formats
9. Bottom-of-Page Formats
10. A Sample Report

- **CHAPTER 10: PERL AND CGI**

1. What is CGI?
2. Web Servers and Browsers
3. HTML
4. HTML Forms
5. Form Elements
6. A Typical CGI Application
7. CGI Input
8. CGI Output
9. Using the `CGI.pm` Module
10. CGI Environment Variables

- **APPENDIX A: COMMAND LINE OPTIONS AND DEBUGGING**

1. Running Perl on the Command Line

2. Summary of Command Line Options
3. The Perl Debugger
4. Perl Debugger Commands
5. Other Debugging Aids
6. The `strict` Module

- **APPENDIX B: MORE ABOUT REGULAR EXPRESSIONS**

1. Remembered Matches
2. Greedy Regular Expressions
3. Nested Remembered Patterns
4. Matching for Multiple Occurrences in a Loop

- **APPENDIX C: REFERENCES AND DATA STRUCTURES**

1. References
2. Syntactic Sugar
3. Anonymous Arrays
4. Higher Dimensional Arrays
5. References and Subroutines

- **APPENDIX D: COMPARING PERL 4 AND 5**

1. Operators
2. Packages
3. Typeglobs

- **APPENDIX E: ADVANCED PERL**

1. What is in the Advanced Perl Course
2. A Taste of Object Orientation
3. A Taste of Network Programming

- **APPENDIX F: PERL DEVELOPMENT USING ECLIPSE**

1. Features of the Perl Plugin
2. Creating a Perl Project
3. Running a Perl Program
4. Preferences - EPIC
5. Preferences - Editor
6. Preferences - Content Assist
7. Preferences - Folding
8. Preferences - Mark Occurrences
9. Preferences - Templates
10. Templates
11. Preferences - Source Formatter

12. Preferences - Task Tags
13. Accessing Perl Documentation
14. Project Properties
15. Debugging
16. Testing Regular Expressions

- **APPENDIX G: REFERENCE**

1. Special Perl Variables
2. Regular Expression Meta-Characters
3. Internet Resources