



## PERL Programming

### Overview:

PERL is one of the most popular and prolific scripting languages around in the computing industry. Widely used, it is a handy tool for every engineer in the computing world. The goal of this 3-day program from Cralina is to provide a solid introduction to PERL for participants who wish to learn this programming language. This program will also provide insights into the various aspects of PERL programming that has made PERL a very popular programming language.

### Target Audience:

- Developers/Designers, Senior Developers/Designers
- Engineers who want to learn programming in PERL
- Engineers who know shell scripting and who want to transition to PERL

### Pre-requisite

- Familiarity with programming and operating systems.

### Delivery Method:

Instructor lead and hands on. The program has an emphasis on hands-on coding for almost every topic.

### Program Contents:

1. Introduction to PERL
  - How to Get PERL?
  - Origin of "PERL"
  - How Do I Make a PERL Program
  - A tour of PERL
2. Scalar Data
  - What Is Scalar Data?
  - Numbers, Strings
  - PERL's Built-in Warnings
  - Scalar Variables
  - Output with print
  - Control Structures (if)
  - Getting User Input
  - The chomp Operator
  - The while Control Structure
  - The undef Value
  - The defined Function
3. Lists and Arrays



- Accessing Elements of an Array
- Special Array Indices
- List Literals and List Assignment
- Interpolating Arrays into Strings
- The foreach Control Structure
- PERL's Favorite Default: \$\_
- Scalar and List Context
- <STDIN> in List Context

#### 4. Subroutines

- System and User Functions
- Defining a Subroutine
- Invoking a Subroutine
- Return Values
- Arguments
- Private Variables in Subroutines
- The local Operator
- Variable-length Parameter Lists
- Notes on Lexical (my) Variables
- The use strict Pragma
- The return Operator

#### 5. Hash

- What Is a Hash?
- Hash Element Access
- Hash Functions
- Typical Use of a Hash

#### 6. I/O in PERL

- Input from Standard Input
- Input from the Diamond Operator
- The Invocation Arguments
- Output to Standard Output
- Formatted Output with printf

#### 7. Regular Expressions

- What's a Regular Expressions?
- Using Simple Patterns
- A Pattern Test Program
- Character Classes
- General Quantifiers
- Anchors
- Memory Parentheses
- Precedence



## 8. Working with Regular Expressions

- Matches with `m//`
- Option Modifiers
- The Binding Operator, `=~`
- Interpolating into Patterns
- The Match Variables
- Substitutions with `s///`
- The split Operator
- The join Function

## 9. Control structures

- The unless Control Structure
- The until Control Structure
- Expression Modifiers
- The Naked Block Control Structure
- The elsif Clause
- Autoincrement and Autodecrement
- The for Control Structure
- Loop Controls
- Logical Operators

## 10. Filehandles and File Tests

- What Is a Filehandle?
- Opening a Filehandle
- Fatal Errors with `die`
- Using Filehandles
- Reopening a Standard Filehandle
- File Tests

## 11. Directory operations

- Moving Around the Directory Tree
- Globbing
- An Alternate Syntax for Globbing
- Directory Handles
- Recursive Directory Listing

## 12. Working with files and directories

- Removing Files
- Renaming Files
- Links and Files
- Making and Removing Directories
- Modifying Permissions
- Changing Ownership



- Changing Timestamps
- Using Simple Modules

### 13. Process Management

- The system Function
- The exec Function
- The Environment Variables
- Using Backquotes to Capture Output
- Processes as Filehandles
- Getting Down and Dirty with Fork
- Sending and Receiving Signals

### 14. Working with Strings

- Finding a Substring with index
- Manipulating a Substring with substr
- Formatting Data with sprintf
- Advanced Sorting

Duration : 3 days