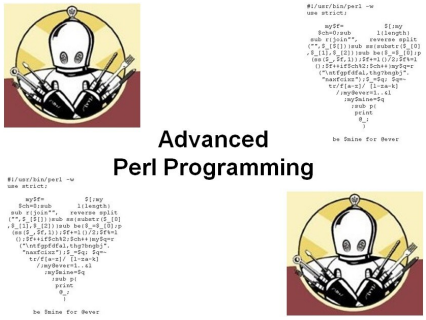
Privé: Advanced Perl Programmeren

Duur: 3 dagen **Prijs: € 1750**

Open Rooster



Doelgroep Cursus Advanced Perl Programmeren

Deze cursus is bedoeld voor ervaren Perl programmeurs die geavanceerde functies van de taal willen leren.

Voorkennis Cursus Advanced Perl

Kennis van en ervaring met programmeren in de programmeertaal Perl is vereist om aan deze cursus te kunnen deelnemen.

Uitvoering Training Advanced Perl Programmeren

De onderwerpen worden besproken aan de hand van presentatie slides. Demo's zorgen voor een nadere verheldering van de behandelde concepten. De theorie wordt afgewisseld met oefeningen. De cursustijden zijn van 9.30 tot 16.30.

Certificering Advanced Perl Programmeren

De deelnemers krijgen na het goed doorlopen van de cursus een officieel certificaat Advanced Perl Programmeren.

Inhoud Privé: Cursus Advanced Perl Programmeren

Cursus Advanced Perl

De cursus Advanced Perl Programmeren is bedoeld voor ervaren Perl programmeurs die de vele geavanceerde onderwerpen die beschikbaar zijn willen verkennen. The Perl taal heeft zich ontwikkeld tot een zeer krachtige multi purpose taal. Modules geschreven voor Perl maken het programmeurs mogelijk om snel programma's te creëren die gebruik maken van complexe datatypes, object-georiënteerde technologieën, grafische gebruikersinterfaces en meer. De onderwerpen die in deze cursus worden behnadeld zijn onder andere het gebruik van Perl packages, de creatie van Perl modules, het gebruik van CPAN en het benaderen van databases met Perl.

SpiralTrain BV
Standerdmolen 10, 2e verdieping
3995 AA Houten

info@spiraltrain.nl
www.spiraltrain.nl
Tel.: +31 (0) 30 – 737 0661

Locaties
Houten, Amsterdam, Rotterdam, Eindhoven, Zwolle, Online

Modules Privé: Cursus Advanced Perl Programmeren

Module 1 : Perl Refresher	Module 2 : Debugging Perl	Module 3 : List Manipulation
Miscellaneous functions Diverse operators grep(), map(), qw// Reference data type Complex data types Advanced regular expressions Modules	Debugging Perl 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 in Perl 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
Module 4 : Blocks and Code References	**Module 5 : Perl Packages**	**Module 6 : Perl Object Orientation**
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	Perl 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 in Perl 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
Module 7 : Tied Variables	**Module 8 : Perl Modules**	**Module 9 : Perl Data Access**
Tied Variables in Perl 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 in Perl 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
Module 10 : Perl SQL Programming	**Module 11 : Perl/Tk**	**Module 12 : Extending Perl**
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	Creating a Perl/Tk Application GUI Programming Overview Adding Widgets Scrolled Widgets Configuring Widgets Menus Using FileSelect Tk::Error and Tk::ErrorDialog Configuring Widgets Geometry Management Geometry Management with grid() The Frame Widget	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