

Pascal Programmeren

Doelgroep Cursus Pascal Programmeren

De cursus Pascal Programmeren is bedoeld voor een ieder die wil leren programmeren in de programmeer taal Pascal.

Voorkennis Cursus Pascal Programmeren

Om aan deze cursus te kunnen deelnemen is basiskennis van programmeren in een andere programmeertaal bevorderlijk voor de begripsvorming maar niet vereist.

Uitvoering Training Pascal Programmeren

De theorie wordt behandeld op basis van presentatie slides. De theorie wordt verduidelijkt door middel van demo's. Na bespreking van een module is er de mogelijkheid te oefenen. De cursustijden zijn van 9.30 tot 16.30.

Certificering Cursus Pascal Programmeren

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

Duur: 3 dagen

Prijs: € 1999

[Open Rooster](#)



Pascal Programming



Inhoud Cursus Pascal Programmeren

In de cursus Pascal Programmeren leren de deelnemers de algemeen toepasbare en oorspronkelijk procedurele taal Pascal voor het ontwikkelen van betrouwbare en efficiënte applicaties. Pascal ontstond begin 70-er jaren met als doel programmeren te onderwijzen als een systematische discipline. Pascal is gebaseerd op de Algol-taal en heeft veel aanhang verworven in academische milieus. Pascal ondersteunt ook object georiënteerd programmeren en het GUI Framework Delphi is gebaseerd op Pascal.

Pascal Intro

De cursus gaat van start met een overzicht van de programma structuur van Pascal, de coding style, de indeling in Units en de Scope rules in Pascal. Ook wordt aandacht besteed aan compiler directives, conditional defines en include files.

Pascal Syntax

Vervolgens wordt ingegaan op de taal syntax van Pascal met variabelen, standaard data types en control flow constructies zoals if clauses en for loops. Hierbij komen ook User Defined Types, Enumerated Types en Exception Handling aan de orde.

Procedures en Functies

Speciale aandacht is er voor procedures en functies in Pascal. De definitie en aanroep van procedures wordt besproken evenals de doorgifte van verschillende typen parameters zoals reference, default en constant parameters. En ook recursieve functies en function overloading komen aan bod.

Arrays en Records

Dan is het tijd voor de behandeling van arrays en records in Pascal. Voor wat betreft arrays worden array indexing, array boundaries en static en dynamic arrays behandeld en ten aanzien van records wordt ingegaan op Record Data Types, Variant Records en Records met Methods en Constructors.

Classes en Objects

Classes en Objects in Pascal zijn ook onderdeel van het programma van de cursus. Hierbij wordt de Pascal implementatie van object georiënteerde concepten als encapsulation, inheritance, polymorphism en abstract classes besproken.

Generics

Tenslotte wordt de cursus afgesloten met de behandeling geparametriseerde types die ook bekend staan onder de naam generics. Generic constraints, generic results en ook smart pointers komen dan aan de orde.

Modules Cursus Pascal Programmeren

Module 1 : Pascal Intro	Module 2 : Language Syntax	Module 3 : Procedures and Functions
Pascal Origins Program Structure Unit Names Program Names Units and Scope Installing Pascal Compiler Directives Conditional Defines Include Files Coding Style Indentation White Space Symbolic Identifiers	Variable Assignments Lifetime and Visibility Global Variables Numeric Types Strings and Characters Floating Points User Defined Types Enumerated Types Expressions and Operators Type Conversions If and Case For and While Exception Handling	Forward Declarations Methods Parameter Passing Reference Parameters Constant Parameters Function Overloading Default Parameters Recursive Functions Inlining Procedural Types Return Values External Declarations Ambiguous Calls
Module 4 : Arrays and Records	Module 5 : Classes and Objects	Module 6 : Generics
Array Data Types Static Arrays Array Size Array Boundaries Dynamic Arrays Array Indexing Record Data Types Variant Records Fields Alignment Records with Methods Records and Constructors	Class Definition Creating Objects Private, Protected, Public Class Methods Encapsulation Self Keyword Constructors Inheritance Polymorphism Abstract Classes Interfaces	Parameterized Types Generic Type Rules Generic Constraints Class Constraints Interface Constraints Generic Containers Object Containers Anonymous Methods Generic Dictionaries Smart Pointers Generic Results