

## **Pascal Programming**

#### **Audience Course Pascal Programming**

The course Pascal Programming is intended for anyone who wants to learn programming in the programming language Pascal.

#### **Prerequisites course Pascal Programming**

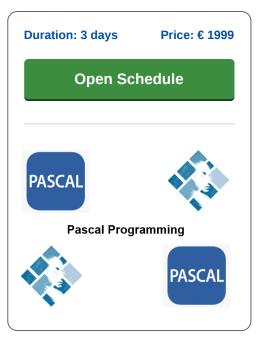
To participate in this course basic knowledge of programming in another programming language is beneficial to the understanding but is not required.

#### **Realization Training Pascal Programming**

The theory is discussed on the basis of presentation slides. The theory is further explained through demos. After discussing a module there is the possibility to practice. Course times are from 9.30 to 16.30.

#### **Certification Pascal Programming**

After successful completion of the course participants receive an official certificate Pascal Programming.



### **Content Course Pascal Programming**

In the Pascal Programming course participants learn the generally applicable and original procedural language Pascal for developing reliable and efficient applications. Pascal was created in the early 1970s with the aim of teaching programming as a systematic discipline. Pascal is based on the Algol language and has gained a lot of following in academic circles. Pascal also supports object-oriented programming and the GUI Framework Delphi is based on Pascal.

#### **Pascal Intro**

The course starts with an overview of the program structure of Pascal, the coding style, the division into Units and the Scope rules in Pascal. Attention is also paid to compiler directives, conditional defines and include files.

#### **Pascal Syntax**

Next the language syntax of Pascal is discussed with variables, standard data types and control flow constructs such as if clauses and for loops. User Defined Types, Enumerated Types and Exception Handling are also treated.

#### **Procedures en Functies**

Special attention is paid to procedures and functions in Pascal. The definition and invocation of procedures is explained as well as the passing of different types of parameters such as reference, default and constant parameters. Also recursive functions and function overloading are treated.

#### **Arrays en Records**

Then it's time to consider arrays and records in Pascal. With regard to arrays, array indexing, array boundaries and static and dynamic arrays are explained and with regard to records Record Data Types, Variant Records and Records with Methods and Constructors are discussed.

#### Classes en Objects

Classes and Objects in Pascal are also part of the program of the course. The Pascal implementation of object oriented concepts such as encapsulation, inheritance, polymorphism and abstract classes is treated.

#### Generics

Finally the course ends with the explanation of parameterized types, also known as generics. Generic constraints, generic results and also smart pointers are part of the subject matter.



# **Modules Course Pascal Programming**

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