

Python Programmeren

Doelgroep voor de Cursus Python Programmeren

De cursus Python Programmeren is bedoeld voor developers en systeembeheerders die willen leren programmeren in Python en andere personen die Python code willen begrijpen.

Voorkennis voor de Training Python Programmeren

Kennis en ervaring met programmeren is niet strikt noodzakelijk om deel te nemen aan deze cursus. Ervaring in programmeren is wel bevorderlijk voor een goede begripsvorming.

Uitvoering Training Python

De theorie in de cursus Python Programmeren wordt behandeld aan de hand van presentatie slides. Illustratieve demo's verduidelijken de concepten. De theorie wordt afgewisseld met oefeningen. De cursustijden zijn van 9.30 tot 16.30.

Certificaat Python Programmeren

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

Duur: 4 dagen

Prijs: € 1999

[Open Rooster](#)



Python Programming



Inhoud Python Programmeren

In de cursus Python Programmeren leren de deelnemers te programmeren in de object georiënteerde scripttaal Python. [Python](#) is een taal die vaak wordt gebruikt voor data analyse, installatie scripts en prototypes van grote applicaties.

Python Fundamentals

Na een inleiding over de installatie en de verschillende manieren om scripts uit te voeren, worden de basisbegrippen zoals declaraties, variabelen en control flow structures besproken.

Data Structuren en Functies

Ook wordt aandacht besteed aan de collection structures, zoals Lists, Tuples en Dictionaries. Vervolgens wordt de focus gericht op het gebruik van functies met de verschillende methoden van parameter passing, zoals by value en by reference. Ook de scope van variabelen en lambda functies worden hierbij besproken.

Modules en Packages

Vervolgens wordt aandacht besteed aan de opdeling van Python software in modules en komt het gebruik van namespaces en packages aan de orde. Comprehensions in Python en functioneel programmeren komen aan de orde evenals de afhandeling van fouten in scripts met behulp van exception handling.

File en Database Access

Vervolgens staat de functionaliteit van diverse Python library functies voor het benaderen van files op het programma en wordt aandacht besteed aan database access met de Python Database API.

Classes en Objects

Ook object georiënteerd programmeren met classes en objects wordt behandeld. In dit opzicht worden concepten als properties, constructors en encapsulation belicht.

Python Libraries

Tenslotte wordt als de tijd het toelaat optioneel aandacht besteed aan verschillende libraries voor Reguliere Expressies, unit testing en date and time. De verdieping op deze cursus is de cursus [Advanced Python Programmeren](#).

Modules Python Programmeren

Module 1 : Python Intro	Module 2 : Variables and Types	Module 3 : Data Structures
What is Python? Python Features Getting Started Setting up PATH Environment Variables Running Python Interactive Mode Script Mode Identifiers Reserved Words Lines and Indentation Multi Line Statements Quotes	Variables Data Types Python Numbers Numerical Types Number Type Conversions Conversion Functions Built-in Number Functions Python Strings String Operations String Formatting Triple Quotes Raw and Unicode Strings Built-in String Functions	Sequences and Lists Accessing and Updating Lists Multidimensional Lists List Operations List Functions and Methods Tuples Accessing Values in Tuples Tuple Functions Bytes and Byte Arrays Sets and Dictionaries Accessing Values in Dictionaries Properties of Dictionary Keys Dictionary Methods
Module 4 : Control Flow	Module 5 : Functions	Module 6 : Modules
Control Flow Constructs if Statement else Statement elif Statement while Loop for Loop break Statement continue Statement Loop with else Combination pass Statement Python Operators Operator Precedence	Function Syntax Calling Functions Pass by Value Pass by Reference Overwriting References Function Arguments Keyword Arguments Default Arguments Variable Length Arguments Lambda Functions return Statement Scope of Variables	import Statement from...import Statement Locating Modules Creating and Using Modules dir Function Python Packages Explicit Import Modules Implicit Import Modules Namespaces and Scoping globals and locals Functions reload Function Test Harness
Module 7 : Comprehensions	Module 8 : Exceptions	Module 9 : Python IO
Functional Programming Map and Filter Reduce and Lambda List Comprehensions Filtered List Comprehension Syntactic Sugar Dictionary Construction with Zip Dictionary Comprehension Dictionary from Keys Set Comprehension	Unexpected Errors Typed Exception Handling Exception Handling with Else except Clause Multiple Exceptions Standard Exceptions try-finally Clause Exception Arguments Raising Exceptions Example raising Exceptions User Defined Exceptions	Input and Output IO Module Opening Files File Open Modes Reading and Writing Binary Files Reading and Writing Text Files File Positions Renaming and Deleting Files Directory Methods Creating Directories
Module 10 : Database Access	Module 11 : Python Classes	Module 12 : Python Libraries
Python DB API Using with Inserting Data Prepared Statements Last inserted row id Retrieving Data Fetching Rows Parameterized Queries Transactions	Object Orientation Creating Classes Class Members Creating and Using Objects Accessing Attributes Property Syntax Built-in Class Attributes Constructors and Destructors Encapsulation	Regular Expressions match Function Matching versus Searching Search and Replace Unit Testing Unit Test Example Date and Time Handling Time Tuple Calendar Functions