

Python-Kurs: Objektorientierte Skriptsprache - Live-Online-Training

Ziele - Ihr Nutzen

Der Python-Kurs schafft für Sie eine fundierte Basis für die Programmierung mit Python. Dazu gehört insbesondere der Umgang mit komplexen Datentypen und die objektorientierte Programmierung mit Python.

Darüber hinaus werden die wichtigsten Python-Bibliotheken der Standardbibliothek (z.B. os, sys, pickle, json, datetime, tkinter, sqlite3) und von Drittanbietern (z.B. pytest, requests, numpy, pandas, Flask, plotly) in Theorie und Praxis vorgestellt.

Durch eine große Zahl praktischer Übungen bekommen Sie ein Gefühl für die Sprache und das effektive Entwickeln und Testen von Programmen.

Der Python-Kurs vermittelt - vereinfachend gesprochen - mehr als die Grundlagen zu allen Bereichen, die für die professionelle Entwicklung in Python notwendig sind. Im Kurs wird Python 3 praktiziert. Abweichungen zu Python 2 werden an den entsprechenden Stellen kurz erklärt.

Teilnehmer

Der Python-Kurs richtet sich an Softwareentwickler mit Programmiererfahrung.

Voraussetzungen

Programmierkenntnisse; Grundkenntnisse einer objektorientierten Sprache (z.B. C++, C# oder Java) sind für die Teilnahme am Python-Kurs von Vorteil.

Live Online Training

19.01. – 21.01.2026 1.950,00 € 3 Tage

* Preis je Teilnehmer, in Euro zzgl. USt.

Anmeldecode: L-PYTHON

Präsenz-Training - Deutsch

Dauer

3 Tage

Präsenz-Training - Englisch

Dauer

3 Tage

Python-Kurs: Objektorientierte Skriptsprache - Live-Online-Training

Inhalt

Grundlagen in Python

- Historie, Motivation, grundlegende Konzepte
- Programmaufbau
- Datentypen

- Operatoren
- Ein- und Ausgabe
- Kontrollstrukturen
- Sequenzielle Datentypen
- Funktionen
- Funktionale Programmierung
- Objektorientierte Programmierung
- Vererbung und Mehrfachvererbung
- Operatorüberladung
- Exceptions

Python Ökosystem

- Definition eigener Module
- Module der Standardbibliothek (Kurzübersicht zu os, sys, pickle, json, math, datetime, re)
- Module von Drittanbietern (Virtual Environments, Kurzübersicht zu requests, numpy, pandas, Flask, plotly, pytest)