

Kurs: Python-Einführung

© T. Birnthaler, OSTC GmbH (www.ostc.de)

(Version 1.2 vom 27.2.2017)

Dauer: 4-5 Tage

Motivation: Python gehört zu den wichtigsten Programmiersprachen der Welt und gewinnt zunehmend Anhänger. Google, Yahoo!, CERN, NASA und YouTube nutzen Python, weil die Sprache einfach, übersichtlich und gleichzeitig performant ist.

Python ist eine ideale Programmiersprache für Programmieranfänger, weil die Syntax besonders einfach und leicht verständlich ist. Ebenso ist Python eine ideale Programmiersprache für Programmierprofis, weil sie alle Eigenschaften einer modernen Programmiersprache besitzt, eine große Standardbibliothek zur Verfügung steht, sowie portabel und performant ist.

Ziel: Mittels vieler praktischer Übungen lernen Sie die Elemente von Python gründlich kennen und erstellen einfache sowie auch komplexere Programme (Skripte), die gründlich ausgetestet werden. Der Kurs legt viel Wert auf sorgfältig und klar geschriebene Programme und gibt auch Tipps, wie man übersichtlichen Programmcode formuliert.

Hinweis: Der Kurs ist für Linux/UNIX- und für MS-Windows-Systeme geeignet, als Programmierumgebung wird die Kommandozeile und ein beliebiger Editor verwendet.

Inhalt:

- Entstehung und Philosophie von Python
 - Unterschied Python 2.x und Python 3.x
- Interaktive Python-Shell "IDLE" und Python-IDE "Eric"
 - Hilfe zu Python
- Python-Installation
- Python-Skript: Aufbau und Aufruf

- Syntax und Einrückung
 - Kommentare
 - Dokumentation per "DocString" (pydoc)
- Eingebaute Datentypen
 - NoneType, Boolean, Ganzzahl, Fließkommazahl, Komplexe Zahl
 - Sequenz: String, Tupel, Liste
 - Collection: Dictionary, Set, FrozenSet
- Referenzen
 - Flaches und tiefes Kopieren
- Kontrollstrukturen:
 - Verzweigung
 - Schleife
 - Vorzeitiger Abbruch
- Standard-Ein/Ausgabe
 - Formatierte Ausgabe
- Dateizugriff
- Funktionen
 - Parameter (Werteübergabe)
 - Rückgabewert
 - Lokale und globale Variable
 - Rekursive Funktion
- Exception Handling
- Module und Pakete
- Standardbibliothek (häufig verwendete Module)
 - Fest eingebaute Funktionen (builtins)

- Zeichenketten-Funktionen (string)
- Betriebssystem-Schnittstelle (os, os.path, shutil)
- Systemspezifische Funktionen (sys)
- Reguläre Ausdrücke (re)
- Zufallszahlen (random)
- Mathematische Funktionen (math)
- Datum- und Zeitfunktionen (datetime)
- Zeitfunktionen (time)
- Objektorientierte Programmierung (OOP)
 - Klasse, Objekt, Attribut und Methode
 - Konstruktor, Destruktor
 - Klassenattribut und -methode
 - Kapselung
 - Vererbung
 - Polymorphie
 - Überladen von Operatoren und Methoden

Schulungsunterlagen:

- Skript
- Lösungsblätter zu allen Übungen

Voraussetzungen:

- Muss: Allgemeine Computer-Kenntnisse (Tastatur, Maus, Grafische Oberfläche, Editor)
- Muss: Kenntnis einer beliebigen Programmiersprache
- Optional: Kenntnis einer Integrierten Entwicklungsumgebung (IDE)
- Optional: Linux/UNIX-Kenntnisse

Abgrenzung:

- Objektorientierung wird nur rudimentär behandelt (extra Kurs)
- Web-Programmierung wird nicht behandelt (extra Kurs)
- GUI-Programmierung wird nicht behandelt (extra Kurs)
- Datenbank-Programmierung wird nicht behandelt (extra Kurs)