

Python für Einsteiger

Einführung in die Programmierung

Python zählt derzeit zu den beliebtesten und zugänglichsten Programmiersprachen. In diesem Python-Einsteigerkurs nehmen wir uns ausreichend Zeit, um die Grundkonzepte der Programmierung mit Python mithilfe einfacher Beispiele zu erarbeiten.

So erarbeiten Sie sich das erforderliche Grundwissen, das Sie später in Bereichen wie Datenanalyse, künstliche Intelligenz oder IT-Automatisierung einsetzen und vertiefen können.

Zudem erwerben Sie dadurch die Fähigkeit zur besseren Kommunikation in Projekten, bei denen es um Programmierung geht.

Bitte beachten Sie, dass dieser Kurs speziell für Anfänger und Programmier-Neulinge konzipiert ist! Falls Sie bereits Vorkenntnisse in anderen Programmiersprachen besitzen, empfehlen wir Ihnen unseren 3 tägigen Kurs Python für Programmierer – Grundlagen für den schnellen Umstieg.

Kursinhalt

- Programmiersprachen und -paradigmen: Einordnung von Python
- Variablen und Datentypen
- Operatoren und deren Anwendung
- Kontrollstrukturen (Verzweigungen und Schleifen)
- Built-In Funktionen
- Selbstgeschriebene Funktionen
- Best Practices
- Grundlagen Debugging
- Fehlerbehandlung
- Kommunikation mit REST-APIs
- Einführung in die Objektorientierung

Zielgruppe

Dieser Kurs richtet sich an alle, die die Grundlagen der Programmierung erlernen möchten, um selbst aktiv zu programmieren, oder lediglich ein besseres Verständnis für bestehenden Code aufbauen möchten.

Voraussetzungen

Teilnehmer benötigen für diesen Kurs keine Programmierkenntnisse. Eine Affinität zu logischen Fragestellungen und die Fähigkeit, sich zügig in neue technische Themenfelder einzuarbeiten, setzen wir voraus.

Kursziel

Sie erlernen die Grundlagen der Programmierung mit Python. Sie verstehen Variablen und Datentypen sowie Operatoren, Kontrollstrukturen und Funktionen. Sie wenden Best Practices an. Sie beherrschen Debugging und Fehlerbehandlung, nutzen einfache REST-APIs und ordnen Objektorientierung ein.

Dieser Kurs im Web



Alle tagesaktuellen Informationen und Möglichkeiten zur Bestellung finden Sie unter dem folgenden Link: www.experteach.ch/go/PYFE

Vormerkung

Sie können auf unserer Website einen Platz kostenlos und unverbindlich für 7 Tage reservieren. Dies geht auch telefonisch unter 06074 4868-0.

Garantierte Kurstermine

Für Ihre Planungssicherheit bieten wir stets eine große Auswahl garantierter Kurstermine an.

Ihr Kurs maßgeschneidert

Diesen Kurs können wir für Ihr Projekt exakt an Ihre Anforderungen anpassen.

Premium Print Paket



Zu diesem Kurs können sie optional das hochwertige Premium Print Paket zum Preis von € 225,- (zzgl. MwSt) erwerben.

| Training | Preise zzgl. MwSt. |
|-------------------------------|-----------------------------|
| Termine in Deutschland | 5 Tage CHF 2.855,- |
| Termine in Österreich | 5 Tage CHF 2.855,- |
| Termine in der Schweiz | 5 Tage CHF 3.290,- |
| Online Training | 5 Tage CHF 2.855,- |
| Termin/Kursort | Kurssprache Deutsch |
| 08.06.-12.06.26 Online | 28.09.-02.10.26 Düsseldorf |
| 13.07.-17.07.26 Frankfurt | 28.09.-02.10.26 Online |
| 13.07.-17.07.26 Online | 02.11.-06.11.26 Online |
| 17.08.-21.08.26 Berlin | 02.11.-06.11.26 Wien |
| 17.08.-21.08.26 Hamburg | 07.12.-11.12.26 München |
| 17.08.-21.08.26 Online | 07.12.-11.12.26 Online |
| 17.08.-21.08.26 Online | 07.12.-11.12.26 Zürich |

Stand 02.06.2026



EXPERTeach



Inhaltsverzeichnis

Python für Einsteiger – Einführung in die Programmierung

| | | |
|---|--|---|
| 1 Grundlagen der Programmierung | 6.4 Verzweigung in Python: if | 12.1.3 Das OS – Modul |
| 1.1 Was bedeutet programmieren? | | 12.1.4 Das Shutil – Modul |
| 1.2 Algorithmus | 7 Namensräume und String- sowie Listenmethoden | 12.1.5 Import |
| 1.2.1 Darstellung von Algorithmen | 7.1 Namensräume | 12.1.6 Sockets |
| 1.2.2 Pseudocode | 7.2 String Methoden | 12.1.7 Regular Expressions |
| 1.3 Einordnung von Programmiersprachen | 7.3 Listen und Arrays | 12.2 Pip und PyPI |
| 1.4 Compiler | 7.3.1 Liste | 12.3 Projektdokumentation |
| 1.5 Interpreter | 7.3.2 Liste | 12.3.1 Dokumentation in Python – Sphinx |
| 1.6 Grundelemente einer Programmiersprache | 7.3.3 Tupel | 12.3.2 Dokumentation in Python – Sphinx |
| | 7.3.4 Sequenz | 12.3.3 Dokumentation in Python – Sphinx |
| | 7.3.5 Slicing | 12.4 HTTP Abfragen in Python – requests |
| 2 Das Python Projekt | 7.4 Komplexe Datentypen in Python | 12.5 JSON |
| 2.1 Was ist Python? | | 12.6 XML |
| 2.2 Download, Installation und Lieferumfang | 8 Schleifen / Wiederholte Codeausführung | 12.7 Jupyter Notebooks |
| 2.3 Python-Shell | 8.1 Schleifen | 12.8 Wissenschaftliche Berechnungen in Python mit Numpy und Pandas |
| 2.4 IDLE | 8.1.1 While | 12.9 Datenvisualisierung in Python mit Matplotlib |
| | 8.1.2 While-Schleife | 12.10 Weitere Anwendungsmöglichkeiten |
| 3 Zahlen, Zeichenfolgen und Zuweisungen | 8.2 For Schleife Konzept | |
| 3.1 Zahlen | 8.3 For Schleife | 13 Schöner Code |
| 3.2 Zeichenfolgen | 8.3.1 Iterationen II | 13.1 Einfacher Typehint |
| 3.3 Zuweisungen | 8.3.2 For-Details | 13.2 Typehints bei Containern |
| 3.3.1 Verwendung von Variablen | 8.3.3 Comprehensions | 13.3 Typehint in der Funktionssignatur |
| | | 13.4 Documentation mit Docstrings |
| 4 Visual Studio Code und Jupyter | 9 Routinen und Funktionen | 13.5 Übersicht schöner Code |
| 4.1 Was ist Visual Studio Code? | 9.1 Routine – Prozeduren und Funktionen | |
| 4.1.1 Visual Studio Code vs. Visual Studio | 9.2 Funktionen II – Eigene Funktionen | 14 Objektorientierte Programmierung |
| 4.1.2 Installation | 9.3 Verschachtelung von Funktionen | 14.1 Grundprinzipien der OOP |
| 4.1.3 Die Oberfläche von VS Code | 9.4 Namensräume: Globale und lokale Variablen | 14.1.1 Module und Systeme |
| 4.1.4 Die Oberfläche von VS Code - Activity Bar und Sidebar | 9.5 Arten von Argumenten | 14.1.2 Strukturelle Elemente objektorientierter Software |
| 4.1.5 Das Search Tool | 9.6 Anonyme Funktionen | 14.1.3 Beziehungen zwischen Objekten |
| 4.1.6 Das Python-Extension Pack für VS Code | | 14.1.4 Design Pattern (Entwurfsmuster) |
| 4.1.7 Neue Dateien in VS Code anlegen | 10 Ausnahmebehandlung und Dateihandling | 14.2 Objektorientierung und Python |
| 4.1.8 Mit einzelnen Codedateien in Visual Studio Code arbeiten | 10.1 Exceptions in Python | 14.2.1 Basisbeispiel |
| 4.1.9 Visual Studio Code einen Ordner hinzufügen | 10.2 Hierarchie der Builtin Exceptions (Ausschnitt) | 14.2.2 Sichtbarkeiten |
| 4.1.10 Debugging in Visual Studio Code | 10.3 Exception Beispiel | 14.2.3 Destruktor |
| 4.2 Jupyter Notebooks | 10.4 Fehlermeldung Erzeugen | 14.2.4 Statische Elemente |
| 4.2.1 Interaktive Code-Zellen | 10.5 Lesen einer einfachen Text Datei | 14.2.5 Mehrfachvererbung |
| 4.2.2 Exporte in andere Formate und Hilfen | 10.6 Dateien lesen mit dem Kontext Manager | |
| 4.3 Jupyter Notebooks im VSC | 10.7 Dateien erstellen | 15 Python – Cheatsheets |
| | | 15.1 Methoden und Operatoren |
| 5 Built-in Funktionen | 11 Datentypen - Sammlungen | 15.2 Stringmethoden |
| 5.1 Funktionen | 11.1 Sammlungen | 15.3 Listenmethoden |
| 5.2 Eingebaute Funktionen | 11.2 Map | 15.4 Dictionaries |
| 5.3 Anwendung Eingebauter Funktionen | 11.3 Eine Menge – Set | 15.5 Schleifenkonstruktionen |
| 5.3.1 Die Hilfe-Funktion | 11.4 Dictionaries | 15.6 Die Try – Except Konstruktion |
| | 11.5 Dictionaries – Methoden | 15.7 Ein- und Ausgabemöglichkeiten |
| 6 Verzweigungen/Bedingte Codeausführung | 12 Python – Pakete und Anwendungen | 15.8 Modul-Importe |
| 6.1 Grundlagen der Verzweigung | 12.1 Die Standardbibliothek | 15.9 Editoren – Visual Studio Code – VSC |
| 6.2 Komplexe Verzweigungen | 12.1.1 Import | |
| 6.3 Wahrheitswerte und Vergleiche | 12.1.2 Das SYS – Modul | |

