|  |
| --- |
| IU Internationale Hochschule |
| Einführung in das Programmieren mit Python |
| Kurscode |

# Übergeordnete Lernziele

Python ist innerhalb kurzer Zeit zu einer der weltweit beliebtesten und gängigsten Programmiersprachen für die Softwareentwicklung geworden. Vor diesem Hintergrund soll Ihnen der Kurs **Einführung in das Programmieren mit Python** sowohl die Gründe für die wachsende Verbreitung dieser Programmiersprache als auch ein umfassendes Verständnis ihrer Vor- und Nachteile vermitteln. Dabei erfahren Sie nicht nur, wie Sie Python herunterladen und installieren, sondern lernen auch anhand von Beispielen verschiedene Einsatzbereiche kennen, die mit den Themenschwerpunkten Ihres Studiengangs korrespondieren.

Nach Abschluss des Kurses werden Sie mit grundlegenden Python-Konzepten wie Variablen, Konstanten, Strings und Collections sowie Datei-Eingaben und -Ausgaben vertraut sein. Darüber hinaus sind Sie in der Lage, verschiedene Arten von Python-Anweisungen zu erläutern und zu verwenden, darunter Zuweisungen, Ausdrucksanweisungen und Funktionsanweisungen. Zudem können Sie die Nutzung von bedingten Ausdrücken, Schleifen und Iteratoren erklären und am Beispiel demonstrieren.

Ergänzend werden im weiteren Verlauf des Kurses die Einsatzmöglichkeiten und Zwecke von Funktionen in Python sowie die für die Erstellung eigener Funktionen geltenden Syntaxregeln erörtert. Auf diese Weise erlangen Sie die nötigen Kenntnisse, um Funktionen mit verschiedenen Parametern und Rückgabetypen zu implementieren und den Sichtbarkeitsbereich von Variablen und Funktionen zu beschreiben.

Des Weiteren wird es Ihnen möglich sein, die Verwendung von Fehlerbehandlungsmethoden und Logdateien zu erklären und zu demonstrieren sowie die wichtigsten Gründe für die Nutzung von Modulen, Namensräumen und verschiedenen datenwissenschaftlichen Paketen zu nennen.

# Lektion 1 – Einführung in Python

Lernziele

Nach Abschluss dieser Lektion werden Sie in der Lage sein, …

… die Gründe für die wachsende Beliebtheit von Python in der Softwareentwicklung zu nennen.

… die Stärken und Schwächen von Python zu erläutern.

… Python herunterzuladen und zu installieren.

… die verschiedenen Komponenten einer Python-Entwicklungsumgebung zu beschreiben und einzusetzen.

# 1. Einführung in Python

### Aus der Praxis

Mehmet und Lea sind seit Jahren befreundet und unterhalten sich heute so angeregt wie lange nicht. Obwohl die beiden in der Vergangenheit bereits verschiedene Geschäftsideen diskutiert haben, schien ihnen bislang keine davon überzeugend genug, um sie in die Tat umzusetzen. Doch dieses Mal ist alles anders. Als sie beim gemeinsamen Mittagessen über ihren neuesten Einfall sprechen, herrscht Aufbruchsstimmung. Denn Mehmet und Lea haben die feste Absicht, diese Idee zu verwirklichen.

Worum genau geht es dabei? Da Mehmet und Lea beide im Fußballteam ihrer Universität gespielt haben, kreisen viele ihrer Geschäftsideen aus den letzten Jahren um das Thema „Fußball“. Dabei hat sich keiner ihrer diesbezüglichen Einfälle als marktfähig und profitabel erwiesen – bis jetzt! Mehmet möchte aus Leas Statistikkenntnissen und seinem eigenen Know-how im Bereich Softwareentwicklung Kapital schlagen, um eine Softwareanwendung zur Spielerentwicklung zu erstellen. Ein solches Programm würde Spieler- und Trainer:innen dabei helfen, die erfassten Daten auf Trends und Tendenzen hin zu analysieren und individuelle Verbesserungsschwerpunkte zu identifizieren.

Die beiden haben keinen Zweifel daran, dass diese Idee als Grundlage für ein erfolgreiches Start-up-Unternehmen taugt, weil die fertige Anwendung für viele Aktive interessant und damit äußerst einträglich sein wird. Doch damit sich die Software möglichst breit vermarkten lässt, sollte sie problemlos auf verschiedenen Plattformen laufen. Außerdem stellt sie hohe Anforderungen in puncto Rechenleistung, da sie umfangreiche Datenanalysen und komplexe statistische Berechnungen durchführen muss. Um sicherzustellen, dass ihre Anwendung diesen Ansprüchen gerecht wird und es keine Probleme mit der Funktionsfähigkeit und Umsetzung gibt, halten Mehmet und Lea verschiedene Fragen fest, die vor dem Start ihres Entwicklungsprojekts beantwortet werden müssen:

**Programmiersprache**

Python ist eine weitverbreitete interpretierte Programmiersprache, die im Lehrbetrieb sowie für datenwissen­schaftliche Projekte und zahlreiche andere Zwecke eingesetzt wird.

* Welche **Programmiersprache** eignet sich am besten für die Erstellung der Anwendung und aus welchen Gründen?
* Wie kann eine Anwendung plattformübergreifend lauffähig sein?
* Was genau bedeutet „Plattform“ in diesem Zusammenhang eigentlich?
* Sind manche Programmiersprachen besser für datenwissenschaftliche und mathematische Anwendungen geeignet als andere? Falls ja, welche Sprachen weisen hier besondere Stärken auf und sollten deshalb von Mehmet und Lea in Betracht gezogen werden?
* Wo können die beiden nach der Wahl einer geeigneten Programmiersprache die dazugehörigen Tools herunterladen? Und wie sollten sie ihr Projekt angehen?

## 1.1 Warum Python?

**Softwareentwicklung**

Dieser Begriff bezeichnet sämtliche Entwurfs-, Design-, Programmier- und Testprozesse zur Erstellung neuer Softwareanwendungen oder -komponenten.

Herzlichen Glückwunsch zu Ihrem Einstieg in die **Softwareentwicklung** mit Python! Wenn Sie eine Karriere in diesem Bereich anstreben, stehen Ihnen hervorragende Möglichkeiten offen. Nicht umsonst wird eine Laufbahn in der Softwareentwicklung seit zehn Jahren durchgängig in den von Wirtschaftsfachleuten und Marktforschungsunternehmen erstellten Rankings der besten Karriereoptionen aufgeführt. Im *US News & World Report 2019* belegte die Sparte sogar den Spitzenplatz, gemessen an der Zahl der offenen Stellen, am Gehalt, an der Beschäftigungsquote und am prognostizierten Wachstum der Branche (US News & World Report 2019).

Doch auch wenn Sie sich nicht für eine Karriere in der Informatik interessieren, lohnt die Beschäftigung mit dem Thema. Erfolgreiche Branchengrößen wie Bill Gates und Steve Jobs haben immer wieder betont, dass Angestellte mit Kenntnissen im Bereich Softwareentwicklung für ihre Unternehmen von großem Nutzen sind – unabhängig davon, welchen Karriereweg sie einschlagen.

In jedem Fall ist Python ein guter Einstiegspunkt, vor allem dann, wenn Sie bisher noch keine Erfahrungen im Programmieren gesammelt haben. Die Sprache wird von vielen Bildungsinstitutionen im Rahmen der Ausbildung genutzt und hat im Jahr 2014 Java als gängigste Programmiersprache für einführende Informatikkurse an US-Universitäten abgelöst (Guo, 2014).



Weitere Belege für diesen Aufwärtstrend liefert der Popularity of Programming Languages Index (PYPL-Index), dessen Programmiersprachenranking auf der Zahl der Google-Suchen nach einschlägigen Tutorials basiert. Er ist ein guter Indikator für die Zahl der Personen, die die verschiedenen Programmiersprachen erlernen möchten. Im August 2019 belegte Python hier den ersten Platz, vor Java, JavaScript und C# (PYPL-Index 2019). Zudem zeigt der Vergleich der jährlichen PYPL-Rankings, dass der Anteil von Python zwischen August 2018 und August 2019 um 4,5 Prozent gestiegen ist und Python damit als einzige Programmiersprache unter den Top Ten einen Zuwachs verzeichnete. Das deutet darauf hin, dass immer mehr Menschen Python erlernen möchten und die Programmiersprache ihren Konkurrenten aus der Spitzengruppe weitere Marktanteile abjagt.

Allerdings sollten Sie aus diesen Zahlen nicht den Schluss ziehen, dass Python nur zu Lernzwecken eingesetzt wird und sich vor allem für einführende Informatikkurse eignet. Denn Python ist auch in der Praxis äußerst beliebt und zählt zu den gängigsten Programmiersprachen der Softwareentwicklungsbranche. Das belegt beispielsweise der TIOBE Programming Community Index – ein weltweites Ranking der beliebtesten Sprachen für die Softwareentwicklung, in das unter anderem Angaben zur Zahl der in der Sprache ausgebildeten Fachleute, zum Angebot an einschlägigen Einführungskursen und zur Anzahl kommerzieller Anbieter von Softwarelösungen für die jeweilige Sprache einfließen (TIOBE 2019). Hier konnte Python in den letzten Jahren konstant zulegen und kletterte schließlich im September 2018 vom vierten auf den dritten Platz, hinter Java und C. Außerdem zeigt sich im Jahresvergleich, dass Python im August 2019 weiterhin den dritten Platz (hinter Java und C) belegte, mit einem um 3,03 Prozent gestiegenen Anteil.

Damit verzeichnete die Programmiersprache in der Gruppe der Top 20 das höchste Wachstum, gefolgt von Groovy (auf Platz 13, mit einer Wachstumsrate von 1,04 Prozent).



Wie ist das zu erklären? Weshalb ist Python eine so hervorragende Option für alle, die das Programmieren lernen möchten? Auf diese Frage gibt es viele Antworten, von denen die folgenden besonders häufig genannt werden:

* Python und viele dazugehörige Tools, Module und Bibliotheken sind kostenlos erhältlich.
* Verglichen mit vielen anderen Programmiersprachen ist die **Syntax** von Python knapper und präziser. Dadurch lassen sich mehr Funktionen mit weniger Zeilen realisieren, was das Programmieren spürbar vereinfacht.

**Syntax**

Die Regeln und Symbole, die bei der Erstellung von Programmen in einer bestimmten Programmiersprache zur Anwendung kommen, bilden die Syntax dieser Sprache.

* Python-Code wurde gezielt so konzipiert, dass er der englischen Alltagssprache ähnelt. Daher lässt er sich leichter lesen als Programme in vielen anderen Sprachen.
* Die umtriebige Python-Community stellt vielfältige Ressourcen für angehende und erfahrene Entwickler:innen bereit.
* Python unterstützt prozedurale, objektorientierte und funktionale Programmierparagdigmen.
* Da Python eine **interpretierte Programmiersprache** ist, können Neuentwicklungen sofort getestet werden, ohne dass die betreffende Anwendung zuerst kompiliert werden muss.

**Interpretierte Programmiersprache**

Python wird als interpretierte Programmiersprache bezeichnet, weil Python-Code direkt ausgeführt werden kann und nicht erst durch einen Compiler in Befehle einer Maschinensprache übersetzt werden muss.

Angesichts der zahlreichen Vorteile, die Python für den Einstieg in das Programmieren bietet, lässt sich leicht nachvollziehen, warum diese Programmiersprache häufig an Universitäten für einführende Informatikkurse genutzt wird und den Spitzenplatz im PYPL-Index belegt. Doch warum ist Python auch bei Fachleuten und kommerziellen Anbietern so beliebt? Hier sind nicht nur die oben aufgeführten Gründe, sondern auch die im Folgenden genannten Punkte ausschlaggebend:

* Python-Code lässt sich nicht nur relativ einfach schreiben und lesen, sondern ist außerdem extrem robust und leistungsstark.
* Als benutzerfreundliche, kompakte Programmiersprache ist Python eine naheliegende Option für die schnelle Erstellung funktionsfähiger Prototypen.
* Python ist die Sprache der Wahl im Bereich **Data Science** und kommt bei vielen Projekten rund um die künstliche Intelligenz zum Einsatz.

**Data Science (Datenwissenschaften)**

In diesem akademischen Feld werden wissenschaftliche Ansätze für die Organisation, Darstellung und Analyse von Daten mithilfe von Algorithmen und Rechenkapazitäten erarbeitet.

* Wegen der weiten Verbreitung der Sprache in den Bereichen Data Science und KI sind zahlreiche Python-Bibliotheken für mathematische und statistische Anwendungen sowie Deep Learning und andere Varianten des maschinellen Lernens verfügbar.
* Es gibt außergewöhnlich leistungsstarke Bibliotheken und Frameworks für die Entwicklung von Webanwendungen in Python (wie beispielsweise Django und Flask).
* Da sich Python sowohl in der akademischen Lehre als auch in der beruflichen Praxis einer wachsenden Beliebtheit erfreut, steigt die Zahl gut bezahlter Stellen für Python-Entwickler:innen. Somit lohnt sich das Erlernen der Programmiersprache auch unter finanziellen Aspekten!

1.2 Download und Installation von Python

Nachdem wir nun bestens über die zahlreichen Vorteile von Python für den Einstieg in das Programmieren und die Entwicklung kommerzieller Lösungen informiert sind, können wir loslegen! Grundsätzlich ist die neueste Version der Programmiersprache mitsamt diversen nützlichen Ressourcen auf der Python-Website zum Download verfügbar. Doch im Rahmen dieser Lehrveranstaltung gehen wir einen anderen Weg, weil wir Python im Studiengang Data Science immer wieder für datenwissenschaftliche Anwendungen nutzen werden. Aus diesem Grund müssen wir nicht nur Python, sondern auch einige Python-Bibliotheken für das maschinelle Lernen und datenwissenschaftliche Anwendungen installieren.

Eine der gängigsten Python-Distributionen für diesen Bereich ist Anaconda. In ihr sind unter anderem drei wichtige Tools enthalten, die im Studiengang Data Science durchgängig Verwendung finden:

* NumPy: eine äußerst effiziente Bibliothek für die Handhabung großer Datensätze und mehrdimensionaler Arrays
* SciPy: eine für wissenschaftliche Berechnungen konzipierte Bibliothek mit Funktionen für lineare Algebra, Interpolation und die Signal- und Bildverarbeitung
* Pandas: eine Bibliothek mit Funktionen für die Verwaltung und Analyse von numerischen Tabellen und Zeitreihendaten

Neben Anaconda gibt es weitere Optionen für den datenwissenschaftlichen Einsatz von Python. Eine mögliche Alternative besteht darin, dass Sie zunächst nur Python installieren und dann NumPy, SciPy oder andere gewünschte Bibliotheken hinzufügen. Beispielsweise steht Ihnen mit Python(x,y) eine weitere Bibliothek für wissenschaftliche Projekte und numerische Berechnungen zur Verfügung, während Enthought leistungsstarke Tools für die Datenvisualisierung und -analyse bereitstellt. Abgesehen davon hat die ebenso umfangreiche wie umtriebige Python-Community viele weitere Open-Source-Lösungen entwickelt, die eventuell erwägenswert sind. Für diesen Kurs beschränken wir uns jedoch auf die Installation der Anaconda-Distribution.

Zu diesem Zweck besuchen Sie zunächst die Anaconda-Website und klicken dort auf den Download-Button. Anschließend wählen Sie das Betriebssystem Ihres Geräts (Windows, MacOS oder Linux), wie auf dem nachstehenden Screenshot dargestellt:



Auf dem nächsten Bildschirm (siehe unten) können Sie dann zwischen verschiedenen Versionen von Python wählen. Im August 2019 standen hier Python 3.7 und Python 2.7 zur Verfügung:



Da wir im Rahmen dieses Kurses und für das gesamte Data-Science-Studium Python 3.x nutzen, sollten Sie unbedingt Version 3.7 herunterladen (und nicht Version 2.7). Denn wie jede moderne Software wird auch Python in regelmäßigen Abständen aktualisiert. Und weil sich Version 3 in mancherlei Hinsicht grundlegend von Version 2 unterscheidet, werden viele der Beispiele aus diesem Kurs nicht richtig funktionieren, wenn Sie eine falsche Version verwenden.

Sobald Sie die passende Version von Anaconda heruntergeladen haben, führen Sie das Installationsprogramm aus. Der nun folgende Abschnitt illustriert die Installation auf einem Windows-Gerät, wobei die Installationsprozesse auf MacOS- und Linuxsystemen ähnlich ablaufen. Hier wie dort können Sie den Vorgang starten, indem Sie auf dem Einrichtungsbildschirm auf „Next“ klicken:



Daraufhin wird Ihnen die Lizenzvereinbarung angezeigt, der Sie (selbstverständlich erst nach eingehender Lektüre) zustimmen:



Anschließend müssen Sie sich für eine der beiden Installationsarten entscheiden (Freigabe des Programms nur für das eigene Konto oder Installation für alle Benutzer):





Auf dem nächsten Bildschirm akzeptieren Sie die Standardeinstellung und klicken dann auf „Install“:



Nach dem Ende des Installationsprozesses klicken Sie auf den verbleibenden Bildschirmen auf „Next“. Damit haben Sie die Einrichtung von Anaconda Python erfolgreich abgeschlossen!