IBB BUSINESS AKADEMIE
WEITER MIT BILDUNG

Berufs-begleitend weiterbilden!

Python Programmer

Ein kurzer Überblick

Python ist die Nummer 1 der Programmiersprachen für Machine Learning und Data Science. Ob Web-Applikationen, Webseiten oder APIs: Oft sind grundlegende Programmierkenntnisse dafür bereits ausreichend.

Unser Kooperationspartner StackFuel hat ein Training entwickelt, das die Grundlagen der Programmierung mit Python legt. Sie machen sich mit den wichtigsten Programmierkonzepten wie Variablen, Typen, Funktionen und Methoden vertraut, entwickeln Klassen und Module und führen Unit Tests durch. Darauf aufbauend widmen Sie sich fortgeschrittenen Konzepten der objektorientierten Programmierung wie Vererbung und Komposition. Sie beenden das Training mit einem unternehmensrelevanten Projekt zur Konfiguration eines Passwortmanagers und qualifizieren sich mit erfolgreichem Abschluss für die Jobrolle als Python Programmer.

Eine Weiterbildung in Python ermöglicht Ihnen den optimalen Quereinstieg in die Softwareentwicklung. Know-how in diesem Bereich kann zudem der Grundstein für eine spätere Spezialisierung in weitere Felder auf dem hochattraktiven IT-Arbeitsmarkt sein – beispielsweise Full Stack Development oder Data Security.

Sie haben Interesse an einem AZAV - zertifizierten Angebot? Dann schauen Sie sich einmal unter diesem Link die Angebote an.

Kursinhalte

MODUL 1 – PYTHON BASICS

Kapitel 1 - Data Types

- In diesem Kapitel bewegen Sie sich zum ersten Mal in unserer Programmierumgebung – dem Data Lab – und führen erste Code-Befehle aus.

Kursnummer
BA-3054-B-1

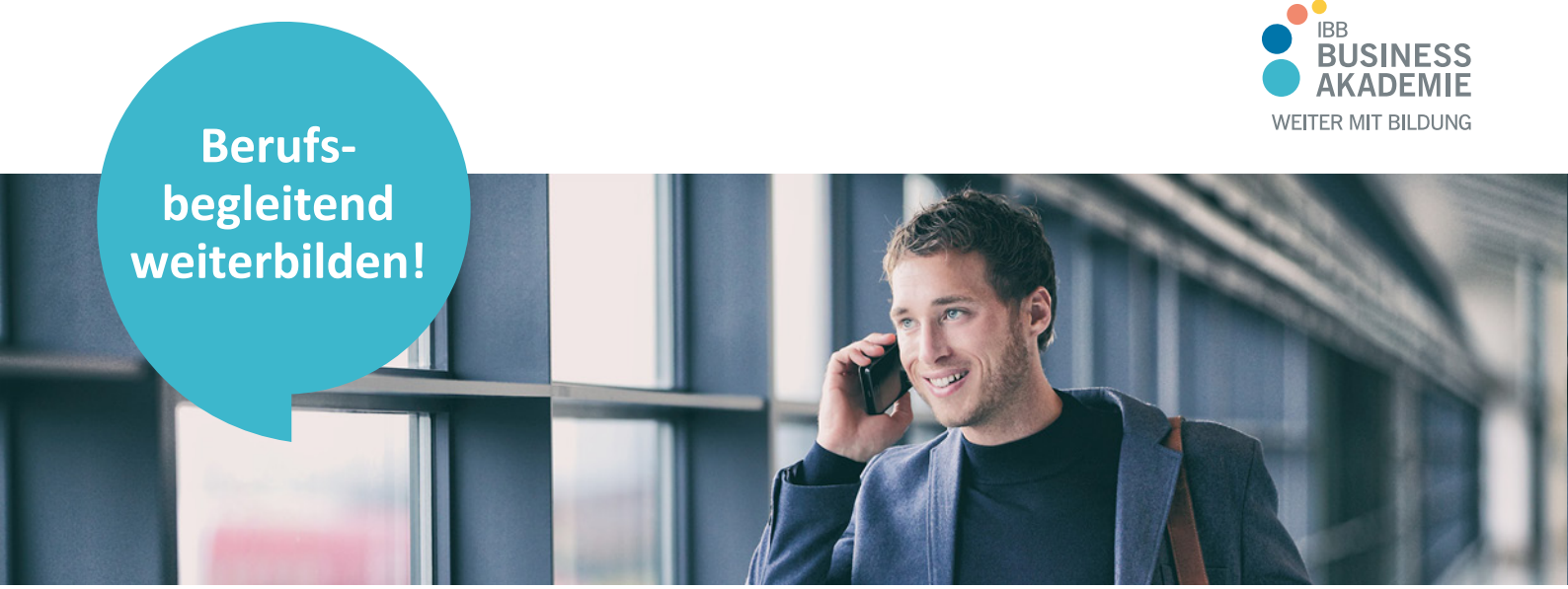
Beratung und Anmeldung
Telefon: 04161 5165-89
E-Mail: akademie@ibb.com

Die nächsten Starttermine

27.05.24 - 15.09.24	08.07.24 - 27.10.24
19.08.24 - 08.12.24	30.09.24 - 19.01.25
11.11.24 - 02.03.25	02.12.24 - 23.03.25

Trainingsdauer
Seminardauer: 4 Monate

Teilnahmegebühr
ab 4.990,00€



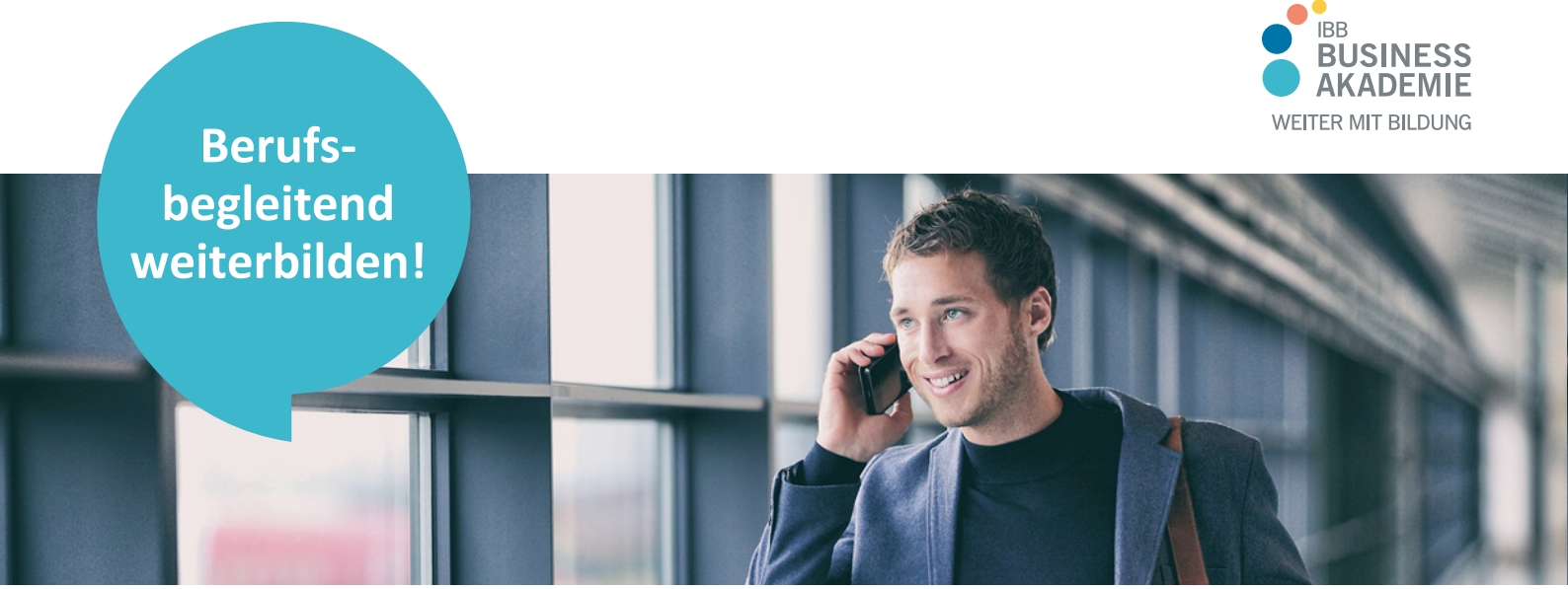
Dabei lernen Sie einleitend, was Daten oder Texte in Python repräsentieren.
Anhand eines Regelwerks, erfahren Sie wie Daten in Variablen erzeugt, zugewiesen und erprobt werden. Im Anschluss behandeln Sie die Auslesung
gängiger Fehlermeldungen und üben, wie Sie sie produktiv im Arbeitsalltag
nutzen. Sie lernen außerdem grundlegende Python-Standardfunktionen wie
z. B. type() oder str() kennen und nutzen diese in Anwendungsbeispielen.
Nachdem Sie das if-Statement kennengelernt hast und damit den Ablauf Ihres Codes an Bedingungen knüpfen können, schließen Sie das Kapitel mit
dem ersten Teil eines zweistündigen Miniprojekts ab: Sie programmieren ein
User Interface, das flexibel auf die Eingaben der User:innen reagiert.

Kapitel 2 - Flow Control

- Im zweiten Kapitel erarbeiten Sie zwei essenzielle Vorgänge, um Ihre Codes noch flexibler zu machen: Sie lernen Listen und for-Schleifen kennen. Listen erlauben mehr Flexibilität im Speichern von Daten und sind Voraussetzung für fortgeschrittenes Programmieren. Sie lernen sie zu erstellen, auszulesen und zielgerichtet zu verändern. Mit Listen erweitern Sie
die Funktionalität Ihrer User Interface aus dem ersten Kapitel und schließen
damit das erste Miniprojekt ab. Im Anschluss arbeiten Sie mit for-Schleifen,
mit denen Sie Ihren Code automatisch mehrmals ausführen können und erreichen so die nächste Stufe des Programmierens.

Kapitel 3 - Functions, Modules and Methods

- Im dritten Kapitel runden Sie Ihre Programmierfähigkeiten in Python ab und erlernen fortgeschrittene Techniken. Dazu gehören Funktionen und Methoden. Sie lernen, selbstständig Funktionen zu definieren und so Ihren Code besser zu strukturieren. Außerdem kombinieren Sie



die einzelnen Programmierelemente wie Bedingungen, Schleifen und Funktionen in Programmen. Sie erarbeiten anhand unterschiedlicher Methoden, wie Sie in richtiger Schreibweise Python-Module importieren und dabei Funktionalität im Ganzen sicherstellen. Verknüpfend daran lernen Sie, wie Sie Daten als einfachen Test einlesen und exportieren Mithilfe eines einstündigen Miniprojekts mit Telefondaten festigen Sie die gelernten Inhalte des Kapitels.

Kapitel 4 - Python Applications

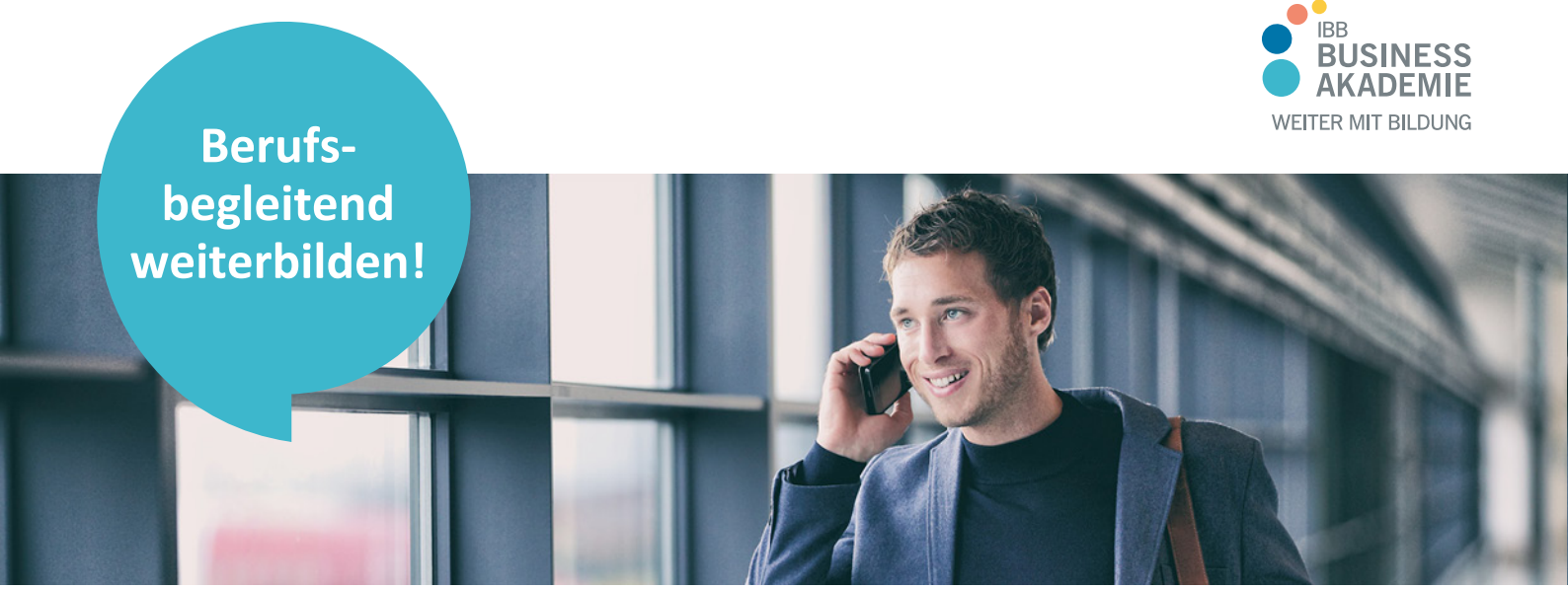
- Im vierten Kapitel werden die Inhalte von Kapitel 1-3 wiederholt und mit weiterführenden Inhalten abgerundet. Sie erfahren, was Dictionaries sind, und wie sie Ihren Code noch effizienter machen. Sie machen dabei Bekanntschaft mit komplexen Datenstrukturen, die Sie für Ihr Abschlussprojekt brauchen. In einem vierstündigen Praxisprojekt zur Erstellung einer komplexen User Interface zur automatisierten Verarbeitung von Kundenanfragen müssen Sie sämtliche Inhalte der bisherigen Kapitel zusammenführen.

MODUL 2 – OBJECT-ORIENTED PROGRAMMING

Kapitel 1: Advanced Python

- Im Einstiegskapitel wiederholen Sie die wichtigsten Inhalte aus dem vorherigen
  Modul, bevor Sie eine Reihe von Konzepten kennenlernen, die Sie ins nächste Level der Python Programmierung katapultieren. Sie betrachten die
  Funktionsdefinition noch einmal tiefergehend und lernen Standardwerte, Typenhinweise und assert-Statements kennen. Funktionen können Sie danach noch besser als Werkzeuge für Ihre Projekte nutzen. Darüber hinaus behandeln Sie die Begriffe List Comprehension und Dictionary Comprehension zur effizienten Erstellung von Listen und Dictionaries. Zum
  Abschluss des Kapitels erfahren Sie, wie Sie Ihren Code anhand von Layout und
  Struktur dem Industriestandard PEP8 anpassen.

Kapitel 2 – OOP Basics



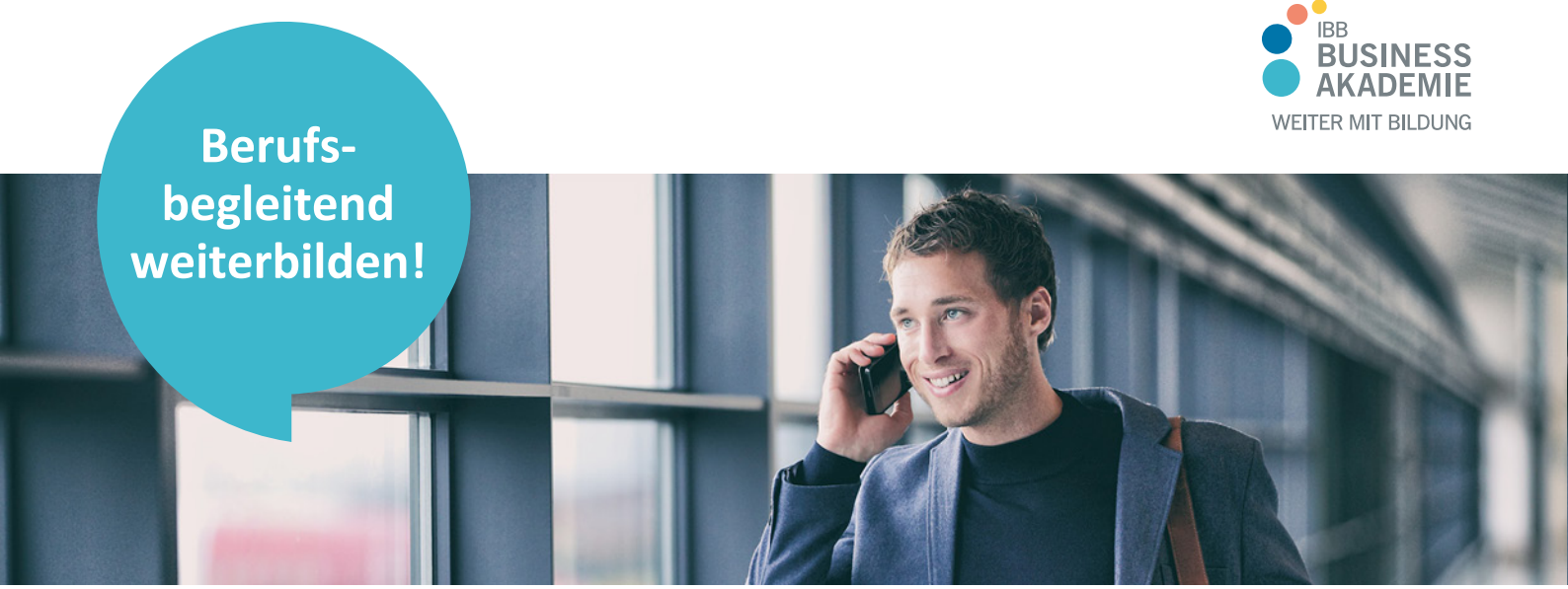
- Im zweiten Kapitel lernen Sie anhand einfacher Beispiele, was OOP ist, welche
  Programmprinzipien darauf aufbauen und welche Schlüsse Sie daraus ableiten können. Im Hauptteil des Kapitels erforschen Sie, wie Klassen und
  Attribute definiert sind und genutzt werden. Dabei untersuchen Sie anhand
  von Beispielen Instanzmethoden sowie deren Nutzung und Definition mit Method Chaining. Sie erfahren, was das self-Keyword ist, sowie Debugging
  von Klassendefinitionen zu unterscheiden. Abschließend testest Sie Ihr bisheriges Fachwissen in einem interaktiven Zwischenprojekt und wiederholen
  die Übungen aus dem Kapitel.

Kapitel 3 – Inheritance and Composition

- Im dritten Kapitel erfahren Sie, was Vererbungen und Kompositionen sind und wie Sie diese Konzepte in Anwendungsfällen nutzen. Neben der einfachen Vererbung lernen Sie auch weiterführende Methoden wie die Vererbungshierarchie und die multiple Vererbung kennen. Dabei üben Sie tiefgehende Methoden zur Vererbung, die bei der Wiederverwendbarkeit von
  Daten von übergeordneten zu untergeordneten Klassen verwendet werden und
  dabei Datenverluste ausgleichen. Zum Abschluss bekommen Sie die wichtigsten
  Best Practices für Unit Tests von uns an die Hand, um Fehler in Ihrem Code zu
  entdecken, bevor Ihre Nutzer:innen sie finden.

Kapitel 4 – Advanced OOP

- Im vierten Kapitel behandeln Sie weiterführende Begriffe der objektorientierten Programmierung, die Sie im Arbeitsalltag begleiten werden. Sie studieren, wie sich Programme und Module unterscheiden und welche Rolle __main__ dabei spielt. Dazu lernen Sie, was Decorators sind und wie Sie Property Decorators optimal nutzen. Sie betrachten



statische und Klassenmethoden und welche speziellen Methoden und Klassenrepräsentationen mit __str_() und _repr_() zum Einsatz kommen können. Darauf aufbauend lernen Sie Darstellungsmöglichkeiten des Operator Overloading sowie weitere wichtige Methoden aus der Python Standard Library kennen und wenden im Anschluss die gelernten Lerninhalte in einem unternehmensrelevanten Zwischenprojekt an.

Kapitel 5 – OOP Applications

- Im fünften Kapitel stellen Sie Ihre Kenntnisse in zwei größeren Projekten unter Beweis, die klassische Anwendungsfälle der objektorientierten Programmierung darstellen. Im ersten Projekt bauen Sie eine eigene Schnittstelle zu einer beliebten Data Science-Bibliothek, die weitere Verwendungsmöglichkeiten für Machine Learning und Daten- oder Textanalysen ermöglicht. Das zweite Projekt behandelt die Programmierung einer eigenen Blockchain, wo Sie die zugrundeliegenden Konzepte näher kennenlernen. Am Ende des Modul 2 sind Sie für die Anwendung von OOP in der Unternehmenswelt gerüstet.

MODUL 3 – FINAL PROJECT

- Im Rahmen des Abschlussprojekts vertiefen Sie die erlernten Inhalte aus den Modulen Python Basics und Object-Oriented Programming und programmieren selbständig einen Passwortmanager. Dafür setzen Sie eine Programmierumgebung auf und füllen Ihre Datei mithilfe eines Terminals mit Datensätzen, die ein Regelwerk schafft, das den Passwortmanager konfiguriert.

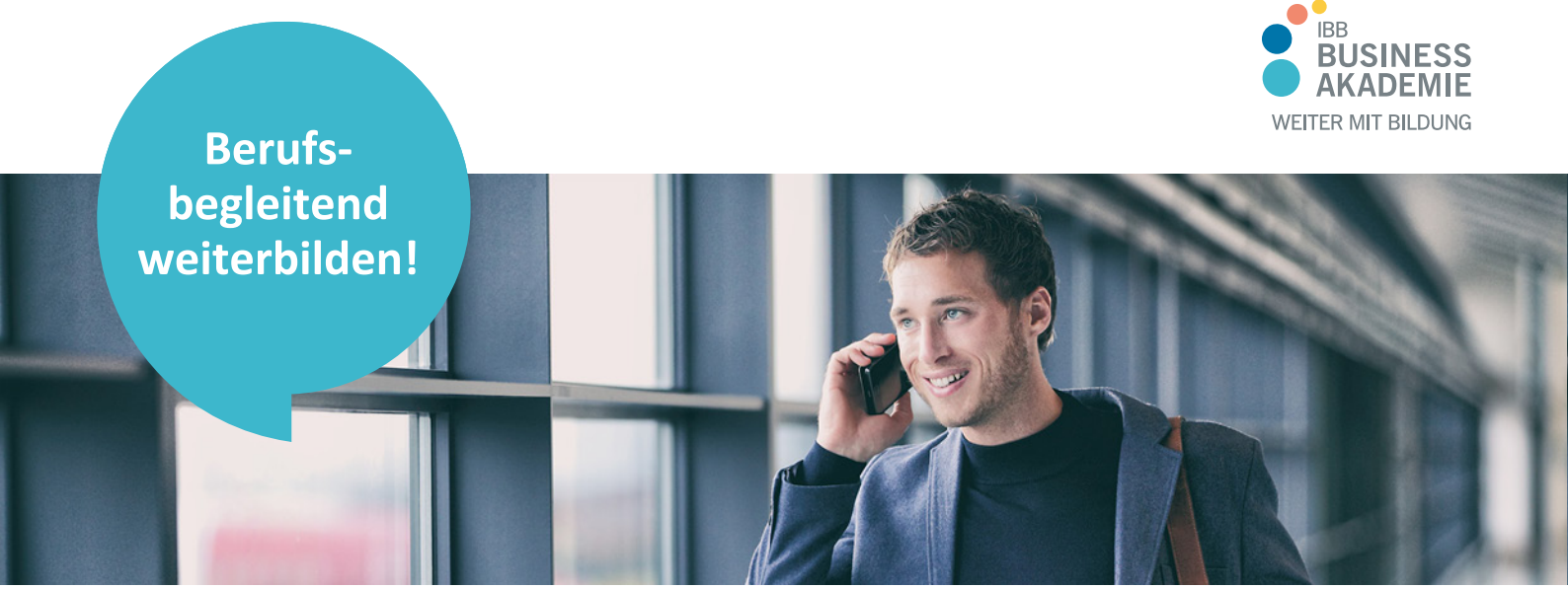
Teilnahmevoraussetzungen

Es werden keine Programmierkenntnisse vorausgesetzt

Allen Interessierten stehen wir in einem persönlichen Gespräch zur Abklärung ihrer individuellen Teilnahmevoraussetzungen zur Verfügung.

Zielgruppe

Die Weiterbildung Python Programmierer eignet sich für alle, die Python als



Programmiersprache erlernen und professionell nutzen möchten. Sie sollten eine Begeisterung für logisches Denken und Lösungen komplexer Probleme mitbringen. Die Weiterbildung ist für Quereinsteiger geeignet und der perfekte Einstieg
in die Softwareentwicklung, um Python-Entwickler/-in oder Data Scientist zu werden.

Ihre Vorteile

- Praxisnahe Lernumgebung
- Moderner Technologie Stack
- Browserbasiert
- Innovatives Data Lab

Herausgeber:

IBB Institut für Berufliche Bildung AG
Bebelstr. 40
21614 Buxtehude

Telefon: 04161 5165-89
E-Mail: akademie@ibb.com

Vorstand
Katrin Witte (Vorsitz)
Lea Tornow
Sabine Ulrichs

Aufsichtsratsvorsitzende
Sigrid Baumann-Tornow

ibb.weiterbildung
IBB_AG
pages/ibbbusinessakademie
company/ibb-business-akademie