Software Engineering Python

Softwareentwicklung (Python, PHP, Java, etc.) · Anbieter: IU Akademie IU Internationale Hochschule

FÖRDERUNG & KOSTEN

Kurspreis: auf Anfrage

Förderfähig über: Bildungsgutschein

AZAV-zertifizierter Bildungsträger — Förderung durch die Agentur für Arbeit möglich

Kursdaten auf einen Blick

Durchführung	Online
Lernformat	Self-Paced
Dauer	4 Monate in Vollzeit oder 8 Monate in Teilzeit
Unterrichtssprache	Deutsch
Kursstart	Startet alle 14 Tage
Vorkenntnisse	Keine Vorerfahrung
Abschluss	Zeugnis auf Hochschulniveau
Inklusive	Berufscoaching, Notebook / Tablet (geliehen)

Bildungsträger & Kontakt

IU Akademie IU Internationale Hochschule

Juri-Gagarin-Ring 152, 99084 Erfurt

Telefon: 030 22027283

E-Mail: weiterbildung@iu-akademie.de

Web: https://www.iu-akademie.de

Kurs online ansehen

01 Kursbeschreibung

Keine Frage: Python ist beliebt – und gilt als eine der benutzerfreundlichsten Programmiersprachen. Sie besticht durch ihre Einfachheit, Vielseitigkeit und ihre breiten Anwendungsgebiete. Von der Webentwicklung über die Datenanalyse bis zur Spieleentwicklung und Künstliche Intelligenz. Mit Deiner Weiterbildung Software Engineering Python hast Du das volle Wissen, um die Tech-Branche mitzugestalten.

Deine Weiterbildung Software Engineering Python gibt Dir alles an die Hand, was Du für eine Software Programmierung mit Python brauchst. Du lernst, Python effektiv zu nutzen, um Anwendungen zu entwickeln, Daten zu analysieren und künstliche Intelligenz zu implementieren. Und Du wirst die Grundlagen der Sprache, ihre Bibliotheken und Frameworks sowie bewährte Praktiken für saubere Python-Codes kennenlernen. Gleichzeitig tauchst Du tief in die Algorithmik ein. Im Bereich Software Engineering befasst Du Dich mit den besten Praktiken für die Planung, Entwicklung und Wartung von Softwareprojekten. Du wirst Konzepte des Software Engineering und agile Methoden näher kennenlernen. Dein neues Wissen wendest Du dann gleich an und setzt eine Desktop-Anwendung in die Praxis um. Hinweis: Durch den erfolgreichen Abschluss dieser Weiterbildung erlangen die Teilnehmenden keinen offiziellen Ingenieur-Titel.

02 Lehrplan & Module

1. Programmieren mit Python

Python ist eine der vielseitigsten und am weitesten verbreiteten Skriptsprachen. Seine klare und übersichtliche Syntax sowie sein geradliniges Design tragen wesentlich zu diesem Erfolg bei und machen ihn zu einer idealen Sprache für die Programmierausbildung. Die Anwendungsgebiete reichen von der Webentwicklung bis hin zum wissenschaftlichen Rechnen. Insbesondere in den Bereichen Datenwissenschaft und künstliche Intelligenz ist sie die gebräuchlichste Programmiersprache, die von allen wichtigen Datenverarbeitungs- und Analyseframeworks unterstützt wird. Dieser Kurs bietet eine gründliche Einführung in die Sprache und ihre Hauptfunktionen sowie Einblicke in die Begründung und Anwendung wichtiger angrenzender Konzepte wie Umgebungen, Tests und Versionskontrolle.
Kursinhalte
Einführung in Python Datenstrukturen
Konditionaler Code
Funktionen
Schleifen
Eingabe/Ausgabe
Programme protokollieren
Module und Pakete
Klassen und Vererbung Namespaces und Scopes
Klassen und Vererbung
Iteratoren und Generatoren
Fehler und Ausnahmen Syntaxfehler
Behandlung und Auslösung von Ausnahmen
Benutzerdefinierte Ausnahmen
Wichtige Python-Bibliotheken Standard-Python-Bibliothek
Wissenschaftliche Berechnungen
Beschleunigung von Python
Visualisierung
Zugriff auf Datenbanken
Arbeiten mit Python Virtuelle Umgebungen
Verwaltung von Paketen
Unit- und Integrationstests
Dokumentation des Codes
Versionskontrolle Einführung in die Versionskontrolle
Versionskontrolle mit GIT
Fakten zum Modul: Niveau: Master · Unterrichtssprache: Deutsch · 5 ECTS-Punkte · Modulprüfung: Hausarbeit

2. Software Engineering: Softwareprozesse

Software-Prozesse und Lebenszyklusmodelle ermöglichen eine strukturierte Vorgehensweise für verschiedene Software-Engineering-Aufgaben. Ziel dieses Moduls ist es, ein Verständnis für diese Struktur zu vermitteln und zu zeigen, wie sie über den gesamten Plan-Build-Run-Lebenszyklus angewendet werden kann. Eine wichtige Grundlage für die Arbeit mit Software-Prozessen ist die Modellierung mit einer geeigneten Notation. Es werden verschiedene Arten von Lebenszyklen besprochen, darunter der plangetriebene und der agile Ansatz sowie Mischformen (Hybridmodelle). Ein besonderes Augenmerk wird auf die unterschiedlichen Umgebungen gelegt, für die sich diese Ansätze am besten eignen.
Kursinhalte
Grundlagen der Softwareprozesse Die Bedeutung von Softwareprozessen und Lebenszyklusmodellen
Historischer Überblick
Softwareprozess: Definition und Modellierung Modellierungsnotationen und Metamodelle
Notationen zur Modellierung der Wechselwirkung zwischen Prozessen
Notationen der Detailebene
Grundlegende Lebenszyklusmodelle Wasserfall-Modelle
Das V-Modell
Komponentenbasierte Modelle
Iterative, inkrementelle und evolutionäre Modelle
Agile und schlanke Entwicklungsprozesse Das Agile Manifest
Scrum
Allgemeine agile Praktiken
Lean Development und Kanban
Skalierung in der agilen Entwicklung
Hybride Prozesse
Der Lebenszyklus von Softwareprodukten Prozessmodelle auf Detailebene
IT-Service-Management und Betrieb
DevOps
Informationssicherheit und Datenschutz
Governance und Management von Softwareprozessen Prozesssteuerung
Prozessdesign und -einführung
Prozessanpassung
Prozessbeurteilung, -verbesserung und -bewertung
Werkzeugunterstützung
Fakten zum Modul: Niveau: Master · Unterrichtssprache: Deutsch · 5 ECTS-Punkte · Modulprüfung: Fachpräsentation

3. Algorithmik

Eine Kernaktivität der Informatik und ähnlicher Bereiche, ist der Entwurf, die Verwendung und die Anwendung von Algorithmen zur Lösung von Problemen. Dieser Kurs führt in gängige Ansätze zum Entwurf von Algorithmen sowie in wichtige Klassen von Algorithmen ein, die zur Lösung häufiger Probleme verwendet werden können. Die erfolgreiche Durchführung dieser Aufgaben erfordert ein gründliches Verständnis der Qualitätsmerkmale von Algorithmen: (partielle und totale) Korrektheit, Genauigkeit, Vollständigkeit und Effizienz. Gleichzeitig gibt es Grenzen für das, was ein Algorithmus theoretisch und praktisch erreichen kann und sollte, und es ist wichtig, diese Grenzen zu erkennen und zu berücksichtigen. Neben Algorithmen, die auf Standardprogrammierparadigmen basieren, gibt es auch verschiedene andere Programmierparadigmen, die zu anderen Arten von Algorithmen führen; daher bietet dieser Kurs auch eine kurze Einführung in Algorithmen für paralleles Rechnen, probabilistische Algorithmen und Quantenalgorithmen.
Kursinhalte
Einführung in die Algorithmik Grundlegende Konzepte und historischer Überblick
Algorithmen, Programmiersprachen und Datenstrukturen
Qualität von Algorithmen: Korrektheit, Genauigkeit, Vollständigkeit, Effizienz
Die Rolle von Algorithmen in der Gesellschaft
Algorithmenentwurf Datenstrukturen
Rekursion und Iteration
Divide-and-Conquer
Algorithmenstrategien: Ausgleichendes, gieriges und dynamisches Programmieren
Wichtige Algorithmen Sortieren und Suchen
Mustererkennung
Der RSA-Algorithmus
Der k-Means-Algorithmus für Daten-Clustering
Korrektheit, Genauigkeit und Vollständigkeit von Algorithmen Partielle Korrektheit
Totale Korrektheit
Sicherstellung der Korrektheit im Programmieralltag
Genauigkeit, Approximation und Fehleranalyse
Berechenbarkeit Modelle der Datenverarbeitung
Das Halteproblem
Unentscheidbare Probleme
Effizienz von Algorithmen: Komplexitätstheorie Modelle der Komplexität
NP-Vollständigkeit
P=NP?
Erweiterte Algorithmik Parallele Rechnern
Probabilistische Algorithmen
Fakten zum Modul: Niveau: Master · Unterrichtssprache: Deutsch · 5 ECTS-Punkte · Modulprüfung: Klausur, 90 Minuten

4. Projekt: Software Engineering

In dieser Lehrveranstaltung wird das in den vorangegangenen Modulen erworbene Wissen auf kleine bis mittelgroße Projekte angewendet. Die Projektteams durchlaufen wichtige Phasen der Software-Entwicklung und erstellen die entsprechenden Artefakte (z.B. Anforderungsspezifikation, Design, Implementierung, Tests, Dokumentation). Die Qualitätssicherung dieser Artefakte wird durch den Tutor und Studierende durchgeführt. Auf diese Weise lernen die Studierenden sowohl die Erstellung als auch die Qualitätssicherung von Artefakten in einem Softwareentwicklungsprozess.
Kursinhalte
Im Software-Engineering werden Projekte in Bezug auf die Rollen der Teammitglieder (z. B. Projektmanager, Qualitätssicherung) und die einzelnen Aktivitäten (z. B. Planung, Analyse, Implementierung) organisiert, um Softwaren zu erstellen, die den Kundenbedürfnissen entsprechen. In Bezug auf die gewählte Vorgehensweise (agil oder plangetrieben) durchlaufen die Studierenden alle typischen Phasen, die für große Projekte relevant sind. Sie sammeln Erfahrungen hinsichtlich organisatorischer Aspekte, wie gängigen Rollen und Aktivitäten, sowie bezüglich praktischer Aufgaben wie die Erstellung der Anforderungsspezifikation, Design, Implementierung, Tests und Dokumentation.
Fakten zum Modul: Niveau: Master · Unterrichtssprache: Deutsch · 5 ECTS-Punkte · Modulprüfung: Portfolio

03 Enthaltene Zertifikate

Trägerzertifikat: Zeugnis auf Hochschulniveau

04 Voraussetzungen

Vorkenntnisse

Niveau: Keine Vorerfahrung

Zulassungsvoraussetzung: gute Deutschkenntnisse auf B2-Niveau.

Die Weiterbildung ist auf dem inhaltlichen Niveau eines Master Studiengangs. Eine Weiterbildung auf Master-Niveau ist anspruchsvoller als auf Bachelor-Niveau. Vorhandenes Grundlagenwissen im gewählten Fachbereich ist deshalb von Vorteil.

Technische Voraussetzungen

Laptop mit Internetzugang.

05 Zielgruppen & Berufsperspektiven

- Arbeitssuchende
- Fachkräfte

Mögliche Berufsbilder: Python Entwickler, Softwareentwickler, Anwendungsentwickler

NÄCHSTER SCHRITT

Kursdetails online ansehen und Anbieter kontaktieren

IU Akademie IU Internationale Hochschule · Juri-Gagarin-Ring 152, 99084 Erfurt
030 22027283 · weiterbildung@iu-akademie.de · https://www.iu-akademie.de

Online ansehen