



Fundacja
ALTERnacja

Brak ocen dla tego dostawcy

Podstaw programowania w języku Python - PCAP: Programming Essentials in Python (dla nauczycieli)

Numer usługi 2024/04/22/165599/2131802

📍 zdalna w czasie rzeczywistym

📄 Usługa szkoleniowa

🕒 72 h

📅 20.08.2024 do 29.10.2024

5 000,00 PLN brutto

5 000,00 PLN netto

69,44 PLN brutto/h

69,44 PLN netto/h

Informacje podstawowe

| | |
|--------------------------------------|--|
| Kategoria | Informatyka i telekomunikacja / Programowanie |
| Identyfikator projektu | Kierunek - Rozwój |
| Sposób dofinansowania | wsparcie dla pracodawców i ich pracowników wsparcie dla osób indywidualnych |
| Grupa docelowa usługi | <p>Szkolenie przeznaczone jest dla osób fizycznych lub pracowników firm, którzy chcieliby zdobyć solidne podstawy programowania przy użyciu języka Python. Grupa docelowa obejmuje osoby, które dopiero rozpoczynają swoją przygodę z programowaniem i nie mają wcześniejszego doświadczenia z tym językiem, jak również tych, którzy chcieliby poszerzyć swoje umiejętności o Pythona. Dla początkujących kurs PCAP stanowi doskonałą platformę do nauki podstawowych koncepcji programowania oraz zapoznania się ze składnią i funkcjonalnościami języka Python. Z uwagi na ogromną popularność języka Python, treści szkoleniowe mogą zostać wykorzystane w wielu obszarach techniki i inżynierii, począwszy od technologii informacyjnych (IT) poprzez analizę danych, sztuczną inteligencję oraz przemysł wytwórczy.</p> <p>Dzięki praktycznym przykładom i zadaniom kursowym, uczestnicy szkolenia będą mogli szybko zastosować zdobytą wiedzę w praktyce, co stanowi kluczowy element skutecznej nauki programowania.</p> |
| Minimalna liczba uczestników | 12 |
| Maksymalna liczba uczestników | 20 |
| Data zakończenia rekrutacji | 19-07-2024 |
| Forma prowadzenia usługi | zdalna w czasie rzeczywistym |

Cel

Cel edukacyjny

Zasadniczym celem edukacyjnym jest zapoznanie uczestników z:

- podstawami programowania i specyfiką języka Python,
- typami danych i operacji,
- instrukcjami warunkowymi, sterującymi i pętlami,
- listami, wycinkami, krotkami, łańcuchami znaków,
- przetwarzaniem list, słowników, krotek, tablic,
- tworzeniem i wykorzystaniem funkcji,
- zagadnieniem zakresu nazw,
- modułami, pakietami,
- wyjątkami i ich obsługą,
- programowaniem obiektowym (OOP),
- dziedziczeniem, domknięciami
- pracą z plikami.

Efekty uczenia się oraz kryteria weryfikacji ich osiągnięcia i Metody walidacji

| Efekty uczenia się | Kryteria weryfikacji | Metoda walidacji |
|--|---|-------------------------|
| <p>Uczestnik rozumie uniwersalne koncepcje programowania komputerowego, obejmujące zmienne, struktury danych, algorytmy, kontrolę przepływu, funkcje i wyjątki.</p> | <p>Test podsumowujący (1/2). Egzamin końcowy.</p> | <p>Test teoretyczny</p> |
| <p>Uczestnik potrafi sprawnie korzystać z narzędzi deweloperskich, środowiska wykonawczego oraz składni i semantyki języka Python.</p> | <p>Test podsumowujący (1/2). Egzamin końcowy.</p> | <p>Test teoretyczny</p> |
| <p>Uczestnik potrafi stosować fundamentalne techniki programowania, najlepsze praktyki, zwyczaje i słownictwo, w tym najczęściej używane funkcje biblioteki standardowej w Pythonie 3.</p> | <p>Test podsumowujący (1/2). Egzamin końcowy.</p> | <p>Test teoretyczny</p> |
| <p>Uczestnik potrafi pisać programy w języku Python przy użyciu standardowej infrastruktury językowej oraz zna sposoby rozwiązywania typowych problemów implementacyjnych.</p> | <p>Test podsumowujący (1/2). Egzamin końcowy.</p> | <p>Test teoretyczny</p> |

| Efekty uczenia się | Kryteria weryfikacji | Metoda walidacji |
|--|---|------------------|
| Uczestnik umie pracować z modułami i pakietami, przetwarzać pliki tekstowe i binarne, oraz korzystać z generatorów, iteratorów i domknięć. | Test podsumowujący (1/2). Egzamin końcowy. | Test teoretyczny |
| Uczestnik rozumie podstawy programowania zorientowanego obiektowo (OOP) oraz sposób, w jaki są one stosowane w języku Python. | Test podsumowujący (1/2). Egzamin końcowy. | Test teoretyczny |

Kwalifikacje

Kompetencje

Usługa prowadzi do nabycia kompetencji.

Warunki uznania kompetencji

Pytanie 1. Czy dokument potwierdzający uzyskanie kompetencji zawiera opis efektów uczenia się?

TAK. Po zadaniu egzaminu dostarczonego przez Python Institute dostępnego w programie Cisco Network Academy, uczestnik otrzyma imienny dokument w języku angielskim Statement of Achievement (Oświadczenie o osiągnięciach), który zawiera osiągnięte kompetencje.

Pytanie 2. Czy dokument potwierdza, że walidacja została przeprowadzona w oparciu o zdefiniowane w efektach uczenia się kryteria ich weryfikacji?

TAK. Dokument potwierdza przeprowadzenie walidacji z wykorzystaniem globalnego systemu edukacyjne Cisco Network Academy, który udostępnia szczegółowe narzędzia egzaminacyjne. Dokument potwierdza osiągnięcia kompetencji zgodnych z poniższymi efektami uczenia.

Pytanie 3. Czy dokument potwierdza zastosowanie rozwiązań zapewniających rozdzielenie procesów kształcenia i szkolenia od walidacji?

TAK, metodyka szkoleń i weryfikacji kompetencji Cisco Network Academy wymaga, aby egzamin końcowy realizowany był przez desygnowanego instruktora Cisco, nie prowadzącego szkolenia dla danej grupy kursowej. System egzaminacyjny Netacad.com automatycznie ocenia uczestnika i generuje imienny dokument.

Program

PCAP: Programming Essentials in Python jest najbardziej rozpoznawalnym kursem podstaw języka Python na świecie, ponieważ została stworzony przez Python Institute (<https://pythoninstitute.org/pcap>) i świadczony jest przez globalną strukturę Cisco Network Academy (<https://www.netacad.com/>), która działa w 190 krajach.

Zawartość merytoryczna kolejnych modułów kursu PCAP została tak dobrana, aby uczestnik szkolenia zapoznawał się kolejno i stopniowo z treściami merytorycznymi. Nie mam zatem wymogu posiadania wiedzy programistycznej w momencie przystępowania do kursu.

Kurs PCAP składa się z sześciu modułów:

Mod.1 Wstęp do języka Python i programowania.

- podstawy programowania: kompilacja a interpretacja
- podstawowe metody formatowania i wyświetlania danych
- operatory numeryczne, ich wzajemnymi relacjami i wiązania
- pojęcie zmiennej, konwencje nazewnicze
- operator przypisania, zasady rządzące budową wyrażeń
- wprowadzenie i konwersja danych

Mod.2 Typy danych, zmienne, podstawowe operacje wejścia/wyjścia, operatory.

- literały w Pythonie
- operatory – narzędzia do manipulacji danymi
- operatory numeryczne, ich wzajemne relacje
- pojęcie zmiennej, konwencje nazewnicze
- operator przypisania, zasady rządzące budową wyrażeń
- wprowadzenie i konwersja danych

Mod.3. Wartości logiczne, wykonywanie warunkowe, pętle, listy, przetwarzanie list, operacje logiczne i bitowe.

- wartości logicznymi (Boolean)
- porównywanie różnych wartości oraz kontrolowania ścieżek wykonania programu przy użyciu instrukcji warunkowych *if* i *if-else*
- wykorzystanie pętli (*while* i *for*) oraz sposób kontrolowania ich zachowania za pomocą instrukcji *break* i *continue*
- różnice między operacjami logicznymi a bitowymi
- pojęcie listy i przetwarzania list, wycinanie, sortowanie
- koncepcja tablic wielowymiarowych

Mod.4 Funkcje, zakresy, krotki, słowniki, przetwarzanie danych.

- definiowanie i użycie funkcji – ich uzasadnieniem, celem, konwencjami oraz pułapkami
- Sposoby przekazywania argumentów do funkcji, ustalenie wartości domyślnej
- zwracanie wyniku funkcji
- zagadnienie zakresu nazw
- rola krotek i słowników w przetwarzaniu danych

Mod.5 Moduły, pakiety, metody list, metody łańcuchów znaków, wyjątki.

- charakterystyka modułów standardowych, zastosowanie instrukcji *try-except* oraz *raise*
- wprowadzić pojęcie łańcuchów znaków i ich specyficznych metod, wraz z ich podobieństwami i różnicami w porównaniu do list

Mod.6 Programowanie obiektowe: klasy, metody, obiekty i standardowe cechy obiektów; obsługa wyjątków i operacje na plikach.

- podstawy programowania zorientowanego obiektowo (OOP)
- różnice między podejściem OOP a klasycznym, podejściem proceduralnym
- standardowe cechy obiektowe: dziedziczenie, abstrakcję, enkapsulację i polimorfizm, wraz z problemami specyficznymi dla Pythona, takimi jak zmienne egzemplarza kontra klasy
- implementacja dziedziczenia
- funkcje generujące (instrukcją *yield*)
- domknięcia - słowo kluczowe *lambda*
- praca z plikami: otwarcie, odczyt i zapis

Harmonogram

Liczba przedmiotów/zajęć: 18

| Przedmiot / temat zajęć | Prowadzący | Data realizacji zajęć | Godzina rozpoczęcia | Godzina zakończenia | Liczba godzin |
|--|-----------------|-----------------------|---------------------|---------------------|---------------|
| 1 z 18 Wprowadzenie do języka Python i programowania komputerowego | Adrianna Piszcz | 20-08-2024 | 18:00 | 21:15 | 03:15 |
| 2 z 18 Pierwszy program | Adrianna Piszcz | 23-08-2024 | 18:00 | 21:15 | 03:15 |
| 3 z 18 Literały - dane same w sobie | Adrianna Piszcz | 27-08-2024 | 18:00 | 21:15 | 03:15 |
| 4 z 18 Operatory arytmetyczne | Adrianna Piszcz | 30-08-2024 | 18:00 | 21:15 | 03:15 |
| 5 z 18 Hierarchia priorytetów | Adrianna Piszcz | 03-09-2024 | 18:00 | 21:15 | 03:15 |
| 6 z 18 Zmienne | Adrianna Piszcz | 06-09-2024 | 18:00 | 21:15 | 03:15 |
| 7 z 18 Typowanie zmiennych | Adrianna Piszcz | 20-09-2024 | 18:00 | 21:15 | 03:15 |
| 8 z 18 Funkcja input() i operatory łańcuchów znaków | Adrianna Piszcz | 24-09-2024 | 18:00 | 21:15 | 03:15 |
| 9 z 18 Łańcuchy c.d. | Adrianna Piszcz | 27-09-2024 | 18:00 | 21:15 | 03:15 |
| 10 z 18 Operatory porównania | Adrianna Piszcz | 01-10-2024 | 18:00 | 21:15 | 03:15 |
| 11 z 18 Wykonanie warunkowe | Adrianna Piszcz | 04-10-2024 | 18:00 | 21:15 | 03:15 |
| 12 z 18 Pętle w Pythonie | Adrianna Piszcz | 08-10-2024 | 18:00 | 21:15 | 03:15 |
| 13 z 18 Pętle w Pythonie cz.2 | Adrianna Piszcz | 11-10-2024 | 18:00 | 21:15 | 03:15 |
| 14 z 18 Pętle w Pythonie cz.3 | Adrianna Piszcz | 15-10-2024 | 18:00 | 21:15 | 03:15 |

| Przedmiot / temat zajęć | Prowadzący | Data realizacji zajęć | Godzina rozpoczęcia | Godzina zakończenia | Liczba godzin |
|--|-----------------|-----------------------|---------------------|---------------------|---------------|
| 15 z 18 Logika komputerowa i listy | Adrianna Piszcz | 18-10-2024 | 18:00 | 21:15 | 03:15 |
| 16 z 18 Funkcje, typy sekwencyjne i mutowalność | Adrianna Piszcz | 22-10-2024 | 18:00 | 21:15 | 03:15 |
| 17 z 18 Programy złożone | Adrianna Piszcz | 25-10-2024 | 18:00 | 21:15 | 03:15 |
| 18 z 18 EGZAMIN | - | 29-10-2024 | 18:00 | 21:15 | 03:15 |

Cennik

Cennik

| Rodzaj ceny | Cena |
|----------------------|--------------|
| Koszt usługi brutto | 5 000,00 PLN |
| Koszt usługi netto | 5 000,00 PLN |
| Koszt godziny brutto | 69,44 PLN |
| Koszt godziny netto | 69,44 PLN |

Prowadzący

Liczba prowadzących: 1



1 z 1

Adrianna Piszcz

Wykładowca akademicki, Wydział Informatyki UKW

Informacje dodatkowe

Informacje o materiałach dla uczestników usługi

Każdy uczestnik szkolenia otrzyma:

- dostęp do platformy elearningowej netacad.com, także po zakończeniu szkolenia. W netacad dostępne są kompletne materiały e-learningowe do kursu.
- dostęp do gotowego środowiska programistycznego, aby na bieżąco tworzyć programy pod nadzorem instruktora,
- dostęp do własnej platformy Fundacji ALTERNACJA celem pobierania dodatkowych zadań.

Informacje dodatkowe

Jako godzinę szkolenia przyjęto godzinę lekcyjną.

Warunki techniczne

Aby uczestniczyć w szkoleniu **PCAP: Programming Essentials in Python**, uczestnik powinien dysponować typowym komputerem stacjonarnym lub laptopem na minimalnych parametrach:

- łącze do Internetu w dowolnej technologii (także LTE) przepustowości przynajmniej 2 Mbit/s
- procesor Intel Core2 Duo lub lepszy
- pamięć RAM: 4GB lub więcej,
- kamera i mikrofon.

Kontakt



Piotr Żmudziński

E-mail piotr@alternacja.pl

Telefon (+48) 695 616 100