



Programowanie w języku Python - podstawy

Numer usługi 2026/02/12/202247/3330895

2 952,00 PLN brutto
2 400,00 PLN netto
84,34 PLN brutto/h
68,57 PLN netto/h
157,50 PLN cena rynkowa ⓘ

JSYSTEMS SPÓŁKA
Z OGRANICZONĄ
ODPOWIEDZIALNOŚ
CIĄ

Brak ocen dla tego dostawcy

- 📄 Usługa szkoleniowa
- 📺 zdalna w czasie rzeczywistym
- 🕒 35:00 h
- 📅 23.11.2026 do 27.11.2026

Informacje podstawowe

Kategoria	Informatyka i telekomunikacja / Programowanie
Grupa docelowa usługi	Szkolenie jest odpowiednie zarówno dla osób początkujących, jak i dla tych, którzy potrzebują ugruntowania podstaw przed przejściem do bardziej zaawansowanych tematów.
Minimalna liczba uczestników	6
Maksymalna liczba uczestników	10
Data zakończenia rekrutacji	19-11-2026
Forma prowadzenia usługi	zdalna w czasie rzeczywistym
Liczba godzin usługi	35
Podstawa uzyskania wpisu do BUR	Certyfikat systemu zarządzania jakością wg. ISO 9001:2015 (PN-EN ISO 9001:2015) - w zakresie usług szkoleniowych

Cel

Cel edukacyjny

Nabycie przez uczestników podstawowych umiejętności programowania w języku Python, umożliwiających samodzielne tworzenie skryptów i prostych aplikacji, automatyzację zadań przy użyciu Pythona oraz przygotowanie do dalszego uczenia się zaawansowanych zastosowań Pythona.

Efekty uczenia się oraz kryteria weryfikacji ich osiągnięcia i Metody walidacji

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Stosuje podstawową składnię Python: zmienne, typy danych, instrukcje sterujące.	Uczestnik implementuje skrypt realizujący wskazany algorytm przy użyciu zmiennych, list, pętli for/while i instrukcji warunkowych, który zwraca poprawny wynik.	Test teoretyczny z wynikiem generowanym automatycznie
Tworzy i stosuje funkcje, moduły i pakiety Pythona. Uczestnik definiuje funkcje z parametrami i wartościami domyślnymi, importuje moduły biblioteki standardowej i organizuje kod w logiczne moduły. Test teoretyczny z wynikiem generowanym automatycznie. Walidację przeprowadza trener prowadzący szkolenie.	Uczestnik definiuje funkcje z parametrami i wartościami domyślnymi, importuje moduły biblioteki standardowej i organizuje kod w logiczne moduły.	Test teoretyczny z wynikiem generowanym automatycznie
Przetwarza dane przy użyciu list, słowników i zbiorów. Czyta i zapisuje dane z/do plików w Pythonie.	Uczestnik rozwiązuje zadanie przetwarzania danych przy użyciu list comprehension, słowników i operacji na zbiorach, uzyskując poprawny wynik w mniej niż 15 liniach kodu. Uczestnik pisze skrypt odczytujący dane z pliku CSV, przetwarzający je i zapisujący wyniki do nowego pliku, z poprawną obsługą kodowania i wyjątków.	Test teoretyczny z wynikiem generowanym automatycznie Test teoretyczny z wynikiem generowanym automatycznie

Kwalifikacje

Kompetencje

Usługa prowadzi do nabycia kompetencji.

Warunki uznania kompetencji

Pytanie 1. Czy dokument potwierdzający uzyskanie kompetencji lub wyraźnie z nim powiązane inne dokumenty związane ze wsparciem zawierają opis efektów uczenia się?

TAK

Pytanie 2. Czy dokument lub wyraźnie z nim powiązane inne dokumenty związane ze wsparciem potwierdzają, że walidacja została przeprowadzona w oparciu o zdefiniowane w efektach uczenia się kryteria ich weryfikacji i zgodnie z zaplanowanymi metodami walidacji?

TAK

Pytanie 3. Czy dokument lub wyraźnie z nim powiązane inne dokumenty związane ze wsparciem potwierdzają zastosowanie rozwiązań zapewniających rozdzielanie procesów kształcenia i szkolenia od walidacji?

TAK

Program

1. Środowisko pracy

- o Instalacja interpretera Python
- o Instalacja IDE - Pycharm
- o Tworzenie projektu
- o Obsługa interfejsu Pycharm
- o Przydatne skróty klawiaturowe

2. "Hello world" i pisanie na konsoli

3. Zmienne i typy danych

- o Zmienne
- o Typy danych
 - Typ tekstowy
 - Typy liczbowe

4. Instrukcje warunkowe

- o Jeden warunek
- o Else
- o Wiele warunków
- o Operatory logiczne w warunkach

5. Pętle

- o Pętla while
- o Pętla for
- o Zagnieżdżanie pętli
- o Instrukcja BREAK
- o Instrukcja CONTINUE

6. Łańcuchy znaków

- o Funkcje wbudowane
 - upper
 - lower
 - title
 - replace
 - len w kontekście ciągów tekstowych
 - count
 - strip
 - split i join - zamiana tekstu na listę i listy na tekst
- o Łańcuchy funkcji

- o Iterowanie po łańcuchach tekstowych
- o Mnożenie tekstu. Ale jak?
- o Wygodne sprawdzanie czy tekst zawiera frazę
- o Czy Python>Java?
- o Cięcia, cięcia - o cięciu łańcuchów tekstowych słów kilka

7. Listy

- o Tworzenie list
- o Pobieranie wartości z list
- o Iterowanie po listach
- o Sprawdzanie czy element znajduje się na liście
- o Modyfikowanie zawartości listy
 - Dodawanie nowych wartości i wstawianie w miejsce istniejących
 - Kasowanie elementów z listy
- o Funkcje wbudowane w listy
 - Sortowanie i odwracanie list
 - Inne ciekawe funkcje i możliwości

8. Krotki

- o Deklaracja i uzupełnianie krotek danymi
- o Pobieranie wartości z krotek

9. Słowniki

- o Tworzenie słowników
- o Pobieranie wartości ze słowników
- o Modyfikacja zawartości słowników

10. Zestawy

- o Tworzenie zestawów i konwersje z innych typów złożonych
- o Modyfikowanie zawartości zestawów
- o Funkcja difference
- o Funkcja intersect
- o Funkcja union

11. Zaawansowane elementy przetwarzania list i zbiorów

12. Wyjątki

- o Obsługa wyjątków

13. Funkcje

- o Deklarowanie funkcji
- o Parametry funkcji
- o Wartości domyślne parametrów

- o Zwracanie wyników z funkcji

- o Dokumentowanie funkcji

14. Moduły

- o Definiowanie modułów

- o Dokumentowanie modułów i sprawdzanie dostępnych funkcji

15. Korzystanie z plików tekstowych

- o Czytanie z plików tekstowych

- read()

- readlines()

- readline()

- o Funkcja seek()

- o Sprawdzanie ilości linii w pliku

- o Zapis w plikach tekstowych

- Tryby otwarcia pliku

- Wprowadzanie danych do pliku

16. Dane zdalne - wykorzystanie usług sieciowych

- o Pobieranie danych za pomocą GET

- o Przesyłanie danych za pomocą POST

17. Wykorzystanie baz danych

- o Tworzenie baz danych, użytkowników i tabel w bazach danych PostgreSQL

- o Łączenie z serwerem PostgreSQL

- o Pobieranie danych z użyciem SELECT

- o Wstawianie, zmiana i kasowanie danych, oraz operacje DDL

18. Wstęp do obiektowości

- o Deklaracja klas

- o Tworzenie obiektów

- o Atrybuty

- o Metody

- o Najczęściej używane metody magiczne

19. Flask - tworzymy aplikację WEB

- o Tworzenie projektu

- o Mapowanie adresów

- o Kod i szablony kodu HTML

- o Przekazywanie danych do widoku i tagi jinja2

- o Odczyt parametrów z paska

- o Pobieranie i umieszczanie danych w sesji

o Obsługa formularzy

o Tworzymy w pełni funkcjonalną aplikację typu TODOS przechowującą dane w bazie

Harmonogram

Liczba pozycji harmonogramu: 16

Przedmiot / temat	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
1 z 16 Środowisko pracy	Przemysław Smaga	23-11-2026	09:00	12:30	03:30
2 z 16 Przerwa Obiadowa	Przemysław Smaga	23-11-2026	12:30	13:00	00:30
3 z 16 "Hello world" i pisanie na konsoli/Zmienne i typy danych	Przemysław Smaga	23-11-2026	13:00	16:00	03:00
4 z 16 Instrukcje warunkowe/Pętle	Przemysław Smaga	24-11-2026	09:00	12:30	03:30
5 z 16 Przerwa Obiadowa	Przemysław Smaga	24-11-2026	12:30	13:00	00:30
6 z 16 Łańcuchy znaków/Listy	Przemysław Smaga	24-11-2026	13:00	16:00	03:00
7 z 16 Krotki/Słowniki	Przemysław Smaga	25-11-2026	09:00	12:30	03:30
8 z 16 Przerwa Obiadowa	Przemysław Smaga	25-11-2026	12:30	13:00	00:30
9 z 16 Zbiory/Zaawansowane elementy przetwarzania list i zbiorów	Przemysław Smaga	25-11-2026	13:00	16:00	03:00
10 z 16 Wyjątki/Funkcje	Przemysław Smaga	26-11-2026	09:00	12:30	03:30
11 z 16 Przerwa Obiadowa	Przemysław Smaga	26-11-2026	12:30	13:00	00:30

Przedmiot / temat	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
12 z 16 Moduły/Korzystanie z plików tekstowych	Przemysław Smaga	26-11-2026	13:00	16:00	03:00
13 z 16 Dane zdalne - wykorzystanie usług sieciowych/ Wykorzystanie baz danych	Przemysław Smaga	27-11-2026	09:00	12:30	03:30
14 z 16 Przerwa Obiadowa	Przemysław Smaga	27-11-2026	12:30	13:00	00:30
15 z 16 Wstęp do obiektowości/Flask - tworzymy aplikację WEB	Przemysław Smaga	27-11-2026	13:00	15:30	02:30
16 z 16 Walidacja	Przemysław Smaga	27-11-2026	15:30	16:00	00:30

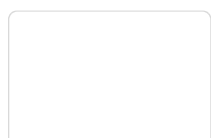
Cennik

Cennik

Rodzaj ceny	Cena
Koszt przypadający na 1 uczestnika brutto	2 952,00 PLN
Koszt przypadający na 1 uczestnika netto	2 400,00 PLN
Koszt osobogodziny brutto	84,34 PLN
Koszt osobogodziny netto	68,57 PLN

Prowadzący

Liczba prowadzących: 1



1 z 1

Przemysław Smaga



Trener posiada wieloletnie doświadczenie w programowaniu w języku Python, przy czym kluczowe kwalifikacje w zakresie programowania Python (podstawy i poziom średniozaawansowany), automatyzacji zadań i tworzenia skryptów zostały zdobyte i są czynnie wykorzystywane w okresie ostatnich 5 lat (od 2021 roku do chwili obecnej). Potwierdzają to zrealizowane projekty developerskie oraz szkolenia z programowania Python w latach 2022–2026.

Informacje dodatkowe

Informacje o materiałach dla uczestników usługi

Informacje o materiałach dla uczestników usługi - Uczestnicy otrzymają komplet materiałów PDF. Każdy uczestnik otrzymuje kod dostępu i

dane logowania do platformy ZOOM na 7 dni przed datą rozpoczęcia szkolenia. Dane

przesyłane są na adres e-mail podany podczas rejestracji.

Warunki uczestnictwa

Umiejętność korzystania z komputera.

Informacje dodatkowe

Warunkiem ukończenia szkolenia i otrzymania zaświadczenia jest uzyskanie minimalnej

frekwencji na poziomie 80% całkowitego czasu trwania usługi. Obecność uczestnika będzie

potwierdzana na podstawie codziennych list obecności lub logów z platformy online.

Warunki techniczne

Uczestnik musi dysponować sprzętem i łączem o parametrach:

- Procesor: min. 4-rdzeniowy (np. Intel i5/i7 lub odpowiednik AMD/M1/M2)
- Pamięć RAM: min. 16 GB
- Dysk: min. 20 GB wolnej przestrzeni
- System operacyjny: Windows 10/11 Pro, Linux lub macOS
- Multimedia: Sprawna kamera internetowa oraz mikrofon (wymagane do komunikacji i weryfikacji obecności)
- Łącze internetowe: Stabilne połączenie o minimalnej prędkości 10 Mbps (download) / 5 Mbps (upload)
- Oprogramowanie: Uprawnienia administratora pozwalające na instalację narzędzi

Kontakt

Biuro Obsługi Klienta

E-mail biuro@jssystemy.pl



Telefon (+48) 534 506 503