



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Wprowadzenie do programowania w Python [S1DSwB1>WdPP]

Przedmiot

Kierunek studiów

Data Science w biznesie

Rok/Semestr

1/1

Studia w zakresie (specjalność)

–

Profil studiów

ogólnoakademicki

Poziom studiów

pierwszego stopnia

Język oferowanego przedmiotu

polski

Forma studiów

stacjonarne

Wymagalność

obligatoryjny

Liczba godzin

Wykład

30

Laboratorium

0

Inne

0

Ćwiczenia

30

Projekty/seminaria

0

Liczba punktów ECTS

5,00

Koordynatorzy

dr inż. Marcin Nowak

marcin.nowak@put.poznan.pl

dr Grzegorz Nowak

grzegorz.nowak@put.poznan.pl

Wykładowcy

Wymagania wstępne

Student posiada podstawową wiedzę z matematyki oraz informatyki. Potrafi efektywnie się uczyć oraz analitycznie rozwiązywać problemy. Ma świadomość konieczności poszerzania swoich kompetencji

Cel przedmiotu

Celem przedmiotu jest wprowadzenie studentów do fundamentalnych zasad programowania w języku Python. Studenci zdobędą wiedzę na temat podstawowej składni i semantyki języka, sposobów organizacji kodu oraz kluczowych koncepcji, takich jak typy danych, operatory, instrukcje sterujące, funkcje i struktury danych. Przedmiot ma na celu rozwinięcie umiejętności analitycznego myślenia oraz efektywnego rozwiązywania problemów algorytmicznych. Studenci nauczą się pisać, testować i debugować programy w Pythonie, a także korzystać z podstawowych bibliotek i narzędzi programistycznych. Zdobyte kompetencje będą stanowiły solidną podstawę do dalszego rozwijania umiejętności programistycznych w bardziej zaawansowanych obszarach informatyki.

Przedmiotowe efekty uczenia się

Wiedza:

1. Opisuje podstawowe koncepcje programowania, składnię języka Python oraz jego kluczowe struktury danych [DSB1_W02].
2. Wyjaśnia podstawowe mechanizmy debugowania, testowania kodu oraz obsługi API w Pythonie [DSB1_W05].

Umiejętności:

1. Implementuje programy w języku Python, stosując zmienne, operatory, struktury danych oraz instrukcje sterujące [DSB1_U02].
2. Tworzy i wykorzystuje funkcje oraz moduły do organizacji kodu w Pythonie [DSB1_U08].
3. Przetwarza pliki tekstowe, JSON, XML oraz inne formaty danych z wykorzystaniem Pythona [DSB1_U04].
4. Stosuje biblioteki Pandas i NumPy do analizy i przetwarzania danych [DSB1_U09].
5. Debuguje i testuje kod w Pythonie, wykorzystując narzędzia do analizy błędów i zapewnienia poprawności działania programów [DSB1_U07].
6. Argumentuje i ocenia różne podejścia do rozwiązywania problemów programistycznych, bazując na analizie kodu [DSB1_U11].
7. Samodzielnie rozwija swoje umiejętności programistyczne, korzystając z dostępnych źródeł i kursów online [DSB1_U15].

Kompetencje społeczne:

1. Dokonuje krytycznej analizy poprawności kodu, optymalizując jego działanie i zgodność z dobrymi praktykami programistycznymi [DSB1_K01].
2. Współpracuje w zespołach programistycznych, stosując narzędzia do kontroli wersji i wspólnej edycji kodu [DSB1_K02].
3. Stosuje technologie automatyzacji i skrypty Pythona do usprawniania pracy oraz optymalizacji procesów biznesowych [DSB1_K04].

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Wykład: Dwa kolokwia, za które studenci otrzymują oceny formułujące w postaci punktów - po 50 punktów za każde kolokwium. Ocena końcowa stanowi sumę punktów z dwóch ocen formułujących. Pierwsze kolokwium odbywa się w połowie kursu, a drugie na jego zakończenie. Próg zaliczeniowy to 50 punktów łącznie z obu kolokwiów.

Laboratoria: Dwa kolokwia, za które studenci otrzymują oceny formułujące w postaci punktów - po 50 punktów za każde kolokwium. Ocena końcowa stanowi sumę punktów z dwóch ocen formułujących. Pierwsze kolokwium odbywa się w połowie kursu, a drugie na jego zakończenie. Próg zaliczeniowy to 50 punktów łącznie z obu kolokwiów.

Treści programowe

Kurs obejmuje podstawowe koncepcje programowania oraz wprowadzenie do języka Python, w tym jego składnię, typy danych i operatory. Omawiane są kluczowe struktury danych oraz instrukcje sterujące, takie jak pętle i warunki. Uczestnicy poznają zasady programowania modularnego i funkcyjnego oraz uczą się pracy z plikami i strumieniami wejścia-wyjścia. Kurs obejmuje również wprowadzenie do bibliotek Pythona oraz obsługę API. Dodatkowo poruszane są zagadnienia debugowania, testowania kodu i automatyzacji zadań przy użyciu skryptów Pythona.

Tematyka zajęć

Podstawowe pojęcia związane z programowaniem
Wprowadzenie do Pythona i środowiska programistycznego
Podstawy składni Pythona - zmienne, operatory, typy danych
Struktury danych - listy, krotki, słowniki, zbiory
Instrukcje sterujące - pętle i warunki
Funkcje i programowanie modularne
Obsługa plików i strumieni wejścia-wyjścia
Programowanie funkcyjne - lambda, map, filter, reduce
Zaawansowana obsługa plików - JSON, XML, pickle

Wprowadzenie do bibliotek Pythona
Pandas i Numpy jako kluczowe biblioteki analizy danych
Obsługa API i pobieranie danych z internetu
Programowanie wielowątkowe
Debugowanie i testowanie kodu
Automatyzacja zadań i skrypty Pythona

Metody dydaktyczne

Wykłady: wykład problemowy, prezentacja case studies
Laboratoria: zadania problemowe, studium przypadku, praca w grupach

Literatura

Podstawowa:

Mertz, D. (2024). Lepszy kod w Pythonie, Promise

Matthes, E. (2023). Python. Instrukcje dla programisty, Helion, Gliwice

Uzupełniająca:

Dawson, M. (2023). Python dla każdego. Podstawy programowania, Wydanie 3, Helion, Gliwice.

Downey, A.B. (2025). Myśl w języku Python! Nauka programowania, Helion

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	125	5,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	62	2,50
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	63	2,50