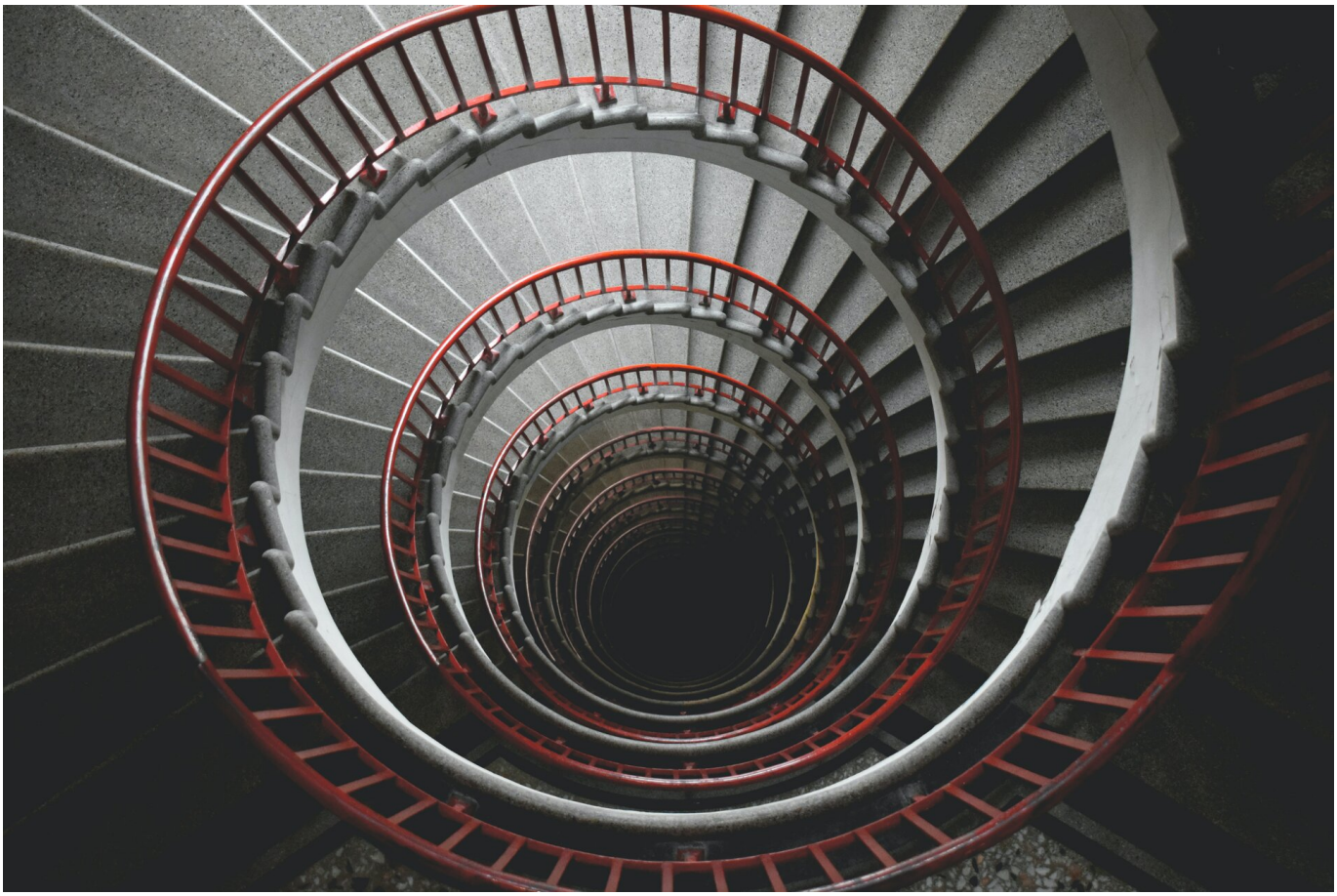
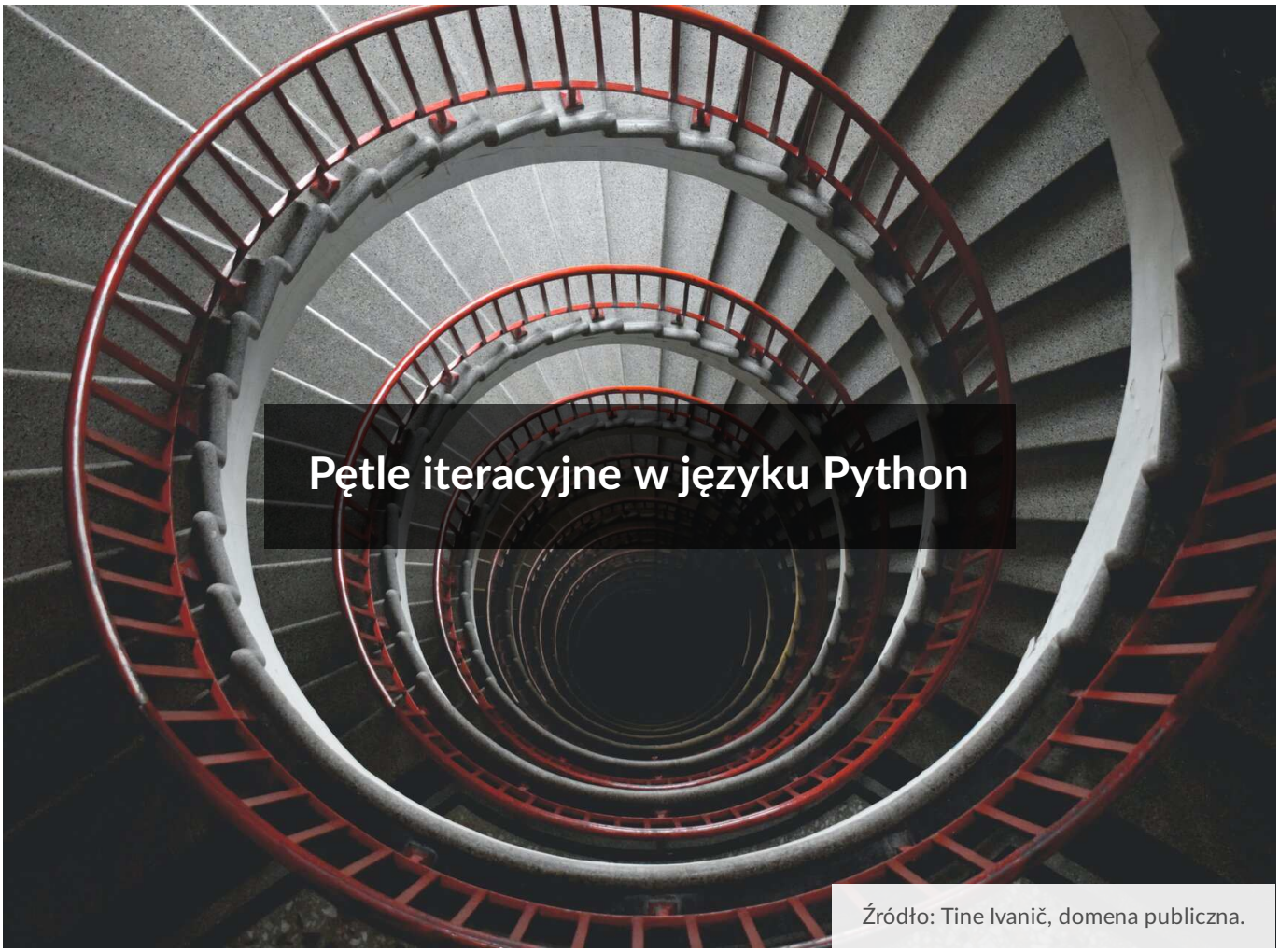


Pętle iteracyjne w języku Python

- Wprowadzenie
- Przeczytaj
- Gra edukacyjna
- Sprawdź się
- Dla nauczyciela



Źródło: Tine Ivanič, domena publiczna.

> W tym e-materiale powtarzamy wiadomości ze szkoły podstawowej.

Poznaliśmy już pojęcie iteracji i algorytmy iteracyjne, omówiliśmy także pętlę `for`. Wiemy już, że wykorzystuje się ją do operowania na kolejnych elementach zbioru lub do wykonania fragmentu kodu określoną ilość razy.

W tym e-materiale omówimy stosowanie pętli `for` w języku Python.

Ciekawi cię, jak wyglądają implementacje w innych językach programowania? Możesz się z nimi zapoznać w dwóch pozostałych e-materiałach z tej serii:

- Pętle iteracyjne w języku C++,
- Pętle iteracyjne w języku Java.

Więcej zadań? Sięgnij do Pętle iteracyjne – zadania maturalne.

> **Twoje cele**
>
> - Poznasz i zrozumiesz składnię i działanie pętli `for` w języku Python.
> - Wykorzystasz pętlę `for` do rozwiązywania prostych problemów.

Przeczytaj

Zmienna wewnętrzna

Pseudokod pętli `for` wygląda następująco:

```
for (zmienna wewnętrzna) in (zbiór):
    instrukcja
```

Zmienna wewnętrzna to nazwa zmiennej, która będzie przechowywała aktualnie analizowany element ze zbioru, który przechodzimy (iterujemy). Jeśli zbiorem jest lista uczniów, to do zmiennej wewnętrznej przy każdej iteracji będzie przypisywany kolejny uczeń.

Zbiorem może być tablica, ciąg znaków lub zakres stworzony za pomocą funkcji `range()`.

> **Ważne!**
>
> Funkcja `range()` tworzy sekwencję liczb w podanym zakresie. Domyślnie pierwszą liczbą jest `0`, a kolejne są domyślnie zwiększane o `1`.
>
> `range(start, stop, krok)`
>
> Na przykład, `range(6)` zwróci zbiór liczb `0, 1, 2, 3, 4, 5`. Funkcja `range(2, 6)` zwróci liczby `2, 3, 4, 5`, a `range(2, 9, 2)` zwróci `2, 4, 6, 8`.

Pętla operująca na danym zbiorze (na przykład imion) przedstawia się tak:

```
uczniowie = ["Maciek", "Magda", "Asia"]

for imie in uczniowie:
    print(imie)
```

Tak zapisana pętla wykona instrukcje wewnątrz niej trzy razy, ponieważ tyle jest elementów w tablicy. Każde takie przejście nazywane jest **iteracją**. Zatem można powiedzieć, że w pierwszej iteracji wypiszemy imię `Maciek`, w kolejnej `Magda` a w ostatniej `Asia`.

Przykłady wykorzystania pętli `for`

Wyliczanka

Napiszmy prosty program drukujący `10` razy ten sam napis.

```
1 for x in range(10):
2     print("Witaj, swiecie!")
```

Zwróćmy uwagę na funkcję `range()`, która jest bardzo często wykorzystywana w pętlach. W powyższym kodzie została wykorzystana skrócona składnia funkcji z jednym argumentem, jakim jest liczba elementów w zbiorze. Zmodyfikujmy powyższy kod, aby przed tekstem: `Witaj, swiecie!`, drukował po kolei numery z zakresu `range()`:

```
1 for x in range(10):
2     print(str(x) + ". Witaj, swiecie!")
```

Wynikiem będzie dziesięciokrotne wypisanie `(numer). Witaj swiecie!`, ale numerowane od `0` do `9`. W programowaniu przyjęło się, że wyliczanie elementów rozpoczyna się od `0`, a nie – jak przywykliśmy – od `1`.

Jeśli jednak chcemy numerować od `1` do `10`, należy zmienić parametry funkcji `range()`, aby rozpoczynała zakres od `1` i kończyła na `11`.

```
1 for x in range(1, 11):
2     print(str(x) + ". Witaj, swiecie!")
```

> **Ważne!**
>
> Do drukowania wyniku wykorzystaliśmy funkcję `str()`, która zamienia typ `integer` na typ `string`. Gdybyśmy chcieli bezpośrednio połączyć liczbę całkowitą z ciągiem znaków, interpreter zwróciłby błąd.

Suma liczb

Napiszmy program liczący sumę liczb z podanego zakresu. W tym celu zdefiniujemy zakres od `1` do `10`, a także zmienną `suma`.

```
1 suma = 0
2
3 for x in range(1, 11):
4     suma = suma + x
5
```

```
6 print(suma)
```

W trakcie każdej iteracji kolejna liczba jest dodawana do sumy. Po zakończeniu pętli, drukujemy wynik.

Słownik

> **tablica**
>
> zbiór elementów, w którym można przechowywać dane; w języku Python tablice są często nazywane listami
> **funkcja**
>
> wydzielona część programu, która przetwarza dane i ewentualnie zwraca na ich podstawie pewną wartość; przykładową funkcją w języku Python jest `print()`

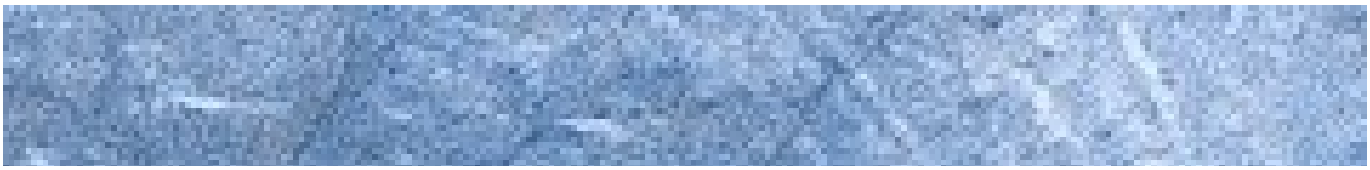
Gra edukacyjna

Polecenie 1

Przetestuj zaprezentowany algorytm zliczania sumy elementów z listy. Następnie zbuduj algorytm, w którym lista będzie zawierała kilka imion, a algorytm zliczy sumę liter we wszystkich imionach.

Polecenie 2

Rozwiąż quiz dotyczący różnych zastosowań pętli `for` w języku Python. Sprawdź swoją wiedzę.



Test

Pętle iteracyjne w języku Python --- quiz

Poziom trudności:

łatwy

Limit czasu:

5 min

Twój ostatni wynik:

-

Uruchom

Polecenie 3

Zapisz algorytm z **polecenia 1**, używając listy kroków.

Polecenie 4

Zaproponuj trzy inne pytania, które mogłyby pojawić się w teście sprawdzającym wiedzę.

Sprawdź się

Pokaż ćwiczenia:

Ćwiczenie 1

Napisz program, który wypisze po kolei imiona uczniów znajdujących się w zbiorze `uczniowie`, a zwróci liczbę uczniów. W tym celu dodaj instrukcje składające się na funkcję `wypisz(uczniowie)`. Wykorzystaj pętlę `for` — przy każdym przebiegu pętli zwiększaj licznik.

Twoje zadania

1. Wypisz imiona uczniów
2. Zwróć liczbę uczniów

Dla nauczyciela

Autor: Jakub Kamiński
Przedmiot: Informatyka

Temat: Pętle iteracyjne w języku Python

Grupa docelowa:

Liceum ogólnokształcące i technikum, liceum ogólnokształcące, technikum

Podstawa programowa:

Cele kształcenia – wymagania ogólne

> II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera oraz innych urządzeń cyfrowych: układanie i programowanie algorytmów, organizowanie, wyszukiwanie i udostępnianie informacji, posługiwanie się aplikacjami komputerowymi.

Treści nauczania – wymagania szczegółowe

> II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych.
>
> > Zakres rozszerzony. Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:
> >
> > > 2) stosuje zasady programowania strukturalnego i obiektowego w rozwiązywaniu problemów;

Kształtowane kompetencje kluczowe:

- kompetencje cyfrowe;
- kompetencje osobiste, społeczne i w zakresie umiejętności uczenia się;
- kompetencje matematyczne oraz kompetencje w zakresie nauk przyrodniczych, technologii i inżynierii.

Cele operacyjne (językiem ucznia):

- Poznasz i zrozumiesz składnię i działanie pętli `for` w języku Python.
- Wykorzystasz pętlę `for` do rozwiązywania prostych problemów.

Strategie nauczania:

- konstruktywizm;

- konektywizm.

Metody i techniki nauczania:

- dyskusja;
- rozmowa nauczająca z wykorzystaniem multimedium i ćwiczeń interaktywnych;
- ćwiczenia praktyczne.

Formy pracy:

- praca indywidualna;
- praca w parach;
- praca w grupach;
- praca całego zespołu klasowego.

Środki dydaktyczne:

- komputery z głośnikami, słuchawkami i dostępem do internetu;
- zasoby multimedialne zawarte w e-materiale;
- tablica interaktywna/tablica, pisak/kreda.

Przebieg lekcji

Przed lekcją:

1. **Przygotowanie do zajęć.** Nauczyciel loguje się na platformie i udostępnia e-materiał: „Pętle iteracyjne w języku Python". Nauczyciel prosi uczniów o zapoznanie się z treściami w sekcji „Przeczytaj".

Faza wstępna:

1. Przedstawienie tematu zajęć oraz wspólne z uczniami ustalenie kryteriów sukcesu.

Faza realizacyjna:

1. **Praca z tekstem.** Uczniowie przystępują do cichego czytania tekstu zawartego w sekcji „Przeczytaj", jeśli nauczyciel - na podstawie raportu na platformie - uważa, że przygotowanie uczniów jest wystarczające, może pominąć tę czynność.
2. **Praca z multimedium.** Nauczyciel wyświetla zawartość sekcji „Gra edukacyjna". Uczniowie testują zaprezentowany algorytm zliczania sumy elementów z listy. Następnie samodzielnie budują algorytm, w którym lista będzie zawierała kilka imion, a algorytm zliczy sumę liter we wszystkich imionach.
3. **Ćwiczenie umiejętności.** Uczniowie realizują indywidualnie ćwiczenia nr 1, po ich wykonaniu porównują otrzymane wyniki z inną osobą.

Faza podsumowująca:

1. Nauczyciel zadaje pytania podsumowujące, np.
    - kiedy używamy pętli `for`?
    - w jakim celu używamy pętli `for`?
    - podaj przykład algorytmu z zastosowaniem pętli `for`.

Praca domowa:

1. Uczniowie wykonują polecenie nr 2 z sekcji „Gra edukacyjna”.

Materiały pomocnicze:

- Oficjalna dokumentacja techniczna dla języka Python 3 (lub nowszej wersji).
- Oficjalna dokumentacja techniczna dla oprogramowania PyCharm lub IDLE.

Wskazówki metodyczne:

- Multimedia w sekcjach: „Przeczytaj”, „Gra edukacyjna”, „Sprawdź się” można wykorzystać jako materiał służący powtórzeniu materiału.