

Wstęp do programowania w języku wysokiego poziomu PYTHON

Aby w języku Python wyświetlać na ekranie komunikaty i wyniki, stosujemy instrukcję wyjścia – funkcję `print()`. Argumentami funkcji (podanymi w nawiasach) mogą być teksty, wyrażenia arytmetyczne lub zmienne oddzielone przecinkami.

Przykłady	Wyniki
<code>print("Informatyka")</code>	komputer wyświetla na ekranie napis „Informatyka”
<code>print(23 + 17)</code>	komputer wyświetla na ekranie wartość wyrażenia $23 + 17$, czyli 40
<code>print(wzrost)</code>	komputer wyświetla na ekranie wartość zmiennej <code>wzrost</code>
<code>print("Suma liczb 345 i 36 wynosi:", 345 + 36)</code>	komputer wyświetla na ekranie napis „Suma liczb 345 i 36 wynosi:” i w tym samym wierszu obliczoną wartość wyrażenia (czyli 381)

Tabela 2. Przykłady stosowania funkcji `print()`

Zadanie 1:

Wprowadź powyższe przykłady do swojego pliku w aplikacji IDLE. UWAŻAJ na pisownię.

Zadanie 2:

Zapisz samodzielnie kod który zwróci wynik: „Wynik dzielenia liczb 40 i 4 wynosi 10”.

Zadanie 3:

W szóstej linijce swojego kodu otrzymaj zapis: „Aktualnie pracuję na lekcji informatyki w trybie skryptowym”.

Czym jest zmienna?

Zmienna w programie komputerowym to nazwana część pamięci operacyjnej komputera (RAM) o unikatowym adresie – **komórka (komórki) pamięci**. W każdym języku programowania musimy określić sposób korzystania ze zmiennych i nadawania im wartości.

Zmienna ma zawsze jakąś wartość i określone miejsce w pamięci komputera (adres komórki pamięci). Jeśli w tym samym programie zapiszemy w tej samej zmiennej nową wartość (wykonamy tzw. instrukcję przypisania), to poprzednia wartość znika.

Komórka pamięci

Część pamięci operacyjnej komputera (RAM) o unikatowym adresie. W komórkach pamięci przechowuje się dane reprezentujące instrukcje procesora lub dane dla tych instrukcji.

W języku Python w nazwach zmiennych można używać tylko liter, znaku podkreślenia i cyfr. Nazwa nie może zaczynać się od cyfry. Przyjęte jest, że w nazwach zmiennych nie stosuje się liter z polskimi znakami diakrytycznymi. Należy nadawać nazwy określające znaczenie danej zmiennej. Małe i wielkie litery w nazwach zmiennych (także instrukcji) mają różne znaczenie, np. zmienne `Suma` i `suma` to dwie różne zmienne.

Zmiennej możemy w programie nadać konkretną wartość za pomocą instrukcji przypisania. Znak (operator) `=` oznacza przypisanie zmiennej zapisanej po lewej stronie znaku wartości umieszczonej po jego prawej stronie.

Przykład	Opis
<code>rok = 2021</code>	zmiennej <code>rok</code> przypisujemy wartość, która jest liczbą całkowitą, czyli zmienna <code>rok</code> jest typu całkowitego (<code>int</code>)
<code>wzrost = 159.5</code>	zmiennej <code>wzrost</code> przypisujemy wartość, która jest liczbą rzeczywistą, czyli zmienna <code>wzrost</code> jest typu rzeczywistego (<code>float</code>)
<code>nazwisko = "Kowalski"</code>	zmiennej <code>nazwisko</code> przypisujemy tekst (ciąg znaków), czyli zmienna <code>nazwisko</code> jest typu tekstowego (<code>str</code>)
<code>rok = rok + 1</code>	zmiennej <code>rok</code> przypisujemy wartość zmiennej <code>rok</code> zwiększoną o 1
<code>iloczyn = a * b</code>	zmiennej <code>iloczyn</code> przypisujemy iloczyn wartości pamiętanych w zmiennych <code>a</code> i <code>b</code>

Tabela 1. Przykłady przypisywania zmiennej wartości w języku Python



W większości języków programowania (w tym w języku Python) część dziesiętną liczby zapisujemy po kropce (a nie po przecinku, jak w matematyce), np. 34.5, a nie 34,5.

Zmiennej możemy nadać wartość za pomocą instrukcji przypisania, wprowadzając wartość z klawiatury po uruchomieniu programu. W tym celu stosujemy instrukcję wejścia – funkcję `input()`.

Argumentem funkcji (podanym w nawiasach) jest tekst (ciąg znaków) zapisany w apostrofach, który pojawi się na ekranie jako swego rodzaju „zaproszenie” do wpisania ciągu znaków z klawiatury. Jeśli użyjemy funkcji `input()` w instrukcji przypisania, wpisany przez użytkownika ciąg znaków komputer zapamięta w zmiennej podanej po lewej stronie:

```
nazwa_zmiennej = input("Zaproszenie do wprowadzenia danej z klawiatury: ")
```

Aby komputer mógł zapamiętać wprowadzoną za pomocą instrukcji `input()`

W obliczeniach z użyciem zmiennych używamy operatorów arytmetycznych zapisanych w tabeli 2.

Operator	Działanie	Przykład	Wynik
+	dodawanie	23 + 56	79
-	odejmowanie	987 - 233	754
*	mnożenie	432 * 6	2592
//	dzielenie całkowite (z obcięciem części ułamkowej)	55 // 3	18
/	dzielenie zmiennoprzecinkowe (z zachowaniem części ułamkowej)	55 / 3	18.333333333333332
%	obliczenie reszty z dzielenia dwóch liczb całkowitych	37 % 4	1

Tabela 2. Podstawowe operatory arytmetyczne w języku Python

Przykład	Opis
<code>nazwisko = input("Wprowadź nazwisko: ")</code>	jeśli wpiszemy z klawiatury nazwisko „Kowalski”, komputer w zmiennej <code>nazwisko</code> zapamięta ciąg znaków „Kowalski”
<code>a = input("Wprowadź liczbę: ")</code>	jeśli wpiszemy z klawiatury liczbę 346, komputer w zmiennej <code>a</code> zapamięta ciąg znaków „346”, a nie liczbę 346
<code>a = int(input("Wprowadź liczbę: "))</code>	jeśli wpiszemy z klawiatury liczbę 346, komputer w zmiennej <code>a</code> zapamięta liczbę całkowitą 346
<code>srednia = float(input("Podaj średnią ocen: "))</code>	jeśli wpiszemy z klawiatury liczbę 4.3, komputer w zmiennej <code>srednia</code> zapamięta 4,3

Tabela 3. Przykłady wprowadzania z klawiatury wartości zmiennej w języku Python

Przykład zastosowania zmiennych w obliczeniach matematycznych:

```
a = int(input("Podaj liczbę: "))  
b = int(input("Podaj liczbę: "))  
suma = a + b  
print("Suma wynosi:", suma)
```

Zadanie 4: Wprowadź i przetestuj kod powyższego programu. UWAŻAJ na pisownię.

Zadanie 5. Przygotuj program wykonujący odejmowanie na zmiennych liczb x i y

Zadanie 6. Przygotuj program obliczający resztę z dzielenia dowolnych liczb całkowitych dla zmiennych liczb R i G

Zadanie 7. Przygotuj program obliczający wynik dzielenia całkowitego dwóch liczb dla zmiennych d i f