

Python

ćwiczenia

```
position:absolute;z-index:999;copy
px 5px #ccc}.gbrtl .gbm{-moz-bo
;display:block;position:absolute
opacity:1;*top:-2px;*left:-5px;
/;top:-4px\0/;left:-6px\0/;rig
ne-box;display:inline-block;fo
display:block;list-style:none;
ne-block;line-height:27px; padd
pointer;display:block;text-de
ative;z-index:1000}.gbtm{*disp
padding-right:9px)#gbz .gbzt,
```

Typy danych

W Pythonie mamy kilka podstawowych typów danych:

- Liczby całkowite (**int**): np. `10`, `-5`, `1000`
- Liczby zmiennoprzecinkowe (**float**): np. `3.14`, `-0.5`
- Łańcuchy znaków (**str**): np. `"Python"`, `"Jestem uczniem"`
- Typ logiczny (**bool**): np. `True`, `False`

ĆWICZENIE 1

```
liczba = 5
```

```
pi = 3.14
```

```
tekst = "Cześć, Python!"
```

```
flaga = True
```

```
print(liczba, pi, tekst, flaga)
```

Operatory

Operatory służą do wykonywania operacji na danych. W Pythonie mamy różne rodzaje operatorów:

- Operatory arytmetyczne: ``+``, ``-``, ``*``, ``/``, ``//`` (dzielenie całkowite), ``%`` (reszta z dzielenia), ``**`` (potęgowanie)
- Operatory porównania: ``==``, ``!=``, ``<``, ``>``, ``<=``, ``>=``
- Operatory logiczne: ``and``, ``or``, ``not``

ĆWICZENIE 2

Napisz prosty program, który obliczy pole prostokąta. Wprowadź dwie zmienne: długość i szerokość prostokąta, a następnie oblicz pole.

```
dlugosc = 5
```

```
szerokosc = 3
```

```
pole = dlugosc * szerokosc
```

```
print("Pole prostokąta wynosi:", pole)
```

ĆWICZENIE 3. *Napisz program, który sprawdzi, czy liczba zapisana w zmiennej `x` jest większa od 10. Jeśli tak, wyświetli komunikat „Liczba jest większa od 10”, w przeciwnym razie wyświetli „Liczba jest mniejsza lub równa 10*

```
x = float(input("Podaj liczbę: "))  
  
# Sprawdzanie, czy liczba jest większa od 10  
  
if x > 10:  
    print("Liczba jest większa od 10")  
else:  
    print("Liczba jest mniejsza lub równa 10")
```

ĆWICZENIE 4. *Napisz program, który wypisuje liczby od 1 do 10 (włącznie) za pomocą pętli `for`.*

Pętla for wypisująca liczby od 1 do 10

```
for i in range(1, 11):  
    print(i)
```

Instrukcja warunkowa (if)

Instrukcja `if` służy do wykonywania kodu tylko wtedy, gdy warunek jest spełniony.

ĆWICZENIE 5.

```
wiek=int(input("podaj wiek "))
```

```
if wiek >= 18:
```

```
    print("Jesteś pełnoletni.")
```

```
else:
```

```
    print("Nie jesteś pełnoletni.")
```


PODZIELNOŚĆ

LICZB

Operator **%** w języku Python nazywany jest **operatorem reszty z dzielenia** lub **operatorem modulo**. Jego głównym zadaniem jest zwrócenie reszty z dzielenia jednej liczby przez drugą.

Składnia:

$a \% b$

- **a** – liczba dzielona (dzielna)
- **b** – dzielnik (liczba przez którą dzielimy)

W wyniku działania operatora % otrzymujemy resztę z dzielenia liczby a przez liczbę b.

Przykład:

7 % 3

Działanie:

- 7 podzielić przez 3 daje 2 jako część całkowitą (bo $7 / 3 = 2$), a reszta to 1 (bo $7 - 2 * 3 = 1$).
- **Wynik: 7 % 3 = 1.**

Przykłady:

Przykład 1: Podzielność przez 2

$$10 \% 2$$

- 10 podzielone przez 2 daje 5 i resztę 0.
- Wynik: $10 \% 2 = 0$.

Ponieważ reszta wynosi 0, oznacza to, że 10 jest liczbą **podzielną przez 2**.

Przykład 2: Podzielność przez 3

$$15 \% 3$$

- 15 podzielone przez 3 daje 5 i resztę 0.
- Wynik: $15 \% 3 = 0$.

Ponieważ reszta wynosi 0, oznacza to, że 15 jest liczbą **podzielną przez 3**.

Pytanie? Czy liczba 22 jest podzielna przez 5?

22 % 5

- 22 podzielone przez 5 daje 4 i resztę 2 (bo $22 - 4 * 5 = 2$).
- Wynik: $22 \% 5 = 2$.

Ponieważ reszta wynosi 2, oznacza to, że 22 nie jest liczbą **podzielną przez 5**.

Ćwiczenie: Czy podana liczba 11 jest nieparzysta?

11 % 2

- 11 podzielone przez 2 daje 5 i resztę 1.
- Wynik: $11 \% 2 = 1$.

Ponieważ reszta wynosi 1, oznacza to, że 11 jest liczbą **nieparzystą**.

PODSUMOWANIE - PODZIELNOŚĆ LICZB

zastosowanie operatora %:

1.Sprawdzanie podzielności:

1.Jeśli $a \% b == 0$, to liczba a jest podzielna przez b .

2.Określanie parzystości i nieparzystości:

1.Jeśli $a \% 2 == 0$, to liczba a jest parzysta.

2.Jeśli $a \% 2 != 0$, to liczba a jest nieparzysta.

3.Iteracja przez liczby:

1.Można używać operatora % w pętlach, aby wybrać liczby spełniające określony warunek, np. liczby podzielne przez 3 w danym zakresie.

Ćwiczenie 1:

Sprawdzenie, czy liczba jest podzielna przez 2

Napisz program, który sprawdza, czy podana liczba jest podzielna przez 2. Jeśli tak, wypisz „Liczba jest podzielna przez 2”, w przeciwnym przypadku wypisz „Liczba nie jest podzielna przez 2”.

```
liczba = int(input("Podaj liczbę: "))
```

```
if liczba % 2 == 0:
```

```
    print("Liczba jest podzielna przez 2")
```

```
else:
```

```
    print("Liczba nie jest podzielna przez 2")
```

Ćwiczenie 2: Wyświetlanie liczb podzielnych przez 2 w zakresie

Napisz program, który wypisuje wszystkie liczby podzielne przez 2 w zakresie od 1 do 20.

```
for i in range(1, 21):  
    if i % 2 == 0:  
        print(i)
```

Ćwiczenie 3:

Sprawdzanie liczby parzystej lub nieparzystej

Napisz program, który sprawdza, czy liczba jest parzysta, czy nieparzysta. Użyj instrukcji warunkowej, aby wypisać odpowiedni komunikat.

```
liczba = int(input("Podaj liczbę: "))  
if liczba % 2 == 0:  
    print("Liczba jest parzysta")  
else:  
    print("Liczba jest nieparzysta")
```


ZADANIA Z PODRĘCZNIKA :

1. Ćwiczenie 6/58
2. Ćwiczenie 11/63