

Pętla while, for

Pętla **while** wykonuje instrukcję lub zestaw instrukcji, o ile określony warunek logiczny jest prawdziwy, np.:

```
while True:
    print("Pętla nieskończona")

licznik = 5
while licznik > 2:
    print(licznik)
    licznik -= 1
```

Pętla **for** wielokrotnie wykonuje zestaw instrukcji; służy do iteracji po typach sekwencyjnych (np. lista, słownik, krotka, lub zestaw lub innych obiektów, które są **iterowalne** (np. łańcuchy znaków).

```
slovo = "Python"
for litera in slovo:
    print(litera, end="")
```

Możesz użyć instrukcji **break** oraz **continue** do zmiany przepływu pętli:

- Instrukcja **break** służy do wyjścia z pętli, np.:

```
tekst = "Witam na kursie Pythona"
for litera in tekst:
    if litera == "P":
        break
    print(litera, end="")
```

- Instrukcja **continue** służy do pominięcia bieżącej iteracji i kontynuowania następnej iteracji, np.:

```
tekst = "to jest ten tekst "
for litera in tekst:
    if litera == " ":
        continue
    print(litera, end="")
```

Pętle **while** oraz **for** mogą mieć także gałąź **else**. Gałąź **else** jest wykonywana po zakończeniu wykonywania pętli, o ile nie została przerwana przez instrukcję **break**. Gałąź pętli **else** jest zawsze wykonywana raz, niezależnie od tego, czy pętla została wprowadzona do jej ciała, czy nie.

```
zm1=0
while zm1 != 33:
    zm1=int(input("podaj magiczną liczbę :"))
    if zm1==99:
        continue
    print(zm1)
    if zm1==66:
        break
else:
    print("to jest sekcja 'else'")

print("Koniec petli")
```

Funkcja **range()** generuje ciąg liczb. Akceptuje liczby całkowite i zwraca obiekty zdefiniowanego zakresu. Składnia funkcji **range()** wygląda następująco: **range** (start, stop, step), gdzie:

- **start** jest opcjonalnym parametrem określającym numer początkowy sekwencji (domyślnie **0**)
- **stop** jest opcjonalnym parametrem określającym koniec wygenerowanej sekwencji (**nie jest uwzględniony**),
- **step** jest opcjonalnym parametrem określającym różnicę między poszczególnymi liczbami w sekwencji (domyślnie **1**.)

```
for i in range(100):  
    print(i)
```

```
for i in range(1, 20):  
    if i % 2 == 0:  
        print(i)
```

Zadanie 1

a)

Stwórz pętlę **for**, która liczy od 0 do 10 i wyświetla nieparzyste liczby na ekranie. Użyj szkieletu kodu poniżej:

```
for i in range(1, 11):  
    # linie kodu
```

b)

Stwórz pętlę **while**, która liczy od 0 do 10 i wyświetla nieparzyste liczby na ekranie. Użyj szkieletu kodu poniżej:

```
x = 1  
while x < 11:  
    # linia kodu
```

c)

Stwórz program zawierający pętlę **for** oraz instrukcję **break**. Program powinien iterować przez znaki w adresie e-mail; wychodzić z pętli, gdy natrafi na symbol **@** i wyświetlać w jednej linii tę część adresu, która znajduje się przed **@**. Użyj szkieletu kodu poniżej:

```
for ch in "mariusz.kowalski@organizacja.com.pl":  
    if ch == "@":  
        # linia kodu
```

d)

Stwórz program zawierający pętlę **for** oraz instrukcję **continue**. Program powinien iterować przez ciąg znaków składający się z cyfr, zamienić każde 0 na x i wyświetlić zmodyfikowany ciąg znaków na ekranie. Użyj szkieletu kodu poniżej:

```
for cyfra in "0165031806510":  
    if cyfra == "0":  
        # linia kodu
```