

Pętle

Podstawy programowania w języku Python



Pętle

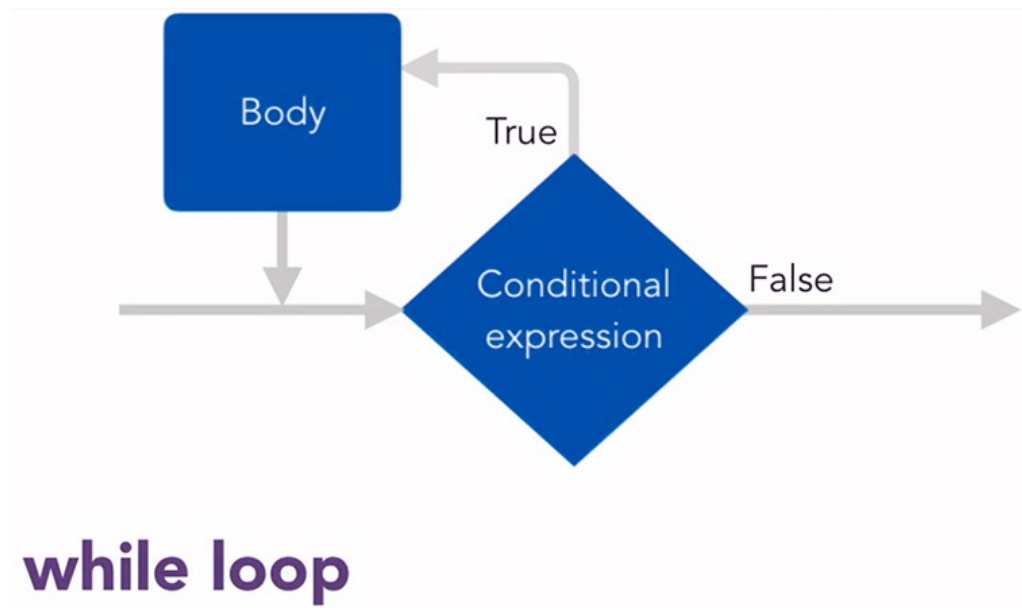
To sekwencje instrukcji, które są powtarzane.

Mogą być powtarzane:

- **aż do osiągnięcia przez program określonego warunku**
Np. program może prosić użytkownika o podanie hasła wyświetlając napis "Podaj hasło".
Napis będzie wyświetlany dotąd, aż użytkownik poda odpowiednie hasło.
- **określoną ilość razy**
Np. program może wyświetlić 3 razy napis "Uwaga".

Pętla "while"

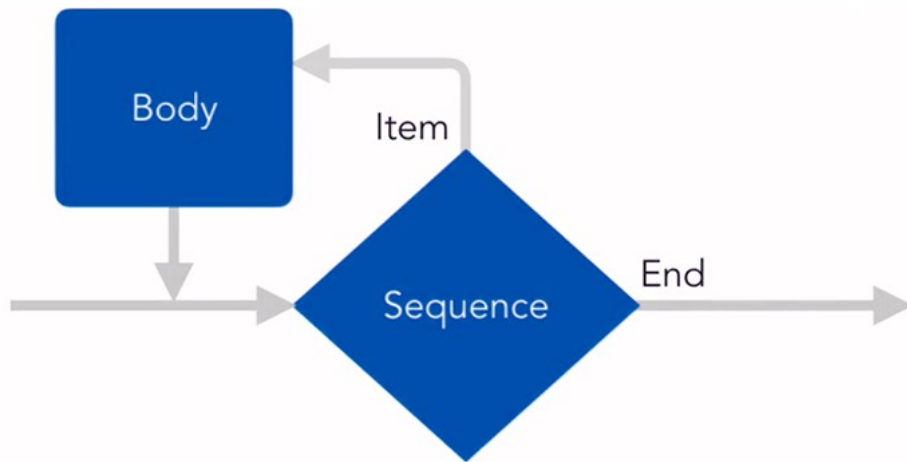
- wykonuje się dopóki spełniony jest pewien warunek logiczny



```
1 # Wypisze 0 1 2 3 4
2 counter = 0
3 while counter < 5:
4     print(counter, end=" ")
5     counter += 1
```

Pętla "for"

- służy w Pythonie do iteracji po sekwencji (np. lista, krotka tuple, łańcuch string) lub innych iterowalnych obiektach



```
1 for i in range(0, 5):  
2     print(i)
```

for loop

Funkcja range()

- zwraca szczególny typ obiektu jakim jest generator
- generator tworzy liczby całkowite
- możemy wykorzystać go w pętli

```
1 for i in range(0, 6, 2):  
2     print(i, end=" ")  
3 #wynik: 0 2 4
```

Funkcja range() posiada 3 argumenty: wartość początkowa, wartość końcowa, krok o jaki zmienia się wartość z każdym obrotem pętli. Jeśli zmiana następuje co 1, nie musimy podawać trzeciego argumentu.

Instrukcje "break" i "continue"

- za pomocą instrukcji **break** wychodzimy z najbliższej pętli while lub for

```
1  for i in range(0, 10):  
2      print(i, end=" ")  
3      if (i >= 3):  
4          break  
5  #wynik: 0 1 2 3
```

- za pomocą instrukcji **continue** przechodzimy do następnego kroku w pętli

```
1  for i in range(0, 10):  
2      if (i % 2 == 0):  
3          continue  
4      print(i, end=" ")  
5  #wynik: 1 3 5 7 9
```

Pętle z gałęzią "else"

- za blokiem kodu zawierającym pętlę możemy użyć instrukcji **else**
- kod zawarty po klauzuli **else** jest wykonywany zawsze po normalnym zakończeniu działania pętli
- wyjątkiem niewykonania się kodu po **else** jest wyjście z pętli za pomocą **break** lub **return**

```
1  for i in range(5):  
2      print(i, end=" ")  
3  else:  
4      print("koniec")  
5  #wynik: 0 1 2 3 4 koniec
```

Pytanie

Jakiego polecenia możemy użyć w pętli **for**, aby przeskoczyć do kolejnej iteracji bez wykonywania reszty instrukcji?

- a) break
- b) continue
- c) return

Odpowiedź: b)

Pytanie

Co będzie wynikiem uruchomienia poniższego skryptu?

- a) 5 10 2
- b) 5 6 7 8 9
- c) 5 7 9
- d) wystąpi błąd

```
1 for i in range(5, 10, 2):  
2     print(i, end=" ")
```

Odpowiedź: c)
Iterowanie od 5 do 9, ale co 2.

Pytanie

Co się stanie, gdy użyjemy polecenia **break** w pętli **while**?

- a) pętla zakończy działanie
- b) przeskoczy do kolejnej iteracji
- c) nic się nie stanie

Odpowiedź: a)