

# Wstęp do języka Python

## Praktyczne ćwiczenia dla studentów 1

### Zadanie 1-1 Instalacja Python'a

Odwiedź stronę <https://www.anaconda.com/download/> i pobierz dystrybucję Anaconda dla wersji Pythona 3.8. Zapoznaj się z notatnikiem jupyter Python, który jest dołączony do instalacji anakondy. Jeśli wolisz, zainstaluj również wybrany IDE/edytor (PyCharm, VS Code, Spyder). Upewnij się, że masz skonfigurowaną i uruchomioną konfigurację Pythona dla samouczków.

### Zadanie 1-2 Listy

Utwórz listę zawierającą wszystkie liczby od 0 do 10.

### Zadanie 1-3 Pętle i tryby warunkowe

Korzystając z utworzonej listy z zadania 1-2, wydrukuj zawarte w niej liczby nieparzyste, używając pętli i warunków. Spróbuj użyć innego rodzaju pętli.

### Zadanie 1-4 Odwzorowanie listy

Teraz wygeneruj listę zawierającą wszystkie liczby od 0 do n, które zostały podniesione do kwadratu za pomocą odwzorowanych list.

### Zadanie 1-5 Funkcje

Napisz funkcję, która przyjmuje liczbę całkowitą n. Funkcja najpierw tworzy listę liczb od 0 do n. Następnie podnieś do kwadratu każdy numer na liście. Następnie każdy z kwadratów jest sprawdzana, czy wynik jest nieparzysty. Wszystkie liczby nieparzyste są następnie dołączane do nowej listy. Funkcja zwraca listę liczb nieparzystych (do kwadratu).

### Zadanie 1-6 Przypisanie

Biorąc pod uwagę listę a = [„ja”, „lubie”, „ciasteczka”] i inną listę b = a. Zastąp na liście b słowo „cookies” słowem „jabłka”. Na koniec wydrukuj obie listy (a i b). Co otrzymasz?

### Zadanie 1-7 Płytką kopia I

Biorąc pod uwagę listę  $a = [„I”, „lubię”, „ciasteczka”]$  i inną listę, która przyjmuje płytką kopię  $a$ ,  $b = a[:]$ . Podobnie jak w poprzednim zadaniu 1-6, zamień na liście  $b$  słowo „cookies” na „jabłka”. Na koniec wydrukuj obie listy ( $a$  i  $b$ ). Co teraz obserwujesz?

### Zadanie 1-8 Płytką kopia II

Teraz otrzymujemy listę  $a = [„ja”, „lubię”, [„czekolada”, „ciasteczka”]]$  oraz inną listę, która przyjmuje płytką kopię  $a$ ,  $b = a[:]$ . Zmień teraz słowo „ciasteczka” na „jabłka” w  $b$ . Teraz wydrukuj ponownie obie listy ( $a$  i  $b$ ). Co obserwujesz?

### Zadanie 1-9 Głęboka kopia

Podobnie jak w poprzednim zadaniu, otrzymujemy listę  $a = [„ja”, „lubię”, [„czekolada”, „ciasteczka”]]$ . Inna lista  $b = \text{deepcopy}(a)$  przyjmuje tym razem głęboką kopię z  $a$ . Zmień teraz słowo „ciasteczka” na „jabłka” w  $b$ . Wydrukuj obie listy ( $a$  i  $b$ ). Co teraz obserwujesz?

*Wskazówka: W przypadku głębokiej kopii najpierw wpisz: **from copy import deepcopy**.*

### Zadanie 1-10 Słowniki I

Utwórz słownik z  $n$  wpisami, w którym klucze są wyliczane od 0 do  $n - 1$ , a wartości są odpowiadającymi im kluczami do kwadratu. Używaj odwzorowania list.

Przykład oczekiwanego wyniku:  $n = 7$ ;  $\{0:0, 1:1, 2:4, 3:9, 4:16, 5:25, 6:36\}$

### Zadanie 1-11 Słowniki II

Użyj słownika z poprzedniego zadania. Napisz **odwzorowaną listę**, aby otrzymać listę wszystkich kluczy słownika.

### Zadanie 1-12 Funkcje Lambda

Write a list comprehension which takes a number  $n$  and returns a list with even numbers, using a lambda function.

### Zadanie 1-13 map

Najpierw napisz funkcję, która pobiera długość w calach i zwraca długość w cm. Biorąc pod uwagę listę  $l$  z długościami w calach:

$l = [4, 4.5, 5, 5.5, 6, 7]$

Utwórz listę odwzorowania, który przyjmuje  $l$  i zwraca listę ze wszystkimi wartościami przekonwertowanymi na cm za pomocą `map`.

### Zadanie 1-14 filter

Utwórz listę odwzorowania, który filtruje listę  $l$  z powyższego zadania, zwracając tylko rozmiary od 4 do 6 cali.

### **Zadanie 1-15 reduce**

Utwórz listę odwzorowania, która redukuje listę 1 przez zsumowanie wszystkich długości.

*Wskazówka: aby skorzystać z **funkcji** reduce, należy ją zaimportować: z `functools import reduce`*