

Ćwiczenia 3

Kontynuacja wprowadzenia do języka Python

1. (`caa.py`) Stwórz nowy plik o podanej nazwie i wykonaj w nim poniższe czynności:
 - stwórz krotkę o 6 elementach różnych typów
 - spróbuj zmienić element o indeksie 2
 - zakomentuj z powyższego podpunktu
 - wyświetl na konsoli element krotki o indeksie 3
 - wyświetl na konsoli element piąty od końca
2. (`cab.py`) Stwórz nowy plik o podanej nazwie i wykonaj w nim poniższe czynności:
 - stwórz zbiór (`set`) z 5 elementami typu `string`
 - spróbuj dodać do zbioru istniejący element w zbiorze (jeśli się nie udaje, zakomentuj kod)
 - sprawdź i jeśli to możliwe wykonaj operacje z zadania 1 dla zbioru.
3. (`cac.py`) Stwórz słownik, który będzie przechowywał jako klucz imiona, a jako wartość jakieś liczby (odpowiadające numerom telefonom). Następnie wykonaj kilka operacji na słowniku.
4. (`cad.py`) Napisz program, który na wejściu od użytkownika będzie pobierał liczbę w systemie rzymskim, a na wyjściu wyświetli na konsoli wartość liczby w systemie arabskim (program wystarczy zrobić dla liczb całkowitych z zakresu 1-100).
5. (`cae.py`) Napisz program, który zamieni wprowadzony przez użytkownika ciąg cyfr na formę tekstową. Znaki nie będące cyframi mają być ignorowane. Wykorzystaj w programie słownik.

Przykładowe wejście:

```
45j2
```

Przykładowe wyjście:

```
cztery pięć dwa
```

6. (`caf.py`) Stwórz program, w którym powstaną 3 zbiory przechowujące nazwy polskich miast. Następnie wyświetl na konsoli:
 - miasta, które są w przynajmniej jednym ze zbiorów (odpowiednik sumy zbiorów),
 - miasta, które są jednocześnie we wszystkich zbiorach (odpowiednik części wspólnej zbiorów),
 - miasta, które są w pierwszym zbiorze, ale nie ma ich w drugim zbiorze (odpowiednik różnicy zbiorów).

W tym punkcie nie korzystaj z instrukcji warunkowych, a wykorzystaj operacje na zbiorach.

7. (`cag.py`) Stwórz program, który od użytkownika pobiera sekwencję znaków a na wyjściu sprawdza czy znaki stanowią numer PESEL i jeśli numer jest poprawny, to wyświetli datę urodzenia i płeć. Pomoc: <https://pl.wikipedia.org/wiki/PESEL>. W programie uwzględnij sytuację:
 - jeśli są znaki nie będące cyframi, ma być info o błędnym formacie.
 - jeśli długość wprowadzonego stringu przez użytkownika jest dłuższa lub krótsza, też ma być info o błędnym formacie.
 - uwzględnij wszystkie możliwe lata urodzeń od 1800 do 2299.

Ostatnia aktualizacja pliku: 2019-03-07 16:19:46.