

Zadanie 1. Utwórz moduł o nazwie np. "text" zawierający trzy funkcje:

1. Wypisującą tekst wyrównany do lewej strony ekranu, przyjmującą jako argument maksymalną liczbę znaków mieszczącą się w linii,
2. Wypisującą tekst wyrównany do prawej strony ekranu, przyjmującą argument taki, jak w funkcji pierwszej,
3. Wypisującą tekst wyrównany do środka ekranu, przyjmującą argument taki, jak w funkcji pierwszej.

Napisz program importujący ten moduł, wypisujący tekst na kształt rozmowy dwóch osób w komunikatorze: wycentrowany nagłówek "Rozmowa z: Mietek", wyrównane do lewej wiadomości Mietka, wyrównane do prawej wiadomości użytkownika.

```
>>>
                                     Rozmowa z: Mietek
Mietek: Witaj, mój ulubiony kolego!
                                     Ja: Witaj, nieznajomy
                                     Ja: Jak mogę ci pomóc tego pięknego dnia?
Mietek: Jeśli nie chcesz mojej zguby
Mietek: To gotowca daj mi, luby!
                                     Ja: Nie.
Mietek: To nie.
>>>
```

Co robi *from* [moduł] *import* [nazwa]? Czym różni się od *import* [moduł]?

<https://docs.python.org/3/tutorial/modules.html>

Zadanie 2. Zmodyfikuj moduł z zadania 1. tak, aby wykonanie pliku z nim powodowało wypisanie opisu i listy funkcji zdefiniowanych w nim (funkcja *dir*), oraz prezentację funkcji w nim zawartych, np.:

```
>>> ===== RESTART =====
>>>
Moduł "text" definiuje funkcje center, left, right i służy do:
Wypisania tekstu wyrównanego do lewej,
                                     do prawej,
                                     lub wycentrowanego.

Dodatkowo, ten moduł zawiera: ['__builtins__', '__doc__', '__file__', '__loader__',
 '__name__', '__package__', '__spec__', 'center', 'left', 'right']
```

Wykorzystaj zmienną `__name__`.

<https://docs.python.org/3/tutorial/modules.html#executing-modules-as-scripts>

Zadanie 3. Napisz moduł o nazwie np. "fibonacci" zawierający dwie funkcje liczące podany wyraz ciągu Fibonacciego, pierwsze iteracyjnie, druga rekurencyjnie. Stwórz pakiet o nazwie np. "pakiet", zawierający ten moduł, oraz moduł z zadań 1. i 2. Napisz program prezentujący działanie wszystkich funkcji tego pakietu.

Napisz plik `__init__.py` pozwalający zaimportować wszystkie moduły pakietu przy użyciu:
`from pakiet import *`

<https://docs.python.org/3/tutorial/modules.html#packages>

<https://docs.python.org/3/tutorial/modules.html#importing-from-a-package>

Zadanie 4. Dane są cztery pliki:

1. 01_kto.txt
2. 02_co_zrobil.txt
3. 03_jaki.txt
4. 04_co.txt

Wczytaj zawartość każdego do osobnej listy (każda linia pliku jako osobny element). Napisz funkcję generującą losowe zdania używając linii wczytanych z powyższych plików. Zdania powinny mieć postać {kto} {co_zrobił} {jaki} {co}. Wykorzystaj funkcję *format*. Pamiętaj o obcięciu znaków nowej linii z końców linii wczytanych plików (np. używając metody *strip*). Program powinien tworzyć plik "wynik.txt" i wpisywać do niego 100 wygenerowanych linii. Wykorzystaj słowo kluczowe *with* przy otwieraniu plików oraz pętlę *for* iterującą po liniach pliku.

[spoiler: strona 3]

<https://docs.python.org/3/tutorial/inputoutput.html#reading-and-writing-files>

Zadanie X. Utwórz plik o nazwie 'pl.py' zawierający słownik o nazwie 'lang':

```
lang={}
lang['hello']='Witaj, nieznajomy!'
lang['info']='Podaj dowolną liczbę, policzę dla ciebie pierwiastek'
lang['error']='Podane dane są nieprawidłowe'
lang['bye']='Żegnaj!'
```

Utwórz kolejny plik o nazwie np. 'en.py' zawierający taki sam słownik, ale z tekstem przetłumaczonym na np. angielski.

Napisz program, który po uruchomieniu prosi użytkownika o wybranie języka. Jeśli użytkownik wpisze 'pl', importowany jest moduł *pl*. Jeśli wpisze 'en', importowany jest moduł *en*. *import* może być używany wewnątrz instrukcji warunkowych. Następnie, po załadowaniu odpowiedniego modułu, program liczy pierwiastek liczby podanej przez użytkownika. Wszystkie wiadomości wypisywane po wyborze języka powinny być odczytywane ze słownika *lang*.

Na początek wypisywane jest powitanie oraz prośba o wpisanie liczby. Jeśli użytkownik wpisze liczbę ujemną, wypisywany jest błąd. W przeciwnym razie, wypisywany jest wynik i pożegnanie.

Ad. Zadanie 4.

```
akto=[]
acozrobil=[]
ajaki=[]
aco=[]

with open('01_kto.txt', 'r') as f:
    for line in f:
        kto, age, lived, sex, height, weight,
        co_zrobil, i, j, r, s, t, u, v, w, x, y, z,
        ajaki, aco = line.split()
        akto.append(kto)
        acozrobil.append(co_zrobil)
        ajaki.append(ajaki)
        aco.append(aco)

    print("100%")

print("{(kto) (co_zrobil) (jaki) (co)".format(
    kto=random.choice(akto),
    co_zrobil=random.choice(acozrobil),
    jaki=random.choice(ajaki),
    co=random.choice(aco),
    ))
```