

Tema de vacanta

Silviu Ojog

***LINK*Academy**

Exercițiul 1

Scrieți o funcție Python care primește o listă de numere ca argument și returnează media acestora

lista = [10, 2, 30, 50, 300, 10]

- rezolvați în 2 moduri

Exercițiul 2

Creați o funcție Python care primește o listă de stringuri ca argument și returnează o nouă listă care conține doar șirurile mai lungi decât 5 caractere

Exercițiul 3

Definiți o funcție Python care primește un string ca argument și returnează un nou string în care toate vocalele au fost eliminate

```
vocale = "aeiouAEIOU"
```

```
input_string = "Salutare, ce mai faci?"
```

Exercițiul 4

Creați o funcție Python care primește o listă de numere ca argument și returnează al doilea cel mai mic număr din acea listă.

numere = [10, 2, 30, 50, 300, 10]

Exercițiul 5

Creați o instrucțiune Python care primește un dicționar cu nume și salariu. Filtrați și salvați într-o listă doar salariile care sunt pe 4000.

```
persoane = {'Andreea' : 100000, 'Bob' : 4900,  
'Cosmin' : 12000, 'Darius' : 3000, 'Ema' : 9000}
```

Exercițiul 6

Creați o instrucțiune Python care o listă pe care o transformă într-un dicționar

```
winners = [ 'Messi', 'Modric', 'Ronaldo', 'Ronaldo', 'Messi',  
'Ronaldo', 'Ronaldo', 'Messi', 'Messi', 'Messi', 'Messi', 'Ronaldo',  
'Messi', 'Kaka' ]
```

Exercițiul 7

Creați o instrucțiune Python pentru

Găsirea numărului maxim dintr-o listă:

Calcularea sumei tuturor numerelor dintr-o listă:

Sortarea unei liste în ordine crescătoare:

Eliminarea duplicată din lista:

Inversarea unei liste:

Exercițiul 8

Creați o instrucțiune Python care să transforme `my_list` în `flattened_list`

```
my_list = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
```

```
flattened_list = [1, 2, 3, 4, 5, 6, 7, 8, 9]
```

Exercițiul 9

Creați următoarea listă într-o singură linie de comandă

```
my_list = ["1", "2", "3", "4"]
```

Exercițiul 10

Creați următoarea matrice într-o singură linie de comandă

matrice = [(0, 0), (0, 1), (0, 2), (1, 0), (1, 1), (1, 2), (2, 0), (2, 1), (2, 2)]

Exercițiul 10

Creați următoarea matrice într-o singură linie de comandă

$$\text{matrice} = [(0, 0), (0, 1), (0, 2), \\ (1, 0), (1, 1), (1, 2), \\ (2, 0), (2, 1), (2, 2)]$$

Exercițiul 11

Aveți următoare colecție

```
cursuri = (('Blockchain', ('MySQL', 310)), ('HTML', ('Python', 10)))
```

Accesați elementul

'Python' din cursuri

Exercițiul 12

Aveți următoare colecție

```
cursuri = ('python', 'blockchain', 'sql', 'testare')
```

Sortați alfabetic. Faceți acest lucru în 3 versiuni.

Sortați după lungimea cuvintelor. Faceți acest lucru în 3 versiuni.

Exercițiul 13

Sortați următoare colecție

cursanți = (('Monica', 29), ('Andrei', 21), ('Ion', 28))

După al doilea parametru (vârsta)

Exercitiul cu suma nr nedublate

- Vi se oferă o listă cu numere întregi. Elementele unice ale unei liste sunt elementele care apar exact o dată în matrice.
- Calculați suma lor

```
Input: nums = [1,2,3,2]  
Output: 4  
Explicație: Elementele unice sunt [1,3], iar suma este 4.
```

Example 2:

```
Input: nums = [1,1,1,1,1]  
Output: 0  
Explicație: Nu există elemente unice, iar suma este 0.
```

Example 3:

```
Input: nums = [1,2,3,4,5]  
Output: 15  
Explicație: Elementele unice sunt [1,2,3,4,5], iar suma este 15.
```


Exercitiul cu numere Romane

- Convertiti din Numere Romane în Întreg

I 1

V 5

X 10

L 50

C 100

D 500

M 1000

Ex 1:

Input: s = "III"

Output: 3

Explicație: III = 3.

Ex 2:

Input: s = "LVIII"

Output: 58

Explicație: L = 50, V= 5, III = 3.

Ex 3:

Input: s = "MCMXCIV"

Output: 1994

Explicație: M = 1000, CM = 900, XC = 90 ,IV = 4.

Exercitiul cu numere Romane - 2

- Convertiti un întreg în Numere Romane

I 1

V 5

X 10

L 50

C 100

D 500

M 1000

Exemplul 1:

Intrare: num = 3749

Ieșire: „MMM DCC XLIX”

Explicație:

3000 = MMM ca 1000 (M) + 1000 (M) + 1000 (M)

700 = DCC ca 500 (D) + 100 (C) + 100 (C)

40 = XL cu 10 (X) mai puțin de 50 (L)

9 = IX ca 1 (I) mai puțin de 10 (X)

Exemplul 2:

Intrare: num = 58

Ieșire: „LVIII”

Explicație:

50 = L

8 = VIII

Notă: 49 nu este cu 1 (I) mai puțin de 50 (L), deoarece conversia se bazează pe zecimale

Exemplul 3:

Intrare: num = 1994

Ieșire: „MCMXCIV”

Explicație:

1000 = M

900 = CM

90 = XC

4 = IV

Exercitiul cu paranteze valide

- Având în vedere un string care conține doar caracterele „(”, „)”, „{”, „}”, „[” și „]”, determinați dacă șirul de intrare este valid.
- Un șir de intrare este valid dacă:
 - 1.Parantezele deschise trebuie să fie închise cu același tip de paranteze.
 - 2.Parantezele deschise trebuie să fie închise în ordinea corectă.
 - 3.Fiecare paranteză închisă are un parantez deschis corespunzător de același tip.

Exemplul 1:

Intrare: s = "()"

Output: true

Exemplul 2:

Intrare: s = „() [{ }”

Output: true

Exemplul 3:

Intrare: s = "(]"

Output: false

Exemplul 4:

Intrare: s = „([])”

Output: true

Exercițiul cu TimeConverter



1. Creați clasa TimeConverter cu două proprietăți(tribute):
 - hours - numărul de ore (int)
 - minutes - numărul de minute (int)
2. Clasa trebuie să conțină următoarele metode publice:
 - toMinutes() - convertește timpul total în minute
 - toHours() - convertește timpul total în ore
 - addTime() - adună încă un interval de timp și returnează rezultatul în minute
 - diffTime() - calculează diferența față de alt interval de timp și returnează rezultatul în minute

Exercițiul cu Temperatura



1. Creați clasa **TemperatureConverter** cu două proprietăți(tribute):
 - **value** - valoarea temperaturii
 - **unit** - unitatea de măsură ('C', 'F', sau 'K')
2. Clasa trebuie să conțină următoarele metode publice:
 - **toCelsius()** - convertește temperatura în grade Celsius
 - **toFahrenheit()** - convertește temperatura în grade Fahrenheit
 - **toKelvin()** - convertește temperatura în Kelvin
 - **computeDiff()** - calculează diferența față de altă temperatură (în aceeași unitate)
 - **isFreezing()** - temperatura este sub 0 grade Celsius.
 - **isBoiling()** - temperatura este peste 100 grade Celsius.