




CODE ONE



LET'S CODE

Lista de exercícios - Python

Questão 1.

Faça um programa que peça ao usuário um número e imprima todos os números de um até o número que o usuário informar.

💡 Exemplo:

Se o usuário informar o número 5, seu programa deverá imprimir: 1 2 3 4 5.

Questão 2.

Crie um programa que leia um valor qualquer e apresente uma mensagem dizendo em qual dos seguintes intervalos $[0,25]$, $(25,50]$, $(50,75]$, $(75,100]$ este valor se encontra. Caso o valor não esteja em nenhum destes intervalos, deverá ser impressa a mensagem "Fora de intervalo". Veja alguns exemplo abaixo:

💡 Entrada: 25.01 | Saída: (25,50]
Entrada: 25.00 | Saída: [0,25]
Entrada: 100.00 | Saída: (75,100]
Entrada: -25.02 | Saída: Fora de intervalo

📌 Lembrando que o `[` ou `]` representa que o valor está contido no intervalo, enquanto o `(` ou `)` representa que o valor associado não está contido no intervalo. Em outras palavras, `(75, 100]` representa o intervalo que vai de maior que 75 (não incluindo o 75) até menor ou igual 100.

Questão 3.

Crie uma função que recebe o valor do raio de um círculo como parâmetro e retorna o valor da área desse círculo. Lembrando que a área de círculo é dada pela equação: $A = \pi r^2$.

💡 Dica: Para utilizar um valor mais preciso do pi (π), você pode importar a biblioteca `math`, e utilizar o `math.pi`, como ilustrado no código abaixo:

```
import math  
  
print(math.pi)
```

Questão 4.

Faça um programa que peça 2 números inteiros e um número real, calcule e mostre:

- a) o produto entre o dobro do primeiro e a metade do segundo.
- b) a soma entre o triplo do primeiro e o terceiro.
- c) o terceiro elevado ao cubo.

Questão 5.

Vamos fazer um programa para verificar quem é o assassino de um crime. Para descobrir o assassino, a polícia faz um pequeno questionário com 5 perguntas onde a resposta só pode ser sim ou não:

1. Mora perto da vítima?
2. Já trabalhou com a vítima?
3. Telefonou para a vítima?
4. Esteve no local do crime?
5. Devia para a vítima?

Cada resposta sim dá um ponto para o suspeito. A polícia considera que os suspeitos com 5 pontos são os assassinos, com 4 a 3 pontos são cúmplices e 2 pontos são apenas suspeitos, necessitando de outras investigações. Valores abaixo de 2 são liberados.

No seu programa, você deve fazer essas perguntas e, de acordo com as respostas do usuário, você vai informar como a polícia o considera.

Questão 6.

Faça uma função que recebe uma lista de números e retorna a soma dos elementos dessa lista.

Questão 7.

Faça um programa que leia as coordenadas de 2 (dois) pontos em um plano cartesiano 2D: a coordenada x do primeiro ponto (x_1), a coordenada y do primeiro ponto (y_1), a coordenada x do segundo ponto (x_2) e a coordenada y do segundo ponto (y_2). Em seguida, calcule a distância euclidiana entre os pontos, utilizando a equação abaixo:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Questão 8.

Calcule a soma de mil termos dos inversos dos fatoriais: $1/(1!) + 1/(2!) + 1/(3!) + 1/(4!) + \dots$

Dica: Use três variáveis:

- um contador;
- uma variável para soma;
- e uma variável para os termos.

Lembre-se de que $4! = 4 * 3 * 2 * 1$, que também é igual a $4 * 3!$.

Questão 9.

Crie (manualmente ou sorteando os números) uma lista de 10 números e imprima:

1. uma lista com os 4 primeiros números;
2. uma lista com os 5 últimos números;
3. uma lista contendo apenas os elementos das posições pares;
4. uma lista contendo apenas os elementos das posições ímpares;








5. a lista inversa da lista sorteada (isto é, uma lista que começa com o último elemento da lista sorteada e termina com o primeiro);
6. uma lista inversa dos 5 primeiros números;
7. uma lista inversa dos 5 últimos números.

Questão 10.

Na música, uma nota tem um tom (sua frequência, resultando em quão grave ou agudo é o som) e uma duração (por quanto tempo a nota soa). Neste problema, estamos interessados apenas na duração das notas.


Marcos está dando os primeiros passos para ser um compositor de jingles. Ele está tendo alguns problemas, mas ao menos ele está criando melodias agradáveis e ritmos atrativos. Um jingle é dividido em uma sequência de compassos, e um compasso é formado de uma série de notas.

A duração de uma nota é indicada pela sua forma. Neste problema, iremos utilizar letras maiúsculas para indicar a duração de uma nota. A seguinte tabela lista todas as notas disponíveis:

Notas							
Identificador	W	H	Q	E	S	T	X
Duração	1	1/2	1/4	1/8	1/16	1/32	1/64

A duração de um compasso é a soma da duração de suas notas. Nos jingles de Marcos, cada compasso tem a mesma duração. Como Marcos é apenas um iniciante, seu famoso professor Johann Sebastian III o ensinou que a duração de um compasso deve ser sempre 1.

Por exemplo, Marcos escreveu uma composição contendo cinco compassos, dentre os quais quatro possuem a duração correta e um está errado. No exemplo abaixo, cada compasso é delimitado com barras e cada nota é representada como na tabela acima.

 /HH/QQQQ/XXXTXTEQH/W/HW/

Marcos gosta de computadores assim como de música. Ele quer que você escreva um programa que determine, para cada uma de suas composições, quantos compassos possuem a duração correta e quais são os compassos com duração incorreta.

Exemplo de Entrada	Exemplo de Saída
/HH/QQQQ/XXXTXTEQH/W/HW/	Qtd. de Corretos: 4 Incorretos: HW
/W/W/SQHES/	Qtd. de Corretos: 3
/WE/TEX/THES/	Qtd. de Corretos: 0 Incorretos: [WE, TEX, THES]

LET'S CODE