

Mundo 1 - Bibliotecas

1) Crie um programa que leia um número real qualquer pelo teclado e mostre na tela a sua porção inteira

a)

```
import math
numero = float(input('Digite um valor: '))
print("O valor digitado foi {} e a sua parte inteira é {}".format(numero, math.trunc(numero)))
```

Nesta primeira solução, importamos a biblioteca inteira de `math` e utilizamos uma fórmula chamada `trunc` onde ela extrai apenas a parte inteira do valor, mas para chamarmos a função, precisamos digitar `<biblioteca>.<função>(parâmetro ou valor)`

b)

```
from math import trunc
numero = float(input('Digite um valor: '))
print("O valor digitado foi {} e a sua parte inteira é {}".format(numero, trunc(numero)))
```

Nesta solução importamos apenas a função `trunc` onde teoricamente ela se torna uma padrão já disponibilizada no python, logo podemos chamá-la apenas digitando `<função>(parâmetro ou valor)`

c)

```
num = float(input('DIGITE UM VALOR: '))
print('Valor digitado foi {} e sua parte inteira é {}'.format(num, int(num)))
```

Nesta solução não utilizamos uma biblioteca mas sim a conversão do valor, onde um dos placeholders do `print` vai mostrar a conversão para inteiro da variável `num`.

2) Faça um programa que leia o comprimento do cateto oposto e do cateto adjacente de um triângulo retângulo, calcule e mostre o comprimento da hipotenusa.

a)

```
o = float(input('VALOR CATETO OPOSTO: '))
a = float(input('VALOR CATETO ADJACENTE: '))
```

```
h = ((o**2)+(a**2))**(1/2)
print('O VALOR DA HIPOTENUSA É {:.2f}'.format(h))
```

Aqui fiz de forma manual, montando a fórmula em si onde $h^2 = co^2 + ca^2$ e peço para mostrar o resultado arredondado em duas casas decimais.

b)

```
import math
o = float(input('VALOR CATETO OPOSTO: '))
a = float(input('VALOR CATETO ADJACENTE: '))
h = math.sqrt(pow(o,2)+pow(a,2))
print('O VALOR DA HIPOTENUSA É {:.2f}'.format(h))
```

Ainda fizemos de forma bruta, mas as operações já estão substituídas por funções, porém existe outra forma de simplificar ainda mais.

c)

```
import math
o = float(input('VALOR CATETO OPOSTO: '))
a = float(input('VALOR CATETO ADJACENTE: '))
h = math.hypot(o,a)
print('O VALOR DA HIPOTENUSA É {:.2f}'.format(h))

ou

from math import hypot
o = float(input('VALOR CATETO OPOSTO: '))
a = float(input('VALOR CATETO ADJACENTE: '))
h = hypot(o,a)
print('O VALOR DA HIPOTENUSA É {:.2f}'.format(h))
```

Existe uma função para calcular a hipotenusa, então esta é a forma mais simplificada para resolver.

3) Faça um programa que leia um ângulo qualquer e mostre na tela o valor do Seno, Cosseno e Tangente desse ângulo

a)

```
import math
an = float(input('DIGITE UM ÂNGULO: '))
seno = math.sin(math.radians(an))
cosseno = math.cos(math.radians(an))
tangente = math.tan(math.radians(an))
```

```
print('O ângulo de {} tem o SENO de {:.2f}'.format(an, seno))
print('O ângulo de {} tem o COSSENO de {:.2f}'.format(an, cosseno))
print('O ângulo de {} tem a TANGENTE de {:.2f}'.format(an, tangente))
```

Para calcularmos o seno, cosseno e tangente, precisamos converter o valor para radianos e depois disso transformar no valor final, então fazendo de forma completa importando a biblioteca toda, ou podemos fazer de forma efetiva assim:

```
from math import sin, cos, tan, radians
an = float(input('DIGITE UM ÂNGULO: '))
seno = sin(radians(an))
cosseno = cos(radians(an))
tangente = tan(radians(an))
print('O ângulo de {} tem o SENO de {:.2f}'.format(an, seno))
print('O ângulo de {} tem o COSSENO de {:.2f}'.format(an, cosseno))
print('O ângulo de {} tem a TANGENTE de {:.2f}'.format(an, tangente))
```

4) Um professor quer sortear um dos seus quatro alunos para apagar o quadro. Faça um programa que ajude ele, lendo os nome deles e escrevendo o nome do escolhido.

a)

```
import random
aluno1 = input('NOME ALUNO(A) 1: ')
aluno2 = input('NOME ALUNO(A) 2: ')
aluno3 = input('NOME ALUNO(A) 3: ')
aluno4 = input('NOME ALUNO(A) 4: ')
lista = [aluno1, aluno2, aluno3, aluno4]
print('O(A) ALUNO(A) ESCOLHIDO(A) FOI {}'.format(random.choice(lista)))
```

Existe a biblioteca random que serve para sortear números aleatórios ou escolher dentro de uma sequência, neste caso nós o utilizamos uma lista e a função choice onde ele percorre a lista e tira um de forma aleatória, outra forma de simplificar seria somente importar a função choice.

```
from random import choice
aluno1 = input('NOME ALUNO(A) 1: ')
aluno2 = input('NOME ALUNO(A) 2: ')
aluno3 = input('NOME ALUNO(A) 3: ')
aluno4 = input('NOME ALUNO(A) 4: ')
lista = [aluno1, aluno2, aluno3, aluno4]
print('O(A) ALUNO(A) ESCOLHIDO(A) FOI {}'.format(choice(lista)))
```

5) O mesmo professor do desafio anterior quer sortear a ordem de apresentação de trabalhos dos alunos. Faça um programa que leia o nome dos quatro alunos e mostre a ordem sorteada.

a)

```
import random
aluno1 = input('NOME ALUNO(A) 1: ')
aluno2 = input('NOME ALUNO(A) 2: ')
aluno3 = input('NOME ALUNO(A) 3: ')
aluno4 = input('NOME ALUNO(A) 4: ')
lista = [aluno1,aluno2,aluno3,aluno4]
random.shuffle(lista)
print('A ordem de apresentaação será {}'.format(lista))
```

A função shuffle serve para embaralhar os valores e determinar uma nova ordem, podemos representar da forma completa ou importar apenas a função de forma efetiva.

```
from random import shuffle
aluno1 = input('NOME ALUNO(A) 1: ')
aluno2 = input('NOME ALUNO(A) 2: ')
aluno3 = input('NOME ALUNO(A) 3: ')
aluno4 = input('NOME ALUNO(A) 4: ')
lista = [aluno1,aluno2,aluno3,aluno4]
shuffle(lista)
print('A ordem de apresentaação será {}'.format(lista))
```

6) Faça um programa em Python que abra e reproduza o áudio de um arquivo MP3.

a)

```
import pygame
pygame.init()
pygame.mixer.music.load('nioce.mp3')
pygame.mixer.music.play()
input()
pygame.event.wait()
```

Existe a biblioteca pygame onde ele não somente reproduz musica mas faz outras coisas relacionadas à criação de jogos, mas neste exemplo utilizamos para tocar este .mp3