

cosi-melchor-practica-2

October 11, 2024

1 2.1 Hola Python

1.1 Ejercicio 1

```
[6]: pi = 3.14159  
    diametro = 3
```

Crea una variable llamada radio iguala la mitad del diametro.

```
[7]: radio = diametro / 2  
    print(radio)
```

1.5

Crea una variable llamada area, usando la formula para el área de un circulo: $\pi \cdot \text{radio}^2$

```
[8]: area = pi * radio**2  
    print(area)
```

7.0685775

1.2 Ejercicio 2

Agregue código a la siguiente celda para intercambiar las variables a y b (de modo que a se refiera al objeto al que antes hacía referencia b y viceversa).

```
[9]: a = [1, 2, 3]  
    b = [3, 2, 1]  
    print("a = ",a)  
    print("b = ",b)  
    temp = a  
    a = b  
    b = temp  
    print("a = ",a)  
    print("b = ",b)
```

```
a = [1, 2, 3]  
b = [3, 2, 1]  
a = [3, 2, 1]  
b = [1, 2, 3]
```

1.3 Ejercicio 3a

Agrega paréntesis a la siguiente expresión donde crees necesario para que el resultado te de 1

```
[10]: (5 - 3) // 2
```

```
[10]: 1
```

1.4 Ejercicio 3b

Agregue paréntesis a la siguiente expresión para que evalúe a 0.

```
[11]: 8 - (3 * 2) - (1 + 1)
```

```
[11]: 0
```

```
[12]: (8 - 3) * (2 - (1 + 1))
```

```
[12]: 0
```

2 2.3 Booleanos

2.1 Ejercicio 1

Defina una función llamada signo que tome un argumento numérico y devuelva -1 si es negativo, 1 si es positivo y 0 si es 0.

```
[13]: def signo(number):  
      if number > 0:  
          return 1  
      elif number < 0:  
          return -1  
      return 0
```

```
[14]: print(signo(5))  
      print(signo(-5))  
      print(signo(0))
```

```
1  
-1  
0
```

2.2 Ejercicio 2

Hemos visto que llamar a `bool()` en un número entero devuelve `False` si es igual a 0 y `True` en caso contrario.

¿Puedes aprovechar esto para escribir una función sucinta que corresponda a la oración "¿quiere el cliente exactamente un aderezo? Indica si el cliente quiere exactamente uno de los tres ingredientes disponibles en su Hot dog.

```
[15]: def exact_un_ingredient(salsa_de_tomate, mostaza, cebolla):
    suma = int(salsa_de_tomate) + int(mostaza) + int(cebolla)
    if suma == 1:
        return True
    return False

[16]: print("Ingredientes: Ninguno -> {}".format(
    exact_un_ingredient(salsa_de_tomate=False, mostaza = False, cebolla = False))
)
print("Ingredientes: Cebolla -> {}".format(
    exact_un_ingredient(salsa_de_tomate=False, mostaza = False, cebolla = True))
)
print("Ingredientes: Mostaza -> {}".format(
    exact_un_ingredient(salsa_de_tomate=False, mostaza = True, cebolla = False))
)
print("Ingredientes: Mostaza y Cebolla -> {}".format(
    exact_un_ingredient(salsa_de_tomate=False, mostaza = True, cebolla = True))
)
print("Ingredientes: Salsa de Tomate -> {}".format(
    exact_un_ingredient(salsa_de_tomate=True, mostaza = False, cebolla = False))
)
print("Ingredientes: Salsa de Tomate y Cebolla -> {}".format(
    exact_un_ingredient(salsa_de_tomate=True, mostaza = False, cebolla = True))
)
print("Ingredientes: Salsa de Tomate y Mostaza -> {}".format(
    exact_un_ingredient(salsa_de_tomate=True, mostaza = True, cebolla = False))
)
print("Ingredientes: Salsa de Tomate, Mostaza y Cebolla -> {}".format(
    exact_un_ingredient(salsa_de_tomate=True, mostaza = True, cebolla = True))
)
```

```
Ingredientes: Ninguno -> False
Ingredientes: Cebolla -> True
Ingredientes: Mostaza -> True
Ingredientes: Mostaza y Cebolla -> False
Ingredientes: Salsa de Tomate -> True
Ingredientes: Salsa de Tomate y Cebolla -> False
Ingredientes: Salsa de Tomate y Mostaza -> False
Ingredientes: Salsa de Tomate, Mostaza y Cebolla -> False
```

3 2.4 Ciclos y Listas

3.1 Ejercicio 1

En el juego de Mario Kart se cuenta con el Purple Shell. Cuando se usa, deforma al corredor del último lugar al primer lugar y al corredor del primer lugar al último lugar. Complete la siguiente función para implementar el efecto de Purple Shell.

Dada una lista de corredores, establezca el corredor del primer lugar (en el frente de la lista) para el último lugar y viceversa.

```
[17]: def purple_shell(racers):  
    temp = racers[-1]  
    racers[-1] = racers[0]  
    racers[0] = temp  
    return racers
```

```
[18]: r=["Mario", "Bowser", "Luigi"]
```

```
[19]: purple_shell(r)
```

```
[19]: ['Luigi', 'Bowser', 'Mario']
```

4 2.5 Cadenas y Diccionarios

4.1 Ejercicio 1

Escribir una función para ayudar a limpiar los datos del código postal de Mexico Dada una cadena, debe devolver si esa cadena representa o no un código postal válido. Para nuestros propósitos, un código postal válido es cualquier cadena que consta de exactamente 5 dígitos.

```
[20]: def es_valid_zip(zip_cod):  
    valid_zip = str.isdigit(zip_cod) and len(zip_cod) == 5  
    return valid_zip
```

```
[21]: zip = "73300"  
is_zip = es_valid_zip(zip)  
print("{} is a valid zip?: {}".format(zip.rjust(7),is_zip))  
  
zip = "846547"  
is_zip = es_valid_zip(zip)  
print("{} is a valid zip?: {}".format(zip.rjust(7),is_zip))  
  
zip = "12345"  
is_zip = es_valid_zip(zip)  
print("{} is a valid zip?: {}".format(zip.rjust(7),is_zip))  
  
zip = "125"  
is_zip = es_valid_zip(zip)  
print("{} is a valid zip?: {}".format(zip.rjust(7),is_zip))  
  
zip = "4534R"  
is_zip = es_valid_zip(zip)  
print("{} is a valid zip?: {}".format(zip.rjust(7),is_zip))
```

```
zip = "letra"
is_zip = es_valid_zip(zip)
print("{} is a valid zip?: {}".format(zip.rjust(7),is_zip))
```

```
73300 is a valid zip?: True
846547 is a valid zip?: False
12345 is a valid zip?: True
125 is a valid zip?: False
4534R is a valid zip?: False
letra is a valid zip?: False
```