

Universidad Nacional del Altiplano
Facultad de Ingeniería Estadística e Informática

Docente:

Ing. Coyla Idme Leonel

Alumno:

Ticona Miramira Roberto Angel

Desarrollo de Prácticas

» PRÁCTICA 1

1.- Crear un Código Python para resolver el índice de masa corporal. Usar clases, atributos, acción y objeto.

Tabla de Clasificación del IMC (OMS)

Categoría	Rango del IMC (kg/m^2)
Bajo peso	Menor a 18.5
Normal	18.5 - 24.9
Sobrepeso	25.0 - 29.9
Obesidad grado I	30.0 - 34.9
Obesidad grado II	35.0 - 39.9
Obesidad grado III	40.0 o más

Código

```
1 class IMC:
2     def __init__(self, peso, talla):
3         self.peso = peso
4         self.talla = talla
5
6     def calculo(self):
7         return self.peso / self.talla ** 2
8
9     def clasificar(self):
10        if self.calculo() <= 18.5:
11            return "Bajo peso"
12        elif self.calculo() <= 24.9:
13            return "Normal"
14        elif self.calculo() <= 29.9:
15            return "Sobrepeso"
16        elif self.calculo() <= 34.9:
17            return "Obesidad grado I"
18        elif self.calculo() <= 39.9:
19            return "Obesidad grado II"
20        else:
21            return "Obesidad grado III"
22
23 def main():
24     peso = float(input("Ingrese su peso en kilogramos: "))
25     talla = float(input("Ingrese su talla en metros: "))
26     persona1 = IMC(peso,talla)
27     rpta = persona1.calculo()
28     categoria = persona1.clasificar()
29     print(f"Su IMC es {rpta:.2f} - {categoria}")
```

```
30 if __name__ == "__main__":  
31     main()
```

Ejecución

```
1 Ingrese su peso en kilogramos: 64.4  
2 Ingrese su talla en metros: 1.64  
3 Su IMC es 23.94 - Normal
```

2.- Crear un Código Python para convertir grados Fahrenheit a Celsius. Usar clases, atributos, acción y objeto. Use Encapsulamiento getter y setter.

Fórmulas de Conversión de Temperatura

a. Fahrenheit a Celsius

$$C = \frac{(F - 32) * 5}{9}$$

b. Celsius a Fahrenheit

$$F = (C * \frac{9}{5}) + 32$$

Código

```
1 class Temperatura:  
2     def __init__(self, fahrenheit, celsius):  
3         self.__fahrenheit = fahrenheit  
4         self.__celsius = celsius  
5  
6     def get_fahrenheit(self):  
7         return self.__fahrenheit  
8  
9     def get_celsius(self):  
10        return self.__celsius  
11  
12    def set_fahrenheit(self, nuevo_temp):  
13        if nuevo_temp > 0:  
14            self.__fahrenheit = nuevo_temp  
15        else:  
16            print("Temperatura invalida")  
17  
18    def set_celsius(self, nuevo_c):  
19        if nuevo_c > 0:  
20            self.__celsius = nuevo_c  
21        else:  
22            print("Temperatura invalida")  
23  
24    def calcular(self):  
25        return ((self.__fahrenheit - 32) * 5) / 9  
26  
27    def calcular2(self):  
28        return ((self.__celsius * (9/5)) + 32)  
29  
30 def main():  
31     dato = float(input("Ingrese la temperatura en grados fahrenheit: "))
```

```
32     dato2 = float(input("Ingrese otra temperatura en grados celsius: "))
33     temperatura = Temperatura(dato,dato2)
34     print("La temperatura en grados celcius es: ", temperatura.calcular())
35     print("La temperatura en grados fahrenheit es: ", temperatura.calcular2())
36
37 if __name__=="__main__":
38     main()
```

Ejecución

```
1 Ingrese la temperatura en grados fahrenheit: 98.6
2 Ingrese otra temperatura en grados celsius: 25
3 La temperatura en grados celcius es: 37.0
4 La temperatura en grados fahrenheit es: 77.0
```

» PRÁCTICA 2

1.- Clase Libro: Mostrar información y destruir objeto

Problema:

Crea una clase Libro que contenga:

- Atributos: titulo, autor, anio.
- Constructor que reciba los tres atributos y muestre el mensaje: "Libro [titulo] de [autor] creado".
- Método mostrar_info() que muestre los datos del libro.
- Destructor que imprima: "Libro [titulo] eliminado de la biblioteca".

Luego, crea un objeto de esta clase, llama al método y destrúyelo

Código

```
1 class Libro:
2     def __init__(self, titulo, autor, anio):
3         self.titulo = titulo
4         self.autor = autor
5         self.anio = anio
6         print("Libro creado")
7
8     def mostrar_info(self):
9         print(f"Libro {self.titulo} escribió por {self.autor} en el año {self.anio}")
10
11     def __del__(self):
12         print("Objeto destruido")
13
14 def main():
15     titulo = input("Ingrese el nombre del libro: ")
16     autor = input("Ingrese el nombre del autor: ")
17     an = int(input("Ingrese el año del libro: "))
18     miLibro = Libro(titulo,autor,an)
19     miLibro.mostrar_info()
20 if __name__=="__main__":
21     main()
```

Ejecución

```
1 Ingrese el nombre del libro: Los perros hambrientos
2 Ingrese el nombre del autor: Ciro Alegría
3 Ingrese el año del libro: 1939
```

```
4 Libro creado
5 Libro Los perros hambrientos escribió por Ciro Alegría en el año 1939
6 Objeto destruido
```

2.- Clase CuentaBancaria: Depósito y retiro

Problema:

Crea una clase CuentaBancaria con:

- Atributos: titular, saldo.
- Constructor que inicialice los datos y muestre un mensaje de bienvenida.
- Método depositar (monto) que aumente el saldo.
- Método retirar (monto) que reste del saldo si hay fondos suficientes.
- Destructor que muestre: "Cuenta de [titular] cerrada".

Crea un objeto, realiza un depósito y un retiro, y luego destrúyelo.

Código

```
1 class CuentaBancaria:
2     def __init__(self, titular, saldo):
3         self.titular = titular
4         self.saldo = saldo
5         print(f"Cuenta creada, bienvenido {self.titular}")
6
7     def depositar(self, monto):
8         self.saldo += monto
9         print(f"Depósito de {monto} realizado. Saldo actual: {self.saldo}")
10
11    def retirar(self, monto):
12        if monto <= self.saldo:
13            self.saldo -= monto
14            print(f"Retiro de {monto} realizado. Saldo actual: {self.saldo}")
15        else:
16            print("Fondos insuficientes para retirar esa cantidad.")
17
18    def __del__(self):
19        print(f"Cuenta de {self.titular} cerrada")
20
21
22 def main():
23     titular = input("Ingrese el nombre del titular: ")
24     saldo = float(input("Ingrese su saldo inicial: "))
25
26     cuenta = CuentaBancaria(titular, saldo)
27
28     # Hacemos un depósito
29     cuenta.depositar(200)
30
31     # Hacemos un retiro
32     cuenta.retirar(100)
33
34     # Eliminamos el objeto (forzamos el destructor)
35     del cuenta
36
37
38 if __name__ == "__main__":
39     main()
```

Ejecución

```
1 Ingrese el nombre del titular: Roberto
2 Ingrese su saldo inicial: 650
3 Cuenta creada, bienvenido Roberto
4 Depósito de 200 realizado. Saldo actual: 850.0
5 Retiro de 100 realizado. Saldo actual: 750.0
6 Cuenta de Roberto cerrada
```

3.- Clase Temperatura: Conversión de Celsius a Fahrenheit.

Problema:

Crea una clase Temperatura con:

- Atributo: celsius.
- Constructor que reciba la temperatura en °C.
- Método a_fahrenheit() que convierta y retorne la temperatura en °F.
- Destructor que indique: "Objeto Temperatura destruido"

Crea un objeto, muestra la temperatura en ambas escalas, y elimina el objeto.

Código

```
1 class Temperatura:
2     def __init__(self, celsius):
3         self.celsius = celsius
4         print("Objeto Temperatura creado")
5
6     def a_fahrenheit(self):
7         f = self.celsius * (9 / 5) + 32
8         return f
9
10    def __del__(self):
11        print("Objeto Temperatura destruido")
12
13 def main():
14     cel = float(input("Ingrese la temperatura en grados Celsius: "))
15     temp = Temperatura(cel)
16     rpta = temp.a_fahrenheit()
17     print(f"Temperatura en grados fahrenheit: {rpta}")
18
19 if __name__ == "__main__":
20     main()
```

Ejecución

```
1 Ingrese la temperatura en grados Celsius: 25
2 Objeto Temperatura creado
3 Temperatura en grados fahrenheit: 77.0
4 Objeto Temperatura destruido
```