

ELECTIVA ESPECÍFICA II

PYTHON

Diego Andrés Restrepo Leal

`diego.restrepoleal@campusucc.edu.co`

Programa de ingeniería electrónica

Facultad de ingeniería

Universidad Cooperativa de Colombia

TABLA DE CONTENIDO

1. Estructuras de datos
2. Azúcar sintáctico
 - List comprehension
 - Dictionary comprehension
 - Generadores comprehension
 - Decoradores
3. Manejo de archivos
4. Programación orientada a objetos
5. Ejercicios
6. Taller 01

ESTRUCTURAS DE DATOS



```
def fibon(n):  
    a = 0  
    b = 1  
  
    for i in range(n):  
        yield a  
        a, b = b, a + b
```

Ejemplo Fibonacci con generadores: [Ir a gitlab.](#)

Ejemplo Fibonacci con listas: [Ir a gitlab.](#)

AZÚCAR SINTÁCTICO

- Azúcar sintáctico
 - Más legible.
 - Más fácil de expresar.
 - Ahorra líneas de código.

LIST COMPREHENSION

```
def pares_for():  
  
    pares = []  
  
    for num in range(11):  
        if num % 2 == 0:  
            pares.append(num)  
  
def pares_list():  
    pares = [num for num in range(11) if num % 2 == 0]
```

Ejemplo: Ir a gitlab.

DICCIONARY COMPREHENSION

```
def cubos_for():  
    cubos = {}  
  
    for num in range(11):  
        cubos[chr(num + 65)] = num**3  
  
def cubos_diccionario():  
    cubos = {chr(num + 65): num**3 for num in range(11)}
```

Ejemplo: Ir a [gitlab](#).

GENERADORES COMPREHESION

```
def run():  
    generador = (x**2 for x in range(10000))  
  
    for i in generador:  
        print(i)  
  
if __name__ == '__main__':  
    run()
```

Ejemplo: Ir a gitlab.

De forma general, se pueden definir los decoradores como una función que recibe otra función y regresa una tercera función.

```
def funcion_2(funcion_1):  
    def funcion_3(#parámetro o no):  
        #procesando a la función 1  
        return #función 1 procesada  
    return funcion_3
```

EJEMPLO: NORMALIZACIÓN

Normalizar los números del intervalo $[1, 10]$ al intervalo $[-1, 1]$ Para normalizar, implemente la siguiente ecuación:

$$x_{norm} = \frac{2(x - x_{min})}{x_{max} - x_{min}} - 1$$

Donde:

x son los datos a normalizar.

x_{norm} son los valores normalizados.

Ejemplo: Sin decoradores.

Ejemplo: Con decoradores.

MANEJO DE ARCHIVOS



- Leer y escribir archivos
- Regresa un objeto de tipo file.
- Se debe especificar el modo de operación:
 - 'r' : read.
 - 'w' : write.
 - 'a' : append.
 - 'r+' : read and write.

EJEMPLO: FUNCIÓN OPEN

```
# Escribir archivos:  
with open('numeros.txt', 'w') as f:  
    f.write('x x^2 x^3\n')  
  
# Leer archivos  
with open('numeros.txt', 'r') as f:  
    print(f.read())
```

Ejemplo write: Ir a gitlab.

Ejemplo read: Ir a gitlab.

PROGRAMACIÓN ORIENTADA A OBJETOS

- Encapsular código.
- Definir tipos propios.
- Manejo de tipos de datos y lógica en un sólo contenedor.
- Clases: modelos para crear otros objetos.

- **Personas:**

- Estado:
 - Apariencia
 - Edad
- Acciones:
 - Baila
 - Cocinar

- **Termo**

- ¿Estado?
- ¿Acciones?

- Los objetos poseen atributos.
- Las clases pueden tener diversas variables.
- Python no cuenta con variables privadas, pero es una práctica común definir las con un guión bajo.

- Los métodos son funciones que existen sólo dentro de las clases.
- Las clases deben se inicializadas.
- Existe la herencia entre clases.

EJEMPLO PERSONA: PERSONA.PY

```
class Persona:

    def __init__(self, nombre, edad):
        self.nombre = nombre
        self.edad = edad

    def presentarse(self):
        print(f"Hola, mi nombre es {self.nombre} y tengo {self.edad} de edad.")
```

Ejemplo: Ir a gitlab.

EJEMPLO PERSONA: MAINPERSONA.PY

```
from persona import Persona

def run():
    persona0 = Persona(nombre = "María", edad = 25)
    persona0.presentarse()

    persona1 = Persona(nombre = None, edad = None)
    persona1.nombre = "Juan"
    persona1.edad = 11
    persona1.presentarse()

if __name__ == '__main__':
    run()
```

Ejemplo: Ir a gitlab.

Ejemplo Lampara:

- Ejemplo lamp.py: Ir a gitlab.
- Ejemplo mainlampara.py: Ir a gitlab.
- Ir a ascii art.

EJERCICIOS



EJERCICIO 01

Escriba un programa que le pida al usuario que ingrese el ancho y el largo de una habitación. Una vez leídos los valores, su programa debería calcular y mostrar el área de la habitación. La longitud y el ancho se ingresarán como números de punto flotante. Incluya unidades en su mensaje de solicitud y salida; pies o metros.

EJERCICIO 02

Cree un programa que lea la longitud y el ancho del campo de un agricultor del usuario en pies. Mostrar el área del campo en acres. 43.560 pies cuadrados en un acre.

EJERCICIO 03

Escriba un programa que lea un número entero ingresado por el usuario. Y luego, muestre un mensaje que indique si el número entero es par o impar.

EJERCICIO 04

Escriba un programa que solicite una letra del alfabeto e indique sí es una vocal o una consonante, si no es ninguna de las dos, entonces debe indicar que ese carácter no es válido.

EJERCICIO 05

Escrita un programa que de cómo resultado un archivo de texto con una matriz que muestre todas las combinaciones de multiplicación de los número 1 hasta 10.

EJERCICIO 06

Escriba una función que solicite tres números como parámetros y devuelva el valor mediano de los tres.

Nota: El valor mediano es el medio de los tres valores cuando se ordenan en orden ascendente.

EJERCICIO 07

Escriba un programa que lea enteros del usuario y los almacene en una lista. Su programa debe continuar leyendo valores hasta que el usuario ingrese 0. Luego, debe mostrar todos los valores ingresados por el usuario (excepto el 0) en orden de menor a mayor.

EJERCICIO 08

Escriba un programa que lee palabras del usuario hasta que el usuario ingresa la palabra fin. Su programa debe mostrar cada palabra ingresada por el usuario exactamente una vez.

EJERCICIO 09

Escriba un programa que cree dos archivos en formato txt: el primero debe imprimir los números enteros desde 0 hasta 100, el segundo debe imprimir los números enteros desde 50 hasta 200. Luego escriba otro programa que lea ambos archivos y genere un archivo nuevo que contenga sólo números pares organizados de mayor a menor, estos números no deben estar repetidos.

EJERCICIO 10

Elija un diseño de [ascii art](#). y escriba un programa para emular el comportamiento del objeto elegido.

TALLER 01



Taller 01.