### Guía de sintaxis de Python

Esta guía está diseñada para proporcionar ejemplos claros y básicos de las estructuras y sintaxis de programación en Python. Incluye desde estructuras de control hasta el uso de librerías específicas como Graphviz para generar archivos PNG.

## Índice

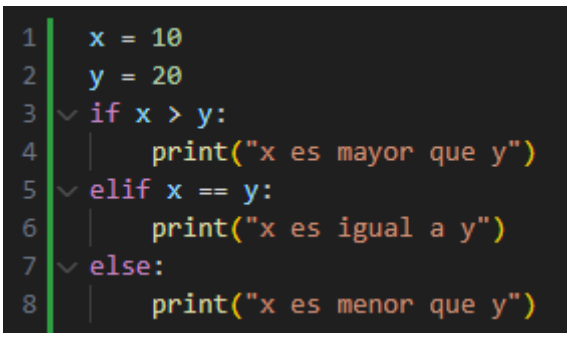


# 1. Estructuras de Control

## If, Elif, Else

La estructura de control `if`, `elif` y `else` se utiliza para tomar decisiones.

**Sintaxis:**



```python
x = 10
y = 20
if x > y:
    print("x es mayor que y")
elif x == y:
    print("x es igual a y")
else:
    print("x es menor que y")
```

### Bucles: For y While

#### For

Usamos `for` para iterar sobre elementos de una lista o rango.

**Sintaxis:**

```python
for i in range(5):
    print("Número:", i)
```

#### While

El bucle `while` se ejecuta mientras una condición sea verdadera.

**Sintaxis:**

```python
count = 0
while count < 5:
    print("Cuenta:", count)
    count += 1
```

## 2. Manejo de Listas y Diccionarios

### Listas

Son estructuras ordenadas que almacenan múltiples elementos.

**Sintaxis:**

```python
numeros = [1, 2, 3, 4]
numeros.append(5)
print(numeros)
print(numeros[0])
```

### Diccionarios

Son colecciones de pares clave-valor.

**Sintaxis:**

```python
alumnos = {"Juan": 8, "Ana": 9}
alumnos["Carlos"] = 7
print(alumnos)
```

# 3. Manejo de Pila (Stack)

Se utiliza para manejar los autómatas en el proyecto.

**Sintaxis:**

```python
stack = []
stack.append("a")
stack.append("b")
print(stack.pop())  # Devuelve 'b'
```

# 4. Manejo de Archivos

## Crear un archivo PNG con Graphviz

Usamos la librería Graphviz para generar grafos.

**Sintaxis:**

```python
from graphviz import Digraph
g = Digraph("Ejemplo")
g.node("A", "Inicio")
g.node("B", "Fin")
g.edge("A", "B", label="Transición")
g.render("grafo", format="png")
```

# 5. Manejo de Excepciones

Usamos try, except para manejar errores de forma segura.

**Sintaxis:**

```python
try:
    x = int(input("Ingrese un número: "))
    print(10 / x)
except ZeroDivisionError:
    print("No se puede dividir por cero.")
```

# 6. Funciones y Métodos

## Definición de Funciones

```python
def suma(a, b):
    return a + b
print(suma(2, 3))
```

**Métodos de Clase y @staticmethod**

```python
class Ejemplo:
    @staticmethod
    def mostrar():
        print("Método estático")
Ejemplo.mostrar()
```

## 7. Clases y Objetos

```python
class Persona:
    def __init__(self, nombre):
        self.nombre = nombre
    def saludar(self):
        print(f"Hola, soy {self.nombre}")
juan = Persona("Juan")
juan.saludar()
```

## 8. Expresiones Regulares

```python
import re
patron = r"a+b*"
texto = "aaab"
if re.fullmatch(patron, texto):
    print("Coincide")
```

## 9. Sintaxis de Graphviz para Grafos

```python
# Agregar Nodo:
g.node("q0", "Estado Inicial")

# Agregar Arista con Etiqueta:
g.edge("q0", "q1", label="a")

# Guardar como PNG:
g.render("automata", format="png")
```