

## Laboratorio 18

### Sesión #18 Introducción a Python, sintaxis básica y tipos de datos

**Título del Laboratorio:** Sintaxis Básica y Tipos de Datos en Python

**Duración:** 2 horas

**Objetivos del Laboratorio:**

1. *Comprender la sintaxis básica de Python.*
2. *Familiarizarse con la declaración y uso de variables de diferentes tipos de datos.*
3. *Aplicar operaciones matemáticas y manipulación de cadenas.*
4. *Trabajar con estructuras de datos como listas.*

**Materiales Necesarios:**

1. *Computadora con Python instalado (versión 3.6 o superior).*
2. *Un editor de texto (por ejemplo, Visual Studio Code, PyCharm o Jupyter Notebook).*
3. *Acceso a internet para consultar documentación y recursos adicionales.*

**Documentos de Ayuda:**

1. *Guía rápida de sintaxis de Python.*
2. *Referencias sobre tipos de datos en Python.*
3. *Ejemplos de código comentados.*

**Estructura del Laboratorio:**

1. Declaración de Variables:

- Crea variables de tipo ``string``, ``int``, ``float``, y ``boolean``.
- Captura por teclado los valores
- Imprime cada una de las variables en la consola.  
`nombre=input("Ingrese nombre:")`

2. Operaciones Matemáticas:

- Captura por teclado dos números enteros y realiza las operaciones de suma, resta, multiplicación, y división.
- Imprime los resultados.  
`numero=int(input("Ingrese un numero:"))`

### 3. Manipulación de Cadenas:

- Crea dos variables `string` y concaténalas para formar una frase completa.
- Captura por teclado los textos de las frases

```
frase1=input("Ingrese la frase 1:")
```

- Usa f-strings para crear una frase similar.

### 4. Trabajar con Listas:

- Crea una lista vacía, que contenga al menos diez elementos de diferentes tipos.

```
Lista=[]
```

- Accede y modifica elementos de la lista.
- Agrega un nuevo elemento a la lista y luego imprime la lista completa.
- Elimina un nuevo elemento a la lista y luego imprime la lista completa.

### 5. Uso de Diccionarios:

- Crea un diccionario que contenga información sobre una persona (nombre, edad, ciudad, y profesión).

- Accede y modifica los valores dentro del diccionario.

```
#modificar valores
```

```
dicci["edad"]=32
```

- Agrega un nuevo par clave-valor al diccionario y luego imprime el diccionario completo.

### 6. Conclusiones