

MODULO 1. INTRODUCCIÓN AL DESARROLLO DE SOFTWARE CON PYTHON

1. PROGRAMACIÓN ESTRUCTURADA CON PYTHON

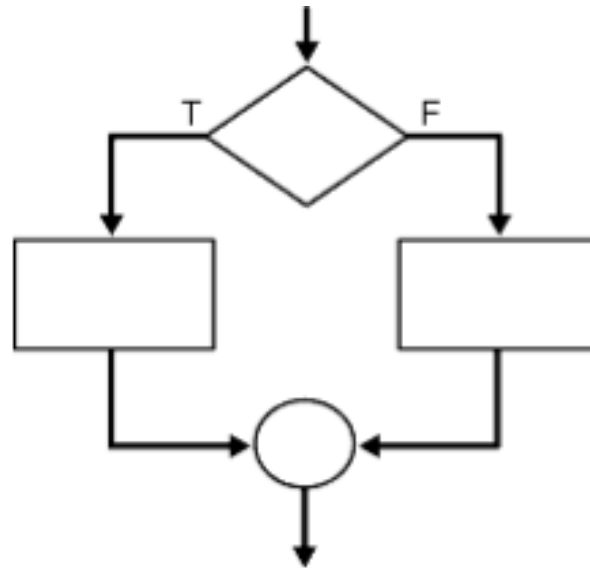
CONTENIDOS A DESARROLLAR

1. Estructuras de control
 - a) Decisión
 - b) Iteración
2. Manejo de colecciones de datos
 - a) Listas
 - b) Tuplas
 - c) Set (Frozenset)
 - d) Diccionarios
3. Manejo de excepciones
4. Funciones

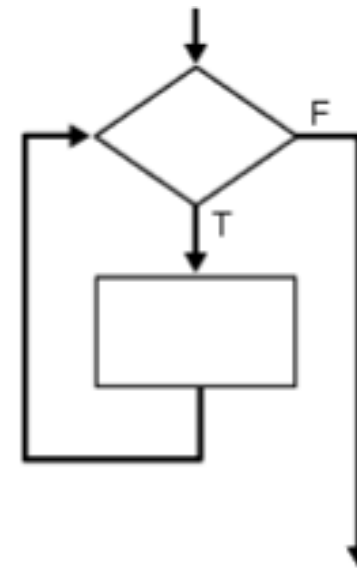
ESTRUCTURAS DE CONTROL

- Las estructuras de control, permiten **cambiar el flujo de ejecución de un programa**, haciendo que ciertos bloques de código se ejecuten si y solo si se dan unas condiciones particulares o se repitan una cantidad determinada o indeterminada de veces.

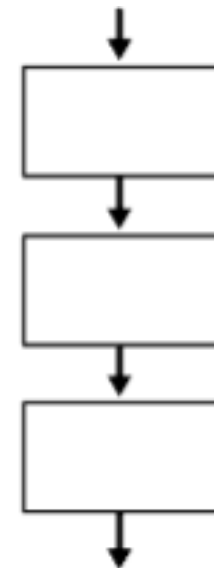
- If
- elif
- Else
- For
- while



Selection



Iteration



Sequence

MUTABILIDAD

Mutable: Si permiten ser modificados una vez creados.

Immutable: Si no permiten ser modificados una vez creados.

Son mutables los siguientes tipos: Listas, Bytearray, Memoryview, Diccionarios, Sets, Y clases definidas por el usuario

Y son inmutables: Booleanos, Enteros, Float, Frozenset, Cadenas, Tuplas, Range, Bytes

PRINCIPALES DIFERENCIAS

Las principales diferencias entre tipos mutables e inmutables son las siguientes:

- Los tipos inmutables son generalmente más rápidos de acceder. Por lo que si no piensas modificar una lista, es mejor que uses una tupla.
- Los tipos mutables son perfectos cuando quieres cambiar su contenido repetidas veces.
- Los tipos inmutables son caros de cambiar, ya que lo que se hace en realidad es hacer una copia de su contenido en un nuevo objeto con las modificaciones.

COLECCIONES DE DATOS

- Una colección de datos en programación almacena 2 o más elementos en un arreglo con distintos números de índice, por lo que nos ayuda a agrupar elementos que tengan algo que ver unos con los otros. Existen cuatro tipos de colecciones de datos en el lenguaje Python
- Listas
- Tuplas
- Set (Frozenset)
- Diccionarios
- Una colección permite agrupar varios objetos bajo un mismo nombre. Por ejemplo, si necesitamos almacenar en nuestro programa los nombres de los alumnos de un curso de programación, será más conveniente ubicarlos a todos dentro de una misma colección de nombre alumnos, en lugar de crear los objetos alumno1, alumno2, etc.

LISTAS

- Listas en Python
- Las listas en Python son un tipo de dato que permite almacenar datos de cualquier tipo. Son mutables y dinámicas, lo cual es la principal diferencia con los sets y las tuplas.
- `lista = [1, 2, 3, 4]`
- Una lista sea crea con `[]` separando sus elementos con comas `,`. Una gran ventaja es que pueden almacenar tipos de datos distintos.
- `lista = [1, "Hola", 3.67, [1, 2, 3]]`

PROPIEDADES DE LAS LISTAS

- Algunas propiedades de las listas:
- Son ordenadas, mantienen el orden en el que han sido definidas
- Pueden ser formadas por tipos arbitrarios
- Pueden ser indexadas con [i].
- Se pueden anidar, es decir, meter una dentro de la otra.
- Son mutables, ya que sus elementos pueden ser modificados.
- Son dinámicas, ya que se pueden añadir o eliminar elementos.

HACIENDO USO DE LAS LISTAS

El bucle for tiene una variante muy especial que puede procesar las listas de manera muy efectiva. Echemos un vistazo a eso.

```
miLista = [10, 1, 8, 3, 5]
```

```
suma = 0
```

```
for i in miLista:
```

```
    suma += i
```

```
print(suma)
```

TUPLAS

- **Tupla (tuple)**

Las tuplas en Python son un tipo o estructura de datos que permite almacenar datos de una manera muy parecida a las listas, con la salvedad de que son inmutables.

- lo que significa que no pueden ser modificadas una vez declaradas, y en vez de inicializarse con parentesis se hace con ().
- Dependiendo de lo que queramos hacer, las tuplas pueden ser más rápidas.

SET (FROZENSET)

- Los sets en Python son una estructura de datos usada para almacenar elementos de una manera similar a las listas, pero con ciertas diferencias.
- Los elementos de un set son únicos, lo que significa que no puede haber elementos duplicados.
- Los set son desordenados, lo que significa que no mantienen el orden de cuando son declarados.
- Sus elementos deben ser inmutables.
- El frozenset es inmutable

DICCIONARIOS

- Un diccionario en Python es una colección de elementos, donde cada uno tiene una llave key y un valor value.
- Los diccionarios se pueden crear con llaves `{}` separando con una coma cada par key: value.

EN RESUMEN

- **Lista** es una colección ordenada y modificable. Permite datos duplicados.
- **Tuple** es una colección ordenada e inmutable. Permite datos duplicados.
- **Set** es una colección que no posee un orden ni un índice. No existen datos duplicados.
- **Dictionary** es una colección sin orden, modificable e indexada. No permite datos duplicados.