

## INTRODUCCIÓN A LA PROGRAMACIÓN DESDE PYTHON

### INICIO

- ¿Qué es programar?
- ¿Por qué lo hacemos?
- Formas de programar (de la agenda al lenguaje ensamblador)
- Distintos lenguajes de programación
- Nociones de arquitectura de hardware (memorias y pilas)
- Juguemos un poco programando el movimiento de una pelotita en Python (utilizando <https://gobstones.github.io/>) como didáctica de soporte web, se aprenderán y aplicarán conceptos básicos de programación como variables, condiciones, funciones, agrupamiento y estructuras en un lenguaje muy amigable

### HERRAMIENTAS DE DESARROLLO

- ¿Qué es IDE? Ejemplos - Spyder, VisualStudio Code, IDLE, Pycharm.
- Instalación de Python Ejecutar IDEs en tu computadora y descripción de las herramientas principales incluidas
- ¿Cómo usar la consola interactiva?
- ¿Cómo crear y ejecutar archivos Python en IDEs?
- Debug de un programa
- Google Colab

### DATOS Y VARIABLES

- Tipos de datos
- Concepto de Variable
- Reglas que debes seguir para asignar nombres apropiados a las variables o Introducción a las Cadenas de Caracteres (strings)
- Función print ()
- Uso de la función len () para obtener el tamaño de una cadena de caracteres
- Estructura interna de las cadenas de caracteres
- Uso de índices para acceder a sus caracteres a través de la indexación
- Métodos de cadenas de caracteres y cómo usarlos
- Características principales de los tipos de datos numéricos (int y float)
- Uso de la función type ()
- Características de los valores booleanos (True y False)
- Función input ()
- Conversión de valores

### OPERADORES

- Operadores Aritméticos: suma, resta, multiplicación, división, división entera, módulo, exponenciación.
- Operadores Lógicos: and, or, y not.
- Operadores Relacionales: <=, >, >=, ==, !=
- Sentencias condicionales en Python (if, else, elif)
- Comentarios
- Practica y repaso general

### COLECCIONES DE DATOS: TUPLAS, LISTAS, DICCIONARIOS Y SETS

- Colecciones de datos o Tuplas o Listas o Diccionarios o Sets
- Ciclos for
- Cómo funcionan los ciclos for en Python.
- Uso de la función range () en los ciclos for
- Función y uso de las variables de control
- Uso de ciclos for para iterar sobre iterables (cadenas de caracteres, listas, tuplas, y diccionarios)

### CICLOS WHILE/BUCLE

- Utilización de ciclos while/bucle
- Tratamiento de errores
- Introducción a los conceptos de error y excepción.
- Error de Sintaxis (SyntaxError).
- Error de Índice (IndexError).
- Error de Clave (KeyError).
- Error de Nombre (NameError).
- Error de División por Cero (ZeroDivisionError)
- Error de Recursión (RecursionError)
- Control de errores (try, except, else y finally)

### FUNCIONES

- Concepto de programación funcional
- Ventajas
- Definición y llamado
- Parámetros y argumentos
- Retornando uno o varios valores
- Práctica y Repaso General

### LIBRERIAS

- Concepto de módulo y su importancia
- Uso de la sentencia import para importar módulos
- Cómo importar todos los elementos de un módulo
- Cómo importar elementos específicos de un módulo
- Cómo asignar un nombre específico a un módulo importado

## TRABAJAR CON ARCHIVOS EXTERNOS

---

- Uso de la sentencia with para abrir y trabajar con archivos.
- Modos de apertura de archivos en Python (leer, escribir, y crear)
- Documentación Python
- Cómo acceder a la documentación oficial
- Breve recorrido de las secciones buscadas frecuentemente
- Ejemplos prácticos de cómo encontrar información en la documentación
- Práctica y Repaso General

## INTRODUCCION A LA PROGRAMACION ORIENTADA A OBJETOS

---

- Introducción a Programación Orientada a Objetos
- Propósito de las clases en Python
- Cómo definir clases, atributos, y métodos en Python
- Concepto de instancia
- Cómo crear instancias y asignar valores iniciales a sus atributos
- Cómo acceder a los atributos de una instancia
- Cómo llamar a un método a través de una instancia
- Práctica y Repaso general