



## Curso de Python para Precolegiados – Syllabus

### 1. Introducción a la programación

- a. ¿Qué es la programación?
- b. ¿Qué es un programa?
- c. ¿Qué es un lenguaje de programación?
- d. Tipos de lenguajes
  - i. Lenguajes interpretados
  - ii. Lenguajes compilados
  - iii. Lenguajes semi-interpretados/compilados
- e. Proceso de ejecución de un programa
- f. Aplicaciones comunes de los lenguajes de programación

### 2. Introducción a Python

- a. Características generales del lenguaje
  - i. Sintaxis simple y clara
  - ii. Lenguaje multipropósito
  - iii. Alto nivel de abstracción
  - iv. Multiplataforma
  - v. Comunidad y ecosistema
  - vi. Programación orientada a objetos
- b. Principales áreas de aplicación
  - i. Análisis de datos
  - ii. Inteligencia Artificial y Machine Learning
  - iii. Automatización y scripting
  - iv. Desarrollo web, móvil y de escritorio
  - v. Testing y hacking ético
- c. Comparativa sintáctica con otros lenguajes
  - i. C#, JavaScript y Python
- d. Versiones de Python
  - i. Python 2 (legado)
  - ii. Python 3 (actual y recomendada)

### 3. Linting (análisis estático de código)

- a. ¿Qué es el linting?
- b. Cómo funciona un linter
- c. Detección de errores y advertencias
- d. Uso de la pestaña "Problems" en VS Code
- e. Ejemplos prácticos con linters populares (pylint, flake8)

### 4. Tipos de datos

- a. Literales y variables
- b. Tipos de datos no secuenciales
  - i. Booleanos
  - ii. NoneType
- c. Tipos de datos secuenciales, mutables e inmutables
  - i. Determinación de la longitud de una secuencia
  - ii. Índices
  - iii. Porciones (*slices*)



- iv. Valores literales y valores por referencia
- v. Métodos asociados a tipos de datos
- vi. Cadenas de caracteres
- vii. Listas
- viii. Tuplas
- ix. Diccionarios
- d. Operaciones y herramientas adicionales sobre datos
  - i. Función type
  - ii. Función isinstance
  - iii. Conversión de tipos de datos (*type casting*)

## 5. Operadores

- a. Operadores aritméticos
- b. Operadores de asignación
- c. Operadores de comparación
- d. Operadores lógicos
- e. Operadores adicionales
  - i. Operador de identidad
  - ii. Operador de pertenencia
  - iii. Operador de asignación por defecto

## 6. Estructuras condicionales

- a. Condicional simple
- b. Condicional completo
- c. Condicional completo compuesto
- d. Condicional ternario
- e. Conmutador (switch)
- f. Observaciones relevantes
  - i. Opcionalidad de estructuras condicionales
  - ii. Anidamiento de estructuras condicionales
  - iii. Cancelación del flujo de ejecución
  - iv. Confusiones léxicas

## 7. Estructuras de iteración

- a. Iteración de rango cerrado (sobre secuencia)
- b. Iteración de rango abierto (sobre condición)
- c. Estructuras generadas por iteración
  - i. List comprehension
  - ii. Tuple comprehension (generador)
  - iii. Dictionary comprehension
  - iv. Filtrado de comprensiones

## 8. Funciones

- a. Definición de funciones personalizadas
- b. Devolución de valores
- c. Argumentos
  - i. Funciones de argumentos fijos
    - 1. Argumentos posicionales
    - 2. Argumentos por defecto
  - ii. Funciones de argumentos variables



1. Argumentos posicionales
2. Argumentos por defecto

d. Docstrings

## **9. Clases**

- a. Atributos
- b. Métodos
  - i. Constructor de clase
  - ii. Métodos definidos por el usuario
  - iii. Métodos mágicos
- c. Mostrar atributos y métodos
- d. Visibilidad de atributos y métodos
- e. Decoradores
  - i. Getters
  - ii. Setters
- f. Herencia de clase

## **10. Excepciones**

- a. Traceback
- b. Manejo de excepciones
- c. Lanzamiento de excepciones

## **11. Librerías**

- a. Uso de librerías
  - i. Módulos de una librería
  - ii. Modificación de la variable de importación
- b. Matplotlib
- c. Numpy
- d. SciPy