



## PYTHON DE PRINCIPIANTE A EXPERTO – NIVEL INTERMEDIO

### I. INFORMACIÓN GENERAL

**Nombre del Curso:** Python de Principiante a Experto – Nivel Intermedio

**Duración:** 22 horas

**Profesor:** Alejandro Castro

**Requisitos:**

- Python de Principiante a Experto – Nivel Básico
- Conocimientos básicos de sistemas operativos: Windows, Linux o Mac.
- Matemáticas de nivel secundario.

### II. DESCRIPCIÓN DEL CURSO

El presente curso aborda los temas más comunes de nivel intermedio/avanzado del lenguaje Python, partiendo de los generadores, programación orientada a objetos, decoradores y librerías para el manejo de datos y gráficos hasta la aplicación de los conocimientos adquiridos en la implementación de un proyecto final.

### III. SUMILLA

Curso con un enfoque práctico donde el alumno aplicará de forma incremental los conocimientos adquiridos sobre programación utilizando el lenguaje de programación Python, finalizando con implementación de un proyecto final. Los temas abarcarán desde conceptos de nivel intermedio/avanzado tales como programación orientada a objetos, generadores y decoradores hasta las principales librerías de Python utilizadas para el manejo de datos.

### IV. OBJETIVOS

El principal objetivo del curso es darle herramientas potentes al alumno para que se pueda desempeñar como programador de Python capaz de implementar y/o trabajar en proyectos de mediana complejidad en dicho lenguaje, haciendo uso de técnicas modernas y las herramientas de trabajo más utilizadas y requeridas en la actualidad.



## V. PROGRAMA ANALÍTICO

### UNIDAD 1: GENERADORES (2h)

1. Generadores: definición y características
2. Comprensión de generadores
3. Sentencia yield en funciones

### UNIDAD 2: PROGRAMACIÓN ORIENTADA A OBJETOS (8h)

1. Clases, instancias, atributos y métodos
2. Constructores y destructores
3. Principios de la programación orientada a objetos
4. Herencia y polimorfismo
5. Sobrecarga de operadores
6. Variables y métodos de clase
7. Métodos estáticos
8. Creación de librerías con clases

### UNIDAD 3: DECORADORES (6h)

1. Decoradores: definición y características
2. Decoradores de funciones
3. Decoradores de clases
4. Principales decoradores predefinidos en Python
5. Getters y setters
6. Interfaces y clases abstractas

### UNIDAD 4: MANEJO DE DATOS CON NUMPY Y PANDAS (4h)

1. Numpy: utilidad y principales tipos de datos
2. Arreglos: definición y principales métodos
3. Indexando un arreglo
4. Funciones importantes de Numpy
5. Pandas: utilidad y principales tipos de datos
6. Series y dataframes: definición y principales métodos
7. Indexando un dataframe
8. Principales funciones para limpieza de datos



## UNIDAD 5: GENERACIÓN DE GRÁFICOS CON MATPLOTLIB (2h)

1. Principales tipos de gráficos
2. Manejo de estilos con rc.Params y estilos predefinidos
3. Manipulando título, leyendas y ejes
4. Mejorando el formato de los gráficos

### VI. METODOLOGÍA

La metodología que se seguirá en el presente curso será la siguiente: primero se desarrollará cada uno de los temas de manera teórica, con apoyo de herramientas digitales gráficas, luego se reforzarán estos conceptos mediante la implementación y la solución de retos relacionados a los temas de clase y los precedentes.

### VII. EVALUACIÓN

Fórmula para el cálculo de la nota final. Cada una de las 5 unidades tendrá tareas de programación que una vez finalizado el curso podrán juntarse y complementarse para desarrollar un proyecto final planteado por el docente:

$$NOTA FINAL = \frac{1}{5} \sum_{i=1}^5 E_i + P$$

NOTA FINAL: Promedio final del curso

$E_i$ : Nota correspondiente a la  $i$ -ésima tarea de programación. Se califican de 0 a 12.

$P$ : Nota correspondiente a la presentación final del proyecto. Se califica de 0 a 8.

El docente seleccionará el medio de recepción de las evaluaciones y tomará en cuenta la asistencia y participación si así lo desea.