



# Curso Superior de Python

Aprende los conocimientos y herramientas necesarias para entender el **funcionamiento y aplicaciones de Python**, uno de los **lenguajes de programación** más importantes en la industria. A lo largo del curso, se realizarán ejemplos financieros para ilustrar cada concepto, desde la **creación de modelos de inversión** hasta el análisis de grandes **datasets financieros**.

## Aspectos destacables y objetivos

- ✓ Comprender los fundamentos de Python, desde los conceptos básicos hasta sus **aplicaciones prácticas en el mundo financiero**
- ✓ Conocer cómo definir y utilizar variables y asignaciones para **gestionar y manipular datos de manera eficiente**
- ✓ Entender y manejar las entradas y salidas estándar para **interactuar con el usuario y procesar datos**
- ✓ Aplicar **técnicas de modularidad** y funciones para **crear código reutilizable** y bien estructurado
- ✓ Explorar el **control de flujo en Python**, incluyendo condicionales y bucles, para **desarrollar algoritmos dinámicos**
- ✓ Introducirse en la **programación orientada a objetos** para estructurar el código de forma clara y manejable
- ✓ Gestionar y manejar excepciones para **crear programas robustos y libres de errores**
- ✓ Descubrir las **capacidades de NumPy**, una biblioteca fundamental para el cálculo numérico y la manipulación de grandes volúmenes de datos
- ✓ Familiarizarse con **Pandas**, una herramienta esencial para la manipulación y análisis de datos financieros



Modalidad Online



Título de Curso Superior



275€

## Claustro

**Jose Lope Alba**

Head of Quantitative Analysis, Wind to Market, S.A.

## Programa

### 1. Introducción a Python

- 1.1. ¿Qué es Python? Historia y aplicaciones prácticas
- 1.2. Configuración del entorno de desarrollo
- 1.3. Primer programa: ¡Hola Mundo!
- 1.4. Información adicional sobre recursos y comunidades en línea

### 2. Fundamentos de Python

- 2.1. Sintaxis básica y tipos de datos fundamentales (números, strings, booleanos)
- 2.2. Tipos de datos compuestos: listas, diccionarios, tuplas, conjuntos
- 2.3. Operadores aritméticos, comparativos, lógicos y de asignación
- 2.4. Casting y modificación de tipos: importancia y consideraciones

### 3. Variables y Asignaciones

- 3.1. Definición y asignación de variables
- 3.2. Convenciones de nombres y buenas prácticas
- 3.3. Uso común de None

### 4. Entradas y Salidas Estándar

- 4.1. Lectura de datos desde el teclado
- 4.2. Impresión de resultados y formateo de salida
- 4.3. Uso de print() y formateo avanzado de strings (f-strings, método format(), operador %)

### 5. Funciones y Modularidad

- 5.1. Creación y llamada de funciones: definiciones y llamadas
- 5.2. Parámetros, argumentos y valores de retorno
- 5.3. Funciones lambda para operaciones rápidas y anónimas

### 6. Control de Flujo

- 6.1. Uso de if, else y elif para decisiones
- 6.2. Bucles for y while para iteraciones
- 6.3. Comprensiones de listas

### 7. Programación Orientada a Objetos

- 7.1. Introducción a clases y objetos
- 7.2. Atributos, métodos y propiedades
- 7.3. Herencia y polimorfismo

### 8. Manejo de Excepciones

- 8.1. Conceptos básicos de las excepciones
- 8.2. Uso de bloques try y except para control de errores
- 8.3. Definición de excepciones personalizadas

### 9. Introducción a NumPy

- 9.1. Instalación y activación de la librería numpy
- 9.2. Creación y manipulación de ndarrays: creación, tipos de datos, operaciones aritméticas
- 9.3. Indexación y slicing en ndarrays
- 9.4. Operaciones avanzadas: transposición, concatenación, división y uso de funciones numpy

### 10. Introducción a Pandas

- 10.1. Instalación y activación de la librería pandas
- 10.2. Creación y manipulación de Series y DataFrames
- 10.3. Operaciones aritméticas, transposición, indexación y slicing en Series y DataFrames
- 10.4. Lectura de ficheros: uso de pd.read\_excel y pd.read\_csv para leer archivos .xlsx y .csv

## A quién va dirigido

-  Estudiantes **universitarios de informática, ingeniería, ciencias de datos, matemáticas, finanzas, economía**, y otras áreas que pueden beneficiarse del conocimiento en programación y análisis de datos
  -  Profesionales que **trabajen en finanzas, consultoría, marketing, operaciones o recursos humanos**, y que deseen adquirir habilidades técnicas para mejorar la automatización de procesos, análisis de datos, o manejo de grandes volúmenes de información
  -  Emprendedores interesados en **desarrollar sus propias aplicaciones** o que deseen comprender mejor cómo usar la tecnología en sus startups o empresas para optimizar procesos o crear productos digitales
  -  **Ingenieros, analistas de datos, y desarrolladores** que estén familiarizados con la tecnología y busquen formalizar o expandir sus conocimientos en Python para aplicar a desarrollo web, inteligencia artificial, análisis de datos o automatizaciones
  -  **Académicos e investigadores** que necesiten Python para análisis de datos, simulaciones, o procesamiento de información
- Personas que deseen aprender Python para hacer una transición a trabajos más orientados a la tecnología, como desarrollo de software, ciencia de datos o análisis financiero