

# INTRODUCCIÓN A LA PROGRAMACIÓN EN PYTHON

Duración: 35 Horas | 10 Unidades

Modalidad: Online Teleformación (Asíncrono)

Requisitos: desarrolladores con al menos 1 o 2 años de experiencia en departamentos TI y de desarrollo.

Objetivos:

- Adquirir los conocimientos y destrezas necesarias para empezar a programar con Python.
- Adquirir conceptos necesarios para el análisis, diseño y desarrollo de algoritmos básicos y su respectiva implementación en el lenguaje de programación Python.
- Identificar los elementos del lenguaje, manejar la sintaxis y prototipar tus propios proyectos.
- Desarrollar la capacidad para analizar, diseñar e implementar soluciones computacionales de baja y media complejidad.

Presentación:

Python es un lenguaje de programación de propósito general muy popular. Es fácil de aprender y rápido para implementar prototipos. Python es un lenguaje de programación fácil e intuitivo pensado para que su sintaxis sea lo más parecido posible al lenguaje humano. Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, también, programación funcional. Es un lenguaje Multiplataforma ya que puede ser ejecutado en distintos sistemas operativos, también es Software libre: con lo esto implica. mPosee también una gran comunidad de desarrollo tanto en inglés como en español.

Todo esto hace que programadores con conocimientos en Python pueden aumentar sus posibilidades para trabajar en empresas u organizaciones que empleen nuevas tecnologías como el desarrollo de aplicaciones web y de escritorio, software testing, videojuegos, animación 3D, arquitectura, administración de sistemas y redes, seguridad informática, Big Data, Business Intelligence (Inteligencia de Negocio), Sistemas de Información Geográfica, ciencia aplicada...

## TEMARIO

### MÓDULO 1: PYTHON 3. INTRODUCCIÓN Y ENTORNOS

- Python 3: Introducción y entornos
- Introducción
- Sobre el curso
- ¿Qué es Python?
- Características
- Entorno para el trabajo con Python
- Linux
- Python en Windows
- Intérprete de Python
- Visual Studio Code
- Instalación

- Proceso de instalación en Linux
- Proceso de instalación en windows
- Espacio de trabajo para Python 3
- Extensiones para el desarrollo de Python
- Iniciar un nuevo programa Python
- Introducción del código Python
- Running Python Code
- Cómo vas a trabajar
- Requisitos y nomenclaturas para ejecución de los ejercicios
- Nombres de archivos
- Reglas básicas
- Resumen

## MÓDULO 2: CONCEPTOS BÁSICOS DE PROGRAMACIÓN EN PYTHON

- Conceptos básicos de programación en Python
- Definiciones y conceptos
- Algoritmos
- Diagramas de flujo
- Pseudocódigo
- Programador
- Programa Informático
- Diferencia entre algoritmo y programa informático
- Tipos de lenguajes de programación
- Bajo nivel
- Alto nivel
- Compiladores e intérpretes
- Compiladores
- Intérpretes
- Python3: sesiones interactivas
- Iniciar sesión interactiva
- Finalizar una sesión interactiva
- Operadores aritméticos
- Sumas y Restas
- Multiplicación y división
- Otros operadores
- Precedencia de los operadores aritméticos
- Operadores lógicos
- Operadores de Comparación
- Tabla de todos los operadores aritméticos, lógicos y de comparación
- Variables
- Reglas y convención de nombres
- Asignación compacta
- Resumen

### MÓDULO 3: ENTRADA, SALIDA Y TIPOS DE DATOS

- Entrada, Salida y Tipos de Datos
- Tipos de datos
- Comentarios
- Números
- Operaciones
- Algunos métodos para trabajar con números
- Textos
- Manipular textos
- Print()
- Print() -> Texto
- Print() -> Números
- Print() -> Números y Textos
- Cadenas "f"
- Input()
- Ejercicio guiado 1
- El problema
- Paso 1: Abrir el programa
- Paso 2: Crear archivo .py
- Paso 3: Resolver el problema
- Paso 4: Programar la solución
- Paso 5: Ejecutar
- Analizando el código
- Ejercicio guiado 2
- 1.- Crear el archivo
- 2.- Resuelve en problema
- 3.- Programamos la solución
- Analizando el código
- Resumen

### MÓDULO 4: ESTRUCTURA DE DATOS

- Estructura de datos
- Listas
- Slicing
- Especificación de índices para manejo de los elementos de una lista
- Funciones básicas para manejo de listas
- Operadores para manejo de listas
- Funciones especiales para manejo de listas
- Tuplas
- Slicing
- Especificación de índices para manejo de los elementos de una tupla
- Manejo de tuplas
- Funciones especiales para manejo de tuplas

- Operadores para manejo de tuplas
- Diccionarios
- Métodos para Diccionarios
- Operadores de identidad
- Resumen

## MÓDULO 5: CONTROL DE FLUJO EN PYTHON I (IF, ELIF, ELSE)

- Control de flujo en Python I (if, elif, else)
- Control de flujo
- Estructuras de Decisión
- Identación
- Ejecución condicionada de un bloque IF
- Ejecución selectiva entre dos bloques de instrucciones IF - ELSE
- Ejecución selectiva entre dos bloques de instrucciones IF - ELSE II. Decisiones anidadas
- Decisiones consecutivas IF ELIF ELSE
- Resumen

## MÓDULO 6: CONTROL DE FLUJO EN PYTHON II (FOR - WHILE)

- Control de flujo en Python II (for - while)
- FOR
- For controlado con secuencia a través de una lista
- For controlado con secuencia a través de un rango
- Ejercicios guiados
- FOR (Tabla de multiplicar)
- FOR II (Números primos)
- While
- Ejemplo de uso del bucle WHILE I (como contador)
- Ejemplo de uso del bucle WHILE (evento)
- Analiza el código
- WHILE infinito
- Ejemplo de bucle infinito por error
- Bucles anidados
- Analiza el código
- Break, Continue, Else & Pass
- Break
- Continue
- Else
- Pass
- Resumen

## MÓDULO 7: FUNCIONES EN PYTHON

- Funciones en Python
- Definir funciones en Python
- Documentar funciones en Python
- Los parámetros
- Orden de los parámetros (keywords)
- Parámetros por defecto
- Parámetros múltiples indefinidos
- Parámetros múltiples por clave-valor
- Ámbitos
- Ámbito global
- Ámbito nonlocal
- Ámbitos locales
- Búsqueda de nombres entre ámbitos
- La expresión global
- Retorno de datos
- Retorno múltiple
- Funciones de orden superior
- Filter y Map
- Funciones Lambda
- Funciones Recursivas
- Funciones generadoras e iteradores
- Sugerencias generales para programar con funciones
- Resumen

## MÓDULO 8: USO DE MÓDULOS Y PAQUETES EN PYTHON

- Uso de módulos y paquetes en Python
- Módulos
- Localización de módulos
- Archivos compilados de Python
- Creando un módulo
- `__name__`
- Importación de módulos
- Estructura de un Módulo
- Funciones `help()` y `dir()`
- Ejercicio guiado para la realización y utilización de módulos
- `usuario.py`
- `contrasena.py`
- `validador.py`
- Paquetes
- Módulos de Python
- Módulo `Random`
- Módulo `datetime` y `time`

- Módulos
- Resumen

## MÓDULO 9: PROGRAMACIÓN ORIENTADA A OBJETOS CON PYTHON

- Programación orientada a objetos en Python
- Pensar en POO
- Cualidades de un objeto
- El objeto es (CUALIDADES)
- El objeto tiene
- Objetos que comparten características
- El objeto puede (CAPACIDADES)
- Algo de vocabulario
- Programación Orientada a Objetos o POO
- Abstracción
- Encapsulamiento
- Polimorfismo
- Herencia
- La Clase
- Atributos
- Métodos
- Método `__init__`
- El Objeto
- Acceso al interior de un objeto
- Herencia
- Subclase
- Métodos y atributos de la subclase
- El nuevo objeto
- Herencia Múltiple
- Manteniendo el código privado (Encapsulación)
- Métodos especiales
- Método `__str__`
- Métodos para operar matemáticamente
- Métodos Get Set & Delete
- `@Property` en python
- Método `__len__`
- Resumen

## MÓDULO 10: MANEJO DE ERRORES, EXCEPCIONES Y ARCHIVOS

- Manejo de errores, excepciones y archivos
- Errores
- Error de sintaxis
- Excepciones o errores lógicos
- Análisis de una excepción

- La función
- Llamada sin errores
- Llamada con errores
- Analizando el error
- La ruta
- Tipo y causa del error
- Levantar excepciones
- Manejando excepciones
- Excepción básica o general
- Utilización de banderas
- Excepciones por tipo
- Detalles de error
- Excepciones propias
- La expresión finally
- Manejo de Archivos
- Modos al abrir un archivo
- Abriendo el archivo
- Métodos para manejo de archivos
- Ejemplo de gestión de archivos
- Analiza el código
- Últimas anotaciones sobre manejo de archivos
- with
- Borrar mover y renombrar archivos
- Resumen