

PROGRAMACIÓN ORIENTADA A OBJETOS PHYTON INTERMEDIO

RES.CD.:150 /24



✳ PRESENTACIÓN

Bienvenidos al curso de Python Intermedio. Este curso está diseñado para aquellos que ya tienen una base en Python y desean profundizar sus conocimientos para desarrollar aplicaciones más complejas y robustas.

✳ MODALIDAD

- ✓ Presencial

✳ OBJETIVOS

- Profundizar en la Programación Orientada a Objetos (OOP): Comprender conceptos avanzados como herencia múltiple, mixins y patrones de diseño.
- Manejo Avanzado de Datos: Trabajar con estructuras de datos más complejas y técnicas avanzadas de manipulación de datos.
- Desarrollo de Aplicaciones: Crear aplicaciones más sofisticadas utilizando módulos y librerías avanzadas.
- Automatización y Scripts Avanzados: Desarrollar scripts más complejos para automatizar tareas y procesar datos de manera eficiente.
- Conexión a Bases de Datos: Conectar y manipular bases de datos utilizando bibliotecas como sqlite3 y SQLAlchemy.
- Programación Asíncrona: Introducción a técnicas de programación asíncrona para mejorar el rendimiento y la eficiencia de tus aplicaciones

✱ TEMARIO

➤ **Clase 1: Introducción a la Programación Orientada a Objetos**

Presentación del curso

¿Qué es la OOP?

Comparación con la programación procedural

Ventajas de la OOP

Instalación y configuración

Instalación de Python y un IDE (como PyCharm o VSCode)

Configuración del entorno

Primeros pasos en Python

Sintaxis básica

Variables y tipos de datos

Operadores básicos

Ejercicios prácticos

Escribir y ejecutar scripts simples en Python

➤ **Clase 2: Clases y Objetos**

Concepto de clases y objetos

Definición de clase

Creación de objetos

Atributos y métodos

Atributos de instancia

Métodos de instancia

El constructor `__init__`

Uso y propósito

Ejemplos prácticos

Ejercicios prácticos

Crear clases y objetos simples

Definir y utilizar atributos y métodos

CAPACITACION EXTRACURRICULAR FACULTAD REGIONAL VENADO TUERTO

➤ **Clase 3: Herencia**

Concepto de herencia

Definición y propósito

Sintaxis en Python

Herencia simple y múltiple

Ejemplos de herencia simple

Introducción a la herencia múltiple

Sobrescritura de métodos

Cómo y por qué sobrescribir métodos

Ejemplos prácticos

Ejercicios prácticos

Crear una jerarquía de clases

Implementar herencia y sobrescritura

➤ **Clase 4: Polimorfismo y Encapsulamiento**

Polimorfismo

Concepto y ejemplos

Polimorfismo en métodos y operadores

Encapsulamiento

Visibilidad de atributos y métodos

Uso de getters y setters

Propiedades (@property)

Uso y ventajas de las propiedades en Python

Ejemplos prácticos

Ejercicios prácticos

CAPACITACION EXTRACURRICULAR FACULTAD REGIONAL VENADO TUERTO

Implementar polimorfismo en una jerarquía de clases

Encapsular atributos y utilizar getters y setters

➤ **Clase 5: Composición y Agregación**

Composición

Definición y ejemplos

Diferencias con la herencia

Agregación

Concepto y ejemplos

Relaciones entre objetos

Diagrama de clases UML

Introducción a UML

Cómo representar relaciones de composición y agregación

Ejercicios prácticos

Implementar composición y agregación en un proyecto simple

Dibujar un diagrama de clases UML

➤ **Clase 6: Proyectos Integradores y Buenas Prácticas**

Proyecto integrador

Presentación de un proyecto final que integre todos los conceptos vistos

Desarrollo del proyecto en clase

Buenas prácticas en OOP

Principios SOLID

Documentación y comentarios de código

Uso de herramientas de control de versiones (Git)

Resolución de dudas y preguntas

CAPACITACION EXTRACURRICULAR FACULTAD REGIONAL VENADO TUERTO

Espacio para resolver dudas y preguntas de los estudiantes

Evaluación final

Presentación y discusión del proyecto final

Evaluación del curso y feedback

*** DÍAS DE CURSADA**

- Dos días por semana

*** DURACIÓN**

- **Carga horaria semanal:** 4 hs.
- **Duración total en clases:** 6 clases.
- **Duración total en horas:** 24 hs.

*** DOCENTE**

- A confirmar.

*** MATERIAL DIDÁCTICO**

- Apuntes a cargo del docente.

*** REQUISITOS**

- Sin requisitos previos.

*** RÉGIMEN DE ASISTENCIA Y CALIFICACIÓN**

- **Promoción:** Asistencia y participación en clase. Entrega de Prácticos.
- **Asistencia al 80 % de las clases.**
- **Calificación:** Aprobado o Desaprobado.