

1 Introducción

Este curso, es una completa introducción al lenguaje de programación Python, la biblioteca estándar y técnicas de programación Python. Aunque el curso no asume ninguna experiencia previa con Python, el curso se centra en gran medida de las aplicaciones prácticas que incluyen secuencias de comandos, análisis de datos, matemáticas y administración de sistemas.

2 Silabus

El objetivo del curso es cubrir los módulos y bibliotecas del lenguaje Python, para escribir programas útiles. Un enfoque principal del curso es sobre el uso de Python para realizar análisis de datos (estructuras de datos, realizar cálculos, archivo de E/S, formatos de archivos, procesamiento de texto ...)

1. **Introducción a Python** Una introducción al lenguaje de programación Python. Cubre los detalles de cómo iniciar y detener el intérprete y la escritura de programas. Presenta los tipos de datos básicos de Python, archivos, funciones y manejo de errores.
2. **Trabajando con datos** Un recorrido detallado de la forma de representar y trabajar con datos en Python. Cubre tuplas, listas, diccionarios y conjuntos. Aprenderemos cómo utilizar con eficacia las listas por comprensión. Esta sección cubre aspectos críticos del modelo de objetos subyacente de Python, incluyendo variables, el recuento de referencias, copia y verificación de tipos.
3. **Programas y Funciones** Información acerca de cómo organizar programas más grandes en funciones. Un aspecto importante de esta sección es sobre la forma de diseñar las funciones que son confiables y pueden ser fácilmente reutilizadas en otros entornos. Abarca también los detalles técnicos de las funciones incluyendo reglas de ámbito y cadenas de documentación.
4. **Módulos y Librerías** Esta sección trata de cómo organizar los programas en módulos y los detalles sobre el uso de los módulos como una herramienta para la creación de programas extensibles. Concluye con un recorrido por algunos de los módulos de biblioteca más utilizados, incluyendo las relacionadas con la administración de sistemas, procesamiento de texto, subprocessos y bases de datos.
5. **Python Numérico** Conocer las herramientas de cómputo científico de Python, módulos para generar código en paralelo con GPU'S **PyOpenCL** y paralelizar con MPI.
6. **Clases y Objetos** Una introducción a la programación en Python orientado a objetos. Se describe cómo crear nuevos objetos, sobrecarga de operadores, y utilizar métodos especiales de Python. También se menciona los principios básicos de la programación orientada a objetos, incluyendo la herencia y composición.
7. **Dentro del sistema de objetos Python** Una revisión detallada de cómo se implementan los objetos en Python. Los temas principales incluyen la representación de objetos, atributos, herencia, administración de la memoria vinculante, y las propiedades especiales de las clases incluyendo propiedades, slots y atributos privados.
8. **Prueba, depuración, y la práctica de desarrollo de Software** Esta sección trata sobre muchos asuntos que se consideran importantes para el desarrollo de software en Python. Esto incluye el uso eficaz de las cadenas de documentación, pruebas del programa utilizando tanto

los módulos *doctest* y *unittest*, y el uso eficaz de las aserciones. También se describen el depurador de Python y el *profile*.

9. **Iteradores y Generadores** Cubre el protocolo de iteración, objetos iterables, generadores y expresiones generadoras. Un aspecto importante de esta sección se refiere a la utilización de generadores de crear tuberías de procesamiento de datos - una técnica eficaz para tratar una amplia variedad de problemas de programación común de sistemas (por ejemplo, el procesamiento de grandes archivos de datos, manejo de flujos de datos infinitos, etc.)
10. **Otros Tópicos** Una variedad de temas de programación más avanzadas, incluyendo funciones anónimas (lambda), los cierres (closures), funciones de argumentos variables, los decoradores, los métodos de clase estáticos, y los paquetes.
11. **Integración con otros lenguajes** Un estudio de la forma en que Python es capaz de interactuar con programas escritos en otros lenguajes de programación. Los temas incluyen la programación de redes, el acceso a código C, las extensiones COM, Jython y IronPython.