

Python Intermedio	Código	Emisión	Actualización	Versión
			30/01/2022	
Gerencia	Equipo	Cargo Objetivo del Curso	Duración Propuesta del Curso	Páginas
Operaciones	IT CLOUD Learning	Programación Media Python (Partes I y II)	18 hs. de clases virtuales sincrónicas	2

Descripción	Motivación
<p>El curso de Programación Media Python está planificado para continuar los cursos de programación de nivel inicial. Además de continuar con el paradigma de la Programación Estructurada, el curso introduce al paradigma de la Programación Funcional y también se ven conceptos muy básicos del paradigma de la Programación Orientada a Objetos.</p> <p>Python fue elegido para el curso porque tiene el mayor crecimiento en la última década, y cuenta con una enorme comunidad de programadores para su documentación, capacitación y creación de librerías de los más diversos temas.</p> <p>Recorre 5 conceptos populares del nivel medio: Estructuras de datos y de control, funciones y módulos, excepciones y archivos, tipos de datos complejos y programación funcional.</p> <p>Finalizado el curso, el estudiante podrá iniciarse en otros lenguajes de programación y entornos de trabajo (Framework) de la más variada orientación: Web, backend, frontend, app móviles, juegos, sistemas embebidos, ciencia de datos (data science), big data, inteligencia artificial, etc.</p>	<p>¿Por qué aprender a programar en Python?</p> <p>Desde principios de los años 90 del siglo XX, Python viene creciendo y evolucionando a la par de otros lenguajes populares como Java y C++. Pero Python es el lenguaje de mayor crecimiento en la última década, y muy pronto estará entre los lenguajes de mayor aplicación.</p> <p>Otra característica atractiva de Python es la de ser versátil en cuanto a su especialidad. Y es que Python creció tanto en los últimos años, que prácticamente no hay área en la cual no se hayan desarrollado librerías, Framework, entornos de trabajos, etc. Actualmente las áreas de programación más populares encuentran en Python un aliado estratégico muy importante: Big Data, Data Science, IA, Web Apps, Games, Robótica, y un largo etc.</p> <p>Es fácil observar que gran parte del código actual y el que se va a escribir en el futuro inmediato, será codificado en lenguaje Python.</p> <p>El futuro, es HOY.</p>

Contenidos del Curso

Tema 1 - Excepciones y archivos

1. Errores de ejecución - Excepciones
2. Manejo de excepciones - La estructura try - except - finally
3. Generación de excepciones - raise
4. Afirmaciones (puntos de test) - assert
5. Archivos
6. Función open() y método .close()
7. Modos de apertura
8. Métodos .read() , .readline() y .readlines()
9. Método .write() y .writelines()
10. Uso de archivos con try/finally o con with

Práctica: Ejercicios sencillos para fijar los contenidos y desafíos integradores.

Tema 2 - Otros tipos de datos

1. Objeto vacío - None
2. Diccionarios - Método .get() - Valor alternativo
3. Tuplas
4. Segmentos
5. Listas inteligentes (listas generadas por reglas y condiciones)
6. Formateo de cadenas
7. Métodos de cadenas: .join() .split() .replace() .startswith() .endswith() .upper() .lower()
8. Funciones numéricas: min() max() abs() sum()
9. Funciones de listas: all() any() enumerate()
10. Conjuntos - Métodos .add() .remove() y .pop()
11. Operaciones con conjuntos: unión, intersección, diferencia y diferencia simétrica.

Práctica: Ejercicios sencillos para fijar los contenidos y desafíos integradores.

Tema 3 - Programación funcional

1. Funciones Lambda (funciones anónimas)
2. Asignación de funciones Lambda
3. Funciones de orden superior de los iterables: map() y filter()
4. Generadores - yield
5. Decoradores @
6. Recursividad
7. Módulo itertools: funciones count() cycle() repeat() takewhile() chain() accumulate()
8. Funciones combinatorias: product() permutations()

Práctica: Ejercicios sencillos para fijar los contenidos y desafíos integradores.