

CTOP: AN4554 - Iniciación a la programación en Python	Mapeo
	CP2-2526-Fundamentos de Python 1 CP2-2526-Fundamentos de Python 2
RA1. Comprende los conceptos básicos de programación.	
a) Se han identificado correctamente los componentes de Python.	PYTHON 1 – Mod. 1.1.3
b) Se han descrito con precisión las diferencias entre los paradigmas de programación.	PYTHON 2 – Mod. 3
c) Se ha aplicado la sintaxis básica en ejemplos prácticos en Python.	PYTHON 1 – Mod. 2
d) Se han resuelto ejercicios teóricos sobre conceptos fundamentales.	PYTHON 1 – Mod. 1.1
e) Se ha participado activamente en debates sobre el uso de lenguajes de programación. (foro)	Proponer debates en clase
RA2. Desarrolla algoritmos para resolver problemas simples.	
a) Se han creado diagramas de flujo que representen algoritmos de manera efectiva.	PYTHON 1 – Mod. 3.1.9
b) Se ha escrito código en Python que implemente algoritmos diseñados.	PYTHON 1 – Mod. 3.1.9
c) Se han realizado pruebas para validar la funcionalidad de los algoritmos.	PYTHON 1 – Mod. 3.1.10, 4.7.10, 4.7.11
d) Se han identificado y solucionado errores en algoritmos.	PYTHON 1 – Mod. 3.1.8 y 4.7.8
e) Se ha explicado el proceso de desarrollo de un algoritmo de forma clara.	PYTHON 1 – Mod. 3.1.9
RA3. Crea programas sencillos aplicando estructuras de control en Python.	
a) Se han implementado correctamente estructuras de control en programas en Python.	PYTHON 1 – Mod. 3.1.7, 3.1.11, 3.1.12
b) Se han realizado entradas y salidas de datos de forma adecuada.	PYTHON 1 – Mod. 2.1 y 2.6
c) Se han depurado errores en programas de manera eficiente.	PYTHON 1 – Mod. 4.7.11
d) Se ha evaluado el rendimiento de los programas desarrollados.	PYTHON 1 – Mod. 4.7.10
e) Se ha documentado el código de manera clara y comprensible.	PYTHON 1 – Mod. 2.5
RA4. Implementa estructuras de datos como listas, tuplas y diccionarios.	
a) Se han declarado y manipulado listas, tuplas y diccionarios	PYTHON 1 – Mod. 3.4 y 4.6
b) Se han aplicado métodos y funciones integradas para el manejo de estructuras de datos (append(), pop(), keys(), etc.).	PYTHON 1 – Mod. 3.4, 3.5, 3.6 / 4.6
c) Se han implementado estructuras de datos en programas para almacenar y procesar información.	PYTHON 1 – Mod. 3.4 y 4.6 PYTHON 2 – Mod. 3.2.2
d) Se ha comprobado la eficiencia de las estructuras de datos en diferentes escenarios.	PYTHON 1 – Mod. 3.4 y 4.6
e) Se han integrado estructuras de datos avanzadas (como listas de listas o diccionarios anidados) para resolver problemas más complejos de almacenamiento y acceso a la información	PYTHON 1 – Mod. 3.7 PYTHON 2 – Mod. 4.6.5
RA5. Desarrolla aplicaciones básicas utilizando programación orientada a objetos e integración con bases de datos.	
a) Se han definido clases y creado objetos utilizando el paradigma de programación orientada a objetos.	PYTHON 2 – Mod. 3.1
b) Se han implementado atributos y métodos en clases, aplicando encapsulación.	PYTHON 2 – Mod. 3.3 y 3.4
c) Se ha utilizado herencia para optimizar la reutilización del código.	PYTHON 2 – Mod. 3.5
d) Se ha conectado Python con bases de datos SQL mediante SQLite3 u otro gestor.	Contenidos y ejercicios
e) Se han implementado consultas básicas (CRUD) en bases de datos desde un programa en Python.	Contenidos y ejercicios
f) Se han aplicado principios de diseño orientado a objetos para mantener la modularidad y escalabilidad del código, asegurando una fácil integración con bases de datos y otros sistemas.	Contenidos y ejercicios