

PLAN DOCENTE DE LA ASIGNATURA

Curso académico: 2026/2027

Identificación y características de la asignatura					
Código	402295				
Denominación (español)	Python para Ingenieros Introductorio				
Denominación (inglés)	Python for Engineers Introductory				
Titulaciones	Máster Universitario en Informática Industrial y Robótica				
Centro	Escuela de Ingenierías Industriales				
Módulo	UEX – Sistemas Ciberfísicos				
Carácter	Optativa	ECTS	3	Semestre	1º
Profesor					
Apellidos, Nombre		Despacho		Correo-e	
Herrero Agustín, José Luis		B2.8		jherrero@unex.es	
Área de conocimiento	Lenguajes y Sistemas Informáticos				
Departamento	Ingeniería de Sistemas Informáticos y Telemáticos				

Resultados de aprendizaje
<u>Conocimientos o contenidos</u> OPT-CON4 - Identificar las estructuras de datos más adecuadas para las diferentes aplicaciones. <u>Habilidades o destrezas</u> OPT-HAB4 - Programar en Python tanto desde una perspectiva de la programación estructurada como desde el punto de vista de la orientación a objetos. <u>Competencias</u> OPT-COMP4 - Representar gráficamente datos en 2D.

Contenidos
Qué es Python. intérpretes. Operadores, variables, expresiones y sentencias. Tipos de datos básicos. Control de flujo. E/S básica. Colecciones. Comprensiones de listas. Iteradores. Generadores. Funciones vs. orientación a objetos. Clases y objetos. Herencia. Excepciones. Ficheros. Librerías para la programación eficiente con arrays. Visualización de datos en 2D.
Temario
Denominación del tema 1: Contenidos del tema 1: Introducción a Python 1.1. Qué es Python. 1.2. intérpretes. Descripción de las actividades prácticas del tema 1: Introducción al IDE de Python (1 hora)
Denominación del tema 2: Contenidos del tema 2: Estructuras de datos I 2.1 Operadores, variables, expresiones y sentencias. 2.2 Tipos de datos básicos. 2.3 Control de flujo. 2.4 E/S básica Descripción de las actividades prácticas del tema 2: Desarrollo de programas básicos en python (1,5 horas)
Denominación del tema 3: Contenidos del tema 3 Estructuras de datos II 3.1 Colecciones. 3.2 Comprensiones de listas. 3.3 Iteradores. 3.4 Generadores Descripción de las actividades prácticas del tema 3: Utilización de colecciones (2 horas)
Denominación del tema 4: Contenidos del tema 4: Programación Orientada a Objetos (POO) 4.1 Funciones vs. orientación a objetos. 4.2 Clases y objetos. 4.3 Herencia. 4.4 Excepciones Descripción de las actividades prácticas del tema 3: Desarrollo de programas POO (2 horas)
Denominación del tema 5: Contenidos del tema 5: Ficheros 5.1 Lectura de ficheros 5.2 Escritura de ficheros Descripción de las actividades prácticas del tema 3: Manipulación de ficheros (2 horas)
Denominación del tema 6: Contenidos del tema 6: Librerías para la programación eficiente con arrays 6.1 Numpy 6.2 Jupyter 3.3 Pandas Descripción de las actividades prácticas del tema 3: Utilización de librerías (2 horas)
Denominación del tema 7: Contenidos del tema 5: Visualización de datos en 2D. 7.1 Librería Matplotlib Descripción de las actividades prácticas del tema 3: Desarrollo de gráficos 2D (2 horas)

Actividades formativas							
Horas de trabajo del alumno/a por tema		Horas gran grupo	Actividades prácticas			Actividad de seguimiento	No presencial
Tema	Total	GG	L	O	S	TP	EP
1	7	1		1			5
2	9,5	1,5		2			6
3	12	2		2			8
4	9	2		2			5
5	10	2		2			6
6	10	2		2			6
7	10,5	2		1			7,5
Evaluación	7	1					6
Defensa de trabajos	7	1					6
TOTAL	75	13,5		12			49,5

GG: Grupo Grande (85 estudiantes).

L: prácticas laboratorio o campo (15 estudiantes).

O: prácticas sala ordenador o laboratorio de idiomas (20 estudiantes).

S: clases problemas o seminarios o casos prácticos (40 estudiantes).

TP: Tutorías Programadas (seguimiento docente, tipo tutorías ECTS).

EP: Estudio personal, trabajos individuales o en grupo, y lectura de bibliografía.

Metodologías docentes

De entre las metodologías docentes incluidas en el plan de estudios del título para la asignatura, se utilizan las siguientes (marcadas con una "X" en la tabla):

Metodologías docentes	
Método expositivo / lección magistral Exposición oral de contenidos complementada con medios audiovisuales y la introducción de preguntas al alumnado. La lección magistral o conferencia es aquella impartida por un/a docente en ocasiones especiales, con un contenido original.	X
Método práctico laboratorio Realización de actividades de carácter práctico (demostraciones, ejercicios, experimentos e investigaciones).	X
Aprendizaje basado en proyectos o cooperativo La clase se organiza en pequeños grupos en los que el alumnado trabaja conjuntamente en la resolución de tareas asignadas por el profesorado. En el modo proyecto, estas tareas se enfocan a un trabajo de mayor complejidad, pudiendo extenderse a más de una materia o asignatura, de forma coordinada.	
Método de auto-información y aprendizaje autónomo Actividades para fomentar en el alumnado la realización de una búsqueda de recursos adecuados para poder evaluar su progreso.	X
Evaluación Realización de pruebas escritas u orales.	X

Sistemas de evaluación

Criterios de evaluación:

CR1: Desarrollar aplicaciones en Python.

Relacionado con las competencias: OPT-HAB4.

CR2: Aplicar técnicas para el procesamiento de la información en la resolución de problemas de ingeniería

Relacionada con: OPT-CON4, OPT-COMP4

Actividades de evaluación:

De entre las actividades de evaluación incluidas en el plan de estudios del título, en la presente asignatura se utilizan las siguientes:

Actividad de evaluación	Rango establecido	Convocatoria ordinaria	Convocatoria extraordinaria	Evaluación global
Pruebas periódicas y/o examen final	20%–70%	40%	40%	40%
Evaluación de trabajos y proyectos académicamente dirigidos	0%–60%	60%	60%	60%
Evaluación de prácticas	0%–60%			
Evaluación continua, asistencia y participación en actividades	0%–20%			

Descripción de las actividades de evaluación:

Evaluación continua

AEC1. Pruebas periódicas y/o examen final. El alumno realizará una defensa del proyecto desarrollado durante el curso. Esta actividad es recuperable en la convocatoria extraordinaria.

AEC2: Evaluación de trabajos y proyectos académicamente dirigidos. Desarrollo por parte del alumno de un proyecto en el que aplique los contenidos aprendidos durante la asignatura. El trabajo podrá desarrollarse de manera individual o en grupo y será entregado al finalizar el curso para su evaluación por parte del profesor. Esta actividad es recuperable en la convocatoria extraordinaria.

Evaluación global

AEG1. Pruebas periódicas y/o examen final. El alumno realizará una defensa de un proyecto relacionado con la materia de la asignatura. Esta actividad es recuperable en la convocatoria extraordinaria.

AEG2: Evaluación de trabajos y proyectos académicamente dirigidos. Desarrollo por parte del alumno de un proyecto en el que aplique los contenidos aprendidos durante la asignatura. El trabajo podrá desarrollarse de manera individual y será entregado al finalizar el curso para su evaluación por parte del profesor. Esta actividad es recuperable en la convocatoria extraordinaria.

Bibliografía	
Bibliografía básica	
Apuntes de la asignatura (publicados en el Campus Virtual)	
Bibliografía complementaria	
- Fundamentos de Programación: Algoritmos, Estructuras de Datos y Objetos. L. Joyanes. McGraw-Hill, 4ª ed., 2008 - Fundamentos de Programación: Algoritmos, Estructuras de Datos y Objetos. Libro de Problemas. L. Joyanes, L. Rodríguez y M. Fernández. McGraw-Hill, 2ª ed., 2003 - Introducción a la Informática. Prieto y otros. McGraw-Hill, 4ª ed., 2006 - Metodología de la programación. E. Alcalde y M. García, McGraw-Hill, 2ª ed., 1992 - Python 3. Cuevas Álvarez, Alberto. RA-MA S.A. Editorial y Publicaciones	
Otros recursos y materiales docentes complementarios	
- Asignatura en el Campus Virtual: http://campusvirtual.unex.es - Documentación oficial sobre Python: https://docs.python.org/es/3/tutorial/index.html - Documentación en español sobre Python: https://es.python.org/aprende-python/	