



IES ARCIPRESTE DE HITA. DEPARTAMENTO DE INFORMÁTICA

Programación didáctica del módulo: Programación

Ciclo formativo: Desarrollo de Aplicaciones Web

Curso 2025/2026

# **Programación didáctica del módulo**

## **Programación**

**Ciclo formativo:**  
**Desarrollo de Aplicaciones Web**

**Curso: 2025/2026**

**Profesor: César San Juan Pastor**



## Índice

1	Introducción .....	4
2	Legislación aplicable .....	6
3	Ubicación .....	8
4	Objetivos y Resultados del aprendizaje.....	11
4.1	Objetivos comunes del ciclo formativo .....	11
4.2	Objetivos específicos del módulo (Resultados de aprendizaje) .....	14
5	Contenidos.....	15
	Unidad de Trabajo 1: Introducción.....	15
	Unidad de Trabajo 2: Elementos de un programa informático. ....	16
	Unidad de Trabajo 3: Estructuras de control. ....	16
	Unidad de Trabajo 4: Conceptos de P.O.O.....	17
	Unidad de Trabajo 5: Programación bajo POO (I).....	18
	Unidad de Trabajo 6: Programación bajo POO (II).....	18
	Unidad de Trabajo 7: Proyecto.....	19
	Unidad de Trabajo 8: Gestión de datos.....	20
	Unidad de Trabajo 9: Interfaces gráficos. .....	20
6	Concordancia de las unidades de trabajo con los resultados del aprendizaje .....	21
7	Temporalización .....	22
8	Metodología .....	23
8.1	Alumnado pendiente .....	25
9	Evaluación.....	26
9.1	El proceso de evaluación .....	27
9.1.1	Evaluación inicial .....	27
9.1.2	Procedimientos para evaluar el proceso de aprendizaje del alumnado..	27
9.1.3	Evaluación sumativa .....	28
9.1.4	Procedimiento de Evaluación Pendientes.....	28
9.2	Criterios de evaluación .....	28



9.3	Resultados de aprendizaje y criterios de evaluación necesarios para la formación en empresa .....	33
9.4	Criterios de calificación.....	35
9.5	Recuperación .....	36
9.5.1	Planificación de las actividades de recuperación de los módulos no superados .....	37
9.6	Promoción al siguiente curso o repetición de módulo.....	38
9.7	Pérdida de la evaluación continua.....	39
9.7.1	Sistemas e instrumentos de evaluación para los alumnos que han perdido el derecho a la evaluación continua.....	40
9.7.2	Procedimiento de notificación de la pérdida de la evaluación continua.	40
9.7.3	Casos específicos .....	41
9.8	Autoevaluación del profesorado .....	41
10	Alumnado con necesidades específicas de apoyo educativo .....	43
11	Material didáctico.....	43
12	Actividades extraescolares .....	45
13	Bibliografía.....	45



## 1 Introducción

La Formación Profesional está orientada tanto al desarrollo y satisfacción personal del alumno como a la obtención de unos conocimientos de tipo técnico y/o humanístico que han de ser preparatorios para el mundo laboral o la Universidad.

La reforma educativa promulgada por la L.O.G.S.E. (Ley Orgánica de Ordenación General del Sistema Educativo) supuso un cambio radical en el sistema educativo existente hasta entonces. La Formación Profesional tradicional pasó a denominarse Ciclos Formativos, quedando estructurada en familias y niveles. Así, los Ciclos Formativos de Grado Medio permiten obtener el título de Técnico, mientras que los Ciclos Formativos de Grado Superior permiten obtener el título de Técnico Superior.

Posteriormente, la L.O.E. (Ley Orgánica de la Educación) estableció una nueva ordenación de los ciclos formativos, estableciendo el nuevo catálogo de la formación profesional, las unidades de competencia y los módulos formativos asociados del Catálogo Modular de Formación Profesional. Este nuevo marco formativo no hace sino acercar la Formación Profesional a las necesidades actuales de la sociedad del conocimiento, donde la movilidad laboral, las nuevas tecnologías, la cohesión e inserción laboral exigen un nuevo planteamiento del mercado laboral. Así pues, se pretende proporcionar a las personas la formación requerida por el sistema productivo y de acercar los títulos de formación profesional a la realidad del mercado laboral. Los Ciclos Formativos ofertados por la LOE están separados por familias, siendo una de ellas la Informática.

Con la entrada en vigor de la LOMCE en el curso 2014-2015 la FP Básica vino a sustituir a los PCPI, o Programas de Cualificación Profesional Inicial, desvinculando la Formación Profesional Básica de la obtención del Título de ESO. En este centro se lleva impartiendo la formación Básica en la rama de “Informática y Comunicaciones” desde el curso 2014-2015. Con la promulgación de la Ley Orgánica 3/2022, de 31 de marzo, de ordenación e integración de la Formación Profesional la Formación Básica pasa a denominarse Ciclo Formativo de Grado Básico



De acuerdo a la nueva redacción dada por la Ley Orgánica 3/2020, de 29 de diciembre, por la que se modifica la Ley Orgánica 2/2006, de 3 de mayo, de Educación, se establecen las titulaciones de los cursos de especialización, cuyo acceso requiere como mínimo de una titulación de grado superior.

A partir del curso 2024/2025, en Castilla-La Mancha se implantarán, con carácter obligatorio y de forma progresiva, las medidas establecidas en el Real Decreto 659/2023, de 18 de julio, que desarrolla la Ley Orgánica 3/2022, de 31 de marzo, de ordenación e integración de la formación profesional.

En este curso 2025/2026, el Departamento de Informática, impartirá los siguientes cursos:

**a) Ciclos formativos:**

**1. Grado Medio**

- Sistemas Microinformáticos y Redes (primer y segundo curso en turnos de mañana y vespertino).

**2. Grado Superior**

- Administración de Sistemas Informáticos en Red (primer y segundo curso).
- Desarrollo de Aplicaciones Web (primer y segundo curso en turnos de mañana y vespertino).
- Desarrollo de Aplicaciones Web (primer y segundo curso) en la modalidad E-learning).

**3. Grado Básico**

- “Informática y Comunicaciones” (Primer y segundo curso)

**b) Cursos de Especialización (en horario vespertino):**

1. Ciberseguridad en Entornos de las Tecnologías de la Información.
2. Inteligencia Artificial y Big Data.

**c) Las siguientes asignaturas en Bachillerato y la ESO**

1. Digitalización. (4º ESO)



## 2. Desarrollo Digital. (1º Bachillerato)

d) Además, el departamento también será encargado de llevar a cabo las tareas:

1. Responsable de Formación y TIC
2. Jefatura de estudios adjunta de FP
3. Responsable de aula ATECA
4. Responsable de aula APE

Dado el extraordinario auge de la informática, y su gran implantación en la gran mayoría de trabajos actualmente, no es de extrañar que estos ciclos formativos sean considerados por los alumnos como una buena alternativa profesional para su futuro.

Para la inserción de los alumnos en el mundo laboral de modo rápido y eficaz, el alumno debe aprender las técnicas y métodos más adecuados que garanticen la adquisición de los conocimientos y destrezas para desenvolverse en el sector informático.

Esta programación está referida al módulo de “Programación” del ciclo formativo “Desarrollo de Aplicaciones Web” en el centro I.E.S. Arcipreste de Hita de Azuqueca de Henares (Guadalajara).

## 2 Legislación aplicable

La legislación en la que se basa esta programación didáctica es la siguiente:

1. Ley 5/2002, de 19 de junio, donde se establece el sistema integral de la Formación Profesional.
2. Ley Orgánica 2/2006, de 3 de mayo, donde se regula la Formación Profesional en el sistema educativo, organizándola en ciclos formativos de grado medio y grado superior.
3. Real Decreto 1538/2006, de 15 de diciembre, por el que se establece la ordenación general de la Formación Profesional del sistema educativo, incluyendo los aspectos básicos de la evaluación y efectos de los títulos de Formación Profesional.



4. Orden de 29/07/2010, de la Consejería de Educación, Ciencia y Cultura, por la que se regula la evaluación, promoción y acreditación académica del alumnado de formación profesional inicial del sistema educativo de la Comunidad Autónoma de Castilla-La Mancha [2010/14361].
5. Orden de 12 de marzo de 2010, de la Consejería de Educación y Ciencia.
6. Ley 3/2012, de 10 de mayo, de autoridad del profesorado [2012/7512].
7. Ley Orgánica 3/2020, de 29 de diciembre, por la que se modifica la Ley Orgánica 2/2006, de 3 de mayo, de Educación.
8. Orden de 30/07/19, de la Conserjería. de Educación, Cultura y Deportes, por la que se modifican varias órdenes que regulan la evaluación de alumnado que cursa enseñanzas de FP y otras, para adecuar las fechas de evaluación anuales al calendario de evaluaciones.
9. Ley Orgánica 3/2022, de 31 de marzo, de ordenación e integración de la formación profesional.
10. RD 659/2023, de 18 de julio, por el que se desarrolla la ordenación del Sistema de Formación Profesional.
11. Real Decreto 500/2024, de 21 de mayo, por el que se modifican determinados reales decretos por los que se establecen títulos de Formación Profesional de grado superior y se fijan sus enseñanzas mínimas.
12. Real Decreto 405/2023, de 29 de mayo, por el que se actualizan los títulos de la formación profesional del sistema educativo de Técnico Superior en Desarrollo de Aplicaciones Multiplataforma y Técnico Superior en Desarrollo de Aplicaciones Web, de la familia profesional Informática y Comunicaciones, y se fijan sus enseñanzas mínimas.
13. 686/2010, de 20 de mayo, por el que se establece el título de Técnico Superior en Desarrollo de Aplicaciones Web y se fijan sus enseñanzas mínimas. (B.O.E. de 12 de junio del 2010)



14. Real Decreto 230/2011, de 28 de julio, por el que se establece el currículo de Ciclo Formativo de Grado Superior correspondiente al título de Técnico o Técnica Superior en Desarrollo de Aplicaciones Web, en la Comunidad Autónoma de Castilla-La Mancha [2011/11276].
15. Decreto 80/2024, de 5 de noviembre, por el que se modifican determinados decretos que establecen los currículos de los ciclos formativos de grado superior correspondientes a los títulos de Técnico o Técnica Superior de Formación Profesional en la comunidad autónoma de Castilla-La Mancha. [2024/8907].

### 3 Ubicación

Tradicionalmente, el alumnado que se matricula de los ciclos formativos es consciente de que las enseñanzas que va a recibir están muy ligadas a un entorno laboral, y que el objetivo principal de los ciclos formativos es **formar trabajadores en un campo específico**. Al tratarse de enseñanzas dedicadas a la informática, los alumnos tienen claro que el trabajo fundamental se desarrolla con ordenadores, aunque desgraciadamente asocian los contenidos con la ofimática, en lugar de la informática.

El curso de “Desarrollo de Aplicaciones Web” lleva impartiéndose en el centro desde el curso 2013-2014. En el primer curso de este ciclo los grupos que se forman suelen ser bastante heterogéneos, contando mayoritariamente con alumnos procedentes de bachillerato y de otros ciclos formativos de grado superior, aunque también y en menor medida suelen estar formados por alumnos procedentes de grado medio o de la universidad. Pero independientemente de la forma de acceso, en general todos ellos suelen mostrar interés por la informática y por las posibilidades laborales que ofrecen, aunque se prevé que algunos alumnos tengan más problemas para seguir los módulos que otros, según su forma de acceso. Los alumnos procedentes del bachillerato de ciencias suelen poseer una mentalidad más lógica, los de letras una mayor capacidad de síntesis, los procedentes del ciclo formativo de grado



mediante una mejor comprensión de términos técnicos, y los que provienen del mercado laboral un mayor interés, responsabilidad y curiosidad. Por todo ello el grupo de 1º de DAW es proclive a desarrollar grupos de alumnos con distintos niveles de comprensión para los que es necesario adecuar distintos tipos de metodologías.

Desde el curso 2017-2018 se está impartiendo el mismo ciclo de DAW, pero en la modalidad virtual, los contenidos son los mismos para ambos ciclos, sin embargo, la formación profesional a distancia permite realizar los mismos estudios de FP a través de Internet. Esta posibilidad de estudiar, está pensada para gente que no disponga de un horario para asistir a las correspondientes clases. De este modo, la idea es que se pueda seguir la formación desde su domicilio o desde su lugar de trabajo.

Las ventajas que tiene la matriculación en unos estudios online son principalmente:

- Autonomía y flexibilidad en el aprendizaje
- El acceso a los contenidos se realizará desde el lugar deseado, con el único requisito de tener conexión a Internet.
- Estructura modular que permite decidir a cada alumno de qué módulos quiere matricularse en función de sus necesidades personales y su disponibilidad.
- Disponibilidad de la Plataforma Educativa on-line donde se encuentran los materiales y desde donde se comunicará de forma permanente con su profesor.
- Horario libre accesible las 24 horas del día para una mejor adaptación del alumnado matriculado.

En general, esta oferta está dirigida sobre todo a las personas que, por diferentes razones no pueden cursar los estudios en los centros educativos en el horario presencial. Por lo general será el colectivo de personas adultas el destinatario de esta modalidad de enseñanza.

El Departamento de Informática dispone de las siguientes aulas:

**a) Aulas para ciclos y cursos de especialización:**



- a. Formado por 6 aulas situadas en el aulario en las que se imparten los seis cursos de Formación Profesional (dos aulas para el ciclo de SMR, dos para el ciclo de ASIR y dos para el ciclo de DAW) de aproximadamente 50 metros cuadrados cada una de ellas.
- b. El tamaño de las aulas no es el adecuado para realizar clases teóricas y prácticas cuando el grupo de alumnos es superior a 26 alumnos.
- c. Para el grupo de virtual, no será necesaria la utilización de ningún aula, pero si sería útil que el profesor pudiera tener una sala disponible con conexión a Internet donde pudiera trabajar.
- d. Los cursos de especialización se imparten en horario de tarde y ocupan las mismas aulas que los grados superiores.

**b) Aula APE**

- a. La asignatura de Bachillerato y de la ESO se imparte en el aula APE del centro o en aulas tradicionales con el apoyo de ordenadores portátiles.

**c) Aulas para CFG Básico**

- a. La formación básica se imparte en otra aula independiente de los ciclos.
- b. El aula de primero está en la planta baja del aulario
- c. El aula de segundo está en el edificio principal del instituto, un aula situada entre las aulas de APE y ATECA.

**d) Aula ATECA.**

- a. Aula de dotación europea para el desarrollo de proyectos de innovación.

En la mayoría de las aulas debido al gran número de alumnos matriculados en algunos cursos (principalmente en los cursos de primero), las aulas están formadas por hileras de ordenadores para intentar aprovechar el espacio de la forma más óptima posible. Aunque en algunos casos cuando hay pocos alumnos es posible distribuirlas en



forma de U para realizar las clases prácticas, permitiendo un control visual rápido de los ordenadores por parte del profesor, y en el centro de la clase disponer de mesas adicionales para realizar las clases teóricas.

Adicionalmente, el número de ordenadores es escaso, teniendo los alumnos que compartir ordenador (grupos de dos, o incluso tres) y no disponiendo de material de recambio en caso de rotura. En grado superior se permite a los alumnos traer su ordenador portátil en caso de que lo soliciten, rellenando siempre una solicitud que exime de responsabilidad al centro en caso de rotura, extravío o robo del portátil. El aula dispone de un proyector conectado al ordenador del profesor.

## 4 Objetivos y Resultados del aprendizaje

Son objetivos comunes los descritos en el Proyecto educativo del centro y en la PGA, en los que respecta a la convivencia, integración, trabajo en equipo y respeto mutuo entre los integrantes de la comunidad docente.

### 4.1 *Objetivos comunes del ciclo formativo*

Adicionalmente, los objetivos comunes para este ciclo formativo son los descritos en el Real Decreto 686/2010:

- a) Ajustar la configuración lógica analizando las necesidades y criterios establecidos para configurar y explotar sistemas informáticos.
- b) Identificar las necesidades de seguridad verificando el plan preestablecido para aplicar técnicas y procedimientos relacionados.
- c) Instalar módulos analizando su estructura y funcionalidad para gestionar servidores de aplicaciones.
- d) Ajustar parámetros analizando la configuración para gestionar servidores de aplicaciones.
- e) Interpretar el diseño lógico, verificando los parámetros establecidos para gestionar bases de datos.



- f) Seleccionar lenguajes, objetos y herramientas, interpretando las especificaciones para desarrollar aplicaciones web con acceso a bases de datos.
- g) Utilizar lenguajes, objetos y herramientas, interpretando las especificaciones para desarrollar aplicaciones web con acceso a bases de datos.
- h) Generar componentes de acceso a datos, cumpliendo las especificaciones, para integrar contenidos en la lógica de una aplicación web.
- i) Utilizar lenguajes de marcas y estándares web, asumiendo el manual de estilo, para desarrollar interfaces en aplicaciones web
- j) Emplear herramientas y lenguajes específicos, siguiendo las especificaciones, para desarrollar componentes multimedia.
- k) Evaluar la interactividad, accesibilidad y usabilidad de un interfaz, verificando los criterios preestablecidos, para Integrar componentes multimedia en el interfaz de una aplicación.
- l) Utilizar herramientas y lenguajes específicos, cumpliendo las especificaciones, para desarrollar e integrar componentes software en el entorno del servidor web.
- m) Emplear herramientas específicas, integrando la funcionalidad entre aplicaciones, para desarrollar servicios empleables en aplicaciones web.
- n) Evaluar servicios distribuidos ya desarrollados, verificando sus prestaciones y funcionalidad, para integrar servicios distribuidos en una aplicación web.
- o) Verificar los componentes de software desarrollados, analizando las especificaciones, para completar el plan de pruebas.
- p) Utilizar herramientas específicas, cumpliendo los estándares establecidos, para elaborar y mantener la documentación de los procesos.
- q) Establecer procedimientos, verificando su funcionalidad, para desplegar y distribuir aplicaciones.
- r) Programar y realizar actividades para gestionar el mantenimiento de los recursos informáticos.



- s) Analizar y utilizar los recursos y oportunidades de aprendizaje relacionadas con la evolución científica, tecnológica y organizativa del sector y las tecnologías de la información y la comunicación, para mantener el espíritu de actualización y adaptarse a nuevas situaciones laborales y personales.
- t) Desarrollar la creatividad y el espíritu de innovación para responder a los retos que se presentan en los procesos y organización de trabajo y de la vida personal.
- u) Tomar decisiones de forma fundamentada analizando las variables implicadas, integrando saberes de distinto ámbito y aceptando los riesgos y la posibilidad de equivocación en las mismas, para afrontar y resolver distintas situaciones, problemas o contingencias.
- v) Desarrollar técnicas de liderazgo, motivación, supervisión y comunicación en contextos de trabajo en grupo para facilitar la organización y coordinación de equipos de trabajo.
- w) Aplicar estrategias y técnicas de comunicación adaptándose a los contenidos que se van a transmitir, la finalidad y a las características de los receptores, para asegurar la eficacia en los procesos de comunicación.
- x) Evaluar situaciones de prevención de riesgos laborales y de protección ambiental, proponiendo y aplicando medidas de prevención personales y colectivas, de acuerdo a la normativa aplicable en los procesos del trabajo, para garantizar entornos seguros.
- y) Identificar y proponer las acciones profesionales necesarias para dar respuesta a la accesibilidad universal y al diseño para todos
- z) Identificar y aplicar parámetros de calidad en los trabajos y actividades realizados en el proceso de aprendizaje para valorar la cultura de la evaluación y de la calidad y ser capaces de supervisar y mejorar procedimientos de gestión de calidad.



- aa) Utilizar procedimientos relacionados con la cultura emprendedora, empresarial y de iniciativa profesional, para realizar la gestión básica de una pequeña empresa o emprender un trabajo.
- bb) Reconocer sus derechos y deberes como agente activo en la sociedad, teniendo en cuenta el marco legal que regula las condiciones sociales y laborales para participar como ciudadano democrático

## **4.2 Objetivos específicos del módulo (Resultados de aprendizaje)**

En el decreto 230/2011, de 28/07/2011 modificado por el decreto 80/2024 de 5 de noviembre, por el que se establece el currículo del ciclo formativo se establecen los siguientes resultados de aprendizaje para este módulo:

1. Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.
2. Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.
3. Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.
4. Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.
5. Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librerías de clases.
6. Escribe programas que manipulen información seleccionando y utilizando tipos avanzados de datos.
7. Desarrolla programas aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.
8. Utiliza bases de datos orientadas a objetos, analizando sus características y aplicando técnicas para mantener la persistencia de la información.



9. Gestiona información almacenada en bases de datos manteniendo la integridad y consistencia de los datos.

La formación del módulo contribuye a alcanzar los objetivos generales f), g), j) y q) del ciclo formativo y las competencias profesionales, personales y sociales e), f), y j) del título.

## 5 Contenidos

### ***Unidad de Trabajo 1: Introducción***

Contenidos	Objetivos
<ol style="list-style-type: none"><li>1. Origen de la Informática</li><li>2. Conceptos Informáticos Fundamentales</li><li>3. Clasificación del Software</li><li>4. Qué es la programación</li><li>5. Qué es un algoritmo</li><li>6. Documentación de los programas</li><li>7. Ciclo de Vida del Software</li></ol>	<ol style="list-style-type: none"><li>1. Comprender el origen y la evolución de la informática para valorar su impacto en la sociedad y en el desarrollo de las tecnologías actuales.</li><li>2. Identificar y explicar los conceptos informáticos fundamentales, como hardware, software, datos, información, y sistemas operativos, para establecer una base sólida de conocimiento.</li><li>3. Reconocer y clasificar los distintos tipos de software (sistema, aplicación, programación), comprendiendo sus características y funciones en un sistema informático.</li><li>4. Definir qué es la programación y su importancia en el desarrollo de soluciones tecnológicas, comprendiendo su relación con el software y los lenguajes de programación.</li><li>5. Entender el concepto de algoritmo y su papel esencial en la resolución de problemas mediante técnicas lógicas y estructuradas.</li><li>6. Valorar la importancia de la documentación técnica y funcional de los programas, reconociéndola como una herramienta clave en el mantenimiento y comprensión del software.</li><li>7. Conocer las fases del ciclo de vida del software, desde la concepción hasta el mantenimiento, para entender cómo se planifica y desarrolla un proyecto informático de forma estructurada.</li></ol>

### **Resultados y Criterios de Evaluación asociados a los Contenidos y Objetivos:**

R.A. 1: a

***Unidad de Trabajo 2: Elementos de un programa informático.***

Contenidos	Objetivos
<ol style="list-style-type: none"><li>1. Introducción</li><li>2. Historia de Python</li><li>3. Características de Python</li><li>4. Un vistazo rápido al interior de Python</li><li>5. Instalación de Python</li><li>6. Entornos IDE</li><li>7. Elementos de un programa</li></ol>	<ol style="list-style-type: none"><li>1. Conocer la historia y evolución del lenguaje Python, valorando su crecimiento, comunidad, y relevancia en el desarrollo de software actual.</li><li>2. Identificar las principales características de Python, como su sintaxis simple, legibilidad, tipado dinámico e interpretado, para entender por qué es uno de los lenguajes más utilizados.</li><li>3. Explorar la estructura interna básica de un programa en Python, reconociendo cómo se organizan los scripts, bloques de código, indentación, y flujo lógico.</li><li>4. Realizar correctamente la instalación de Python en distintos sistemas operativos, asegurando la configuración adecuada del entorno de ejecución.</li><li>5. Conocer y utilizar diferentes entornos de desarrollo (IDE) como IDLE, PyCharm, VS Code, o Jupyter, para facilitar la escritura, depuración y ejecución de programas.</li><li>6. Identificar los elementos fundamentales de un programa en Python, incluyendo variables, operadores, estructuras de control y funciones básicas.</li></ol>

**Resultados y Criterios de Evaluación asociados a los Contenidos y Objetivos:**

R.A. 1: b, c, d, e, f, g, h, i

***Unidad de Trabajo 3: Estructuras de control.***

Contenidos	Objetivos
<ol style="list-style-type: none"><li>1. Introducción</li><li>2. Condiciones</li><li>3. Estructuras de selección o alternativas</li><li>4. Estructuras de repetición</li><li>5. Estructuras de salto</li><li>6. Prueba y depuración de programas</li><li>7. Documentación del código del programa</li><li>8. Cuestiones de Estilo</li><li>9. Introducción a la programación modular</li></ol>	<ol style="list-style-type: none"><li>1. Introducir al estudiante en los principios básicos de la programación estructurada, promoviendo una comprensión sólida del flujo lógico de los programas.</li><li>2. Aplicar condiciones lógicas y operadores relacionales para resolver problemas mediante la toma de decisiones dentro de los programas.</li><li>3. Utilizar estructuras de selección o alternativas (como if, if-else, elif) para controlar el flujo del programa de manera lógica y eficiente.</li><li>4. Emplear estructuras de repetición (while, for) para realizar tareas repetitivas, comprendiendo sus diferencias y aplicaciones.</li><li>5. Reconocer y aplicar estructuras de salto (break,</li></ol>



	<p>continue, pass) en contextos adecuados, mejorando la eficiencia y control del programa.</p> <p>6. Realizar pruebas y depuración de programas, utilizando herramientas del entorno de desarrollo (IDE) para detectar y corregir errores lógicos y de sintaxis.</p> <p>7. Documentar correctamente el código del programa, utilizando comentarios y convenciones que faciliten su mantenimiento y comprensión por parte de otros desarrolladores.</p> <p>8. Adoptar buenas prácticas de estilo en la escritura de código, siguiendo guías como PEP 8, para favorecer la legibilidad, la claridad y la coherencia.</p> <p>9. Introducir los fundamentos de la programación modular, promoviendo el uso de funciones como forma de organizar el código en bloques reutilizables y mantenibles.</p>
--	---

**Resultados y Criterios de Evaluación asociados a los Contenidos y Objetivos:**

R.A. 3: a, b, c, d, e, f, g, h, i

**Unidad de Trabajo 4: Conceptos de P.O.O.**

Contenidos	Objetivos
1. Introducción 2. Orígenes 3. Características de la POO 4. Conceptos Fundamentales POO 5. Lenguajes	1. Comprender la importancia y finalidad de la Programación Orientada a Objetos, reconociendo su uso en el desarrollo moderno de software. 2. Conocer los orígenes de la POO y su evolución histórica, identificando los lenguajes clave que contribuyeron a su desarrollo (como Smalltalk, C++, Java, Python). 3. Identificar las características principales de la POO, tales como encapsulamiento, herencia, polimorfismo, abstracción y modularidad, y entender cómo estas mejoran la calidad del software. 4. Comprender y aplicar los conceptos fundamentales de la POO, incluyendo clases, objetos, atributos, métodos, instancia, visibilidad, y relaciones entre clases. 5. Reconocer y comparar los principales lenguajes de programación orientados a objetos, evaluando sus diferencias, ventajas y contextos de uso (Python, Java, C++, C#, entre otros).

**Resultados y Criterios de Evaluación asociados a los Contenidos y Objetivos:**

R.A. 2: a, b, c, d, e, f, g, h, i

***Unidad de Trabajo 5: Programación bajo POO (I).***

Contenidos	Objetivos
<ol style="list-style-type: none"><li>1. Clases</li><li>2. Declaración de una clase</li><li>3. Estructura y miembros de una clase</li><li>4. Objetos</li><li>5. Visibilidad</li><li>6. Paquetes</li><li>7. Destrucción de objetos y Liberación de memoria</li><li>8. Ejemplos de clases</li></ol>	<ol style="list-style-type: none"><li>1. Comprender el concepto de clase como estructura básica de la POO, que permite modelar entidades del mundo real mediante atributos y métodos.</li><li>2. Aprender a declarar correctamente una clase en un lenguaje de programación orientado a objetos, respetando su sintaxis y convenciones.</li><li>3. Identificar y definir adecuadamente los miembros de una clase, tales como atributos (datos) y métodos (comportamientos), y conocer su estructura interna.</li><li>4. Entender el concepto de objeto como instancia de una clase, y ser capaz de crear, utilizar y manipular objetos en un programa.</li><li>5. Aplicar los modificadores de visibilidad para controlar el acceso a los miembros de una clase, fomentando el encapsulamiento y la seguridad del código.</li><li>6. Organizar el código utilizando paquetes o módulos, para facilitar la modularidad, el mantenimiento y la reutilización del software.</li><li>7. Comprender el proceso de destrucción de objetos y la liberación de memoria, y su gestión por parte del recolector de basura en lenguajes como Python o Java.</li></ol>

**Resultados y Criterios de Evaluación asociados a los Contenidos y Objetivos:**

R.A. 4: a, b, c, d, e, f, g, h, i

***Unidad de Trabajo 6: Programación bajo POO (II).***

Contenidos	Objetivos
<ol style="list-style-type: none"><li>1. Tipos avanzados: Secuencia</li><li>2. Tipos avanzados: Listas</li><li>3. Otros tipos avanzados</li><li>4. Recursividad</li><li>5. Herencia</li><li>6. Excepciones</li><li>7. Estructuras de datos</li><li>8. Creación de casos de prueba</li></ol>	<ol style="list-style-type: none"><li>1. Comprender y utilizar tipos de datos avanzados, con énfasis en las secuencias (tuplas, cadenas, rangos) y sus operaciones fundamentales.</li><li>2. Profundizar en el uso de listas y sus métodos, reconociendo su versatilidad como estructura de datos dinámica en Python.</li><li>3. Conocer y aplicar otros tipos de datos avanzados, como diccionarios, conjuntos (sets) y estructuras personalizadas, adaptándolos a distintos problemas.</li><li>4. Comprender el concepto de recursividad y aplicarlo correctamente, resolviendo problemas que requieren llamadas repetidas a una misma función con condiciones de parada bien definidas.</li><li>5. Aplicar el principio de herencia en la programación orientada a objetos, permitiendo la reutilización y</li></ol>



	<p>extensión del código mediante jerarquías de clases.</p> <ul style="list-style-type: none"><li>6. Gestionar excepciones de forma adecuada en los programas, utilizando try, except, finally y raise para controlar errores y garantizar estabilidad.</li><li>7. Reconocer y aplicar estructuras de datos adecuadas para almacenar, organizar y manipular información eficientemente, como pilas, colas, listas enlazadas, árboles o tablas hash.</li><li>8. Diseñar y ejecutar casos de prueba para programas, comprendiendo su importancia para verificar el correcto funcionamiento, detectar errores y validar resultados.</li></ul>
--	---

**Resultados y Criterios de Evaluación asociados a los Contenidos y Objetivos:**

R.A. 6: a, b, c, d, e, f, g

R.A. 7: a, b, c, d, e, f, g, h, i, j

**Unidad de Trabajo 7: Proyecto.**

Contenidos	Objetivos
<ul style="list-style-type: none"><li>1. Objetivo</li><li>2. Descripción de la aplicación</li><li>3. Descripción de las etapas</li><li>4. Desarrollo del mismo y su documentación</li></ul>	<ul style="list-style-type: none"><li>1. Integrar los conocimientos teóricos y prácticos adquiridos en las distintas unidades de trabajo de, aplicando conceptos fundamentales, algoritmos, estructuras de datos y programación orientada a objetos para desarrollar una solución completa.</li><li>2. Aplicar técnicas de programación modular y reutilización de código, organizando el proyecto en clases, funciones y módulos bien estructurados y documentados.</li><li>3. Utilizar correctamente los mecanismos de control de flujo, como estructuras condicionales, bucles y gestión de excepciones, para garantizar la robustez y fiabilidad del software desarrollado.</li><li>4. Implementar pruebas, depuración y documentación del código, asegurando la calidad, mantenibilidad y claridad del proyecto.</li><li>5. Fomentar el trabajo colaborativo y la comunicación efectiva, utilizando herramientas de control de versiones y documentación compartida para coordinar el desarrollo en equipo.</li><li>6. Incorporar buenas prácticas de estilo y estándares de programación, asegurando la legibilidad y profesionalidad del código fuente.</li></ul>

**Resultados y Criterios de Evaluación asociados a los Contenidos y Objetivos:**

R.A. 1 al R.A.7: Todos los criterios de todos los resultados de aprendizaje

***Unidad de Trabajo 8: Gestión de datos***

Contenidos	Objetivos
1. Gestión de ficheros 2. Gestión de formatos estructurados: XML, JSON. 3. Bases de Datos: estructuradas y orientadas a objetos	1. Comprender el manejo de ficheros en un lenguaje de programación, incluyendo operaciones básicas como apertura, lectura, escritura y cierre de archivos en diferentes modos (texto y binario). 2. Identificar la utilidad de los formatos estructurados (XML y JSON) como mecanismos estándar para el intercambio de datos entre aplicaciones y servicios web. 3. Comprender los fundamentos de las bases de datos estructuradas, especialmente bases de datos relacionales, y su integración con aplicaciones mediante lenguajes como SQL. 4. Introducir el concepto de bases de datos orientadas a objetos, diferenciando su estructura, modelo de datos y ventajas respecto a las bases de datos tradicionales. 5. Realizar operaciones básicas de conexión, consulta e inserción desde un lenguaje de programación a una base de datos, utilizando bibliotecas y controladores adecuados (por ejemplo, SQLite, MySQL, PostgreSQL, MongoDB). 6. Fomentar la correcta estructuración y almacenamiento de la información, eligiendo el tipo de archivo o base de datos más adecuado según el contexto del problema a resolver.

**Resultados y Criterios de Evaluación asociados a los Contenidos y Objetivos:**

R.A. 5: a, b, c, d, e, f, g, h

R.A. 6: h, i, j

R.A. 8: a, b, c, d, e, f, g, h

R.A. 9: a, b, c, d, e, f, g, h

***Unidad de Trabajo 9: Interfaces gráficos.***

Contenidos	Objetivos
1. Introducción al diseño GUI 2. Diseño dirigido a eventos 3. Componentes 4. Librerías 5. Programación GUI Básica	1. Comprender los principios fundamentales del diseño de interfaces gráficas de usuario (GUI), reconociendo su importancia en la experiencia del usuario y la usabilidad de las aplicaciones. 2. Identificar los conceptos clave del diseño orientado a eventos, donde la interacción del usuario con la interfaz genera respuestas programadas por el sistema. 3. Reconocer y utilizar los componentes básicos de una interfaz gráfica, como botones, cuadros de texto, etiquetas, menús desplegables y cuadros de diálogo. 4. Conocer y aplicar librerías especializadas para el



	desarrollo de GUIs, como Tkinter (en Python) o bibliotecas equivalentes según el lenguaje de programación utilizado. 5. Desarrollar aplicaciones básicas con interfaz gráfica, integrando componentes, eventos y lógica de programación para crear soluciones funcionales y visualmente coherentes.
--	--

**Resultados y Criterios de Evaluación asociados a los Contenidos y Objetivos:**

R.A. 7: a, b, c, d, e, f, g, h, i, j

## 6 Concordancia de las unidades de trabajo con los resultados del aprendizaje

En el siguiente cuadro resumen, se especifica la concordancia entre los objetivos específicos (Resultados de aprendizaje) de este módulo y las unidades de trabajo (la X muestra correspondencia):

Unidad de Trabajo / Resultados del aprendizaje	RA 1	RA. 2	RA. 3	RA. 4	RA. 5	RA. 6	RA. 7	RA. 8	RA. 9
U.T. 1	X								
U.T. 2	X								
U.T. 3			X						
U.T. 4		X							
U.T. 5				X					
U.T. 6			X			X	X		
U.T. 7	X	X	X	X	X	X	X		
U.T. 8					X	X		X	X
U.T. 9							X		



## 7 Temporalización

A continuación, se plantea el calendario de ejecución de las unidades de trabajo ya descritas, la **duración asignada es orientativa** y puede modificarse y adaptarse durante el curso dependiendo del tipo de alumnado, recursos con los que se pueda contar en clase o posibles imprevistos:

Unidad de Trabajo	Duración prevista	Trimestre previsto
U.T. 1	18	1er
U.T. 2	12	1er
U.T. 3	34	1er
U.T. 4	20	1er – 2º
U.T. 5	22	2º
U.T. 6	30	2º-3º
U.T. 7	14	3º
U.T. 8	28	3º
U.T. 9	8	3º
<b>Duración total:</b>	<b>186</b>	



## 8 Metodología

Los aspectos metodológicos que se pretenden aplicar en este módulo descansan en la idea de que el alumno se considere parte activa de la actividad docente, con esto se pretende involucrarlo en el proceso de asimilación de nuevos conceptos y adquisición de capacidades no como un mero contenedor de éstas sino como un productor directo de estos conocimientos y habilidades en sí mismo.

De igual forma se pretende que el alumno respete al profesor y a sus compañeros, respectando igualmente el material de la clase. Dado el poco material disponible para impartir este módulo, esta última premisa se convierte en vital para poder realizar un aprendizaje correcto de la materia.

El módulo de programación es eminentemente procedural y todos los contenidos están interrelacionados en todas las unidades de trabajo del mismo. Teniendo esto en cuenta, se aplicarán los siguientes aspectos en cuanto a metodología didáctica:

- Proponer constantemente la obtención de algoritmos y programas que resuelvan problemas, aplicando las siguientes estrategias:
  - Explicación teórica de los conceptos básicos necesarios
  - Presentación de un problema a resolver
  - Explicación de posibles soluciones
  - Ampliación de conocimientos mediante trabajos de investigación, utilizando distintos recursos y materiales
  - Evaluación y reflexión sobre los resultados obtenidos
  - Posteriormente se realizarán una serie de ejercicios propuestos por el profesor que en las primeras unidades serán resueltos y corregidos por el profesor.
  - Una vez avanzado el curso se propondrán ejercicios que deben realizar los alumnos con el ordenador, bien en clase o en casa.



- En todo momento el profesor asesorará y orientará al alumno, y en caso de petición expresa de este resolverá el ejercicio.
- Si se dispone de tiempo suficiente, al final del curso, el alumno realizará un proyecto integrador. Se tratará de un proyecto en el lenguaje de programación visto donde se integren los conocimientos y procedimientos estudiados en las diferentes unidades temáticas, incluso si es posible podría englobar también los conceptos vistos en el modulo “Entornos de Programación”
- Se utilizará en la medida de lo posible la plataforma Moodle proporcionada por la Junta de comunidades, integrado en Educamos CLM, para proporcionar a los alumnos materiales de consulta, así como ejercicios y tareas.
  - Estructuración de la clase de la forma más óptima posible para aprovechar el espacio según el número de alumnos en el aula.
  - Utilización de la pantalla digital o el proyector para realizar las explicaciones prácticas de software.
  - Agrupación de algunas horas de clase en bloques de 2 sesiones lectivas, con el fin de poder planificar teoría y ejercicios prácticos en el mismo día.
  - Realización de actividades en grupo que permitan, de una forma próxima y fácil, el aporte de distintos puntos de vista sobre un tema concreto.
  - Agrupaciones de alumnos para realizar proyectos o ejercicios conjuntos.
  - Planteamiento de actividades creativas donde el alumno pueda aportar su criterio a los temas comentados.
  - Por otra parte, se plantea la necesidad de motivar e incentivar el interés del alumno por los temas referenciados en clase, esto se concreta en los puntos siguientes:
  - Acercamiento de los temas didácticos al mundo real, aportando publicaciones y documentación de productos lo más conocidos y asequibles posible.



- Desmitificando la teoría más abstracta y convirtiéndola en cosas tangibles. Es decir, analizando el punto de vista práctico de los conceptos expresados en clase.
- Planteando ejemplos de aplicación de los trabajos en clase en el mundo laboral real (o lo más cercano posible) de forma que se vaya formando la imagen, en cada alumno, de su perfil profesional.
- Se utilizará en la medida de lo posible la plataforma Moodle proporcionada por la Junta de comunidades, integrado en Educamos CLM, para proporcionar a los alumnos materiales de consulta, así como ejercicios y tareas.

## **8.1 Alumnado pendiente**

- Se utilizará de forma intensiva la plataforma Moodle, para la comunicación de todos los miembros del módulo, proporcionar materiales, así como ejercicios y tareas:
  - El profesor creará o usará un curso en la plataforma “Educamos” de la junta o añadirá al curso existente del curso a recuperar.
  - Si fuera necesario los alumnos deberán registrarse en la plataforma a principio de curso.
  - El profesor matriculará al alumnado o facilitará a los mismos la forma de matricularse del curso en la plataforma.
  - Se publicará todo el material necesario para desarrollar el plan de recuperación, de forma que los alumnados puedan organizar su tiempo disponible. Si fuera necesario, se podrá incluir material adicional.
  - El profesor facilitará en la plataforma un mecanismo de contacto y quedará a disposición de los alumnos para la resolución de dudas y dificultades.
  - El alumnado podrá vía email solicitar horas de tutoría. Las tutorías podrán realizarse físicamente si existiera un espacio disponible. Es importante destacar, que las tutorías también podrán realizarse telemáticamente si no



existiera espacio disponible o por motivos de incompatibilidad horaria, incluso fuera del horario lectivo para facilitar el acceso a los alumnos pendientes.

- La entrega de las tareas se realizará utilizando la plataforma Educamos o mediante correo electrónico.
- Las pruebas de evaluación podrán consistir:
  - ▶ Micropruebas online (pruebas consistentes en preguntas cortas con un tiempo muy limitado de respuesta aproximadamente 10 minutos para toda la prueba).
  - ▶ Pruebas prácticas a realizar presencialmente.
  - ▶ Trabajos a realizar de manera individual por parte de los alumnos, en este último caso se puede solicitar a los alumnos que realicen una defensa telemática de su trabajo.
- Si por alguna circunstancia la plataforma no estuviera disponible, se buscará una alternativa.
- Por otra parte, se plantea la necesidad de motivar e incentivar el interés del alumno por los temas referenciados en clase, esto se concreta en los puntos siguientes:
  - ▶ Acercamiento de los temas didácticos al mundo real, aportando publicaciones y documentación de productos lo más conocidos y asequibles posible.

## 9 Evaluación

La evaluación será continua, formativa y sumativa, considerándose además de las pruebas objetivas, el trabajo en clase, el progreso, el interés por el módulo, la atención, etc.



## **9.1 *El proceso de evaluación***

### **9.1.1 Evaluación inicial**

Al comienzo de cada Unidad de Trabajo se realizará un pequeño debate que permitirá saber cual es el nivel de conocimientos del alumno sobre cada tema, realizando introducciones sobre aquellos aspectos necesarios para el tema que el alumno no tiene o no ha adquirido completamente, o una pequeña introducción al tema. Se orientará a los alumnos acerca de los contenidos del tema para que los ubiquen dentro de los conocimientos informáticos adquiridos en el curso pasado, o bien en unidades de trabajo anteriores.

En el caso de que Unidades de Trabajo anteriores sirvan como base a una nueva Unidad de Trabajo, los alumnos en esta fase realizarán un repaso de esos conceptos.

### **9.1.2 Procedimientos para evaluar el proceso de aprendizaje del alumnado**

Utilizando la observación y el análisis de los trabajos desarrollados, se utilizarán los siguientes instrumentos de evaluación:

1. El trabajo en equipo
2. La investigación de los contenidos
3. La asistencia regular a clase
4. La puntualidad
5. La correcta utilización del material y equipos
6. Participación en clase
7. Realización y presentación de los trabajos obligatorios solicitados por el profesor.
8. La elaboración de los trabajos optativos
9. Pruebas escritas, con contenidos teóricos y prácticos



Se considera que estos instrumentos de evaluación son adecuados para los criterios de evaluación de este módulo.

### **9.1.3 Evaluación sumativa**

Al final de ciertos bloques de unidades de trabajo, fundamentales para proseguir el desarrollo del módulo, se realizarán pruebas específicas de evaluación escritas llevadas a cabo por el alumno de forma individual.

En ciertas unidades de trabajo se realizarán proyectos o ejercicios de síntesis que deberán ser entregados en una fecha límite que serán calificados en ese trimestre.

### **9.1.4 Procedimiento de Evaluación Pendientes**

Al principio de curso en la plataforma Moodle de la Junta se publicará todo el material necesario para desarrollar el plan de recuperación, de forma que los alumnos puedan organizar su tiempo disponible. La entrega de las tareas se realizará utilizando la plataforma Moodle o cualquier otro medio electrónico que el profesor determine.

Se realizará una prueba de evaluación por cada una de las convocatorias ordinarias. Los alumnos que, después de la primera convocatoria tenga el módulo no superado, accederán a la segunda convocatoria. No obstante, si el alumno no se presenta a la prueba de evaluación para la segunda convocatoria, se entenderá que el alumno renuncia a la misma, sin necesidad de haberlo solicitado previamente.

El acceso a la segunda convocatoria ordinaria se realizará independientemente del tipo de matrícula del alumno.

## **9.2 Criterios de evaluación**

Según el Decreto del currículo y referidos a los correspondientes Resultados de aprendizaje, los criterios de evaluación a tener en cuenta son:

1. Se han identificado los bloques que componen la estructura de un programa informático.
2. Se han creado proyectos de desarrollo de aplicaciones.



3. Se han utilizado entornos integrados de desarrollo.
4. Se han identificado los distintos tipos de variables y la utilidad específica de cada uno.
5. Se ha modificado el código de un programa para crear y utilizar variables.
6. Se han creado y utilizado constantes y literales.
7. Se han clasificado, reconocido y utilizado en expresiones los operadores del lenguaje.
8. Se ha comprobado el funcionamiento de las conversiones de tipo explícitas e implícitas.
9. Se han introducido comentarios en el código.
10. Se han identificado los fundamentos de la programación orientada a objetos.
11. Se han escrito programas simples.
12. Se han instanciado objetos a partir de clases predefinidas.
13. Se han utilizado métodos y propiedades de los objetos.
14. Se han escrito llamadas a métodos estáticos.
15. Se han utilizado parámetros en la llamada a métodos.
16. Se han incorporado y utilizado librerías de objetos.
17. Se han utilizado constructores.
18. Se ha utilizado el entorno integrado de desarrollo en la creación y compilación de programas simples
19. Se ha escrito y probado código que haga uso de estructuras de selección.
20. Se han utilizado estructuras de repetición.
21. Se han reconocido las posibilidades de las sentencias de salto.
22. Se ha escrito código utilizando control de excepciones.
23. Se han creado programas ejecutables utilizando diferentes estructuras de control.
24. Se han probado y depurado los programas.
25. Se ha comentado y documentado el código.



26. Se han creado excepciones.
27. Se han utilizado aserciones para la detección y corrección de errores durante la fase de desarrollo.
28. Se ha reconocido la sintaxis, estructura y componentes típicos de una clase.
29. Se han definido clases.
30. Se han definido propiedades y métodos.
31. Se han creado constructores.
32. Se han desarrollado programas que instancien y utilicen objetos de las clases creadas anteriormente.
33. Se han utilizado mecanismos para controlar la visibilidad de las clases y de sus miembros.
34. Se han definido y utilizado clases heredadas.
35. Se han creado y utilizado métodos estáticos.
36. Se han creado y utilizado conjuntos y librerías de clases
37. Se ha utilizado la consola para realizar operaciones de entrada y salida de información.
38. Se han aplicado formatos en la visualización de la información.
39. Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.
40. Se han utilizado ficheros para almacenar y recuperar información.
41. Se han creado programas que utilicen diversos métodos de acceso al contenido de los ficheros.
42. Se han utilizado las herramientas del entorno de desarrollo para crear interfaces gráficos de usuario simples.
43. Se han programado controladores de eventos.
44. Se han escrito programas que utilicen interfaces gráficos para la entrada y salida de información
45. Se han escrito programas que utilicen matrices (arrays).



46. Se han reconocido las librerías de clases relacionadas con tipos de datos avanzados.
47. Se han utilizado listas para almacenar y procesar información.
48. Se han utilizado iteradores para recorrer los elementos de las listas.
49. Se han reconocido las características y ventajas de cada una de las colecciones de datos disponibles.
50. Se han creado clases y métodos genéricos.
51. Se han utilizado expresiones regulares en la búsqueda de patrones en cadenas de texto.
52. Se han identificado las clases relacionadas con el tratamiento de documentos escritos en diferentes lenguajes de intercambio de datos
53. Se han realizado programas que realicen manipulaciones sobre documentos escritos en diferentes lenguajes de intercambio de datos.
54. Se han utilizado operaciones agregadas para el manejo de información almacenada en colecciones.
55. Se han identificado los conceptos de herencia, superclase y subclase.
56. Se han utilizado modificadores para bloquear y forzar la herencia de clases y métodos.
57. Se ha reconocido la incidencia de los constructores en la herencia.
58. Se han creado clases heredadas que sobrescriben la implementación de métodos de la superclase.
59. Se han diseñado y aplicado jerarquías de clases.
60. Se han probado y depurado las jerarquías de clases.
61. Se han realizado programas que implementen y utilicen jerarquías de clases.
62. Se ha comentado y documentado el código.
63. Se han identificado y evaluado los escenarios de uso de interfaces.
64. Se han identificado y evaluado los escenarios de utilización de la herencia y la composición



65. Se han identificado las características de las bases de datos orientadas a objetos.
66. Se ha analizado su aplicación en el desarrollo de aplicaciones mediante lenguajes orientados a objetos.
67. Se han instalado sistemas gestores de bases de datos orientados a objetos
68. Se han clasificado y analizado los distintos métodos soportados por los sistemas gestores para la gestión de la información almacenada
69. Se han creado bases de datos y las estructuras necesarias para el almacenamiento de objetos.
70. Se han programado aplicaciones que almacenen objetos en las bases de datos creadas.
71. Se han realizado programas para recuperar, actualizar y eliminar objetos de las bases de datos
72. Se han realizado programas para almacenar y gestionar tipos de datos estructurados, compuestos y relacionados.
73. Se han identificado las características y métodos de acceso a sistemas gestores de bases de datos.
74. Se han programado conexiones con bases de datos.
75. Se ha escrito un código para almacenar información en bases de datos.
76. Se han creado programas para recuperar y mostrar información almacenada en bases de datos.
77. Se han efectuado borrados y modificaciones sobre la información almacenada.
78. Se han creado aplicaciones que muestren la información almacenada en bases de datos.
79. Se han creado aplicaciones para gestionar la información presente en bases de datos.
80. Se han realizado las tareas de comprobación de seguridad adecuadas para asegurarse de la seguridad e integridad de los datos almacenados en la base de datos.



### **9.3 Resultados de aprendizaje y criterios de evaluación**

#### **necesarios para la formación en empresa**

Los resultados de aprendizaje y sus correspondientes criterios de evaluación expresados como **requeridos** que se describen en este apartado, deben ser necesariamente alcanzados en su totalidad para poder incorporarse a la fase de formación en empresa u organismo equiparado, de esta forma se garantiza que el desempeño del alumnado en la empresa no va suponer un riesgo para sí mismo, para la seguridad de los trabajadores o trabajadoras, sus instalaciones o para el tratamiento de la información confidencial de la empresa.

#### **RA1. Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.**

- a) Se han identificado los bloques que componen la estructura de un programa informático.
- b) Se han creado proyectos de desarrollo de aplicaciones
- c) Se han identificado los distintos tipos de variables y la utilidad específica de cada uno.
- d) Se ha modificado el código de un programa para crear y utilizar variables.
- e) Se han creado y utilizado constantes y literales.
- f) Se han clasificado, reconocido y utilizado en expresiones los operadores del lenguaje.

#### **RA2. Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.**

- a) Se han identificado los fundamentos de la programación orientada a objetos
- b) Se han escrito programas simples
- c) Se han instanciado objetos a partir de clases predefinidas
- d) Se han utilizado métodos y propiedades de los objetos.
- e) Se han utilizado parámetros en la llamada a métodos.
- f) Se han incorporado y utilizado librerías de objetos.
- g) Se han utilizado constructores
- h) Se han utilizado el entorno integrado de desarrollo en la creación y compilación de programas simples.



**RA3. Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.**

- a) Se ha escrito y probado código que haga uso de estructuras de selección.
- b) Se han utilizado estructuras de repetición
- c) Se han reconocido las posibilidades de las sentencias de salto
- d) Se ha escrito código utilizando control de excepciones.
- e) Se han creado programas ejecutables utilizando diferentes estructuras de control.
- f) Se han probado y depurado los programas.

**RA4. Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.**

- a) Se ha reconocido la sintaxis, estructura y componentes típicos de una clase
- b) Se han definido clases.
- c) Se han definido propiedades y métodos.
- d) Se han creado constructores.
- e) Se han desarrollado programas que instancien y utilicen objetos de las clases creadas anteriormente.
- f) Se han utilizado mecanismos para controlar la visibilidad de las clases y de sus miembros.
- g) Se han definido y utilizado clases heredadas.

**RA5. Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librerías de clases.**

- a) Se ha utilizado la consola para realizar operaciones de entrada y salida de información.
- d) Se han utilizado ficheros para almacenar y recuperar información.

**RA6. Escribe programas que manipulen información seleccionando y utilizando tipos avanzados de datos.**

- a) Se han escrito programas que utilicen matrices (arrays)
- b) Se han reconocido las librerías de clases relacionadas con tipos de datos avanzados.
- c) Se han utilizado listas para almacenar y procesar información.
- d) Se han utilizado iteradores para recorrer los elementos de las listas.

**RA7. Desarrolla programas, aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.**

- a) Se han identificado los conceptos de herencia, superclase, subclase
- b) Se han utilizado modificadores para bloquear y forzar la herencia de clases y métodos



- c) Se ha reconocido la incidencia de los constructores en la herencia.
- d) Se han creado clases heredadas que sobrescriban la implementación de métodos de la superclase.
- e) Se han diseñado y aplicado jerarquías de clases.

**RA8. Utiliza bases de datos orientadas a objetos, analizando sus características y aplicando técnicas para mantener la persistencia de la información.**

Ninguno

**RA9. Gestiona información almacenada en bases de datos relacionales manteniendo la integridad y consistencia de los datos.**

Ninguno

#### **9.4 Criterios de calificación**

Es requisito indispensable para la superación del módulo que el alumno supere cada uno de los resultados de aprendizaje del módulo de acuerdo a los criterios de calificación establecidos. Una vez superados todos los resultados de aprendizaje, la calificación final del módulo se obtendrá sumando la calificación obtenida en cada uno de los RRAA, de acuerdo con los porcentajes de ponderación. Del resultado se tomará la parte entera, redondeando por exceso la cifra si la parte decimal resultase ser igual o superior a 5.

La calificación final del módulo, por lo tanto, se establecerá según los siguientes puntos:

- El rango de calificación será de 1 a 10 valor entero (Delphos)
- El peso de las calificaciones de los RRAA se realizará mediante una media ponderada. (Véase Tabla siguiente)
- El valor mínimo en los RRAA para considerar que las capacidades profesionales han sido alcanzadas será de 5. En el caso, que algún RRAA presente una puntuación inferior a 5, entonces la calificación final del módulo no podrá ser superior a 4.

RESULTADOS DE APRENDIZAJE	% Asignado Evaluación
---------------------------	--------------------------



RA1. Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.	10
RA2. Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.	15
RA3. Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.	15
RA4. Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.	10
RA5. Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librerías de clases.	10
RA6. Escribe programas que manipulen información seleccionando y utilizando tipos avanzados de datos	10
RA7. Desarrolla programas, aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación	15
RA8. Utiliza bases de datos orientadas a objetos, analizando sus características y aplicando técnicas para mantener la persistencia de la información	5
RA9. Gestiona información almacenada en bases de datos relacionales manteniendo la integridad y consistencia de los datos	10
	100%

Cada resultado de aprendizaje está dividido en criterios de evaluación que serán evaluados mediante varios instrumentos de evaluación, pudiendo un instrumento de evaluación evaluar diferentes criterios de evaluación.

El rango de calificación de un CE será de 0 a 10 y el valor mínimo para considerar que el CE está logrado será de 5. Si un CE se evalúa más de una vez, la calificación se obtendrá con un porcentaje en función del momento de desarrollo del contenido asociado al criterio.

## 9.5 Recuperación

Se trata de una evaluación continua como se desprende de la metodología empleada y, por tanto, no existirán pruebas de recuperación específicas (a no ser que el profesor lo considere necesario), realizando planes de refuerzo al alumnado en el



momento que se detecte que un alumno no pueda conseguir los objetivos establecidos.

#### **Acceso a la segunda convocatoria ordinaria**

Los alumnos que, después de la primera convocatoria tengan módulos no superados, accederán a la segunda convocatoria de cada curso académico. No obstante, si el alumno no se presenta a las pruebas de evaluación preparadas por los profesores para la segunda convocatoria, se entenderá que el alumno renuncia a la misma, sin necesidad de haberlo solicitado previamente.

Si un alumno no supera el curso (es decir en la calificación final del curso no obtiene como valor resultante un 5) en la primera evaluación ordinaria, deberá realizar las tareas propuestas para este periodo y presentarse de nuevo a un examen para la evaluación de la segunda evaluación ordinaria.

Se entenderá que supera el módulo si alcanza una nota de cinco tras aplicar la ponderación establecida para la evaluación de los diferentes resultados de aprendizaje.

#### **9.5.1 Planificación de las actividades de recuperación de los módulos no superados**

Dado que se utiliza la plataforma Moodle a lo largo del módulo/asignatura, los alumnos tienen a su disposición el conjunto de ejercicios que les pueden servir de refuerzo para superar el examen de la segunda convocatoria ordinaria.

En el caso de aquellos alumnos que hayan promocionado a 2º y tengan este módulo no superado, se dará acceso a un curso en la plataforma Moodle de la junta, donde el profesor proporciona materiales, así como ejercicios y tareas que deberán realizar los alumnos. La resolución de dudas se realizará utilizando el correo electrónico. Deberán realizar un proyecto propuesto por el profesor cuya nota será del 30% de la nota final y deberán de presentarse a un examen que contendrá el contenido de todo el curso cuya nota valdrá un 70% de la nota final.



La asistencia a clase por parte del alumnado pendiente es muy difícil, por lo que el proceso de evaluación continua se desvirtúa y no es posible. Ante esta situación, el procedimiento de evaluación debe adaptarse a las características de las enseñanzas impartidas, por lo que se evaluará de la siguiente forma.

En el caso de aquellos alumnos que hayan promocionado a 2º que tengan este módulo no superado, deberán realizar las actividades propuestas por el profesor cuya nota será del 30% de la nota final y deberán de presentarse a un examen que contendrá el contenido de todo el curso cuya nota valdrá un 70% de la nota final.

El alumnado recibirá a lo largo de la preparación tareas que podrá realizar y presentar las dudas ante el profesor en horas habilitadas para tal fin.

El examen se realizará unas semanas antes del final del periodo lectivo que tengan los alumnos de segundo curso (Que suelen coincidir entre finales de febrero y marzo).

$$\text{Nota-Recuperación} = \text{Nota-Examen} \times 0,7\% + \text{media_Actividades_Evaluables} \times 0,3\%$$

## **9.6 Promoción al siguiente curso o repetición de módulo**

En la primera convocatoria ordinaria, los alumnos que obtengan una evaluación positiva en todos los módulos, accederán de forma automática al segundo curso del ciclo formativo. El resto de alumnos accederán a la segunda convocatoria anual.

Teniendo los resultados obtenidos por los alumnos en la segunda convocatoria anual se realizará la promoción al siguiente curso, o la repetición del módulo de la siguiente forma:

1. Los alumnos con todos los módulos superados promocionarán al segundo curso.
2. Los alumnos con uno o varios módulos no superados cuya carga horaria sea superior a 300 horas anuales, repetirán todas las actividades programadas para esos módulos y, por tanto, deberán matricularse como alumnos repetidores.



3. Para los alumnos que no han superado uno o varios módulos cuya carga horaria en conjunto sea inferior a 300 horas anuales se decidirá la promoción a segundo según las posibilidades de recuperación que el equipo docente estime.

## **9.7 Pérdida de la evaluación continua**

En el caso de que un alumno no asista a clase, puede perder el derecho a ser evaluado de forma continua. En concreto aquellos alumnos que tengan un 25% de faltas de asistencia injustificadas POR MÓDULO perderán el derecho a la evaluación continua de ese módulo, por lo que deberán presentarse a una prueba objetiva al finalizar el módulo.

En este módulo, el porcentaje de faltas injustificadas que puede tener un alumno antes de perder el derecho a la evaluación continua es: 42 horas.

La pérdida de la evaluación continua se realiza únicamente para el módulo en el que se hayan detectado las faltas de asistencia injustificadas, y no para todo el ciclo formativo.

La justificación válida para los alumnos se realizará mediante un justificante médico expedido por autoridades médicas o por causas de fuerza mayor que el alumno pueda alegar y sean aceptadas por el profesor.

Adicionalmente, para fomentar el cuidado y corresponsabilidad del material de clase y prepararlos para el trabajo en empresa de forma responsable, los alumnos que causen daño intencionado o por negligencia no cuiden el mismo deberán reparar el daño causado al amparo de la Ley de Autoridad del Profesorado. En el caso de que no reparen el daño causado perderán el derecho a la evaluación continua en todos los módulos en los que estén matriculados. Los alumnos volverán a ser evaluados de forma continuada cuando reparen el daño causado.



### **9.7.1 Sistemas e instrumentos de evaluación para los alumnos que han perdido el derecho a la evaluación continua**

En el caso de que un alumno pierda el derecho a evaluación continua, deberá presentarse al examen final del curso que se realizará la última semana del curso. En base a ese examen final se calificará el módulo en la primera sesión de evaluación ordinaria. Aun así, el alumno deberá entregar los trabajos prácticos que considere el profesor PREVIA realización del examen. En el caso de no entregar los trabajos prácticos, el alumno no podrá realizar el examen final.

La calificación final obtenida se calculará según lo descrito en el apartado 9.3 de esta programación didáctica.

### **9.7.2 Procedimiento de notificación de la pérdida de la evaluación continua**

El procedimiento de notificación de la pérdida de la evaluación continua es el siguiente:

1. Una vez el alumno haya perdido el derecho a la evaluación continua, al alcanzar el 25% de las faltas injustificadas, el profesor notificará del hecho al tutor del grupo.
2. El tutor del grupo contactará con el resto de los profesores, por si hubiera algún módulo con alguna circunstancia similar.
3. En el menor tiempo posible se notificará por carta al alumno o a sus tutores legales (en el caso de menores de edad), enviada por el tutor desde la secretaría del centro (con registro de entrada) con el visto bueno de la Dirección del centro. La comunicación se realizará según el modelo establecido en el Anexo I de la orden 29/07/2010 de la Consejería de Educación, Ciencia y Cultura de CLM, por la que se regula la evaluación del alumnado de Formación Profesional.



### 9.7.3 Casos específicos

Aquellos alumnos que tengan este módulo suspendido y hayan pasado de curso deberán igualmente presentar los trabajos prácticos que el profesor le requiera. El alumno deberá ponerse en contacto con el profesor del módulo que ha suspendido para que este le indique los criterios de calificación.

Aquellos alumnos que presenten una justificación a las faltas de asistencia (únicamente debida a causas justificadas), **no perderán el derecho a la evaluación continua**, pero deberán igualmente presentarse a los exámenes parciales y entregar los trabajos prácticos. En el caso de que no lo hagan deberán presentarse al examen final de curso.

Independientemente de lo anterior, es responsabilidad del alumno realizar un seguimiento de las explicaciones realizadas en clase, para poder entregar los proyectos y realizar los exámenes con el resto de la clase.

### 9.8 Autoevaluación del profesorado

La autoevaluación del profesorado está englobada en el Proyecto Educativo del Centro (según su plan de autoevaluación del centro), y se percibe como una forma de mejora y calidad de la enseñanza.

La autoevaluación del profesorado es una práctica constante y continua en el Departamento de Informática, que demuestra a lo largo de cada curso escolar una innovación de metodologías y capacidad de inventiva para poder impartir enseñanzas a pesar de los escasos recursos materiales de los que dispone. Esta autoevaluación del trabajo docente suele ser un proceso interno, de reflexión intrínseca y de necesidad esencial en el trabajo del profesorado. Conviene sin embargo realizar una reflexión escrita de forma periódica, por lo que, una vez terminadas las evaluaciones del primer y segundo trimestre, el profesorado realiza una autoevaluación de su trabajo y metodología empleada. En esa autoevaluación se recogerán los siguientes aspectos:

**Medidas tomadas durante el trimestre que se deben autoevaluar:**



1. Medidas metodológicas (clase magistral, libro de texto, nuevas tecnologías, ...)
2. Organizativas del aula
3. Agrupamientos del alumnado
4. Evaluación
5. Actividades de recuperación
6. Acción tutorial
7. Material
8. Problemas encontrados
9. Correcciones
10. Departamentales

**Medidas que se deben tomar durante el siguiente trimestre:**

1. Medidas metodológicas (clase magistral, libro de texto, nuevas tecnologías, ...)
2. Organizativas del aula
3. Agrupamientos del alumnado
4. Evaluación
5. Actividades de recuperación
6. Acción tutorial
7. Material
8. Problemas encontrados
9. Correcciones

**Resultados académicos:**

1. Porcentaje de alumnos por tramos de calificación.
2. Porcentaje de abandonos o renuncias de convocatorias
3. Número de faltas de asistencia



## 10 Alumnado con necesidades específicas de apoyo educativo

Se realizarán las adaptaciones metodológicas necesarias en los medios y procedimientos de evaluación para el alumnado con necesidades específicas de apoyo educativo, con el fin de garantizar su accesibilidad a las pruebas y que sea evaluado con los medios apropiados a sus posibilidades y características.

En todo caso, en el proceso de evaluación se comprobará que el alumnado ha conseguido los resultados de aprendizaje establecidos para cada uno de los módulos que forman parte del ciclo formativo.

En ningún caso se realizarán adaptaciones curriculares significativas.

## 11 Material didáctico

Los recursos necesarios para impartir este módulo son los siguientes:

### Para las explicaciones de contenidos teóricos:

- Aula con medios audiovisuales:
  - Pizarra.
  - Retroproyector y pantalla o pizarra digital.
  - Ordenador con Windows, Microsoft Office, Acrobat Reader, Winrar, VMWare, PacketTracert, Linux, etc
- Conexión a Internet en el aula.

### Para la resolución de los ejercicios prácticos:

- Pizarra.
- Puestos conectados en red
- Impresoras.

### Para la confección de los trabajos de los alumnos:

- Conexión a Internet en el aula.
- Correo electrónico
- Acceso a Moodle



### Cuidado del material

En la situación actual en la que nos encontramos, con unos presupuestos ajustados y un material escaso, se hace IMPRESCINDIBLE en el Departamento de Informática exigir un cuidado del material a los alumnos. Afortunadamente, esta necesidad viene incluso amparada por ley de CLM, por lo que, en el caso de rotura del material por parte de un alumno, se exigirá el cumplimiento de la Ley de Autoridad del Profesorado, donde se especifica, en su Artículo 7:

*“Artículo 7. Responsabilidad y reparación de daños.*

*Los alumnos/as o personas con él relacionadas que individual o colectivamente causen, de forma intencionada o por negligencia, daños a las instalaciones, equipamientos informáticos, incluido el software, o cualquier material del centro, así como a los bienes de los miembros de la comunidad educativa, quedarán obligados a reparar el daño causado o hacerse cargo del coste económico de su reparación o restablecimiento, cuando no medie culpa in vigilando de los/as profesores/as. Asimismo, deberán restituir los bienes sustraídos, o reparar económicamente el valor de estos.*

*2. En todo caso, quienes ejerzan la patria potestad o la tutela de los menores de edad serán responsables civiles en los términos previstos por la legislación vigente.”*

En el caso de que un alumno cause daño a las instalaciones o material, se amonestará de la acción por escrito informando a Jefatura de Estudios para que tome las medidas disciplinarias oportunas, y gestione la aplicación del artículo mencionado anteriormente.

Como se ha comentado en el apartado 9.6, los alumnos que causaran daño a las instalaciones o material y no reparen el daño causado perderán el derecho a la evaluación continua.



## 12 Actividades extraescolares

Las actividades extraescolares son muy importantes para la motivación del alumnado, por lo tanto, siempre que sea posible se organizarán salidas que sean provechosas para los alumnos (Como ferias de informática, empresas de informática, etc.). Incluso si es posible se contactará con antiguos alumnos para que den una charla a los alumnos actuales sobre su visión del mundo laboral después de haber obtenido el título.

## 13 Bibliografía

No se usará libro de texto, aportando el profesor parte de los apuntes y recomendando el uso de algunos libros de los citados a continuación, y determinadas páginas de Internet.

- Python 3 Los fundamentos del lenguaje (3a edición), Sébastien CHAZALLET, Ediciones ENI, ISBN 978-2-409-02478-8
- Curso de Programación Python (MANUALES IMPRESCINDIBLES), Arturo Montejo Ráez y Salud María Jiménez Zafra, Grupo Anaya Publicaciones Generales, ISBN 978-8441541160