

**IES ALONSO CANO**  
**DEPARTAMENTO DE INFORMÁTICA**

**Técnico Superior en Desarrollo de Aplicaciones  
Web**

**PROGRAMACIÓN**

**Programación didáctica – 1º DAW**

**Curso 2023-2024**

**Profesor: Diego Alarcón García**

# ÍNDICE

1 Marco legislativo.....	3
2 Introducción: conceptualización y características de la materia, relación con el Plan de centro.....	3
3 Los objetivos, los contenidos y su distribución temporal y los criterios de evaluación, posibilitando la adaptación de la secuenciación de contenidos a las características del centro y su entorno.....	4
3.1 Objetivos.....	4
3.1.1. Objetivos generales del ciclo.....	4
3.1.2. Objetivos generales y competencias del título que se alcanzan con el módulo Programación.....	6
3.2 Contenidos.....	6
3.3 Distribución temporal.....	18
3.4 Criterios de evaluación.....	18
4 Contribución de la materia a la adquisición de las competencias básicas.....	23
4.1 Competencia general.....	23
4.2 Competencias profesionales, personales y sociales.....	23
4.3 Cualificaciones y unidades de competencia del Catálogo Nacional de Cualificaciones Profesionales incluidas en el título.....	24
4.3.1. Cualificaciones profesionales completas:.....	24
4.3.2. Cualificaciones profesionales incompletas:.....	24
4.3.3. Correspondencia del módulo profesional con las unidades de competencia.....	25
5 Principios pedagógicos de la materia.....	25
6 Metodología.....	26
7 Evaluación Inicial.....	27
8 Los procedimientos de evaluación del alumnado y los criterios de calificación, en consonancia con las orientaciones metodológicas establecidas.....	28
8.1 Instrumentos de evaluación.....	28
8.2 criterios de calificación.....	28
9 Medidas de atención a la diversidad.....	33
9.1 Medidas generales.....	33
9.2 Clases de refuerzo/recuperación.....	33
10 Materiales y recursos didácticos.....	33
11 Actividades complementarias y extraescolares relacionadas con el currículo que se proponen realizar por los departamentos de coordinación didáctica.....	34

12 Concreción de planes, programas y proyectos del centro vinculados con desarrollo del currículo de la materia.....	34
12.1 Proyecto de FP DUAL.....	34
12.2 Fomento de la lectura.....	34
13 Evaluación de la Programación didáctica.....	35

# **1 Marco legislativo**

Además del marco legislativo general que se desarrolla en la programación del departamento se tendrán en cuenta las siguientes leyes para el desarrollo de la normativa del ciclo formativo:

- Las enseñanzas mínimas de este módulo fueron establecidas en el **Real Decreto 686/2010, de 20 de mayo**, publicado en el *BOE* n.º 143 de 12 de junio de 2010.
- Orden de 16 de junio de 2011, por la que se desarrolla el currículo correspondiente al título de Técnico Superior en Desarrollo de Aplicaciones Web.

## **2 Introducción: conceptualización y características de la materia, relación con el Plan de centro**

El módulo “Programación” se encuadra en el primer curso del Ciclo Formativo de Grado Superior correspondiente al título de Técnico Superior en Desarrollo de Aplicaciones Web, de la familia profesional de Informática y Comunicaciones, correspondiéndose con el nivel CINE-5b (Clasificación Internacional Normalizada de la Educación). Código Referente europeo 0485.

Este módulo tiene una duración de 256 horas, dando lugar a una distribución horaria semanal de 8 horas. Sin embargo, dado que este ciclo formativo dispone de “aula bilingüe” según el plan de centro, una de esas horas se impartirá en idioma inglés.

En este curso académico arrancamos la formación dual en 1º DAW, y se seguirán las pautas establecidas en el proyecto dual presentado y aprobado el curso pasado tal y como se contempla en el Plan de centro.

Así mismo, el ciclo formativo cuenta con un proyecto de formación Dual para todo el alumnado, en el cual se indica, según se ha redactado en dicho proyecto, que la formación en alternancia en la empresa empezará a desarrollarse a partir de 18 de marzo. Por tanto, parte de los resultados de aprendizaje se adquirirán en la empresa asignada a cada alumno.

También hay que tener en cuenta que si, por alguno de los motivos redactados en el proyecto de DUAL, alguno de los alumnos no pudiera acceder a su formación en alternancia, se continuaría su formación en el centro desarrollando las distintas actividades que se considere necesarias para la adquisición de dichos resultados que el resto de compañeros adquirirán en la empresa.

### **3 Los objetivos, los contenidos y su distribución temporal y los criterios de evaluación, posibilitando la adaptación de la secuenciación de contenidos a las características del centro y su entorno**

#### **3.1 Objetivos**

##### **3.1.1. Objetivos generales del ciclo**

Los objetivos generales de este ciclo formativo son los siguientes:

- 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.
- ab) 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.

### 3.1.2. Objetivos generales y competencias del título que se alcanzan con el módulo Programación

La formación del módulo contribuye a alcanzar los objetivos generales e), j), q) del ciclo formativo y las competencias a), e), f), i), j), v), y) y z) del título, las cuales se desarrollarán en el apartado 5 de esta programación.

## 3.2 Contenidos

Unidad Didáctica 1:	Conceptos básicos
CONTENIDOS	
1.1. Algoritmo	
1.2. Lenguajes de programación	
1.2.1. Lenguajes compilados e interpretados	
1.2.2. Lenguajes multiplataforma	
1.3. NetBeans IDE	
1.4. El programa principal	
1.5. Palabras reservadas	
1.6. Concepto de variable	
1.6.1. Identificadores	
1.7. Tipos primitivos	
1.7.1. Variables de tipo primitivo	
1.7.2. Rangos	
1.8. Variables de objeto	
1.9. Constantes	
1.10. Comentarios	
1.11. API de Java	
1.11.1. Paquetes	
1.11.2. Salida por consola	
1.11.3. Entrada de datos	
1.12. Operaciones básicas	
1.12.1. Operador de asignación	
1.12.2. Operadores aritméticos	
1.12.3. Operadores relacionales	
1.12.4. Operadores lógicos	
1.12.5. Operadores opera y asigna	
1.12.6. Operador ternario	
1.12.7. Precedencia	
1.13. Conversión de tipos	
OBJETIVOS	
<ul style="list-style-type: none"><li>• <b>Establecer</b> todos los conceptos básicos que son necesarios para el</li></ul>	

conocimiento de cualquier lenguaje de programación.

- **Concretar** la importancia de los mecanismos básicos que usan los compiladores, como la declaración y asignación de variables, el establecimiento de tipos, etcétera.
- **Clasificar** los distintos operadores, atendiendo a su uso y al resultado de la evaluación de sus expresiones.
- **Profundizar** en el concepto de tipo primitivo, sus tamaños, rangos y conversiones, así como sus valores literales.
- **Asociar** los tipos primitivos con sus operadores.
- **Asimilar** las distintas formas de interacción con el usuario mediante la salida por consola y la entrada de datos desde el teclado.
- **Exponer** el funcionamiento de herramientas (métodos) que proporciona la API para ciertas clases, la diferencia entre el uso estático y no estático de los métodos de la API, así como la clasificación de clases en los distintos paquetes.

Unidad Didáctica 2:	Condicionales
CONTENIDOS	
2.1. Expresiones lógicas relacionales 2.1.2. Operadores lógicos 2.2. Condicional simple: if 2.3. Condicional doble: if-else 2.3.1. Operador ternario 2.3.2. Anidación de condicionales 2.4. Condicional múltiple: switch	
OBJETIVOS	
<ul style="list-style-type: none"><li>• <b>Escribir</b> y <b>probar</b> código que haga uso de estructuras de selección.</li><li>• <b>Clasificar, reconocer</b> y <b>utilizar</b> los operadores del lenguaje en expresiones.</li><li>• <b>Reconocer</b> la estructura de un programa informático, identificando y relacionando los elementos propios de Java.</li><li>• <b>Diseñar</b> aplicaciones cuya ejecución no es siempre lineal, permitiendo, a partir de la entrada de datos, discriminar entre distintos escenarios.</li></ul>	

Unidad Didáctica 3:	Bucles
CONTENIDOS	



<p>3.1. Bucles controlados por condición</p> <p>    3.1.1. while</p> <p>    3.1.2. do-while</p> <p>3.2. Bucles controlados por contador: for</p> <p>3.3. Salidas anticipadas</p> <p>3.4. Bucles anidados</p> <p>    3.4.1. Bucles independientes</p> <p>    3.4.2. Bucles dependientes</p>
<b>OBJETIVOS</b>
<ul style="list-style-type: none"> <li>• <b>Conocer</b> y utilizar las estructuras de repetición.</li> <li>• <b>Programar</b> aplicaciones que repiten conjuntos de instrucciones mediante el uso de bucles.</li> <li>• <b>Distinguir</b> entre un bucle controlado por contador, un bucle precondición y un bucle poscondición.</li> <li>• <b>Utilizar</b> las estructuras adecuadas de control para conseguir que un programa funcione según las especificaciones funcionales.</li> </ul>

Unidad Didáctica 4:	Funciones
<b>CONTENIDOS</b>	
<p>4.1. Conceptos básicos</p> <p>4.2. Ámbito de las variables</p> <p>4.3. Paso de información a una función</p> <p>    4.3.1. Valores en la llamada</p> <p>    4.3.2. Parámetros de entrada</p> <p>4.4. Valor devuelto por una función</p> <p>4.5. Sobrecarga de funciones</p> <p>4.6. Recursividad</p>	
<b>OBJETIVOS</b>	
<ul style="list-style-type: none"> <li>• Asimilar el concepto de función, las ventajas de su uso y la implicación en la mejora de mantenimiento de aplicaciones.</li> <li>• Entender y usar el concepto de parámetro de entrada, así como el mecanismo para generalizar el comportamiento de las funciones.</li> <li>• Escribir programas que hagan un uso adecuado de las funciones y del valor devuelto por estas.</li> <li>• Resolver problemas mediante el uso de funciones recursivas.</li> </ul>	

Unidad Didáctica 5:	Tablas
<b>CONTENIDOS</b>	
<p>5.1. Variables escalares versus tablas</p>	

- 5.2. Índices
  - 5.2.1. Índices fuera de rango
- 5.3. Construcción de tablas
  - 5.3.1. Longitud y tipo
  - 5.3.2. Variables de tabla
  - 5.3.3. Operador new
- 5.4. Referencias
  - 5.4.1. Recolector de basura
  - 5.4.2. Referencia null
- 5.5. Uso de tablas
  - 5.5.1. Tablas ordenadas
  - 5.5.1. Tablas + indicador
- 5.6. Tablas como parámetros de funciones
- 5.7. Operaciones con tablas: la clase Arrays
  - 5.7.1. Obtención del número de elementos de una tabla
  - 5.7.2. Inicialización
  - 5.7.3. Recorrido
  - 5.7.4. Mostrar una tabla
  - 5.7.5. Ordenación
  - 5.7.6. Búsqueda
  - 5.7.7. Copia
  - 5.7.8. Inserción
  - 5.7.9. Eliminación
  - 5.7.10. Comparación de dos tablas
- 5.8. Tablas n-dimensionales
  - 5.8.1. Tablas bidimensionales
  - 5.8.2. Tablas tridimensionales
  - 5.8.3. Tablas con más dimensiones

### OBJETIVOS

- **Conocer** las tablas, que permiten almacenar múltiples valores en una variable.
- **Crear** tablas de distinto tipo y longitudes.
- **Utilizar** las operaciones básicas que se emplean con las tablas.
- **Diseñar** programas que hagan uso de tablas, donde se almacenan los datos necesarios.
- **Modificar** la longitud de una tabla en tiempo de ejecución sin pérdida de los datos que contiene.
- **Usar** la API de Java relacionada con las tablas y aplicar su uso a la resolución de problemas.

Unidad Didáctica 6:

Cadenas de caracteres

### CONTENIDOS

- 6.1. Tipo primitivo char
  - 6.1.1. Unicode
  - 6.1.2. Secuencias de escape

- 6.1.3. Conversión char ↔ int
- 6.1.4. Aritmética de caracteres
- 6.2. Clase Character
  - 6.2.1. Clasificación de caracteres
  - 6.2.2. Conversión
- 6.3. Clase String
  - 6.3.1. Inicialización de cadenas
  - 6.3.2. Comparación
  - 6.3.3. Concatenación
  - 6.3.4. Obtención de caracteres
  - 6.3.5. Longitud de una cadena
  - 6.3.6. Búsqueda
  - 6.3.7. Comprobaciones
  - 6.3.8. Conversión
  - 6.3.9. Separación en partes
- 6.4. Cadenas y tablas de caracteres

### OBJETIVOS

- **Utilizar** el tipo primitivo char.
- **Conocer** las funcionalidades que proporciona la clase Character para la manipulación de caracteres.
- **Usar** la clase String como parte de las aplicaciones que se construyan.
- **Profundizar** en el uso de operaciones avanzadas con texto y realizar implementaciones acordes a los requisitos del sistema.
- **Plantear** distintas alternativas y elegir la solución óptima en cada caso, según el problema que se necesite resolver.
- **Conocer** la API de Java, que permite codificar aplicaciones que utilizan datos de tipo texto.
- **Exponer** las ventajas e inconvenientes de las posibles herramientas que se usan para la creación y manipulación de texto.

Unidad Didáctica 7:

Clases

### CONTENIDOS

- 7.1. Definición de una clase
- 7.2. Crear una clase desde NetBeans
- 7.3. Atributos
  - 7.3.1. Inicialización
- 7.4. Objetos
  - 7.4.1. Referencias
  - 7.4.2. Variables referencia
  - 7.4.3. Operador new
  - 7.4.4. Referencia null
  - 7.4.5. Recolector de basura

- 7.5. Métodos
  - 7.5.1. Ámbito de las variables y atributos
  - 7.5.2. Ocultación de atributos
  - 7.5.3. Objeto this
- 7.6. Atributos y métodos estáticos
- 7.7. Constructores
  - 7.7.1. this()
- 7.8. Paquetes
  - 7.8.1. Crear un paquete desde NetBeans
- 7.9. Modificadores de acceso
  - 7.9.1. Modificadores de acceso para clases
  - 7.9.2. Modificadores de acceso para miembros
  - 7.9.3. Métodos get/set
- 7.10. Enumerados

#### OBJETIVOS

- **Comprender** y **asimilar** los conceptos de la POO.
- **Escribir** programas que hagan uso de la POO para solucionar problemas, facilitando la tarea del programador.
- **Decidir** si los métodos y atributos serán visibles o no, para clases externas y vecinas.
- **Implementar** métodos que permitan la asignación controlada de los atributos no visibles, así como otros que posibiliten consultar los valores de estos atributos.

Unidad Didáctica 8:

Herencia

#### CONTENIDOS

- 8.1. Subclase y superclase
- 8.2. Modificador de acceso para herencia
- 8.3. Redefinición de miembros heredados
  - 8.3.1. super y super()
  - 8.3.2. Selección dinámica de métodos
- 8.4. La clase Object
  - 8.4.1. Método toString()
  - 8.4.2. Método equals()
  - 8.4.3. Método getClass()
- 8.5. Clases abstractas

#### OBJETIVOS

- **Conocer** la idea y la necesidad de la herencia entre clases.
- **Saber** cuáles son los conceptos de subclase y superclase.
- **Utilizar** la herencia como mecanismo de especialización.

- **Conocer** las limitaciones de acceso a los miembros de una superclase.
- **Comprender** y **usar** el mecanismo de la sustitución de métodos.
- **Utilizar** el acceso a miembros sustituidos de una superclase.
- **Comprender** y **usar** la selección dinámica de métodos en tiempo de ejecución.
- **Reconocer** las principales funcionalidades definidas en la clase Object.
- **Usar** la sustitución de los métodos de Object. En particular, la implementación de toString() y equals().
- **Conocer** y **utilizar** las clases abstractas.
- **Usar** las clases abstractas para la selección dinámica de métodos.

Unidad Didáctica 9:	Interfaces
<b>CONTENIDOS</b>	
9.1. Concepto de interfaz <ul style="list-style-type: none"> <li>9.1.1. Definición de una interfaz</li> <li>9.1.2. Implementación de una interfaz</li> </ul> 9.2. Atributos de una interfaz           9.3. Métodos implementados en una interfaz <ul style="list-style-type: none"> <li>9.3.1. Métodos por defecto</li> <li>9.3.2. Métodos estáticos en una interfaz</li> <li>9.3.3. Métodos privados</li> </ul> 9.4. Herencia           9.5. Variables de tipo interfaz           9.6. Clases anónimas           9.7. Acceso entre miembros de una interfaz           9.8. Sintaxis general           9.9. Un par de interfaces de la API <ul style="list-style-type: none"> <li>9.9.1. Interfaz Comparable</li> <li>9.9.2. Interfaz Comparator</li> </ul>	
<b>OBJETIVOS</b>	
<ul style="list-style-type: none"> <li>• <b>Conocer</b> la idea y la necesidad de las interfaces y distinguirlas de las clases abstractas.</li> <li>• <b>Definir</b> métodos abstractos en una interfaz.</li> <li>• <b>Implementar</b> una o más interfaces en una clase.</li> <li>• <b>Implementar</b> métodos de extensión en una interfaz.</li> <li>• <b>Implementar</b> métodos privados como auxiliares en una interfaz.</li> <li>• <b>Diseñar</b> interfaces para operaciones específicas en clases diversas.</li> <li>• <b>Utilizar</b> variables de tipo interfaz para conseguir la selección dinámica de métodos.</li> </ul>	

- **Conocer** qué son la herencia simple y múltiple de interfaces.
- **Conocer** algunas interfaces importantes de la API.
- **Saber** implementar clases anónimas.
- **Implementar** criterios de comparación y aplicarlos a procesos de búsqueda y ordenación.

Unidad Didáctica 10:	Ficheros de texto
CONTENIDOS	
<p>10.1. Excepciones</p> <p style="padding-left: 20px;">10.1.1. Requisito de captura o especificación</p> <p style="padding-left: 20px;">10.1.2. Excepciones de usuario</p> <p>10.2. Flujos de entrada de texto</p> <p>10.3. Scanner y flujos de entrada</p> <p>10.4. Flujos de salida de texto</p> <p>10.5. Ficheros XML y Java. API JAXB</p>	
OBJETIVOS	
<ul style="list-style-type: none"> <li>• <b>Conocer</b> el concepto de excepción y sus tipos.</li> <li>• <b>Manipular</b> excepciones.</li> <li>• <b>Crear</b> excepciones de usuario.</li> <li>• <b>Saber</b> qué es el concepto de flujos de entrada y salida.</li> <li>• <b>Crear</b> flujos de entrada de texto.</li> <li>• <b>Conocer</b> y <b>aplicar</b> funciones de lectura en flujos de entrada de texto.</li> <li>• <b>Utilizar</b> la entrada formateada desde archivos de texto con Scanner.</li> <li>• <b>Crear</b> flujos de salida de texto.</li> <li>• <b>Conocer</b> y aplicar funciones de escritura en flujos de salida de texto.</li> <li>• <b>Cerrar</b> flujos de entrada y salida de texto.</li> <li>• <b>Leer</b> y <b>escribir</b> entre archivos XML y aplicaciones Java con JAXB.</li> </ul>	

Unidad Didáctica 11:	Ficheros binarios
CONTENIDOS	
<p>11.1. Flujos de salida binarios</p> <p>11.2. Flujos de entrada binarios</p> <p>11.3. Ficheros binarios y objetos complejos</p>	
OBJETIVOS	

- **Abrir** y **cerrar** archivos binarios.
- **Escribir** datos primitivos y objetos en archivos binarios.
- **Leer** datos primitivos y objetos de archivos binarios.
- **Conocer** las distintas excepciones que se pueden arrojar durante la apertura y cierre de archivos binarios.
- **Identificar** las distintas excepciones que se pueden arrojar durante la escritura y lectura de datos en archivos binarios.
- **Gestionar** sistemas complejos donde se guarda y recupera información.

Unidad Didáctica 12:	Colecciones
<b>CONTENIDOS</b>	
<p>12.1. Tipos parametrizados o genéricos</p> <ul style="list-style-type: none"> <li>12.1.1. Clases con parámetros genéricos</li> <li>12.1.2. Interfaces con genéricos</li> <li>12.1.3. Parámetros genéricos limitados</li> <li>12.1.4. Métodos genéricos</li> <li>12.1.5. Comodines</li> <li>12.1.6. Cosas que no se pueden hacer con parámetros genéricos</li> </ul> <p>12.2. Interfaz Collection</p> <ul style="list-style-type: none"> <li>12.2.1. Breve presentación de las listas</li> <li>12.2.2. Métodos básicos de la interfaz Collection</li> <li>12.2.3. Métodos globales de la interfaz Collection</li> <li>12.2.4. Métodos de tabla de la interfaz Collection</li> </ul> <p>12.3. Métodos específicos de la interfaz List</p> <p>12.4. Interfaz Set</p> <p>12.5. Conversiones entre colecciones</p> <p>12.6. Clase Collections</p> <p>12.7. Interfaz Map</p> <ul style="list-style-type: none"> <li>12.7.1. Vistas Collection de los mapas</li> <li>12.7.2. Implementaciones de Map</li> </ul>	
<b>OBJETIVOS</b>	
<ul style="list-style-type: none"> <li>• <b>Implementar</b> clases, interfaces y métodos con tipos genéricos.</li> <li>• <b>Conocer</b> la interfaz Collection y sus métodos básicos y globales.</li> <li>• <b>Conocer</b> la interfaz List y sus métodos.</li> <li>• <b>Utilizar</b> las implementaciones de List: ArrayList y LinkedList.</li> <li>• <b>Conocer</b> la interfaz Set.</li> <li>• <b>Usar</b> las implementaciones de Set: HashSet, TreeSet y LinkedHashSet.</li> <li>• <b>Emplear</b> las conversiones entre distintas implementaciones de List y Set.</li> </ul>	

- **Conocer** la interfaz Map.
- **Usar** los métodos de Map a través de sus implementaciones HashMap, TreeMap y linkedHashMap.
- **Emplear** las vistas Collection de los mapas.

Unidad Didáctica 13:	Conexión a base de datos: JDBC
CONTENIDOS	
13.1. API JDBC 13.2. Driver 13.3. Conexión 13.4. Ejecución de sentencias <ul style="list-style-type: none"> <li>13.4.1. Ejecución de consultas (SELECT)</li> <li>13.4.2. Ejecución de sentencias INSERT, UPDATE o DELETE</li> </ul> 13.5. Clase ResultSet <ul style="list-style-type: none"> <li>14.5.1. Tipos de ResultSet</li> <li>14.5.2. Métodos para mover el cursor</li> <li>14.5.3. Ubicación del cursor</li> </ul> 13.6. SQL Injection 13.7. Sentencias parametrizadas 13.8. Operaciones CRUD 13.9. Objeto de acceso a datos	
OBJETIVOS	
<ul style="list-style-type: none"> <li>• <b>Realizar</b> la persistencia de los datos de una aplicación mediante los servicios de un SGBD.</li> <li>• <b>Configurar</b> el driver JDBC para poder acceder a SGBD de distintos fabricantes.</li> <li>• <b>Crear</b> una conexión entre un programa Java y un SGBD.</li> <li>• <b>Ejecutar</b> instrucciones de un SGBD desde una aplicación que funciona como cliente.</li> <li>• <b>Insertar</b> los datos de una aplicación en una base de datos.</li> <li>• <b>Recuperar</b> información almacenada en una base de datos, para su posterior uso, desde un programa Java.</li> <li>• <b>Realizar</b> el mapeo objeto-relacional de clases simples.</li> </ul>	

Unidad Didáctica 14:	API de persistencia de Java
CONTENIDOS	
14.1. Persistencia <ul style="list-style-type: none"> <li>14.1.1. Persistencia con un SGBD</li> <li>14.1.2. Mapeo objeto-relacional</li> <li>14.1.3. Técnicas de persistencias</li> </ul>	



- 14.1.4. JPA
- 14.2. Entidades
  - 14.2.1. Identificador autogestionado
- 14.3. Unidad de persistencia
- 14.4. Gestor de entidades
  - 14.4.1. Métodos de EntityManager
  - 14.4.2. Transacciones
- 14.5. Operaciones CRUD
  - 14.5.1. Create
  - 14.5.2. Read
  - 14.5.3. Update
  - 14.5.4. Delete
- 14.6. Controlador de JPA
- 14.7. JPQL
  - 14.7.1. SELECT
  - 14.7.2. Ejecución de una consulta
  - 14.7.3. Otras consultas con SELECT
  - 14.7.4. UPDATE y DELETE
  - 14.7.5. Consultas parametrizadas
  - 14.7.6. Consultas con nombre
- 14.8. Herencia
- 14.9. Asociaciones
  - 14.9.1. Uno a uno
  - 14.9.2. Uno a muchos
  - 14.9.3. Muchos a uno unidireccional
  - 14.9.4. Muchos a muchos
  - 14.9.5. Configuración de las asociaciones
  - 14.9.6. JPQL con colecciones
- 14.10. Creación de entidades desde la BD

## OBJETIVOS

- **Establecer** los conceptos de persistencia que son necesarios para salvaguardar y recuperar los datos de cualquier aplicación, así como los conceptos y definiciones propias de Java Persistence API.
- **Concretar** la importancia de los mecanismos de conversión entre datos mediante objetos y datos, utilizando un sistema gestor de base de datos relacional.
- **Clasificar** las relaciones existentes entre clases, según su navegabilidad y cardinalidad.
- **Profundizar** en el concepto de persistencia de objeto, vista en los ficheros, y sus alternativas.
- **Asimilar** las distintas anotaciones y clases de Java Persistence API, así como su lenguaje de consulta Java Persistence Query Language.
- **Conocer** las herramientas que ofrece NetBeans para la gestión de JPA: generación automática de controladores de entidad: gestión gráfica del fichero persistence.xml, asistentes para la

creación de conexiones y entidades a partir de sus respectivos esquemas relacionales.

Unidad Didáctica 15:	Stream
CONTENIDOS	
15.1. Interfaces funcionales y expresiones lambda 15.2. Algunas interfaces funcionales de la API 15.2.1. Referencias a métodos 15.3. Interfaz Stream 15.3.1. Formas de crear un Stream 15.3.2. Tuberías o pipelines	
OBJETIVOS	
<ul style="list-style-type: none"><li>• Saber implementar interfaces funcionales en línea.</li><li>• Implementar expresiones lambda.</li><li>• Conocer algunas interfaces funcionales importantes: Predicate, Function y Consumer.</li><li>• Saber aplicar las referencias a métodos.</li><li>• Conocer la interfaz Stream con sus funcionalidades más importantes.</li><li>• Emplear las operaciones intermedias más importantes de los Stream.</li><li>• Usar tuberías en las operaciones encadenadas de los Stream.</li><li>• Utilizar las operaciones terminales más importantes de los Stream.</li></ul>	

Unidad Didáctica 16:	Interfaces de usuario con AWT y Swing
CONTENIDOS	
16.1. Abstract Windowing Toolkit (AWT) vs. Swing 16.2. Componentes GUI AWT 16.3. Gráficos 16.4. Gestores de Layout 16.5. Componentes Swing 16.6. Componentes Swing 16.7. Applets 16.8. Modelo de delegación de eventos 16.9. Frame Windows 16.10. Uso de clases Inner para Listeners 16.11. Construcción de aplicaciones con GUIs 16.12. Multimedia con Java	
OBJETIVOS	
<ul style="list-style-type: none"><li>• <b>Escribir</b> programas para dibujar elementos gráficos</li><li>• <b>Escribir</b> programas con interfaces de usuario elaboradas</li></ul>	

- **Conocer** la tecnología Applet
- **Conocer** la tecnología Applet
- **Conocer** las API para mostrar imágenes y reproducir sonidos

### 3.3 Distribución temporal

U. D.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
CARGA LECTIVA	9h	9h	9h	9h	9h	9h	24h	15h	16h	15h	15h	24h	15h	26h	26h	26h

La distribución por trimestres de cada una de las unidades didácticas sería la siguiente:

TRIMESTRE	UNIDADES DIDÁCTICAS
1 <sup>ER</sup> TRIMESTRE	1, 2, 3, 4, 5, 6, 7, 8, 9 y 10
2 <sup>O</sup> TRIMESTRE	10, 11, 12, 13 y 14
3 <sup>ER</sup> TRIMESTRE	14, 15 Y 16

Hay que tener en cuenta, para esta distribución horaria, que parte de su formación, debido al proyecto de FP DUAL, la realizarán en la empresa durante el tercer trimestre y, los resultados de aprendizaje correspondientes a las unidades que se citan anteriormente, se evaluarán con las actividades propuestas en la empresa, además de otras que completarán los distintos resultados de aprendizaje.

Así mismo, los alumnos que por las razones expuestas en el proyecto no realicen su formación en la empresa, realizarán las actividades conducentes a la adquisición de los resultados de aprendizaje en el centro educativo según la distribución temporal de esta programación.

### 3.4 Criterios de evaluación

En el ciclo formativo se tiene en cuenta para la evaluación del módulo unos criterios de evaluación englobados dentro de unos resultados de aprendizaje, los cuales son los siguientes:

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

Criterios de evaluación:

- 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 utilizado entornos integrados de desarrollo.
- d Se han identificado los distintos tipos de variables y la utilidad específica de cada uno.
- e Se ha modificado el código de un programa para crear y utilizar variables.
- f Se han creado y utilizado constantes y literales.
- g Se han clasificado, reconocido y utilizado en expresiones los operadores del lenguaje.
- h Se ha comprobado el funcionamiento de las conversiones de tipo explícitas e implícitas.
- i Se han introducido comentarios en el código.

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

Criterios de evaluación:

- 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 escrito llamadas a métodos estáticos.
- f Se han utilizado parámetros en la llamada a métodos.
- g Se han incorporado y utilizado librerías de objetos.
- h Se han utilizado constructores.
- i Se ha utilizado el entorno integrado de desarrollo en la creación y compilación de programas simples.

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

Criterios de evaluación:

- 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.
- g Se ha comentado y documentado el código.

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

Criterios de evaluación:

- 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.
- h Se han creado y utilizado métodos estáticos.
- i Se han definido y utilizado interfaces.

- j Se han creado y utilizado conjuntos y librerías de clases.

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

Criterios de evaluación:

- a Se ha utilizado la consola para realizar operaciones de entrada y salida de información.
- b Se han aplicado formatos en la visualización de la información.
- c Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.
- d Se han utilizado ficheros para almacenar y recuperar información.
- e Se han creado programas que utilicen diversos métodos de acceso al contenido de los ficheros.
- f Se han utilizado las herramientas del entorno de desarrollo para crear interfaces gráficos de usuario simples.
- g Se han programado controladores de eventos.
- h Se han escrito programas que utilicen interfaces gráficos para la entrada y salida de información.

## **6. Escribe programas que manipulen información, seleccionando y utilizando tipos avanzados de datos.**

Criterios de evaluación:

- a Se han escrito programas que utilicen 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.
- e Se han reconocido las características y ventajas de cada una de la colecciones de datos disponibles.
- f Se han creado clases y métodos genéricos.
- g Se han utilizado expresiones regulares en la búsqueda de patrones en cadenas de texto.
- h Se han identificado las clases relacionadas con el tratamiento de documentos XML.
- i Se han realizado programas que realicen manipulaciones sobre documentos XML.

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

Criterios de evaluación:

- a Se han identificado los conceptos de herencia, superclase y 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.
- f Se han probado y depurado las jerarquías de clases.
- g Se han realizado programas que implementen y utilicen jerarquías de clases.
- h Se ha comentado y documentado el código.

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

Criterios de evaluación:

- a Se han identificado las características de las bases de datos orientadas a objetos.
- b Se ha analizado su aplicación en el desarrollo de aplicaciones mediante lenguajes orientados a objetos.
- c Se han instalado sistemas gestores de bases de datos orientados a objetos.
- d Se han clasificado y analizado los distintos métodos soportados por los sistemas gestores para la gestión de la información almacenada.
- e Se han creado bases de datos y las estructuras necesarias para el almacenamiento de objetos.
- f Se han programado aplicaciones que almacenen objetos en las bases de datos creadas.
- g Se han realizado programas para recuperar, actualizar y eliminar objetos de las bases de datos.
- h Se han realizado programas para almacenar y gestionar tipos de datos estructurados, compuestos y relacionados.

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

Criterios de evaluación:

- a Se han identificado las características y métodos de acceso a sistemas gestores de bases de datos relacionales.
- b Se han programado conexiones con bases de datos.
- c Se ha escrito código para almacenar información en bases de datos.
- d Se han creado programas para recuperar y mostrar información almacenada en bases de datos.
- e Se han efectuado borrados y modificaciones sobre la información almacenada.
- f Se han creado aplicaciones que ejecuten consultas sobre bases de datos.
- g Se han creado aplicaciones para posibilitar la gestión de información presente en bases de datos relacionales.

Así mismo, los cada uno de los contenidos del módulo están pensados para alcanzar estos resultados de aprendizaje y, par evaluar dicho logro, se tendrá en cuenta si se ha obtenido una evaluación positiva en los criterios de evaluación asociados. La correspondencia de la unidades didácticas con dichos criterios se esboza en la tabla siguiente:



## **4 Contribución de la materia a la adquisición de las competencias básicas**

El perfil profesional del título de Técnico Superior en Desarrollo de Aplicaciones Web queda determinado por su competencia general, sus competencias profesionales, personales y sociales, y por la relación de cualificaciones y, en su caso, unidades de competencia del Catálogo Nacional de Cualificaciones Profesionales incluidas en el título.

### **4.1 Competencia general**

La competencia general de este título consiste en desarrollar, implantar, y mantener aplicaciones web, con independencia del modelo empleado y utilizando tecnologías específicas, garantizando el acceso a los datos de forma segura y cumpliendo los criterios de accesibilidad, usabilidad y calidad exigidas en los estándares establecidos.

### **4.2 Competencias profesionales, personales y sociales**

Las competencias profesionales, personales y sociales de este título son las que se relacionan a continuación:

- a** Configurar y explotar sistemas informáticos, adaptando la configuración lógica del sistema según las necesidades de uso y los criterios establecidos.
- b** Aplicar técnicas y procedimientos relacionados con la seguridad en sistemas, servicios y aplicaciones, cumpliendo el plan de seguridad.
- c** Gestionar servidores de aplicaciones adaptando su configuración en cada caso para permitir el despliegue de aplicaciones web.
- d** Gestionar bases de datos, interpretando su diseño lógico y verificando integridad, consistencia, seguridad y accesibilidad de los datos.
- e** Desarrollar aplicaciones web con acceso a bases de datos utilizando lenguajes, objetos de acceso y herramientas de mapeo adecuados a las especificaciones.
- f** Integrar contenidos en la lógica de una aplicación web, desarrollando componentes de acceso a datos adecuados a las especificaciones.
- g** Desarrollar interfaces en aplicaciones web de acuerdo con un manual de estilo, utilizando lenguajes de marcas y estándares web.
- h** Desarrollar componentes multimedia para su integración en aplicaciones web, empleando herramientas específicas y siguiendo las especificaciones establecidas.
- i** Integrar componentes multimedia en el interface de una aplicación web, realizando el análisis de interactividad, accesibilidad y usabilidad de la aplicación.
- j** Desarrollar e integrar componentes software en el entorno del servidor web, empleando herramientas y lenguajes específicos, para cumplir las especificaciones de la aplicación.
- k** Desarrollar servicios para integrar sus funciones en otras aplicaciones web, asegurando su funcionalidad.
- l** Integrar servicios y contenidos distribuidos en aplicaciones web, asegurando su funcionalidad.
- m** Completar planes de pruebas verificando el funcionamiento de los componentes software desarrollados, según las especificaciones.
- n** Elaborar y mantener la documentación de los procesos de desarrollo, utilizando herramientas de generación de documentación y control de versiones.



- ñ. Desplegar y distribuir aplicaciones web en distintos ámbitos de implantación, verificando su comportamiento y realizando modificaciones.
- o Gestionar y/o realizar el mantenimiento de los recursos de su área en función de las cargas de trabajo y el plan de mantenimiento.
- p Adaptarse a las nuevas situaciones laborales, manteniendo actualizados los conocimientos científicos, técnicos y tecnológicos relativos a su entorno profesional, gestionando su formación y los recursos existentes en el aprendizaje a lo largo de la vida y utilizando las tecnologías de la información y la comunicación.
- q Resolver situaciones, problemas o contingencias con iniciativa y autonomía en el ámbito de su competencia, con creatividad, innovación y espíritu de mejora en el trabajo personal y en el de los miembros del equipo.
- r Organizar y coordinar equipos de trabajo, supervisando el desarrollo de este, con responsabilidad, manteniendo relaciones fluidas y asumiendo el liderazgo, así como, aportando soluciones a los conflictos grupales que se presentan.
- s Comunicarse con sus iguales, superiores, clientes y personas bajo su responsabilidad utilizando vías eficaces de comunicación, transmitiendo la información o conocimientos adecuados, y respetando la autonomía y competencia de las personas que intervienen en el ámbito de su trabajo.
- t Generar entornos seguros en el desarrollo de su trabajo y el de su equipo, supervisando y aplicando los procedimientos de prevención de riesgos laborales y ambientales de acuerdo con lo establecido por la normativa y los objetivos de la empresa.
- u Supervisar y aplicar procedimientos de gestión de calidad, de accesibilidad universal y de diseño para todos, en las actividades profesionales incluidas en los procesos de producción o prestación de servicios.
- v Realizar la gestión básica para la creación y funcionamiento de una pequeña empresa y tener iniciativa en su actividad profesional con sentido de la responsabilidad social.
- x. Ejercer sus derechos y cumplir con las obligaciones derivadas de su actividad profesional, de acuerdo con lo establecido en la legislación vigente, participando activamente en la vida económica, social y cultural.

### **4.3 Cualificaciones y unidades de competencia del Catálogo Nacional de Cualificaciones Profesionales incluidas en el título**

#### **4.3.1. Cualificaciones profesionales completas:**

Desarrollo de aplicaciones con tecnologías Web IFC154\_3 (Real Decreto 1087/2005, de 16 de septiembre), que comprende las siguientes unidades de competencia:

UC0491\_3 Desarrollar elementos software en el entorno cliente.

UC0492\_3 Desarrollar elementos software en el entorno servidor.

UC0493\_3 Implementar, verificar y documentar aplicaciones web en entornos internet, intranet y extranet.

#### **4.3.2. Cualificaciones profesionales incompletas:**

- a) Programación en lenguajes estructurados de aplicaciones de gestión IFC155\_3 (Real Decreto 1087/2005, de 16 de septiembre).

UC0223\_3 Configurar y explotar sistemas informáticos.

UC0226\_3 Programar bases de datos relacionales.

- b) Programación con lenguajes orientados a objetos y bases de datos relacionales IFC 080\_3 (Real Decreto. 295/2004, de 20 de febrero).

UC0223\_3 Configurar y explotar sistemas informáticos.

UC0226\_3 Programar bases de datos relacionales.

### **4.3.3. Correspondencia del módulo profesional con las unidades de competencia**

El módulo profesional 0485 Programación se considera un módulo transversal y no tiene asociado ninguna unidad de competencia específica.

## **5 Principios pedagógicos de la materia**

Antes de empezar a pensar en los principios pedagógicos del módulo hay que partir de una idea fundamental: El alumno es el principal artífice en el proceso de aprendizaje y el profesor es el que lo acompaña en el proceso, permitiendo establecer relaciones entre los conocimientos y experiencias previas con los nuevos contenidos”.

Además, se tendrán en cuenta las distintas competencias profesionales, personales y sociales que el alumno debe de alcanzar y que se han redactado previamente.

Considerando todo esto, los principios pedagógicos a tener en cuenta en nuestro módulo serían los siguientes:

- El alumno debe ser el centro del aprendizaje y debe asumir un papel activo. El proceso debe tener en cuenta sus intereses, conocimientos previos, características personales, etc. La diversidad es por ello un elemento consustancial, se debe buscar la individualización del proceso: los alumnos deben buscar, analizar, experimentar, aplicar, comunicar y no únicamente recibir y memorizar. Las situaciones deberán ser flexibles y variadas.
- Necesidad de motivar, de despertar la curiosidad, de emocionar a nuestros alumnos y buscar estrategias para desarrollar a motivación intrínseca.
- Aprendizaje significativo, pues no se trata de “transmitir” conocimientos, sino de diseñar procesos que permitan a nuestros alumnos construir y desarrollar sus propios esquemas de conocimiento, a la vez que aprenden cómo funciona su propio proceso de aprendizaje y estrategias para controlarlo y optimizarlo.
- Enfoque centrado en las competencias: debemos tener siempre como referente lograr que nuestros alumnos sean competentes en la vida real, es decir, que no solo tengan conocimientos, sino que sepan aplicarlos en la búsqueda de soluciones a problemas reales, para ello necesitaremos recrear contextos adecuados y propuestas auténticas.
- Importancia de crear en el aula un clima adecuado, seguro y estimulante que favorezca la interacción y potencie los procesos de aprendizaje. El aprendizaje es un proceso social en

gran medida, por lo de que debemos de crear las condiciones para que este proceso se desarrolle. La cooperación entre iguales es una estrategia valiosísima en el aprendizaje: cooperación, diálogo, debate, respeto a las diferencias, saber escuchar, desarrollar la empatía son estrategias fundamentales que se desarrollarán en este contexto.

- Buscar el desarrollo integral de los alumnos, es decir, todas las facetas de la personalidad deben ser tenidas en cuenta en la escuela, evitar centrarnos solo en el aspecto intelectual: la afectividad, las funciones no cognitivas, las emociones, la creatividad, las relaciones sociales, el desarrollo moral... todos son aspectos que debemos cuidar y respetar.
- Uso de recursos didácticos variados que faciliten el proceso de representación y abstracción que permitan ajustar el proceso a las capacidades, estilos y preferencias de todos los alumnos.
- Enfoque globalizador e interdisciplinar: los alumnos no acceden a la realidad de forma parcelada, sino que la entienden como un todo, de ahí la necesidad de presentar los contenidos relacionados entre sí de forma que tengan significado. Lo que implica que el proceso educativo precisa de un proceso de coordinación docente.
- La evaluación como parte fundamental del proceso de enseñanza-aprendizaje, y no solo para comprobar el nivel de desarrollo del proceso sino como herramienta de valoración y mejora del propio proceso.

Como consecuencia de estos principios se hace necesario diseñar metodologías activas e inclusivas.

## 6 Metodología

La metodología será activa, de manera que el aprendizaje resulte de la transmisión de conocimientos por parte del profesorado y de la acción del alumnado, estimulando la indagación personal, el razonamiento, el sentido crítico, la reflexión y la toma de una postura razonada. Por ello deben “aprender haciendo cosas”. Ello implica actividades basadas en la participación, motivación, discusión, flexibilidad, en el contacto con la realidad cotidiana y el trabajo en equipo. Además los contenidos de lo aprendido deben resultar funcionales, aplicables a circunstancias reales de la vida cotidiana.

La impartición del módulo se fundamentará en los siguientes aspectos:

- Para la explicación de cada Unidad se realizará una **exposición teórica de los contenidos** de la unidad por parte del profesor/a, que utilizará un apoyo gráfico gracias a uso de presentaciones de diapositivas, videotutoriales, etc..
- Posteriormente se realizarán una serie de **ejercicios propuestos por el profesor/a y resueltos** por él/ella **en clase**. El objetivo de estos ejercicios es llevar a la práctica los conceptos teóricos que se asimilaron en la exposición teórica anterior.
- El profesor/a realizará la labor de guía a la hora de resolver todas las dudas que pueda tener el alumnado, tanto teóricas como prácticas, de forma que ellos investiguen las posibles soluciones al problema planteado. Si se considerase necesario se realizarán

ejercicios específicos que aclaren los conceptos que más cueste comprender al alumnado.

- El profesor/a propondrá un conjunto de **ejercicios**, de contenido similar a los que ya se han resuelto en clase, **que deberán ser resueltos por el alumnado**, bien en horas de clase o bien en casa.
- Los **ejercicios prácticos** se realizarán en el aula. Las prácticas se resolverán de forma individual o en grupo, depende del número de alumnos/as que haya por ordenador. (No es aconsejable que haya más de dos alumnos/as por cada equipo informático).
- Además se podrá proponer algún **trabajo que englobe conocimientos** de varias unidades de trabajo para comprobar que los conocimientos mínimos exigidos en cada una de las unidades han sido satisfactoriamente asimilados. Sería recomendable un trabajo por cada evaluación.
- Durante todo el curso estará activo Moodle Centros que nos servirá de plataforma de comunicación con el alumnado, favoreciendo la entrega de materiales a los alumnos, la recogida de trabajos, comunicación de fechas importantes, foros, etc.

## **7 Evaluación Inicial**

Este curso académico, el grupo de alumnos en este módulo está formado por 19 alumnos y una alumna y, una vez desarrollada la evaluación inicial se observa que es un grupo trabajador y con bajos conocimientos en lo que a programación se refiere, con algunas excepciones. El nivel es parecido en todos los alumnos, aunque el alumnado que viene del CFGM de SMR se nota algo más preparado en la materia de informática en general pero no es una preparación propia de la materia de programación.

Por todo ello, se partirá desde una introducción básica a la programación y se irán introduciendo gradualmente el resto de contenidos a tratar. Como se ha comentado antes, el alumnado parece trabajador y está atento a las explicaciones de clase, por ello creo que la dinámica de las clases será adecuada.

Algún alumnado presenta algunas más dificultades que otro, con lo cual, se tiene que prestar más atención y adaptar la metodología para que los conceptos y su posterior aplicación práctica sea lo más asequible posible, todo ello teniendo en cuenta que todos deben alcanzar todos los resultados de aprendizaje para obtener una calificación positiva en el módulo.

## 8 Los procedimientos de evaluación del alumnado y los criterios de calificación, en consonancia con las orientaciones metodológicas establecidas

### 8.1 Instrumentos de evaluación

En este módulo se emplearán distintos instrumentos para evaluar la consecución de los distintos resultados de aprendizaje definidos en el apartado siguiente. Entre los distintos instrumentos son:

Exámenes orientados a evaluar los distintos criterios de evaluación

Ejercicios prácticos, que se realizarán de manera individual o en grupo, pudiéndose utilizar para ver su resolución:

- Rúbricas
- Listas de cotejo.

Observación directa.

### 8.2 criterios de calificación

%	Resultados de aprendizaje	Criterios de evaluación	Peso
12%	<b>RA1. Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación</b>	a) Se han identificado los bloques que componen la estructura de un programa informático.	11%
		b) Se han creado proyectos de desarrollo de aplicaciones	5%
		c) Se han utilizado entornos integrados de desarrollo.	5%
		d) Se han identificado los distintos tipos de variables y la utilidad específica de cada uno.	13%
		e) Se ha modificado el código de un programa para crear y utilizar variables.	13%
		f) Se han creado y utilizado constantes y literales.	13%
		g) Se han clasificado, reconocido y utilizado en expresiones los operadores del lenguaje.	13%

	<b>utilizado.</b>	h) Se ha comprobado el funcionamiento de las conversiones de tipo explícitas e implícitas.	<b>14%</b>
		i) Se han introducido comentarios en el código.	<b>13%</b>
	<b>Subtotal RA1</b>		<b>100%</b>
<b>15%</b>	<b>RA2. Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.</b>	a) Se han identificado los fundamentos de la programación orientada a objetos.	<b>11%</b>
		b) Se han escrito programas simples.	<b>12%</b>
		c) Se han instanciado objetos a partir de clases predefinidas.	<b>12%</b>
		d) Se han utilizado métodos y propiedades de los objetos.	<b>12%</b>
		e) Se han escrito llamadas a métodos estáticos.	<b>12%</b>
		f) Se han utilizado parámetros en la llamada a métodos.	<b>12%</b>
		g) Se han incorporado y utilizado librerías de objetos.	<b>12%</b>
		h) Se han utilizado constructores.	<b>12%</b>
		i) Se ha utilizado el entorno integrado de desarrollo en la creación y compilación de programas simples.	<b>5%</b>
	<b>Subtotal RA2</b>		<b>100%</b>
<b>12%</b>	<b>RA3. Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.</b>	a) Se ha escrito y probado código que haga uso de estructuras de selección.	<b>14%</b>
		b) Se han utilizado estructuras de repetición.	<b>15%</b>
		c) Se han reconocido las posibilidades de las sentencias de salto.	<b>12%</b>
		d) Se ha escrito código utilizando control de excepciones.	<b>14%</b>
		e) Se han creado programas ejecutables utilizando diferentes estructuras de control.	<b>15%</b>
		f) Se han probado y depurado los programas.	<b>14%</b>
		g) Se ha comentado y documentado el código.	<b>16%</b>
	<b>Subtotal RA3</b>		<b>100%</b>
<b>12%</b>	<b>RA4. Desarrolla programas organizados</b>	a) Se ha reconocido la sintaxis, estructura y componentes típicos de una clase.	<b>10%</b>
		b) Se han definido clases.	<b>10%</b>

	<b>en clases analizando y aplicando los principios de la programación orientada a objetos.</b>	c) Se han definido propiedades y métodos.	<b>10%</b>
		d) Se han creado constructores.	<b>10%</b>
		e) Se han desarrollado programas que instancien y utilicen objetos de las clases creadas anteriormente.	<b>10%</b>
		f) Se han utilizado mecanismos para controlar la visibilidad de las clases y de sus miembros.	<b>10%</b>
		g) Se han definido y utilizado clases heredadas.	<b>10%</b>
		h) Se han creado y utilizado métodos estáticos.	<b>10%</b>
		i) Se han definido y utilizado interfaces.	<b>10%</b>
		j) Se han creado y utilizado conjuntos y librerías de clases.	<b>10%</b>
		<b>Subtotal RA4</b>	<b>100%</b>
<b>8%</b>	<b>RA5. Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librerías de clases.</b>	a) Se ha utilizado la consola para realizar operaciones de entrada y salida de información.	<b>12,5%</b>
		b) Se han aplicado formatos en la visualización de la información.	<b>12,5%</b>
		c) Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.	<b>12,5%</b>
		d) Se han utilizado ficheros para almacenar y recuperar información.	<b>12,5%</b>
		e) Se han creado programas que utilicen diversos métodos de acceso al contenido de los ficheros.	<b>12,5%</b>
		f) Se han utilizado las herramientas del entorno de desarrollo para crear interfaces gráficos de usuario simples.	<b>12,5%</b>
		g) Se han programado controladores de eventos.	<b>12,5%</b>
		h) Se han escrito programas que utilicen interfaces gráficos para la entrada y salida de información.	<b>12,5%</b>
		<b>Subtotal RA5</b>	<b>100%</b>
<b>14%</b>	<b>RA6. Escribe programas que manipulen información,</b>	a) Se han escrito programas que utilicen arrays.	<b>12%</b>
		b) Se han reconocido las librerías de clases relacionadas con tipos de datos avanzados.	<b>11%</b>
		c) Se han utilizado listas para almacenar y procesar información.	<b>11%</b>
		d) Se han utilizado iteradores para recorrer los elementos de las listas.	<b>11%</b>

	<b>seleccionando y utilizando tipos avanzados de datos.</b>	e) Se han reconocido las características y ventajas de cada una de la colecciones de datos disponibles.	11%
		f) Se han creado clases y métodos genéricos.	11%
		g) Se han utilizado expresiones regulares en la búsqueda de patrones en cadenas de texto.	11%
		h) Se han identificado las clases relacionadas con el tratamiento de documentos XML.	11%
		i) Se han realizado programas que realicen manipulaciones sobre documentos XML.	11%
		<b>Subtotal RA6</b>	<b>100%</b>
<b>11%</b>	<b>RA7. Desarrolla programas, aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.</b>	a) Se han identificado los conceptos de herencia, superclase y subclase.	12,5%
		b) Se han utilizado modificadores para bloquear y forzar la herencia de clases y métodos.	12,5%
		c) Se ha reconocido la incidencia de los constructores en la herencia.	12,5%
		d) Se han creado clases heredadas que sobrescriban la implementación de métodos de la superclase.	12,5%
		e) Se han diseñado y aplicado jerarquías de clases.	12,5%
		f) Se han probado y depurado las jerarquías de clases.	12,5%
		g) Se han realizado programas que implementen y utilicen jerarquías de clases.	12,5%
		h) Se ha comentado y documentado el código.	12,5%
		<b>Subtotal RA7</b>	<b>100%</b>
<b>8%</b>	<b>RA8. Utiliza bases de datos orientadas a objetos, analizando sus características y aplicando técnicas para mantener la persistencia de la información.</b>	a) Se han identificado las características de las bases de datos orientadas a objetos.	12,5%
		b) Se ha analizado su aplicación en el desarrollo de aplicaciones mediante lenguajes orientados a objetos.	12,5%
		c) Se han instalado sistemas gestores de bases de datos orientados a objetos.	12,5%
		d) Se han clasificado y analizado los distintos métodos soportados por los sistemas gestores para la gestión de la información almacenada.	12,5%
		e) Se han creado bases de datos y las estructuras necesarias para el almacenamiento de objetos.	12,5%
		f) Se han programado aplicaciones que almacenen objetos en las bases de datos creadas.	12,5%
		g) Se han realizado programas para recuperar, actualizar y eliminar objetos de las bases de datos.	12,5%



		h) Se han realizado programas para almacenar y gestionar tipos de datos estructurados, compuestos y relacionados.	12,5%
		<b>Subtotal RA8</b>	<b>100%</b>
8%	<b>RA9. Gestiona información almacenada en bases de datos relacionales manteniendo la integridad y la consistencia de los datos.</b>	a) Se han identificado las características y métodos de acceso a sistemas gestores de bases de datos relacionales.	13%
		b) Se han programado conexiones con bases de datos.	14,5%
		c) Se ha escrito código para almacenar información en bases de datos.	14,5%
		d) Se han creado programas para recuperar y mostrar información almacenada en bases de datos.	14,5%
		e) Se han efectuado borrados y modificaciones sobre la información almacenada.	14,5%
		f) Se han creado aplicaciones que ejecuten consultas sobre bases de datos.	14,5%
		g) Se han creado aplicaciones para posibilitar la gestión de información presente en bases de datos relacionales.	14,5%
		<b>Subtotal RA9</b>	<b>100%</b>

## **9 Medidas de atención a la diversidad**

### **9.1 Medidas generales**

Hay que considerar que hay alumnos que les cuesta más comprender los contenidos porque tienen algún tipo de dificultad por ser algunos alumnos NEAE. Se ha considerado realizar las clases lo más prácticas posible, tratando de explicar los conceptos mediante ejemplos prácticos en vez de tantos conceptos teóricos y proponiendo varios tipos de ejercicios, partiendo de un nivel de dificultad fácil y que se vayan complicando conforme vayan avanzando. Se dará espacio y tiempo para que consulten con los compañeros y se integren en el grupo, siendo este tipo de actividades prácticas de tipo colaborativo.

También se realizarán proyectos en grupos pequeños, utilizando la técnica “divide y vencerás”, donde, el proyecto grande se divide en proyectos más pequeños y de menor dificultad que se repartirán entre cada uno de los miembros del grupo, para luego unir todos estos pequeños “trozos”, aumentando su dificultad de integración hasta conseguir resolver el problema que se proponía al principio del proyecto.

### **9.2 Clases de refuerzo/recuperación**

Aunque a lo largo del curso se propondrán distintas actividades de refuerzo y recuperación, puede haber alumnos que a finales de mayo no hayan logrado superar todos los resultados de aprendizaje para poder promocionar según indica la normativa. En este caso, los alumnos que no hayan alcanzado los resultados de aprendizaje de este módulo, dispondrán de unas tres semanas más o menos en el mes de junio (del 1 al 22) para realizar actividades de recuperación y poder superar el módulo, según indica la orden de evaluación de 29 de septiembre de 2010.

Asimismo, los alumnos que repitan el módulo por no haber alcanzado los distintos resultados de aprendizaje, asistirán a las clases y realizarán las mismas actividades que el resto de compañeros.

## **10 Materiales y recursos didácticos**

Material a disposición del alumnado:

- Ordenador individual con conexión a red e Internet.
- Sistema operativo Ubuntu instalado.
- Compilador/intérprete de Java.
- Entorno IDE NetBeans.

Material del aula:

- Proyector.

- Ordenador en la mesa del profesor con Ubuntu instalado.
- Eoptes instalado.

Además los alumnos dispondrán de una cuenta en el servidor Proxmox del Departamento con acceso externo.

## **11 Actividades complementarias y extraescolares relacionadas con el currículo que se proponen realizar por los departamentos de coordinación didáctica**

En el departamento se han propuesto una serie de actividades extraescolares y complementarias que se desglosarán en la correspondiente programación del departamento.

## **12 Concreción de planes, programas y proyectos del centro vinculados con desarrollo del currículo de la materia**

### **12.1 Proyecto de FP DUAL**

Este ciclo formativo se desarrolla atendiendo a un proyecto de FP Dual avalado por el claustro de profesores y el consejo escolar del instituto y, además, aprobado por la Consejería de Desarrollo Educativo y Formación Profesional. La redacción de dicho proyecto, en el que se desarrolla la formación en alternancia del alumnado, está disponible en Séneca. Los resultados de aprendizaje a alcanzar por parte del alumno en la empresa están descritos en dicho proyecto.

### **12.2 Fomento de la lectura**

El fomento de la lectura es una tarea distinta de la formación de la habilidad de leer, que se inicia en la familia y en la escuela, y que necesita de personas y lugares que permitan el acercamiento a los libros.

La lectura es una herramienta fundamental del desarrollo de la personalidad, pero también lo es de socialización como elemento esencial para convivir en democracia y desenvolverse en la sociedad de la información

Por todo ello, durante el curso se realizarán distintas actividades destinadas al fomento de la lectura:

1. Se trabajarán artículos de revistas y periódicos impresos y/o online relacionados con cada una de las unidades de trabajo de este módulo.
2. Se potenciará el uso de la biblioteca del centro para la búsqueda de información.
3. Se propondrán lecturas no obligatorias de libros y revistas relacionadas con los intereses de los alumnos.

Así mismo, en el **tercer trimestre**, se potenciará el que los alumnos hagan exposiciones orales, enlazándolo así con lo propuesto en el proyecto de FP Dual donde, se indica, que los alumnos, una

vez estén haciendo su formación en alternancia en la empresa, asistirán cada dos semanas al centro donde expondrán, mediante el uso de presentaciones, las tareas realizadas en la empresa al resto de sus compañeros.

El desarrollo de este plan de lectura se plasma en trimestres de la siguiente forma:

1. Comprensión lectora (primer trimestre): Leeremos artículos de actualidad y los comentaremos en clase, valorando a continuación la comprensión que de los mismos de la hecho.
2. Expresión escrita (segundo trimestre): Después de leer algunas noticias de actualidad se realizará por escrito un comentario sobre las mismas y se valorará la capacidad de expresión mostrada.
3. Expresión oral (tercer trimestre): Después de leer algún texto histórico sobre la evolución de las redes locales se pedirá al alumnado una valoración y opinión oral sobre lo leído.

## **13 Evaluación de la Programación didáctica**

A lo largo del curso se irá evaluando el proceso educativo redactado en esta programación didáctica y poder modificar, si es el caso, alguno de los apartados más relevantes.