



## PROGRAMACIÓN DIDÁCTICA

# 0485. PROGRAMACIÓN

### DESARROLLO APLICACIONES WEB



## ÍNDICE

ÍNDICE.....	2
BASE LEGAL.....	3
MÓDULO PROGRAMACIÓN.....	4
LAS UNIDADES DE TRABAJO.....	5
ORIENTACIONES PEDAGÓGICAS.....	6
Módulo Programación y su integración en el título “Desarrollo de Aplicaciones Web” .....	6
UNIDADES DE TRABAJO.....	7
Unidad 1. Entornos integrados de desarrollo.....	7
Unidad 2. Desarrollo de clases I.....	8
Unidad 3. Elementos de un programa informático. Lenguaje JAVA y BASIC.....	9
Unidad 4. Desarrollo de clases II.....	10
Unidad 5. Estructuras de control I.....	11
Unidad 6. Utilización de objetos I.....	12
Unidad 7. Documentación.....	13
Unidad 8. Desarrollo de clases III.....	14
Unidad 9. E/S de información I.....	15
Unidad 10. GUI y Eventos.....	16
Unidad 11. Estructuras de control II.....	17
Unidad 12. Utilización de objetos II.....	18
Unidad 13. E/S de información II.....	19
Unidad 14. Estructuras de almacenamiento en memoria I.....	20
Unidad 15. Estructuras de almacenamiento en memoria II.....	21
Unidad 16. Mantenimiento de la persistencia de los objetos.....	22
Unidad 17. Gestión de bases de datos relacionales.....	23
TEMPORALIZACIÓN GENERAL DEL MÓDULO 2021/2022.....	24
EVALUACIÓN Y CALIFICACIÓN.....	25
ATENCIÓN A LA DIVERSIDAD.....	29
RECURSOS Y MATERIALES.....	30
ACTIVIDADES COMPLEMENTARIAS-EXTRAESCOLARES.....	31
SEGUIMIENTO Y AUTOEVALUACIÓN DEL PROFESORADO.....	32
MATERIAL BIBLIOGRÁFICO.....	33
ANEXO.....	34



## **BASE LEGAL**

- Ley Orgánica 8/2013, de 9 de diciembre, para la mejora de la calidad educativa. LOMCE. BOE Núm. 295, Martes 10 de diciembre de 2013.
- 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.
- Real Decreto 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
- Orden de 29 de septiembre de 2010, por la que se regula la evaluación, certificación, acreditación y titulación académica del alumnado que cursa enseñanzas de formación profesional inicial que forma parte del sistema educativo en la Comunidad Autónoma de Andalucía.
- Decreto 327/2010, de 13 de julio, por el que se aprueba el Reglamento Orgánico de los Institutos de Educación Secundaria.



## **MÓDULO PROGRAMACIÓN.**

Este módulo profesional contiene parte de la formación necesaria para desempeñar la función de programación de aplicaciones de propósito general en lenguajes orientados a objetos.

La función de programación de aplicaciones de propósito general en lenguajes orientados a objetos incluye aspectos como:

- El desarrollo de programas organizados en clases aplicando los principios de la programación orientada a objetos.
- La utilización de interfaces para la interacción de la aplicación con el usuario.
- La identificación, análisis e integración de librerías para incorporar funcionalidades específicas a los programas desarrollados.
- El almacenamiento y recuperación de información en sistemas gestores de bases de datos relacionales y orientados a objetos.

Las actividades profesionales asociadas a esta función se aplican en el desarrollo y la adaptación de programas informáticos de propósito general en lenguajes orientados a objetos.



## LAS UNIDADES DE TRABAJO

El presente documento, pretende detallar el currículo del módulo profesional, para facilitar el proceso de enseñanza y aprendizaje y maximizar los resultados del alumnado. Para cada unidad de trabajo, se detallarán:

- a) **Resultados de Aprendizaje:** habilidades que el alumno debe ir adquiriendo para ir desarrollando las competencias necesarias para adquirir el título.
- b) **Competencias:** capacidades profesionales, personales y actitudinales para aplicar de forma integrada los contenidos propios del módulo profesional en la actividad profesional.
- c) **Contenidos:** conjunto de conocimientos, habilidades, destrezas y actitudes que contribuyen al logro de los resultados de aprendizaje y a la adquisición de competencias.
- d) **Criterios de evaluación:** son el referente específico para evaluar el aprendizaje del alumnado. Describen aquello que se quiere valorar y que el alumnado debe lograr, tanto en conocimientos como en competencias; responden a lo que se pretende conseguir en cada módulo profesional.
- e) **Metodología didáctica:** conjunto de estrategias, procedimientos y acciones organizadas y planificadas por el profesorado, de manera consciente y reflexiva, con la finalidad de posibilitar el aprendizaje del alumnado y el logro de los objetivos planteados.
- f) **Temporalización:** nos permite ubicar en el tiempo la unidad de trabajo y conocer de antemano el número de sesiones dedicado a la unidad de trabajo.



## ORIENTACIONES PEDAGÓGICAS

### Módulo Programación y su integración en el título “Desarrollo de Aplicaciones Web”.

Tal y cómo se detalla en la ORDEN de 16 de junio de 2011, este módulo profesional contiene parte de la formación necesaria para desempeñar la función de programación de aplicaciones de propósito general en lenguajes orientados a objetos.

La función de programación de aplicaciones de propósito general en lenguajes orientados a objetos incluye aspectos como:

- El desarrollo de programas organizados en clases aplicando los principios de la programación orientada a objetos.
- La utilización de interfaces para la interacción de la aplicación con el usuario.
- La identificación, análisis e integración de librerías para incorporar funcionalidades específicas a los programas desarrollados.
- El almacenamiento y recuperación de información en sistemas gestores de bases de datos relacionales y orientados a objetos.

Las actividades profesionales asociadas a esta función se aplican en el desarrollo y la adaptación de programas informáticos de propósito general en lenguajes orientados a objetos.

La formación del módulo contribuye a alcanzar los objetivos generales de este ciclo formativo que se relacionan a continuación:

- e) Interpretar el diseño lógico, verificando los parámetros establecidos para gestionar bases de datos.
- j) Emplear herramientas y lenguajes específicos, siguiendo las especificaciones, para desarrollar componentes multimedia.
- q) Programar y realizar actividades para gestionar el mantenimiento de los recursos informáticos.

La formación del módulo contribuye a alcanzar las competencias profesionales, personales y sociales de este título 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.
- 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.
- 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.
- 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.

Las líneas de actuación en el proceso de enseñanza-aprendizaje que permiten alcanzar los objetivos del módulo versarán sobre:

- La interpretación y aplicación de los principios de la programación orientada a objetos.
- La evaluación, selección y utilización de herramientas y lenguajes de programación orientados a objetos
- La utilización de las características específicas de lenguajes y entornos de programación en el desarrollo de aplicaciones informáticas.
- La identificación de las funcionalidades aportadas por los sistemas gestores de bases de datos y su incorporación a los programas desarrollados.
- La documentación de los programas desarrollados.



## UNIDADES DE TRABAJO

### Unidad 1. Entornos integrados de desarrollo.

#### RESULTADOS DE APRENDIZAJE

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

#### CRITERIOS DE EVALUACIÓN

- 1.a) Se han identificado los bloques que componen la estructura de un programa informático.
- 1.b) Se han creado proyectos de desarrollo de aplicaciones.
- 1.c) Se han utilizado entornos integrados de desarrollo.

#### CONTENIDOS

##### Contenidos de la unidad

- 1. Lenguajes de Programación. Tipos y aplicaciones.
- 2. Entornos integrados de desarrollo.
  - a. Definición y tipos. Entornos comerciales y de Software libre.
  - b. Instalación y descripción de entornos integrados de desarrollo.
- 3. Creación de proyectos.
  - a. Estructura y componentes.
  - b. Ejecución de proyectos en distintos IDE's y Lenguajes de Programación.
  - c. Trabajo en equipos de desarrollo. Perfiles
- 4. Entornos de desarrollo para programación orientada a objetos.
  - a. Entornos específicos. Android Studio. Netbeans. Eclipse. Gambas, etc.
  - b. Plugins de integración en entornos genéricos.

#### COMPETENCIAS TRABAJADAS AL FINALIZAR LA UNIDAD

a) Configurar y explotar sistemas informáticos, adaptando la configuración lógica del sistema según las necesidades de uso y los criterios establecidos.

#### TEMPORALIZACIÓN

4h

#### METODOLOGÍA

3h. Presentación de la UT. Explicación con apoyo de proyector del contenido teórico y presentación de la instalación de los distintos IDE's que se usarán en el módulo.

1h. Prueba evaluación: Creación de un proyecto sencillo con alguno de los IDE utilizados en la UT.



## Unidad 2. Desarrollo de clases I.

### RESULTADOS DE APRENDIZAJE

2. Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.
4. Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.

### CRITERIOS DE EVALUACIÓN

- 2.a) Se han identificado los fundamentos de la programación orientada a objetos.
- 2.b) Se han escrito programas simples.
- 4.a) Se ha reconocido la sintaxis, estructura y componentes típicos de una clase.
- 4.b) Se han definido clases.
- 4.c) Se han definido propiedades y métodos.

### CONTENIDOS

#### Contenidos de la unidad

1. Concepto de clase y objeto.
2. Estructura y miembros de una clase. Atributos de objetos. Métodos de objetos.

### COMPETENCIAS TRABAJADAS AL FINALIZAR LA UNIDAD

- 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.
- 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.

### TEMPORALIZACIÓN

10h

### METODOLOGÍA

- 2h. Explicación teórico-práctica de la POO: concepto objeto y clase.
- 2h. Proyecto 1: Definición de las clases necesarias para la aplicación SNAKE. Puesta en común y soluciones individuales.
- 2h Proyecto 2: Definición de las clases necesarias para la aplicación SPACE INVADERS. Puesta en común y soluciones individuales.
- 2h. Proyecto 3: Definición de las clases necesarias para la aplicación SPORTY. Puesta en común y soluciones individuales.
- 2h. Prueba evaluación: Creación y programación de la estructura de las clases utilizadas en los distintos proyectos planteados.





## Unidad 3. Elementos de un programa informático. Lenguaje JAVA y BASIC

### RESULTADOS DE APRENDIZAJE

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

### CRITERIOS DE EVALUACIÓN

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

### CONTENIDOS

#### Contenidos de la unidad

1. Estructura y bloques fundamentales.
  - a) Los paquetes en Java. Una clase sencilla.
  - b) Las clases en BASIC. Una clase sencilla.
  - c) Secuencia de ejecución
  - d) Agrupación de sentencias
2. Variables.
3. Tipos de datos.
4. Literales.
5. Constantes.
6. Operadores y expresiones.
7. Conversiones de tipo.
8. Comentarios.

### COMPETENCIAS TRABAJADAS AL FINALIZAR LA UNIDAD

- 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.
- 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.

### TEMPORALIZACIÓN

12h

### METODOLOGÍA

- 3h. Explicación de la estructura fundamental de una aplicación sencilla en Java. Explicación de la estructura fundamental de una aplicación sencilla en Gambas3.
- 5h. Explicación en clase de las distintas estructuras de control del flujo de ejecución de un lenguaje de programación. Ejemplos en Java y Basic
- 3h. Realización de ejercicios sobre tipos de datos, sus operadores y sus conversiones.
- 1h. Prueba evaluación: creación de una calculadora gráfica en BASIC y de consola en Java.



## Unidad 4. Desarrollo de clases II.

### RESULTADOS DE APRENDIZAJE

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

### CRITERIOS DE EVALUACIÓN

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

### CONTENIDOS

#### Contenidos de la unidad

- 1. Creación de atributos.
- 2. Creación de métodos.
  - a) Métodos
  - b) Funciones
- 3. Creación de constructores.
- 4. Utilización de clases y objetos. Java y BASIC.

### COMPETENCIAS TRABAJADAS AL FINALIZAR LA UNIDAD

- 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.
- 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.

### TEMPORALIZACIÓN

12h

### METODOLOGÍA

- 11h. Realización de todas las clases del proyecto Snake en clase. Versión BASIC
- 1h. Prueba escrita. Modificaciones de clases en BASIC



## Unidad 5. Estructuras de control I.

### RESULTADOS DE APRENDIZAJE

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

### CRITERIOS DE EVALUACIÓN

3.a) Se ha escrito y probado código que haga uso de estructuras de selección.

3.b) Se han utilizado estructuras de repetición.

3.c) Se han reconocido las posibilidades de las sentencias de salto.

3.e) Se han creado programas ejecutables utilizando diferentes estructuras de control.

3.f) Se han probado y depurado los programas.

3.g) Se ha comentado y documentado el código.

### CONTENIDOS

#### Contenidos de la unidad

- a) Estructuras de selección.
  - a) If
  - b) if-else. Anidamiento.
  - c) Switch-Select-Case-Brake.
- b) Estructuras de repetición.
  - a) For
  - b) While
  - c) Do-While
- c) Estructuras de salto. Inconvenientes.

### COMPETENCIAS TRABAJADAS AL FINALIZAR LA UNIDAD

- 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.
- 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.

### TEMPORALIZACIÓN

11h

### METODOLOGÍA

6h. Explicación teórico-práctica de la Unidad de trabajo.

4h. Realización de ejercicios de refactoring de estructuras de control IF, FOR y WHILE de las clases programadas para el proyecto Snake.

1h. Prueba evaluación. Programación de los métodos de una clase sencilla haciendo uso de estructuras de control variadas para el proyecto Snake



## Unidad 6. Utilización de objetos I.

### RESULTADOS DE APRENDIZAJE

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

### CRITERIOS DE EVALUACIÓN

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

### CONTENIDOS

#### Contenidos de la unidad

- 1. Características de los objetos. Creación, uso, destrucción.
- 2. Instanciación de objetos.
- 3. Utilización de métodos.
- 4. Utilización de propiedades.
- 5. Primeras pruebas. Logging y debugging.

### COMPETENCIAS TRABAJADAS AL FINALIZAR LA UNIDAD

- 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.
- 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.

### TEMPORALIZACIÓN

25h

### METODOLOGÍA

- 3h. Explicación teórico-práctica de la Unidad de Trabajo.
- 18h. Cambio de toda la aplicación Snake de BASIC a Java usando Libgdx como Framework
- 2h. Mejora y uso de clases de la API proporcionada por distintas librerías de Gambas para incluir mejoras en el proyecto Snake (aumento de las funcionalidades de las distintas clases programando nuevos métodos).
- 1h. Prueba evaluación A. Clase sencilla en Java
- 1h. Prueba evaluación B. Clase “contenedor” en Java.



## Unidad 7. Documentación.

### RESULTADOS DE APRENDIZAJE

1. Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.
3. Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.

### CRITERIOS DE EVALUACIÓN

- 1.i) Se han introducido comentarios en el código.
- 3.h) Se ha comentado y documentado el código.

### CONTENIDOS

#### Contenidos de la unidad

1. Documentación de programas.
  - a) Documentación interna, comentarios.
  - b) Documentación externa, diagramas de clases, requisitos, guías, etc.

### COMPETENCIAS TRABAJADAS AL FINALIZAR LA UNIDAD

- 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.
- 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.

### TEMPORALIZACIÓN

4h

### METODOLOGÍA

- 1h. Explicación. Técnicas de documentación: comentarios en el código. Documentación externa.
- 1h. Autorevisión de los proyectos anteriores.
- 1h. Revisión de los proyectos de un compañero/a.
- 1h. Prueba evaluación. Incluir documentación interna y externa en un proyecto existente previo.



## Unidad 8. Desarrollo de clases III.

### RESULTADOS DE APRENDIZAJE

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

### CRITERIOS DE EVALUACIÓN

- 4.g) Se han definido y utilizado clases heredadas.
- 7.a) Se han identificado los conceptos de herencia, superclase y subclase.
- 7.b) Se han utilizado modificadores para bloquear y forzar la herencia de clases y métodos.
- 7.c) Se ha reconocido la incidencia de los constructores en la herencia.
- 7.d) Se han creado clases heredadas que sobrescriban la implementación de métodos de la superclase.
- 7.e) Se han diseñado y aplicado jerarquías de clases.
- 7.f) Se han probado y depurado las jerarquías de clases.
- 7.g) Se han realizado programas que implementen y utilicen jerarquías de clases.
- 7.h) Se ha comentado y documentado el código.

### CONTENIDOS

#### Contenidos de la unidad

- 1. Concepto de herencia. Tipos. Utilización de clases heredadas.
  - a) Composición de clases.
  - b) Herencia.
  - c) Superclases y subclases.
  - d) Clases y métodos abstractos y finales.
  - e) Sobreescritura de métodos.
  - f) Constructores y herencia.
- 2. Librerías de clases. Creación. Inclusión y uso de interfaces de clases.

### COMPETENCIAS TRABAJADAS AL FINALIZAR LA UNIDAD

- 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.
- 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.

### TEMPORALIZACIÓN

26h

### METODOLOGÍA

- 4h. Explicación teórica de la unidad de trabajo
- 20h. Implementación del proyecto Space Invaders usando herencia y LIBGDX, Android Studio y tres plataformas destino (Web, Android y Desktop)
- 2h. Prueba evaluación. Desarrollar clases mediante composición, herencia, sobreescritura, implementación de interfaces (nuevos y existentes).



## Unidad 9. E/S de información I.

### RESULTADOS DE APRENDIZAJE

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

### CRITERIOS DE EVALUACIÓN

5.a) Se ha utilizado la consola para realizar operaciones de entrada y salida de información.

5.b) Se han aplicado formatos en la visualización de la información.

5.c) Se han reconocido las posibilidades de entrada/salida del lenguaje y las librerías asociadas.

### CONTENIDOS

#### Contenidos de la unidad

1. Tipos de flujos. Flujos de bytes y de caracteres.
2. Clases relativas a flujos. Streams en JAVA. Streams en BASIC
3. Utilización de flujos.
4. Entrada desde teclado.
5. Salida a pantalla.
6. Entrada desde la red. Salida a la red.

### COMPETENCIAS TRABAJADAS AL FINALIZAR LA UNIDAD

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.

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.

### TEMPORALIZACIÓN

14h

### METODOLOGÍA

6h. Explicación de los contenidos de la unidad en clase.

4h. Implementación de varios ejemplos de lectura/escritura sencilla en los flujos de sistema de E/S por defecto

3h. Implementación de una conexión sencilla TCP/IP. Aplicación de mensajes.

1h. Prueba evaluación: añadir funcionalidad de control de E/S en alguna de las aplicaciones realizadas.



## Unidad 10. GUI y Eventos.

### RESULTADOS DE APRENDIZAJE

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.

### CRITERIOS DE EVALUACIÓN

- 4.g) Se han definido y utilizado clases heredadas.
- 4.h) Se han creado y utilizado métodos estáticos.
- 4.i) Se han definido y utilizado interfaces.
- 4.j) Se han creado y utilizado conjuntos y librerías de clases.
- 5.f) Se han utilizado las herramientas del entorno de desarrollo para crear interfaces gráficos de usuario simples.
- 5.g) Se han programado controladores de eventos.
- 5.h) Se han escrito programas que utilicen interfaces gráficos para la entrada y salida de información.

### CONTENIDOS

#### Contenidos de la unidad

1. Interfaces.
2. Concepto de evento.
3. Creación de controladores de eventos.

### COMPETENCIAS TRABAJADAS AL FINALIZAR LA UNIDAD

- 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.
- 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.

### TEMPORALIZACIÓN

14h

### METODOLOGÍA

- 6h. Explicación teórica de los contenidos.
- 2h. Revisión de los proyectos realizados hasta ahora. GUI y clases relacionadas en JAVA y BASIC.
- 5h. Implementación de aplicaciones en Android Studio para Android. Creación del GUI.
- 1h. Prueba evaluación. Añadir funcionalidad a alguna de las aplicaciones realizadas previamente focalizando el trabajo en el interfaz gráfico con el usuario





## Unidad 11. Estructuras de control II.

### RESULTADOS DE APRENDIZAJE

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

### CRITERIOS DE EVALUACIÓN

3.d) Se ha escrito código utilizando control de excepciones.

3.e) Se han creado programas ejecutables utilizando diferentes estructuras de control.

3.g) Se han probado y depurado los programas.

3.h) Se ha comentado y documentado el código.

### CONTENIDOS

#### Contenidos de la unidad

1. Control de excepciones.
2. Depuración de programas.
3. El depurador como herramienta de control de errores.

### COMPETENCIAS TRABAJADAS AL FINALIZAR LA UNIDAD

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.

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.

### TEMPORALIZACIÓN

14h

### METODOLOGÍA

6h. Explicación de los contenidos de la unidad.

7h. Revisión de los proyectos realizados para realizar control de excepciones y mejora de los errores dependientes del contexto de ejecución.

1h. Prueba evaluación: Añadido de control de excepciones: `nullpointerexception`, `ioexception`, `exception` genérica.



## Unidad 12. Utilización de objetos II.

### RESULTADOS DE APRENDIZAJE

2. Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.
4. Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.

### CRITERIOS DE EVALUACIÓN

- 2.c) Se han instanciado objetos a partir de clases predefinidas.
- 2.d) Se han utilizado métodos y propiedades de los objetos.
- 2.e) Se han escrito llamadas a métodos estáticos.
- 2.f) Se han utilizado parámetros en la llamada a métodos.
- 2.g) Se han incorporado y utilizado librerías de objetos.
- 4.d) Se han creado constructores.
- 4.h) Se han creado y utilizado métodos estáticos.
- 4.j) Se han creado y utilizado conjuntos y librerías de clases.

### CONTENIDOS

#### Contenidos de la unidad

1. Utilización de métodos estáticos.
2. Librerías de objetos. Inclusión y uso.
3. Constructores.
4. Destrucción de objetos y liberación de memoria.

### COMPETENCIAS TRABAJADAS AL FINALIZAR LA UNIDAD

- 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.
- 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.

### TEMPORALIZACIÓN

14h

### METODOLOGÍA

- 7h. Explicación de la teoría de la unidad.
- 6h. Revisión de las librerías de objetos utilizadas hasta ahora: LibGdx, OkHttp, Swing...
- 1h. Prueba evaluación: ampliación de la funcionalidad de algún proyecto previo usando una librería conocida.



## Unidad 13. E/S de información II.

### RESULTADOS DE APRENDIZAJE

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

### CRITERIOS DE EVALUACIÓN

5.d) Se han utilizado ficheros para almacenar y recuperar información.

5.e) Se han creado programas que utilicen diversos métodos de acceso al contenido de los ficheros.

### CONTENIDOS

#### Contenidos de la unidad

1. Ficheros de datos. Registros.
2. Apertura y cierre de ficheros. Modos de acceso.
3. Escritura y lectura de información en ficheros.
4. Utilización de los sistemas de ficheros.
5. Creación y eliminación de ficheros y directorios.

### COMPETENCIAS TRABAJADAS AL FINALIZAR LA UNIDAD

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.

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.

### TEMPORALIZACIÓN

20h

### METODOLOGÍA

4h. Explicación de la teoría de la unidad de trabajo.

15h. Adaptación de nuestras apps para uso de persistencia basada en ficheros.

1h. Prueba evaluación: Añadido de persistencia basada en ficheros mediante formato texto en CSV y mediante serialización binaria Java.



## Unidad 14. Estructuras de almacenamiento en memoria I.

### RESULTADOS DE APRENDIZAJE

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

### CRITERIOS DE EVALUACIÓN

- 6.a) Se han escrito programas que utilicen arrays
- 6.b) Se han reconocido las librerías de clases relacionadas con tipos de datos avanzados.
- 6.c) Se han utilizado listas para almacenar y procesar información.
- 6.d) Se han utilizado iteradores para recorrer los elementos de las listas.
- 6.e) Se han reconocido las características y ventajas de cada una de la colecciones de datos disponibles.
- 6.f) Se han creado clases y métodos genéricos.

### CONTENIDOS

#### Contenidos de la unidad

- 1. Estructuras. Definición y uso.
- 2. Concepto de Array. Tipos. Creación de arrays. Recorrido y búsquedas en un array.
- 3. Arrays multidimensionales.
- 4. Cadenas de caracteres. Uso de las cadenas. Recorrido y manipulación. Uso de expresiones regulares en cadenas de texto.
- 5. Concepto de Lista. Tipos. Operaciones.

### COMPETENCIAS TRABAJADAS AL FINALIZAR LA UNIDAD

- 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.
- 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.

### TEMPORALIZACIÓN

20h

### METODOLOGÍA

- 4h. Explicación teórica de la unidad de trabajo.
- 15. Implementación de la App Genoma. Implementación del juego de la vida. Implementación del simulador "Recursos y países". En estas aplicaciones se implementarán los algoritmos de la unidad.
- 1h. Prueba evaluación: Realización de alguna modificación a las clases vistas en la UT.



## Unidad 15. Estructuras de almacenamiento en memoria II.

### RESULTADOS DE APRENDIZAJE

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

### CRITERIOS DE EVALUACIÓN

6.g) Se han utilizado expresiones regulares en la búsqueda de patrones en cadenas de texto.

6.h) Se han identificado las clases relacionadas con el tratamiento de documentos XML.

6.i) Se han realizado programas que realicen manipulaciones sobre documentos XML.

### CONTENIDOS

#### Contenidos de la unidad

1. Aplicación del estándar XML.
2. Concepto de XML Estructura de un documento XML.
3. Especificación de documentos. DTD y XSD.
4. Clases para la creación y manipulación de documentos XML.
5. Otros formatos de texto para transferencia de información: JSON.

### COMPETENCIAS TRABAJADAS AL FINALIZAR LA UNIDAD

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.

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.

### TEMPORALIZACIÓN

18h

### METODOLOGÍA

4h. Explicación teoría de la unidad de trabajo

13h. Programación de una app cliente de servicios opendata en XML y en JSON en la nube.

1h. Prueba evaluación.



## Unidad 16. Mantenimiento de la persistencia de los objetos.

### RESULTADOS DE APRENDIZAJE

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

### CRITERIOS DE EVALUACIÓN

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

### CONTENIDOS

#### Contenidos de la unidad

- 1. Bases de datos orientadas a objetos.
- 2. Características de las bases de datos orientadas a objetos.
- 3. Instalación del gestor de bases de datos.
- 4. Creación de bases de datos.
- 5. Mecanismos de consulta.
- 6. El lenguaje de consultas, sintaxis, expresiones, operadores.
- 7. Recuperación, modificación y borrado de información.
- 8. Tipos de datos objeto; atributos y métodos.
- 9. Tipos de datos colección.

### COMPETENCIAS TRABAJADAS AL FINALIZAR LA UNIDAD

- 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.
- 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.

### TEMPORALIZACIÓN

15h

### METODOLOGÍA

- 4h. Explicación teoría de la unidad de trabajo
- 10h. Adaptación de persistencia con BBDD Orientadas a Objetos de alguna App del curso.
- 1h. Prueba evaluación.



## Unidad 17. Gestión de bases de datos relacionales.

### RESULTADOS DE APRENDIZAJE

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

### CRITERIOS DE EVALUACIÓN

9.a) Se han identificado las características y métodos de acceso a sistemas gestores de bases de datos relacionales.

9.b) Se han programado conexiones con bases de datos.

9.c) Se ha escrito código para almacenar información en bases de datos.

9.d) Se han creado programas para recuperar y mostrar información almacenada en bases de datos.

9.e) Se han efectuado borrados y modificaciones sobre la información almacenada.

9.f) Se han creado aplicaciones que ejecuten consultas sobre bases de datos.

9.g) Se han creado aplicaciones para posibilitar la gestión de información presente en bases de datos relacionales.

### CONTENIDOS

#### Contenidos de la unidad

1. Conexión con bases de datos relacionales. Características, tipos y métodos de acceso.
2. Establecimiento de conexiones. Componentes de acceso a datos.
3. Recuperación de información. Selección de registros. Uso de parámetros.
4. Manipulación de la información. Altas, bajas y modificaciones.
5. Ejecución de consultas sobre la base de datos.

### COMPETENCIAS TRABAJADAS AL FINALIZAR LA UNIDAD

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.

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.

### TEMPORALIZACIÓN

15h

### METODOLOGÍA

4h. Explicación de los contenidos de la unidad de trabajo.

10h. Adaptación de persistencia de alguna APP del curso con BBDD Relacionales. Introducción a los Servlets y acceso a datos desde lado servidor.

1h. Prueba evaluación.



## TEMPORALIZACIÓN GENERAL DEL MÓDULO 2021/2022

**- Primer trimestre:** Total 104

UT1. Tiempo. 4 horas.

UT2. Tiempo. 10 horas.

UT3. Tiempo. 12 horas.

UT4. Tiempo. 12 horas.

UT5. Tiempo. 11 horas.

UT6. Tiempo. 25 horas.

UT7. Tiempo. 4 horas.

UT8. Tiempo. 26 horas.

**- Segundo trimestre:** Total 96 horas

UT9. Tiempo. 14 horas.

UT10. Tiempo. 14 horas.

UT11. Tiempo. 14 horas.

UT12. Tiempo. 14 horas.

UT13. Tiempo. 20 horas.

UT14. Tiempo. 20 horas.

**- Tercer trimestre:** Total 48 horas

UT15. Tiempo. 18 horas.

UT16. Tiempo. 15 horas.

UT17. Tiempo. 15 horas.

Total horas del curso: 248 horas. 8 horas hasta las 256 totales del módulo dedicadas a actividades extraescolares (salidas del centro, charlas, actividades del centro).





## EVALUACIÓN Y CALIFICACIÓN

Los procedimientos mediante los cuales voy a obtener información sobre el aprendizaje y el trabajo de cada alumno serán los siguientes:

### a) Actividades de clase evaluables:

Para mi asignatura, el objetivo principal de las actividades de clase es la asimilación y consolidación de los contenidos conceptuales y procedimentales.

Semanalmente, comprobaré si cada alumno/a me ha enviado todas las actividades propuestas. Algunas serán realizadas en grupo, aunque se entregarán los resultados de la actividad individualmente.

Observaré con detalle cómo ha realizado las actividades, atendiendo a estos criterios:

- Hábito y Método de trabajo.
- Presentación correcta, calidad de acabado y estética de la práctica
- Realización correcta de la actividad, conforme a los objetivos que se le pedían
- Aportación de ideas creativas y soluciones nuevas o alternativas.

Estos trabajos se deberán realizar en clase (o en casa si no les da tiempo a hacerlo en clase), tanto en pequeños grupos como individualmente, dependiendo de la actividad.

Tendrán una fecha limitada para su entrega, que generalmente será en los últimos días de trabajo de cada unidad de trabajo. Los trabajos serán evaluados en una escala de 0 a 10.

### b) Exámenes:

A lo largo del curso se realizarán una serie de pruebas evaluativas, tal y como se expresa en las unidades de trabajo. Cada alumno realizará cada examen de forma individual. Cada examen se valorará de 0 a 10

### Obtención de las notas trimestrales y finales.

Para cada periodo de evaluación (e.d. cada trimestre o para la nota final) se calculará la nota de ese periodo con las calificaciones asignadas a cada Resultado de Aprendizaje (RA en adelante) evaluado hasta el momento (no se tendrán en cuenta aquellos no evaluados en el momento del cálculo de la nota), integrando en la nota progresivamente los distintos RA que se hayan ido evaluando de manera continua y actualizando la nota de cada RA evaluado con las nuevas calificaciones obtenidas progresivamente a lo largo del curso.

Todos los RA evaluados tendrán el mismo peso a la hora del cálculo de la calificación en un periodo determinado, y cada criterio de evaluación también tendrá el mismo peso que los demás criterios que determinan el valor de un determinado RA.

La nota de cada RA se obtendrá del cálculo de las calificaciones de cada criterio de evaluación, que a su vez se medirán en las distintas UT con los instrumentos de evaluación detallados en la siguiente tabla:

Resultado de Aprendizaje	Criterio de Evaluación	Instrumentos/Procedimientos de evaluación
RA1	a) Se han identificado los bloques que componen la estructura de un programa informático.	UT1. Prueba UT3. Prueba UT3. Ejercicios
	b) Se han creado proyectos de desarrollo de aplicaciones.	UT1. Prueba UT3. Ejercicios
	c) Se han utilizado entornos integrados de desarrollo.	UT1. Prueba UT3. Prueba UT3. Ejercicios
	d) Se han identificado los distintos tipos de variables y la utilidad específica de cada uno.	UT3. Prueba UT3. Ejercicios
	e) Se ha modificado el código de un programa para crear y utilizar variables.	UT3. Prueba UT3. Ejercicios
	f) Se han creado y utilizado constantes y literales.	UT3. Prueba UT3. Ejercicios
	g) Se han clasificado, reconocido y utilizado en expresiones los operadores del lenguaje.	UT3. Prueba UT3. Ejercicios
	h) Se ha comprobado el funcionamiento de las conversiones de tipo explícitas e implícitas.	UT3. Prueba UT3. Ejercicios
	i) Se han introducido comentarios en el código.	UT3. Prueba UT3. Ejercicios UT7. Prueba UT7. Ejercicios de revisión



RA2	a) Se han identificado los fundamentos de la programación orientada a objetos.	UT2. Prueba UT6. Prueba A UT6. Ejercicios
	b) Se han escrito programas simples.	UT2. Prueba UT6. Prueba A UT6. Ejercicios
	c) Se han instanciado objetos a partir de clases predefinidas.	UT6. Prueba A UT6. Prueba B UT6. Ejercicios UT12. Prueba
	d) Se han utilizado métodos y propiedades de los objetos.	UT6. Prueba A UT6. Prueba B UT6. Ejercicios UT12. Prueba
	e) Se han escrito llamadas a métodos estáticos.	UT6. Prueba B UT6. Ejercicios UT12. Prueba
	f) Se han utilizado parámetros en la llamada a métodos.	UT6. Prueba A UT6. Prueba B UT6. Ejercicios UT12. Prueba
	g) Se han incorporado y utilizado librerías de objetos.	UT6. Prueba B UT6. Ejercicios UT12. Prueba
	h) Se han utilizado constructores.	UT6. Prueba A UT6. Prueba B
	i) Se ha utilizado el entorno integrado de desarrollo en la creación y compilación de programas simples.	UT6. Prueba A UT6. Prueba B UT6. Ejercicios
RA3	a) Se ha escrito y probado código que haga uso de estructuras de selección.	UT5. Ejercicios UT5. Prueba
	b) Se han utilizado estructuras de repetición.	UT5. Ejercicios UT5. Prueba
	c) Se han reconocido las posibilidades de las sentencias de salto.	UT5. Ejercicios UT5. Prueba
	d) Se ha escrito código utilizando control de excepciones.	UT11. Ejercicios UT11. Prueba
	e) Se han creado programas ejecutables utilizando diferentes estructuras de control.	UT5. Ejercicios UT5. Prueba
	f) Se han probado y depurado los programas.	UT5. Ejercicios UT5. Prueba UT11. Ejercicios UT11. Prueba
	g) Se ha comentado y documentado el código.	UT5. Ejercicios UT5. Prueba UT7. Prueba UT7. Ejercicios de revisión UT11. Ejercicios UT11. Prueba
RA4	a) Se ha reconocido la sintaxis, estructura y componentes típicos de una clase.	UT2. Prueba UT4. Prueba
	b) Se han definido clases.	UT2. Prueba UT4. Prueba
	c) Se han definido propiedades y métodos.	UT2. Prueba UT4. Prueba
	d) Se han creado constructores.	UT4. Prueba UT12. Prueba
	e) Se han desarrollado programas que instancien y utilicen objetos de las clases creadas anteriormente.	UT4. Prueba
	f) Se han utilizado mecanismos para controlar la visibilidad de las clases y de sus miembros.	UT4. Prueba
	g) Se han definido y utilizado clases heredadas.	UT8. Prueba UT10. Ejercicios UT10. Prueba
	h) Se han creado y utilizado métodos estáticos.	UT10. Ejercicios UT10. Prueba UT12. Prueba
	i) Se han definido y utilizado interfaces.	UT10. Ejercicios UT10. Prueba
	j) Se han creado y utilizado conjuntos y librerías de clases.	UT10. Ejercicios UT10. Prueba UT12. Prueba



RA5	a) Se ha utilizado la consola para realizar operaciones de entrada y salida de información.	UT9. Ejercicios UT9. Prueba
	b) Se han aplicado formatos en la visualización de la información.	UT9. Ejercicios UT9. Prueba
	c) Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.	UT9. Ejercicios UT9. Prueba
	d) Se han utilizado ficheros para almacenar y recuperar información.	UT13. Ejercicios UT13. Prueba
	e) Se han creado programas que utilicen diversos métodos de acceso al contenido de los ficheros.	UT13. Ejercicios UT13. Prueba
	f) Se han utilizado las herramientas del entorno de desarrollo para crear interfaces gráficos de usuario simples.	UT10. Ejercicios UT10. Prueba
	g) Se han programado controladores de eventos.	UT10. Ejercicios UT10. Prueba
	h) Se han escrito programas que utilicen interfaces gráficos para la entrada y salida de información.	UT10. Ejercicios UT10. Prueba
RA6	a) Se han escrito programas que utilicen arrays	UT14. Ejercicios UT14. Prueba
	b) Se han reconocido las librerías de clases relacionadas con tipos de datos avanzados.	UT14. Ejercicios UT14. Prueba
	c) Se han utilizado listas para almacenar y procesar información.	UT14. Ejercicios UT14. Prueba
	d) Se han utilizado iteradores para recorrer los elementos de las listas.	UT14. Ejercicios UT14. Prueba
	e) Se han reconocido las características y ventajas de cada una de la colecciones de datos disponibles.	UT14. Ejercicios UT14. Prueba
	f) Se han creado clases y métodos genéricos.	UT14. Ejercicios UT14. Prueba
	g) Se han utilizado expresiones regulares en la búsqueda de patrones en cadenas de texto.	UT15. Ejercicios UT15. Prueba
	h) Se han identificado las clases relacionadas con el tratamiento de documentos XML.	UT15. Ejercicios UT15. Prueba
	i) Se han realizado programas que realicen manipulaciones sobre documentos XML.	UT15. Ejercicios UT15. Prueba
RA7	a) Se han identificado los conceptos de herencia, superclase y subclase.	UT8. Ejercicios UT8. Prueba
	b) Se han utilizado modificadores para bloquear y forzar la herencia de clases y métodos.	UT8. Ejercicios UT8. Prueba
	c) Se ha reconocido la incidencia de los constructores en la herencia.	UT8. Ejercicios UT8. Prueba
	d) Se han creado clases heredadas que sobrescriban la implementación de métodos de la superclase.	UT8. Ejercicios UT8. Prueba
	e) Se han diseñado y aplicado jerarquías de clases.	UT8. Ejercicios UT8. Prueba
	f) Se han probado y depurado las jerarquías de clases.	UT8. Ejercicios UT8. Prueba
	g) Se han realizado programas que implementen y utilicen jerarquías de clases.	UT8. Ejercicios UT8. Prueba
	h) Se ha comentado y documentado el código.	UT8. Ejercicios UT8. Prueba
RA8	a) Se han identificado las características de las bases de datos orientadas a objetos.	UT16. Ejercicios UT16. Prueba
	b) Se ha analizado su aplicación en el desarrollo de aplicaciones mediante lenguajes orientados a objetos.	UT16. Ejercicios UT16. Prueba
	c) Se han instalado sistemas gestores de bases de datos orientados a objetos.	UT16. Ejercicios UT16. Prueba
	d) Se han clasificado y analizado los distintos métodos soportados por los sistemas gestores para la gestión de la información almacenada.	UT16. Ejercicios UT16. Prueba
	e) Se han creado bases de datos y las estructuras necesarias para el almacenamiento de objetos.	UT16. Ejercicios UT16. Prueba
	f) Se han programado aplicaciones que almacenen objetos en las bases de datos creadas.	UT16. Ejercicios UT16. Prueba
	g) Se han realizado programas para recuperar, actualizar y eliminar objetos de las bases de datos.	UT16. Ejercicios UT16. Prueba
	h) Se han realizado programas para almacenar y gestionar tipos de datos estructurados, compuestos y relacionados.	UT16. Ejercicios UT16. Prueba



RA9	a) Se han identificado las características y métodos de acceso a sistemas gestores de bases de datos relacionales.	UT17. Ejercicios UT17. Prueba
	b) Se han programado conexiones con bases de datos.	UT17. Ejercicios UT17. Prueba
	c) Se ha escrito código para almacenar información en bases de datos.	UT17. Ejercicios UT17. Prueba
	d) Se han creado programas para recuperar y mostrar información almacenada en bases de datos.	UT17. Ejercicios UT17. Prueba
	e) Se han efectuado borrados y modificaciones sobre la información almacenada.	UT17. Ejercicios UT17. Prueba
	f) Se han creado aplicaciones que ejecuten consultas sobre bases de datos.	UT17. Ejercicios UT17. Prueba
	g) Se han creado aplicaciones para posibilitar la gestión de información presente en bases de datos relacionales.	UT17. Ejercicios UT17. Prueba

### Recuperación del módulo

Como se van a evaluar de manera continua y periódica todos los criterios de evaluación en distintas unidades de trabajo, se realizarán, trimestralmente, pruebas de recuperación para aquellos criterios de evaluación (y por ende, resultados de aprendizaje) que no obtengan un mínimo de un 5 en su evaluación.

### Nota para la Evaluación Ordinaria:

Se tendrán en cuenta todas las notas de los distintos RA tal y cómo se describe previamente. Se considerará que el módulo está superado si tenemos una evaluación de un 5 o más en cada uno de los resultados de aprendizaje.

### Nota para la Evaluación Extraordinaria:

Los alumnos que no hayan superado la convocatoria ordinaria, tendrán la posibilidad de superar la asignatura mediante una convocatoria extraordinaria. Para ello tendrán que superar un examen que tendrá relación con los criterios de evaluación contenidos en todas las unidades de trabajo.



## ATENCIÓN A LA DIVERSIDAD

Todo el alumnado recibirá una formación básica común; dando respuesta a los diversos problemas o dificultades de aprendizaje, estilos de aprendizaje, intereses y capacidades que pueden darse; creando las condiciones necesarias para que todos alcancen los objetivos planteados. Además se crearán las condiciones que hagan posible una enseñanza personalizada, que tenga en cuenta las peculiaridades de cada estudiante y se adapte a ellas.

El conjunto de medidas para dar respuesta a esta diversidad, podrían concretarse en las siguientes:

- Adaptación del mobiliario y materiales para que puedan ser usados por todo el alumnado.
- Diseño de actividades de diferente grado de dificultad y niveles de realización válidos para los mismos contenidos. En este sentido se ofrecerán actividades de:
  - o Refuerzo: para los alumnos con problemas de aprendizaje.
  - o Ampliación: para los alumnos más interesados y sin problemas de aprendizaje.
- Favorecer las estrategias que potencian la motivación y ampliar los intereses del alumnado.
- Metodología.
- Agrupamientos flexibles.
- Materiales.

Se tratará de adaptaciones no significativas, ya que suponen ajustes que no modifican sustancialmente la programación propuesta para el grupo de referencia y sirve para dar respuesta a las diferencias individuales o dificultades de aprendizaje transitorias.



## RECURSOS Y MATERIALES

- Documentos, imágenes, videos, para este nivel y asignatura, de la plataforma Google Classroom proporcionada por el profesor.
- Pizarra blanca: en su empleo se cuidará la presentación de los contenidos, utilizando una letra grande y clara, borrando cuando se haya leído o escrito, y coordinando la exposición oral con la escrita.
- Pizarra Digital o proyector para mostrar sobre la pared la pantalla del ordenador del profesor, hacer prácticas en pizarra, etc.



## **ACTIVIDADES COMPLEMENTARIAS-EXTRAESCOLARES**

Véase programación del ciclo formativo.



## **SEGUIMIENTO Y AUTOEVALUACIÓN DEL PROFESORADO.**

Véase programación del ciclo formativo.





## **MATERIAL BIBLIOGRÁFICO.**

- Libro “Gambas. Programación Visual con Software Libre”. Daniel Campos y José Luis Redrejo
- Manual: Introducción a la Programación Orientada a Objetos con Java. Rafael Llobet Azpitarte, Pedro Alonso Jordá, Jaume Devesa Llinares, Emili Miedes De Elías, María Idoia Ruiz Fuertes, Francisco Torres Goterris
- Webgrafía.
- Materiales y proyectos realizados por el profesor



## **ANEXO.**

### **CRITERIOS Y PROCEDIMIENTOS PREVISTOS PARA ORGANIZAR LA ATENCIÓN A LA DIVERSIDAD DE LOS ALUMNOS**

- Véase programación del ciclo formativo.

### **ADAPTACIÓN A LA SITUACIÓN PROVOCADA POR LA PANDEMÍA COVID-19**

- En caso necesario, se aplicarán las medidas establecidas en el Plan de Contingencia aprobado en el centro IES Politécnico Jesús Marín.