**Programación didáctica del módulo: *Programación***

**Ciclo formativo: *DAW***

**Curso: 2023/2024**

**Profesor: *César San Juan Pastor***

**Índice**

1. Criterios de evaluación 3

2. Criterios de calificación 8

3. Recuperación 8

# 

## Criterios de evaluación

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

1. Se han creado proyectos de desarrollo de aplicaciones
2. Se han utilizado entornos integrados de desarrollo.
3. Se han identificado los distintos tipos de variables y la utilidad específica de cada uno.
4. Se ha modificado el código de un programa para crear y utilizar variables.
5. Se han creado y utilizado constantes y literales.
6. Se han clasificado, reconocido y utilizado en expresiones los operadores del lenguaje.
7. Se ha comprobado el funcionamiento de las conversiones de tipo explícitas e implícitas.
8. Se han introducido comentarios en el código.
9. Se han creado y utilizado procedimientos y funciones.
10. Se ha utilizado el paso de parámetros en procedimientos y funciones.
11. Se han identificado los fundamentos de la programación orientada a objetos.
12. Se han escrito programas simples.
13. Se han instanciado objetos a partir de clases predefinidas.
14. Se han utilizado métodos y propiedades de los objetos.
15. Se han escrito llamadas a métodos estáticos.
16. Se han utilizado parámetros en la llamada a métodos.
17. Se han incorporado y utilizado librerías de objetos.
18. Se han utilizado constructores y destructores de objetos.
19. Se ha utilizado el entorno integrado de desarrollo en la creación y compilación de programas simples.
20. Se ha escrito y probado código que haga uso de estructuras de selección.
21. Se han utilizado estructuras de repetición.
22. Se han reconocido las posibilidades de las sentencias de salto.
23. Se ha escrito código utilizando control de excepciones.
24. Se han creado programas ejecutables utilizando diferentes estructuras de control.
25. Se han probado y depurado los programas.
26. Se ha comentado y documentado el código.
27. Se ha reconocido la sintaxis, estructura y componentes típicos de una clase.
28. Se han definido clases.
29. Se han definido propiedades y métodos.
30. Se han creado constructores.
31. Se han desarrollado programas que instancien y utilicen objetos de las clases creadas anteriormente.
32. Se han utilizado mecanismos para controlar la visibilidad de las clases y de sus miembros.
33. Se han definido y utilizado clases heredadas.
34. Se han creado y utilizado métodos estáticos.
35. Se han definido y utilizado interfaces.
36. Se han creado y utilizado conjuntos y librerías de clases.
37. Se ha utilizado la consola para realizar operaciones de entrada y salida de información.
38. Se han aplicado formatos en la visualización de la información.
39. Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.
40. Se han utilizado ficheros para almacenar y recuperar información.
41. Se han creado programas que utilicen diversos métodos de acceso al contenido de los ficheros.
42. Se han utilizado las herramientas del entorno de desarrollo para crear interfaces gráficos de usuario simples.
43. Se han programado controladores de eventos.
44. Se han escrito programas que utilicen interfaces gráficos para la entrada y salida de información.
45. Se han escrito programas que utilicen Arrays.
46. Se han reconocido las librerías de clases relacionadas con tipos de datos avanzados.
47. Se han utilizado listas para almacenar y procesar información.
48. Se han utilizado iteradores para recorrer los elementos de las listas.
49. Se han reconocido las características y ventajas de cada una de las colecciones de datos disponibles.
50. Se han instanciado objetos a partir de clases predefinidas.
51. Se han utilizado métodos y propiedades de los objetos.
52. Se han escrito llamadas a métodos estáticos.
53. Se han utilizado parámetros en la llamada a métodos.
54. Se han incorporado y utilizado librerías de objetos.
55. Se han utilizado constructores y destructores de objetos.
56. Se ha utilizado el entorno integrado de desarrollo en la creación y compilación de programas simples.
57. Se ha escrito y probado código que haga uso de estructuras de selección.
58. Se han utilizado estructuras de repetición.
59. Se han reconocido las posibilidades de las sentencias de salto.
60. Se ha escrito código utilizando control de excepciones.
61. Se han creado programas ejecutables utilizando diferentes estructuras de control.
62. Se han probado y depurado los programas.
63. Se ha comentado y documentado el código.
64. Se ha reconocido la sintaxis, estructura y componentes típicos de una clase.
65. Se han definido clases.
66. Se han definido propiedades y métodos.
67. Se han creado constructores.
68. Se han desarrollado programas que instancien y utilicen objetos de las clases creadas anteriormente.
69. Se han utilizado mecanismos para controlar la visibilidad de las clases y de sus miembros.
70. Se han definido y utilizado clases heredadas.
71. Se han creado y utilizado métodos estáticos.
72. Se han definido y utilizado interfaces.
73. Se han creado y utilizado conjuntos y librerías de clases.
74. Se ha utilizado la consola para realizar operaciones de entrada y salida de información.
75. Se han aplicado formatos en la visualización de la información.
76. Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.
77. Se han utilizado ficheros para almacenar y recuperar información.
78. Se han creado programas que utilicen diversos métodos de acceso al contenido de los ficheros.
79. Se han utilizado las herramientas del entorno de desarrollo para crear interfaces gráficos de usuario simples.
80. Se han programado controladores de eventos.
81. Se han escrito programas que utilicen interfaces gráficos para la entrada y salida de información.
82. Se han escrito programas que utilicen Arrays.
83. Se han reconocido las librerías de clases relacionadas con tipos de datos avanzados.
84. Se han utilizado listas para almacenar y procesar información.
85. Se han utilizado iteradores para recorrer los elementos de las listas.
86. Se han reconocido las características y ventajas de cada una de las colecciones de datos disponibles.
87. Se han creado clases y métodos genéricos.
88. Se han utilizado expresiones regulares en la búsqueda de patrones en cadenas de texto.
89. Se han identificado las clases relacionadas con el tratamiento de documentos XML.
90. Se han realizado programas que realicen manipulaciones sobre documentos XML.
91. Se han identificado los conceptos de herencia, superclase, subclase y polimorfismo.
92. Se han utilizado modificadores para bloquear y forzar la herencia de clases y métodos.
93. Se ha reconocido la incidencia de los constructores en la herencia.
94. Se han creado clases heredadas que sobrescriban la implementación de métodos de la superclase.
95. Se han diseñado y aplicado jerarquías de clases.
96. Se han probado y depurado las jerarquías de clases.
97. Se han realizado programas que implementen y utilicen jerarquías de clases.
98. Se ha comentado y documentado el código.
99. Se han identificado las características de las bases de datos orientadas a objetos.
100. Se ha analizado su aplicación en el desarrollo de aplicaciones mediante lenguajes orientados a objetos.
101. Se han instalado sistemas gestores de bases de datos orientados a objetos.
102. Se han clasificado y analizado los distintos métodos soportados por los sistemas gestores para la gestión de la información almacenada.
103. Se han creado bases de datos y las estructuras necesarias para el almacenamiento de objetos.
104. Se han programado aplicaciones que almacenen objetos en las bases de datos creadas.
105. Se han realizado programas para recuperar, actualizar y eliminar objetos de las bases de datos.
106. Se han realizado programas para almacenar y gestionar tipos de datos estructurados, compuestos y relacionados.
107. Se han identificado las características y métodos de acceso a sistemas gestores de bases de datos relacionales.
108. Se han programado conexiones con bases de datos.
109. Se ha escrito código para almacenar información en bases de datos.
110. Se han creado programas para recuperar y mostrar información almacenada en bases de datos.
111. Se han efectuado borrados y modificaciones sobre la información almacenada.
112. Se han creado aplicaciones que ejecuten consultas sobre bases de datos.
113. Se han creado aplicaciones para posibilitar la gestión de información presente en bases de datos relacionales

## Criterios de calificación

Dado el carácter práctico del módulo se establece una calificación mixta entre los contenidos evaluados en proyectos o prácticas y en los exámenes.

* En cada evaluación se realizará al menos un **examen** de tipo teórico - práctico que corresponderá como mínimo con el **70 % de la calificación de la evaluación.**
  + El contenido se adecuará a los de la programación valorándose, al menos, los criterios mínimos para poder superar dicha prueba.
  + No se excluye la inclusión de preguntas teóricas en esta prueba
* Si en dicha evaluación hubiera **Actividades de enseñanza-aprendizaje** (proyectos, ejercicios, prácticas o trabajos realizados por el alumno), las evaluaciones de éstas se corresponderán como máximo con un **30% de la calificación de la evaluación**.
  + En este aspecto se valorará además del trabajo realizado en la práctica la actitud, la asistencia a clase y el trabajo realizado por el alumno.
  + La evaluación de las pruebas prácticas (ejercicios, prácticas) será siempre individual, y la realización de trabajos grupales no conllevará en ningún momento que todos los miembros deban tener la misma calificación.
  + La evaluación debe ser un proceso continuo, con lo que las notas de las prácticas de cada evaluación se tendrán en cuenta en la siguiente para determinar el grado de consecución de los objetivos, no serán de aplicación las pruebas escritas que de otras evaluaciones en las siguientes al considerarse que los objetivos evaluados en las pruebas escritas estarán también contenidos en la siguiente.
  + No se aceptarán trabajos retrasados fuera de plazo, a no ser que el profesor considere justificado el retraso por fuerza mayor y siempre y cuando éstos no hayan sido puestos en común, revisados o resueltos en clase; considerándose, en ese caso, que se renuncia explícitamente a aportarlos como evidencias para una evaluación positiva, con la consiguiente merma o perjuicio en la calificación resultante.

**Calificación\_Evaluacion =**

**nota\_Examen x 0.7 + media\_Actividades\_Evaluables x 0.3**

En el caso de que en alguna evaluación no se requiera ninguna actividad o trabajo la nota final de evaluación corresponderá con el 100% de la nota del examen.

Para superar cada evaluación es necesario:

* **Haber obtenido al menos un 4 en cada uno de los exámenes escritos.**
* **Haber presentado y aprobado todas las tareas****, incluso de evaluaciones anteriores.**
* **Alcanzar la media de 5 puntos.**
* **No haber perdido el derecho a la evaluación continua.**

## Superación del módulo

Para poder **superar el módulo** es necesario:

1. Que las ausencias a clase no superen las establecidas en las normas de convivencia del instituto.

2. Que la actitud hacia el profesor y los compañeros sea correcta.

3. La nota final del módulo sea al menos un 5.

4. Obtener al menos un 4 en la unidad de trabajo 7, proyecto globalizador.

5. La nota final del curso se calculará:

**Calificación\_Curso\_Final = Calificacion\_Tercera\_EV \*0,7 + media\_Actividades\_Evaluables \* 0,3**

## Recuperación

Se trata de una evaluación continua como se desprende de la metodología empleada y, por tanto, no existirán pruebas de recuperación específicas (a no ser que el profesor lo considere necesario), realizando planes de refuerzo al alumnado en el momento que se detecte que un alumno no pueda conseguir los objetivos establecidos.

Acceso a la segunda convocatoria ordinaria

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

Si un alumno no supera el curso, deberá realizar las tareas propuestas para este periodo y presentarse de nuevo a un examen para la evaluación de la segunda evaluación ordinaria. Para eta segunda evaluación se hará un examen global de todo el curso y se obtendrá un aprobado si:

* Se obtiene como mínimo un 5 en este examen
* Se han presentado y aprobado todas las tareas propuestas por el profesor.
* Se han entregado y aprobado todas las tereas del año.

Sin embargo, aunque se obtenga una nota superior a 5 en el examen de recuperación, para que la calificación numérica efectiva en la convocatoria supere el 5, será necesario que así lo haga la siguiente media:

**Calificación\_Evaluacion = nota\_Examen x 0.7 + media\_Actividades\_Evaluables x 0.3**