# TEMA 1. PRÁCTICA 1. MANEJO DE FICHEROS DE TEXTO

## Objetivos:

* Diseño y la manipulación de objetos
* Lectura y escritura en ficheros de texto
* Detección y modelado de relaciones entre clases
* Aplicación de todos los conocimientos sobre programación aprendidos hasta el momento.
* Uso de modificadores de visibilidad
* Uso de excepciones.
* Uso de patrones de desarrollo.

## Enunciado

Desarrolle un programa que simule parte del funcionamiento de un sistema de gestión de vehículos. El funcionamiento normal de este tipo de sistemas se detalla a continuación. Supongamos que llega un empleado delante del ordenador y lo primero que hace es autenticarse (login y password) en el sistema. Una vez autenticado, el empleado puede realizar las siguientes tareas: añadir vehículos, modificar datos de los vehículos o eliminarlos.

El sistema de gestión de empleados se encargará de cargar desde un fichero todos los empleados de la empresa, cuando comience a funcionar el programa. La función principal de dicho subsistema es la comprobación de que el login y password de cada empleado que entra en el sistema son correctos.

Se tienen dos tipos de vehículos, camiones y turismos. Todos estos datos deben de ser leídos de un fichero de texto. Los turismos se identifican por una matrícula, marca, modelo, color, número de puertas y precio. Los camiones se identifican por la matrícula, marca, modelo, color, capacidad de carga y precio. Los dos tipos de vehículos pueden ser añadidos, modificados o eliminados. A parte los turismos pueden tener un extra.

Los extras se identifican por un identificador y tienen descripción. Los extras, se pueden mostrar todos, buscar, añadir, modificar, eliminar o eliminar todos. Todo turismo debe de tener un extra que exista.

Es necesario controlar todos los tipos de comportamientos no deseados. Dichos comportamientos se pueden controlar mediante el manejo de excepciones.

Para mayor ayuda se indican los mensajes y el flujo del programa.

**Bienvenido al programa de gestión de vehículos**

**Introduzca su usuario y contraseña**

En el caso de error, es necesario indicarlo mediante una excepción y volver a pedir que se introduzca el usuario y contraseña.

**Usuario o contraseña erróneos.**

**Introduzca su usuario y contraseña**

Una vez identificado, se debe indicar el siguiente menú.

**Elija que opción desea realizar.**

1. **Mostrar todos los vehículos.**
2. **Buscar un vehículo.**
3. **Añadir un vehículo.**
4. **Modificar un vehículo.**
5. **Eliminar un vehículo.**
6. **Eliminar todos los vehículos.**
7. **Vehículo más caro.**
8. **Vehículo más barato.**
9. **Mostrar todos los extras**
10. **Buscar un extra**
11. **Añadir un extra**
12. **Modificar un extra**
13. **Eliminar un extra**
14. **Eliminar todos los extras**
15. **Log out**

En el caso de añadir se deben indicar todos los datos y preguntar al usuario si lo que se va a añadir es un camión o turismo, debido a que tienen características distintas.

En el caso de modificar, se puede tener la opción de modificar cualquier tipo de datos de los vehículos.

Cuando se quiere añadir un vehículo, se debe de comprobar que el extra existe, sino se debe de volver a pedir al usuario.

En el caso de modificar un extra a un vehículo, se debe de comprobar que, al introducir el nuevo extra, debe de existir.

En el momento de añadir un vehículo o extra, la matrícula o el identificador no se deben de repetir.

En el momento de modificar de un vehículo la matrícula o de un extra el identificador, la matrícula o el identificador no se deben de repetir.

Por serialización se deben guardar los objetos coche y camión. El resto deberán utilizar la clase Scanner o PrintWritter.

**Nota 1:** el sistema no deja nunca de funcionar, salvo cuando se seleccione la opción de **Log Out**.

**Nota 2:** Es necesario recordar que para leer cadenas a través de un objeto de la clase Scanner mediante el método nextLine() es necesario colocar antes otra llamada nextLine().

**Nota 3:** es necesario controlar todos los errores que se puedan producir durante la ejecución.

## Condiciones de la Entrega:

Se recuerda que los alumnos deberán defender en clase de prácticas el trabajo realizado, siendo necesario que asistan ambos miembros de cada pareja de prácticas, en caso de que uno de ellos no se presente, ese alumno no será evaluado y por tanto suspenso en la convocatoria ordinaria. La nota de cada miembro es diferente en función del trabajo realizado y la defensa que se haga.

Igualmente es necesario recordar que es necesario que la documentación entregada contenga, diseño y código fuente. En la fase de diseño deberá entregarse el diagrama de clases.

El funcionamiento de la práctica no garantiza el aprobado de esta, se valorarán otros aspectos como la robustez y eficiencia de esta, el correcto diseño y perfecta utilización de las técnicas y metodologías comentadas en clase de teoría, la documentación entregada y el cumplimiento de los requisitos definidos en el enunciado.

Toda esta información se subirá a través de Campus Virtual en un espacio habilitado para tal efecto, antes de la hora de la defensa para que quede constancia de la entrega. Posteriormente a esta fecha no se permitirá subir ningún otro documento. Toda la documentación deberá ir comprimida en un único archivo con extensión .rar o .zip. Preferiblemente la documentación debería ir en formato PDF y el código fuente en su correspondiente archivo .java. Los archivos binarios .class no son necesarios ya que deberán generarse de forma fácil siguiendo las instrucciones redactadas en el manual de usuario.

También se recuerda que el hecho, tanto de copiar una práctica como dejarla para que otros la copien, es una falta muy grave que será sancionada.

Para facilitar la localización y corrección de los entregables, los ficheros de entrega se enviarán utilizando un nombre en el que conste el número de entregable y el número del equipo. Por ejemplo, para la práctica 5 del equipo GA1, el fichero deberá llamarse P5‐GA1.pdf.

Acompañando a cada entregable se enviará un informe en el que conste el nombre de cada miembro del equipo y el porcentaje de su participación y realización para dicho entregable.

# ANEXO 1. PUNTUACIONES CÓDIGO

En este anexo, se encuentran la Tabla 1 la rúbrica del código fuente, donde se detallan las puntuaciones por cada apartado.

*Tabla 1. Rúbrica Tema 1. Práctica 1. Código Fuente.*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Aspecto** | | | | **Escala de calificación de los ejercicios** | | | |
|  | **Peso** | **10 excelente** | **8 notable** | | **6 bien** | **5 suficiente** | **1 insuficiente** |
| Diseño y manipulación de objetos | 30% | Correcto diseño y manipulación de objetos |  | | Uso de objetos de forma incorrecta en algunos casos |  | No diseña, ni utiliza los objetos correctamente |
| Diseño y manipulación de ficheros. | 30% | Uso correcto de herencia, abstracción y polimorfismo | Uso correcto en la mayoría de las situaciones | |  |  | Uso incorrecto de herencia, abstracción y polimorfismo |
| Detección y modelado de relaciones entre clases | 10% | Relación correcta entre clases |  | | Algunas clases no están bien relacionadas |  | Las clases no están bien relacionadas |
| Diseño y manipulación de Excepciones | 5% | Correcto diseño y manipulación de excepciones | Correcta manipulación de excepciones, pero problemas en su diseño | | Uso de excepciones de forma incorrecta en algunos casos |  | No diseña, ni utiliza las excepciones correctamente |
| Diseño y manipulación de interfaces | 5% | Correcto diseño y manipulación de interfaces |  | |  |  | No diseña, ni utiliza los interfaces correctamente |
| Correcto uso de modificadores de visibilidad | 2,5% | Correcto diseño y manipulación de modificadores |  | |  |  | No diseña, ni utiliza los modificadores correctamente |
| Correcto uso de métodos sobrecargados | 2,5% | Correcto diseño y uso de métodos sobrecargados |  | |  |  | Incorrecto diseño y uso de métodos sobrecargados |
| Aplicación de todos los conocimientos sobre programación aprendidos hasta el momento | 10% | Utiliza correctamente los conocimientos adquiridos anteriormente | Utiliza correctamente los conocimientos adquiridos anteriormente en la mayoría de casos | | Utiliza correctamente los conocimientos adquiridos anteriormente en algunos de casos |  | No utiliza correctamente los conocimientos adquiridos anteriormente |
| Uso de patrones de desarrollo | 5% | Uso correcto |  | |  |  | Uso incorrecto |
| Claridad del código fuente | -10% | El código no está bien estructurado, ni hace uso de comentarios | El código no está bien estructurado o no se hace uso de comentarios | |  | El código fuente no es del todo claro y hace uso de comentarios irrelevantes | El código fuente es claro y hace uso de comentarios descriptivos |

# ANEXO 2. PUNTUACIONES UML

En este anexo, se encuentran la Tabla 2 la rúbrica del UML, donde se detallan las puntuaciones por cada apartado.

*Tabla 2. Tema 1. Rúbrica Práctica 1. UML.*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Aspecto** | | | | **Escala de calificación de los ejercicios** | | | |
|  | **Peso** | **10 excelente** | **8 notable** | | **6 bien** | **5 suficiente** | **1 insuficiente** |
| Diseño de clases | 30% | Correcto diseño de clases |  | | Alguna clase no se corresponde con el diseño |  | Incorrecto diseño de clases |
| Diseño de paquetes | 10% | Correcto diseño de paquetes |  | | Algún paquete no se corresponde con el diseño |  | Incorrecto diseño de paquetes |
| Relación y modelado entre clases | 20% | Relación correcta entre clases |  | | Algunas clases no están bien relacionadas |  | Las clases no están bien relacionadas |
| Diseño de excepciones e interfaces | 20% | Realiza un correcto diseño de excepciones e interfaces |  | |  | Realiza un correcto diseño de excepciones o interfaces | No realiza un correcto diseño de excepciones, ni interfaces |
| Manejo de modificadores de visibilidad | 10% | Utiliza correctamente los modificadores |  | |  |  | No utiliza correctamente los modificadores |
| Manejo de métodos sobrecargados | 10% | Utiliza correctamente los métodos |  | |  |  | No utiliza correctamente los métodos |

# RESULTADOS DE APRENDIZAJE Y CRITERIOS DE EVALUACIÓN A EVALUAR.

RA 1. CE A, CE B, CE C, CE G y CE H.