



Ejercicio taller serializable

Desarrollar una clase llamada Motor que:

- Tenga dos atributos private de tipo int (litros de aceite) y de tipo int (CV).
- Tenga un constructor con un parámetro de tipo int para los CV. Los litros de aceite por defecto serán 0.
- Tenga un getter para cada uno de los atributos.
- Tenga un setter para los litros.
- ToString

Desarrollar una clase llamada Coche que:

- Tenga un atributo private de tipo Motor, un atributo de tipo String (marca), otro de tipo String (modelo) , String matricula, otro String con la avería que presenta y otro de tipo double con el precio que cuesta arreglar la avería.
- Tenga un getter para cada uno de los atributos.
- Un método acumularAveria que incrementará el importe gastado en averías.
- ToString

Desarrollar una clase llamada Taller que:

- Tendrá como atributos, un ArrayList de coches, un double con la ganancia total del taller.
- Tiene un método aceptarCoche que debe solicitar al usuario la información del Coche y generarlo antes de introducirlo, debe añadir la avería que tiene y el precio de arreglar la misma
- Tenga un método arreglarCoche que pide al usuario la matricula del coche y lo saca del array, pondrá la avería del Coche a null y añadirá el precio de la avería a la ganancia del taller
- Sobrecarga el método toString para que muestre la información de todos los coches que hay en el garaje.

Genera dos métodos estáticos en el main del programa:

- Uno llamado guardarTaller que reciba por parámetro un garaje, solicite una ruta nombre para este y lo guarde en la carpeta del proyecto.
- Otro método cargarTaller que solicite el nombre de un garaje y si existe lo cargue.
- Controla los flujos mediante excepciones.

Desarrollar una clase llamada Actividad2 que en su método main:

- Crea un menú que te permita:
 - Añadir coches al taller.
 - Arreglar coche.
 - Consultar info de los coches que hay en el taller.
 - Consultar la ganancia del taller.
 - Salvar el taller
 - Cargar un taller
 - Salir