1. Se quiere implementar un simulador de vehículos. Existen dos clases de **Vehiculos: Coche** y **Camion**.
2. De todos los **Vehiculos** queremos almacenar la matrícula y la velocidad. En el momento de crearlos, la matricula se recibe como parámetro y la velocidad se pone a 0. En su correspondiente método toString() se devolverá la matrícula y la velocidad. Además se puede **acelerar** pasando como parámetro la cantidad en km/h que se tiene que acelerar.
3. Los coches tienen además un atributo para el número de puertas que se recibe también como parámetro a la hora de crearlo. Y contiene un método que **devuelve** el número de puertas.
4. Los camiones tienen un atributo del tipo de **Remolque** que se inicializa a null en el momento de crear el camión para indicar que aún no tiene remolque. Definimos en la clase dos métodos, **ponRemolque** que recibe un Remolque por parámetro y otro **quitaRemolque** que pone el atributo a null. Cuando se muestre la información que lleve remolque, además de matrícula y velocidad debe aparecer información del remolque.
5. En la clase Camion hay que sobrescribir el método de **acelerar,** de forma que si el camión lleva remolque y la velocidad supera los 100 km/h debe levantarse una excepción del tipo **DemasiadoRapidoExcepcion**.
6. Hay que implementar la clase Remolque que tiene un atributo de tipo entero que es el peso del remolque y su valor se le asigna a la hora de crear el Remolque. Tendrá su método toString() como todas las demás clases.
7. Se implementará la clase **DemasiadoRapidoExcepcion.**
8. Crearemos la clase **PruebaVehiculo** con un método main donde se tenga un menú con las siguientes operaciones (utiliza un ArrayList para guardar los vehículos que se creen desde el menú):

**1.- Añadir coche**

**2.- Añadir camión**

**3.- Añadir un remolque (de peso en kg el que indique el usuario) a todos los camiones**

**4.- Muestre la información de todos los vehículos por pantalla.**

**5.- Borre todos los camiones que tengan un remolque de más de 1000 kg**

**6.- Salir**

Para practicar ordenación con interfaces de Java, prueba a mostrar los vehículos ordenados por todos los criterios que se te ocurran.