# Задача: Създайте йерархия от класове за управление на превозни средства

1. Създайте абстрактен клас `Vehicle` със следните членове:  
 - \*\*Абстрактни методи\*\*:  
 - `GetMaxSpeed(): double` - връща максималната скорост на превозното средство.  
 - `GetFuelConsumption(): double` - връща разхода на гориво на превозното средство.  
 - \*\*Виртуални методи\*\*:  
 - `Drive(): string` - този метод трябва да връща съобщение във формата:   
 `$"Driving a {classType.Name}"`, където `classType.Name` е името на класа.  
  
2. Наследете класа `Vehicle` с два дъщерни класа:  
 - `Car`  
 - `Motorcycle`  
  
3. За всеки от тях:  
 - Добавете \*\*полетата\*\*:  
 - За `Car`: `numberOfDoors` (брой врати) и `enginePower` (мощност на двигателя).  
 - За `Motorcycle`: `engineCapacity` (обем на двигателя) и `hasSidecar` (дали има странична количка).  
 - Осигурете \*\*инкапсулация\*\* за тези полета.  
 - Добавете \*\*публичен конструктор\*\*, който инициализира всички полета.  
 - Имплементирайте конкретни методи за `GetMaxSpeed` и `GetFuelConsumption`.  
 - Презапишете метода `Drive`, за да върнете подходящо съобщение за съответния тип превозно средство.  
  
4. Демонстрирайте полиморфизъм:  
 - Създайте масив от обекти от тип `Vehicle`, съдържащ както автомобили, така и мотоциклети.  
 - Обходете масива и извикайте методите `Drive`, `GetMaxSpeed` и `GetFuelConsumption` за всеки обект.