**Aufgabe**

Erstellen Sie eine Solution mit dem Name „Car.sln“. Die Syntax ist C#, Windows, .NET 8 (Core).

Es gibt zwei Marken von Cars: Ford und VW.

Ein Car bildet ein Aggregate-Root. Die Invariaten sind:

Sobald eine Car-Instanz instanziiert ist, hat diese ganz genau vier Räder: Vorne rechts, vorne links, hinten rechts, hinten links. Die Räder haben auch diese Bezeichnung „Vorne rechts“ etc.

Der Ford-Car hat ein Maximum-Speed von 250 kmh.

Der VW-Car hat ein Maximum-Speed von 180 kmh.

Jeder Car hat einen Jahrgang.

Ich als Konsument dieser Schnittstelle möchte folgendes:

Einen Art von „CarStock“ (AutoLager) in das ich Cars der Marke Ford und VW einbuchen kann. Für den CarStock reicht eine InMemory-Datenbank - daher eine reine Struktur im Cache. Die Einbuchung muss nicht via Console User Input geschehen. Sie können hier beim Applikations-Start eine Einbuchung nach Ihrem Gutdünken machen.

Wenn das Einbuchen geschehen ist, dann möchte ich, dass mir der CarStock eine Inventur-Liste erzeugt mit folgender Ausgabe:

Marke, Jahrgang, Max Speed.

Sie können dazu eine normale Console-Application erzeugen und mit Consol-Output arbeiten. Wie Sie dabei die Solution allenfalls strukturieren ist Ihre Entscheidung.

Was von Interesse ist:

Das objektorientierte Design, Testability (Module Tests)

Als Testing/Isolations-Framework kann NSubstitute eingesetzt werden.

Wo allenfalls Unklarheiten bestehen, treffen Sie Annahmen und Kommentieren diese im Code.

Sie können Ihre Aufgabe in einem Git Repository einchecken.

Dabei sollte nach Auscheken, nur noch F5 gedrückt werden können und alles läuft.

**Tip**: Da ich keine Freund von wahllosen NuGet-Packages bin - arbeiten Sie plain mit den im .NET Framework verfügbaren Typen.