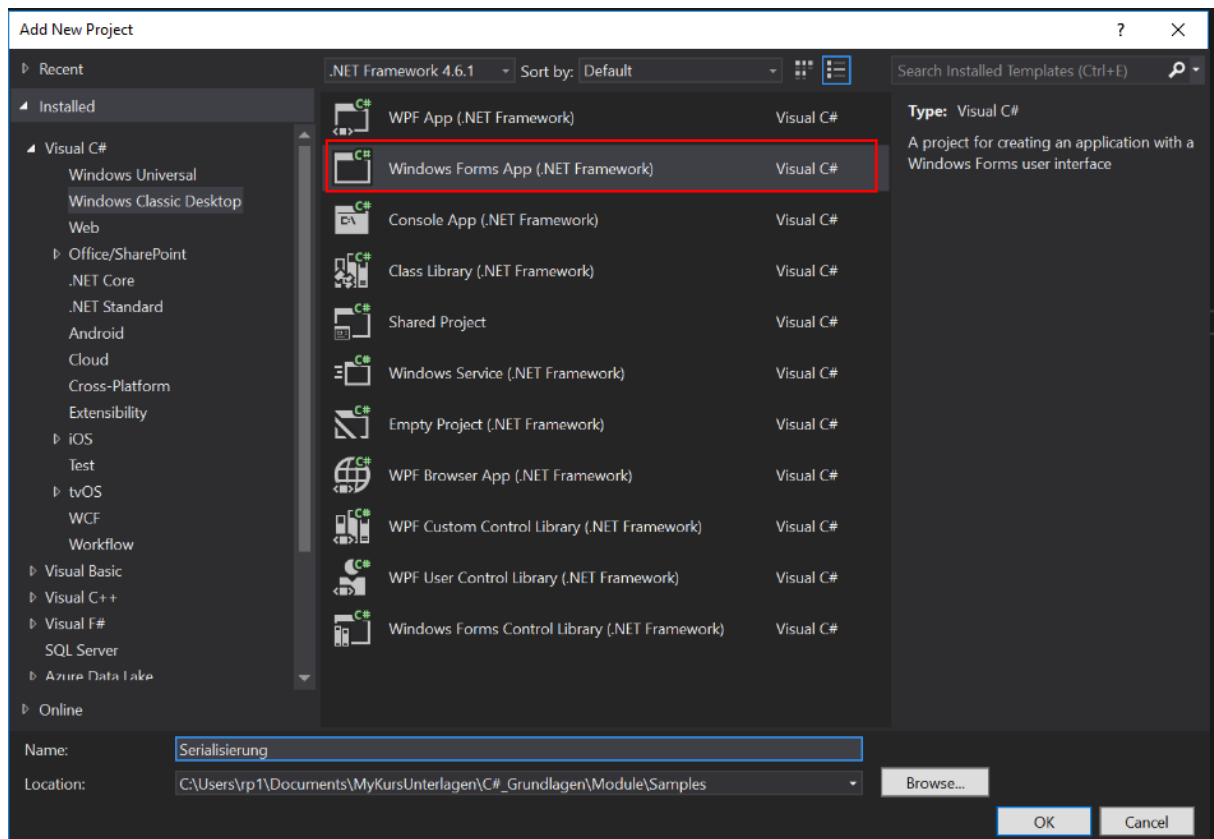


Lab13: Serialisierung

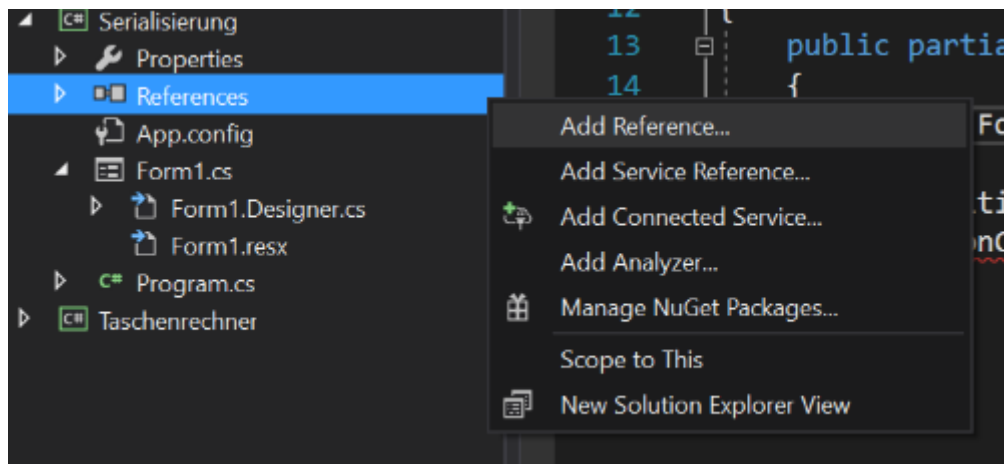
1. Vorbereitung

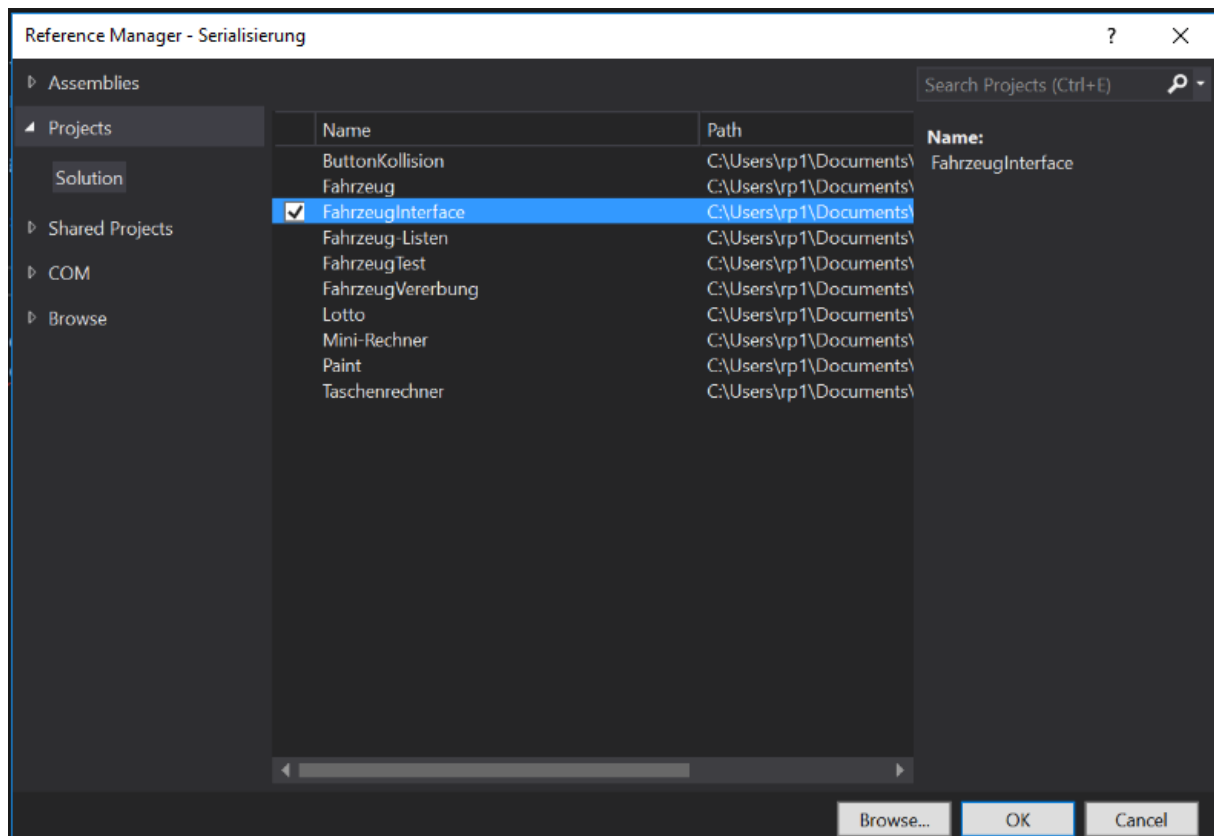
Diese Übung baut auf dem Programm Fuhrpark aus Übung06 auf.

Fügen Sie Ihrer Solution ein neues Windows-Forms-Projekt namens „Serialisierung“ hinzu.

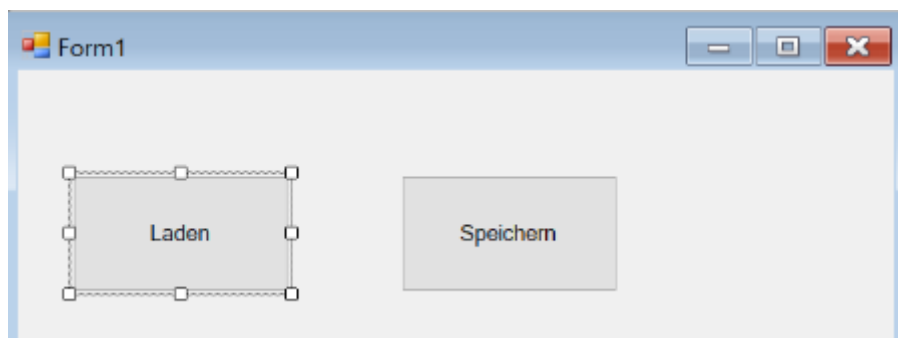


Binden Sie nun unter References Ihr Projekt hinzu, in denen Sie die Fahrzeug-Klassen definiert hatten.





Legen Sie nun 2 Buttons an und beschriften Sie sie mit „Laden“ und „Speichern“.



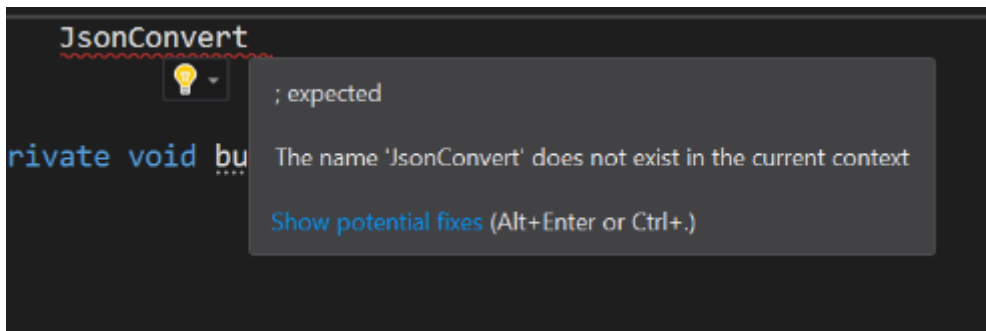
Klicken Sie doppelt auf die Buttons um für beide automatisch Click-EventHandler zu deklarieren.

```
private void button1_Click(object sender, EventArgs e)
{
    ...
}

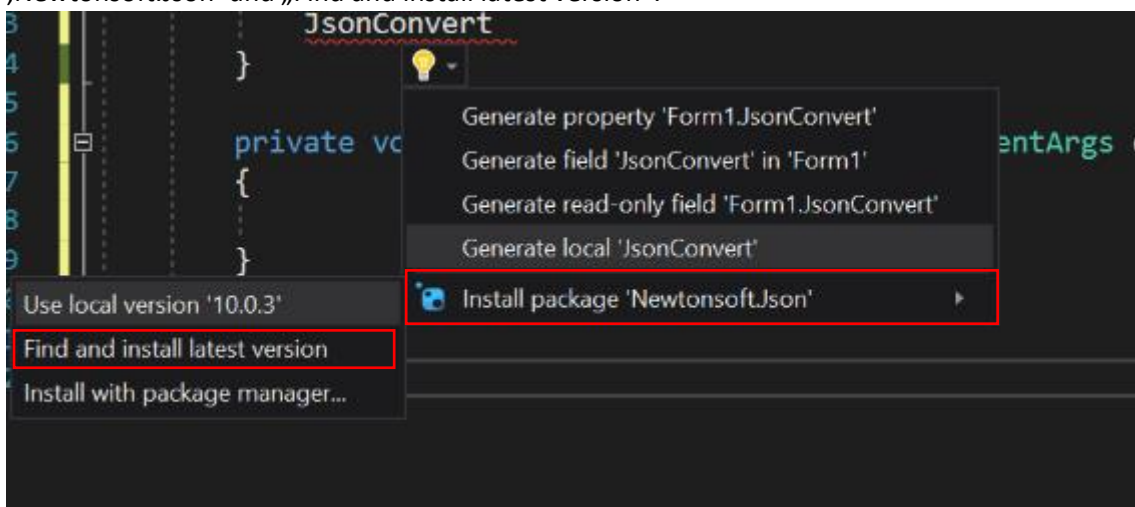
private void button2_Click(object sender, EventArgs e)
{
    ...
}
```

Tippen Sie in eine der beiden Funktionen „JsonConvert“ ein und warten Sie bis das Wort rot unterstrichen angezeigt wird. Fahren Sie dann mit dem Maus-Cursor über das rot unterstrichene

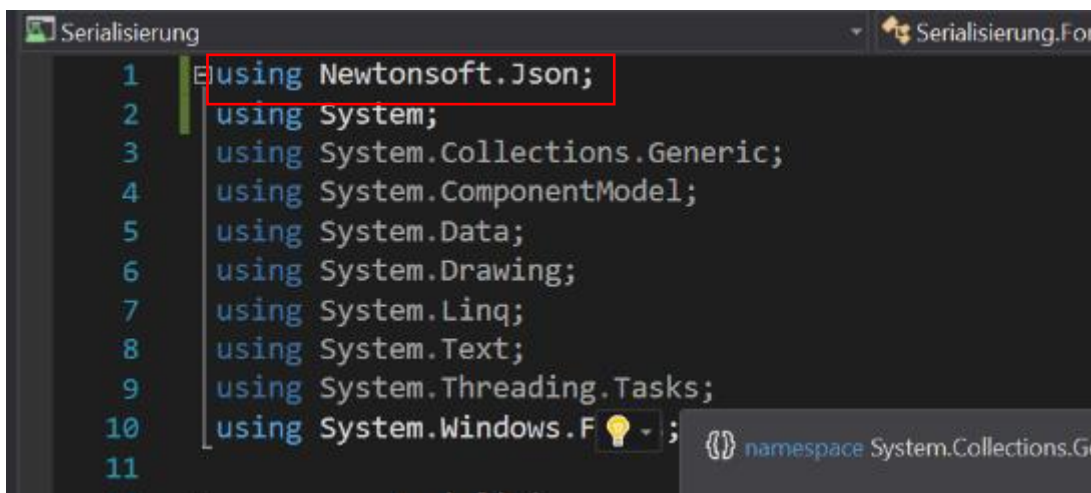
Warten Sie einen Moment, bis sich ein kleiner Kasten eingeblendet wird, innerhalb dem sich das blau hervorgehobene Wort „Show potential fixes“ befindet:



Klicken Sie auf „Show potential fixes“ und anschließend auf „Install package ‚Newtonsoft.Json‘“ und „Find and install latest version“:



Daraufhin sollte in Form1.cs das Statement „using Newtonsoft.Json“ hinzugefügt sein:



Ab jetzt können Sie die statischen Methoden der JsonConvert-Klasse benutzen.

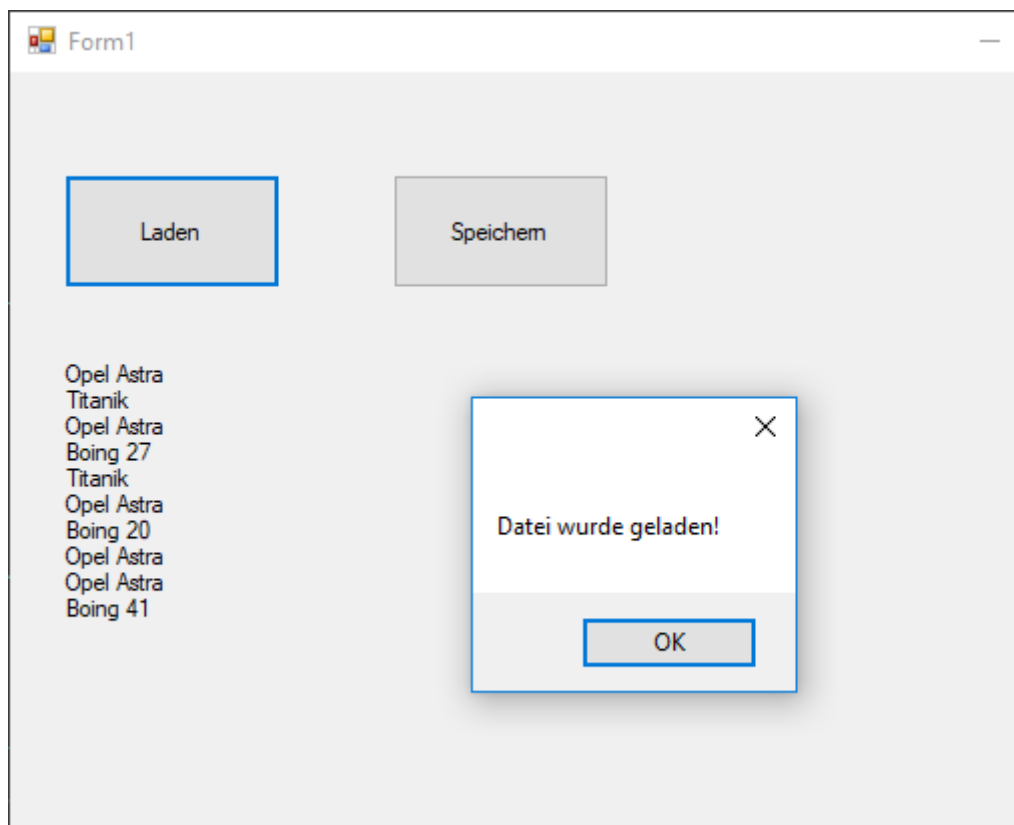
2. Aufgabenstellung

Erzeugen Sie code-seitig eine Liste von zufällig erzeugten Fahrzeug-Objekten, welche man über den Button Speichern mittels JSON-Serialisierung in einer Datei speichern kann, der Nutzer soll dabei über einen SaveFileDialog Ort und Namen der Datei auswählen können.

Analog dazu soll man mittels des Laden-Buttons eine entsprechende Datei öffnen und die darin im JSON-Format gespeicherten Daten über Deserialisierung wieder ins Programm laden können.

Nutzen Sie das Form-Element Label um die Daten der aktuellen Liste anzuzeigen.

So könnte ein fertiges Programm aussehen:



Viel Erfolg!