

Versuchen Sie immer, Ihren Code zu kommentieren!

Aufgabe 1

Gegeben ist ein XML-String (siehe books.xml Datei), der eine komplexe Datenstruktur darstellt. Ohne die Verwendung einer Konverter-Website sollen die passenden Klassen, Datentypen und Attribute ausgewählt werden, um den XML-String korrekt zu deserialisieren.

- Analysiere den gegebenen XML-String und identifiziere die verschiedenen Objekte und deren Eigenschaften
- Erstelle die entsprechenden Klassen in C#, die die Struktur des XML-Strings widerspiegeln
- Verwende die richtigen Datentypen und Attribute (XmlText, XmlElement, XmlAttribute..), um sicherzustellen, dass die Deserialisierung korrekt durchgeführt wird
- Implementiere die Deserialisierung des XML-Strings in C# und überprüfe, ob die Daten erfolgreich in die entsprechenden Klassen geladen werden

Aufgabe 2

Gegeben ist ein JSON-String (siehe car.json Datei), der eine komplexe Datenstruktur darstellt. Ohne die Verwendung einer Konverter-Website sollen die passenden Klassen, Datentypen und Attribute (JsonProperty...) ausgewählt werden, um den JSON-String korrekt zu deserialisieren.

- Analysiere den gegebenen JSON-String und identifiziere die verschiedenen Objekte und deren Eigenschaften
- Erstelle die entsprechenden Klassen in C#, die die Struktur des JSON-Strings widerspiegeln
- Verwende die richtigen Datentypen und Attribute, um sicherzustellen, dass die Deserialisierung korrekt durchgeführt wird
- Implementiere die Deserialisierung des JSON-Strings in C# und überprüfe, ob die Daten erfolgreich in die entsprechenden Klassen geladen wurden

Aufgabe 3

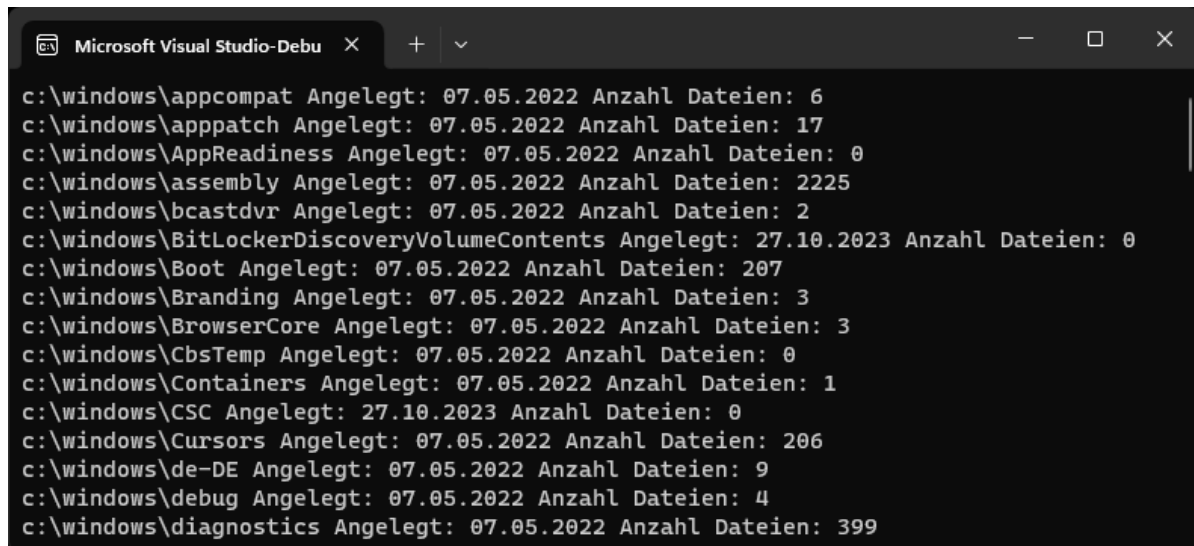
Eine Konsolenanwendung soll für den Ordner „C:\Windows“ alle möglichen Unterverzeichnisse ermitteln und von den Ordnern den Namen, das Erstellungsdatum und die Anzahl der Dateien in dem Ordner ermitteln.

Wichtiger Hinweis: Durch fehlende Berechtigungen können ggf. nicht alle Ordner durchsucht werden. Die Anzahl der Dateien ist dann 0.

Die drei Information sollen in Properties einer zu erstellenden Klasse (z.B. CustomDirInfo) gespeichert werden. Die Klasse soll für die Ausgabe auf der Konsole eine überladene ToString Methode enthalten. Erstellen Sie pro Verzeichnis ein Objekt der Klasse DirInfo und speichern Sie alle Objekte in einer Liste.

Speichern Sie den Inhalt der Liste in einer Json-Datei.

Deserialisieren Sie die Json-Datei anschließend wieder und rufen Sie die ToString Methode aller Objekte für die Ausgabe aus.



```
Microsoft Visual Studio-Debu x + v - □ ×  
c:\windows\appcompat Angelegt: 07.05.2022 Anzahl Dateien: 6  
c:\windows\apppatch Angelegt: 07.05.2022 Anzahl Dateien: 17  
c:\windows\AppReadiness Angelegt: 07.05.2022 Anzahl Dateien: 0  
c:\windows\assembly Angelegt: 07.05.2022 Anzahl Dateien: 2225  
c:\windows\bcastdvr Angelegt: 07.05.2022 Anzahl Dateien: 2  
c:\windows\BitLockerDiscoveryVolumeContents Angelegt: 27.10.2023 Anzahl Dateien: 0  
c:\windows\Boot Angelegt: 07.05.2022 Anzahl Dateien: 207  
c:\windows\Branding Angelegt: 07.05.2022 Anzahl Dateien: 3  
c:\windows\BrowserCore Angelegt: 07.05.2022 Anzahl Dateien: 3  
c:\windows\CbsTemp Angelegt: 07.05.2022 Anzahl Dateien: 0  
c:\windows\Containers Angelegt: 07.05.2022 Anzahl Dateien: 1  
c:\windows\CSC Angelegt: 27.10.2023 Anzahl Dateien: 0  
c:\windows\Cursors Angelegt: 07.05.2022 Anzahl Dateien: 206  
c:\windows\de-DE Angelegt: 07.05.2022 Anzahl Dateien: 9  
c:\windows\debug Angelegt: 07.05.2022 Anzahl Dateien: 4  
c:\windows\diagnostics Angelegt: 07.05.2022 Anzahl Dateien: 399
```