

Übung 5

SERIALISIERUNG / MARSHALLING

Nachfolgend finden Sie eine Übungsaufgabe zur Serialisierung von Daten in verschiedene Datenformate, welche aus einer praktischen Aufgabe und mehreren Verständnisfragen bestehen.

Bitte beachten Sie, dass eine Gruppenarbeit für diese Übung nicht vorgesehen ist.

1. DATENFORMATE & SCHEMATA (Teil 1 von 2)

(a) Erstellen Sie ein XML-Schema (ggf. in einem gewöhnlichen Texteditor) für das Szenario des Datenaustauschs zwischen Flugbuchungsportal und Airline.

Bitte verwenden Sie für Ihre Attribute folgende Datentypen:

- integer
- floating point
- string
- boolean
- date
- time
- Komplexer Datentyp

Bitte ergänzen Sie ihr Schema um drei weitere, eigene Attribute. Davon sollte

- ein Attribut ein komplexer Datentyp sein
- zwei Attribute required*

Speichern Sie Ihre xsd.-Datei unter „<Matrikelnummer_Name>.xsd“ und tauschen Sie diese mit einem Kommilitonen.

(b) Erstellen Sie eine XML-Datei mit fünf Datenobjekten passend zu Ihrer xsd.-Datei der vorherigen Übung (a).

Bitte denken Sie daran in Ihrer XML-Datei auf das Schema zu verweisen!

Speichern Sie Ihre XML-Datei unter „<Matrikelnummer_name>.xml“ und tauschen Sie diese mit einem Kommilitonen.

(c) Recherchieren Sie online nach XML-Parsern, die Ihnen das Validieren einer XML-Datei gegen ein XML-Schema ermöglichen.

Bitte suchen Sie sich nach dem Zufallsprinzip eine (fremde) xsd.-Datei und die entsprechende Objekt-Datei (XML) dazu und validieren Sie diese online.

Machen Sie einen Screenshot des Ergebnisses.

War das analysierte XML-Dokument „wohlgeformt“?

(d) Erstellen Sie eine JSON- Datei (ggf. in einem gewöhnlichen Texteditor) für die in (b) erstellten

Datenobjekte. Stellen Sie hierbei die Objekte in einem Array (nicht nur als Objekte!) dar.

Speichern Sie Ihre JSON-Datei unter „<Matrikelnummer_name>.json“ und tauschen Sie diese mit einem Kommilitonen.

Hinweis: Wichtig ist, dass Sie die Übungen 1 (a) und 2 (a)(b) NICHT mit Ihren eigenen XML und XSD Dateien durchführen.

2. Aufgabe (Teil 2 von 2) – Marshalling / Serialisierung

(a) Lassen Sie die Ihnen zugeteilte XML-Datei über Ihr geschriebenes Programm - in C# oder Java - einlesen und geben Sie die Objekte OHNE XML-SYNTAX aus (Konsole, HTML- Seite - denken Sie an .CSS Dateien oder HTML-MarkUps – Ihnen überlassen). Machen Sie einen Screenshot und senden Sie diesen unter Angabe der Matrikelnummer der bearbeiteten Kommilitonen-Datei.

(b) Lassen Sie die Ihnen zugeteilte JSON-Datei über Ihr geschriebenes Programm einlesen und geben Sie die Objekte aus (Konsole, HTML- Seite – Ihnen überlassen). Machen Sie einen Screenshot und senden Sie diesen unter Angabe der Matrikelnummer des bearbeiteten Kommilitonen.

(c) Beantworten Sie folgende Fragen stichpunktartig:

1. Traten Schwierigkeiten während des Parsens oder der Serialisierens auf, wenn ja, welche?
2. Für welche risikofreie strategische Lösung würden Sie sich als Entwickler entscheiden (oder haben Sie sich entschieden) bezüglich der fehlenden strukturellen Beschreibung erwarteter Daten in JSON (im Gegensatz zu XML i.V.m. XSD Dateien), wenn Sie Daten in einem JSON-Array erwarten und diese in eine objektorientierte Sprache serialisieren?

(d) Recherchieren Sie zwei Beispiele für Objekttypen in C# oder Java (Bps. Threads), die sich nicht serialisieren lassen. Erläutern Sie stichpunktartig Problem und Lösung (darf abgeschrieben sein).

Technische Dokumentation

Die **Abgabe** der Übung erfolgt über einen Post im Forum (Moodle), der die nachfolgenden Informationen enthalten sollte:

- Vor- und Nachname sowie Matrikelnummer
- Dateinamen zu bearbeiteten Dateien der Kommilitonen
- Link zu Online Validierung XML Dateien
- kurze Antworten auf die Fragen (Stichpunkte)
- Terminalausgabe (Auszug) oder Link / Screenshot des Web Frontends
- Vollständiger und dokumentierter (!) Code in Datei(en)