

RMI - Übersicht und Einführung

Die folgenden Aufgaben sollten in Teams zu je zweien gemacht werden, wobei jeweils ein Teammitglied für die Serverseite, das andere für die Clientseite zuständig ist.

Datumserver

Schreiben Sie ein RMI Client / Serversystem, bei dem der Client mit Hilfe eines remote Methodenaufrufs ein Date Objekt erhält mit dem aktuellen Datum plus Uhrzeit.

MwSt-Server

Schreiben Sie ein RMI Client / Serversystem, bei dem der Client mit Hilfe eines remote Methodenaufrufs folgende mathematische Berechnungen durchführen kann:

- a) Berechnung der Mehrwertsteuer (7.5%)
Eingabe : ein Betrag mit maximal zwei Kommastellen
Rückgabe : die Mehrwertsteuer
Hinweis : der Mehrwertsteuersatz soll auf dem Server extern abgespeichert werden und vor der Berechnung eingelesen werden, aus einer Textdatei
- b) Erweiterte Berechnung der Mehrwertsteuer
Eingabe : ein Betrag mit maximal zwei Kommastellen, eine Mehrwertsteuerkategorie
Rückgabe : die Mehrwertsteuer gemäss der Steuerklasse
Hinweis : wie in a) werden in einer Textdatei die Mehrwertsteuersätze zu den Kategorien abgespeichert, im Format <Kategorie><Prozentsatz>
Beispiel für eine Mehrwertsteuertabelle
 "Dienstleistung"; 7.5
 "Verbrauchsmaterial"; 4.5
 ...
- c) Erweiterte Berechnung der Mehrwertsteuer
Eingabe : ein Array mit Kategorien und Beträgen ("Kategorie" und Preis)
Ausgabe : ein Array mit der Mehrwertsteuer pro Preis

Warenlager-Server

Schreiben Sie ein RMI Client / Server System, welches die Verfügbarkeit von Online angebotenen Artikeln überprüft.

Eingabe : ein Einkaufskorb in Form einer Liste (als Vector : Artkle plus Bestellmenge)

Ausgabe : eine Liste mit der Verfügbarkeit pro Artikel

Zusätzlich soll eine Backorder Liste generiert und auf dem Server abgespeichert werden. In diese Liste werden jene Artikel (und Artikelmenen) eingetragen, welche sich im Einkaufskorb befanden, die aber nicht lieferbar sind (Bestand = 0 oder kleiner der Anzahl bestellter Artikel), pro Artikel jeweils eine Zeile mit dem Total ausstehender Artikel.

Hinweise: serverseitig wird ein Lager (einfache Textdatei) geführt. Am Anfang sollen alle Artikel erhältlich sein. Nach der Bearbeitung der Bestellung sollen jeweils entsprechend der Bestellung weniger Artikel an Lager sein. Der Einfachheit halber nehmen wir an, dass das Lager *nicht* aufgefüllt wird.

Projektaufgabe

Flugreservationssystem, gemäss Skript und Programmfragmenten.