
Lehrveranstaltung "Objektorientierte Programmierung"

Übungsblatt 5

Hinweise:

Dieses Übungsblatt ist zur Zulassung zu der Klausur erfolgreich zu bearbeiten ("*Erfolgreich*" bedeutet: Keine Programmabstürze bzw. Endlosschleifen, Aufgabenstellung einschl. der Nebenbedingungen müssen eingehalten sowie Kommentierung und Einrückung korrekt sein!).

Die Aufgaben werden überwiegend in den Übungszeiten bearbeitet und dort auch abgegeben. Allerdings genügt die Zeit hierfür unter Umständen nicht, so dass Sie auch außerhalb dieser Zeiten die Aufgaben bearbeiten müssen. Der Abgabetermin für diese Aufgabe ist der **09. Dezember 2022**.

Aufgabe: In der 5. Übungsaufgabe des Projektes „Belegungen“ soll die Klasse `CBookings` als Containerklasse für alle Daten des Hauptprogramms erstellt werden. Ferner sollen alle Daten jetzt aus einer einfachen XML-Datei namens "`belegungen.xml`" eingelesen werden.

Die Klasse `CBookings` erhält als private Eigenschaften jeweils Vektoren mit Zeigern auf Blöcke, Räume, Studiengänge, Fächer, Belegungen sowie Studenten und Dozenten – also fast alle Daten, die bisher im Hauptprogramm waren; jetzt nur als Vektoren von Zeigern auf die entsprechenden Daten.

Für diese Klasse sollen dann für alle Vektoren jeweils eine `find`-Methode erstellt werden, z.B. eine `findRoom`-Methode, die anhand des übergebenen Raumnamens das entsprechende Objekt im Räume-Vektor sucht und den Zeiger auf das gefundene Raum-Objekt zurückgibt (bzw. einen `NULL`-Zeiger, wenn der Raum nicht gefunden wird).

Der Konstruktor erhält den Dateinamen als Parameter und öffnet die angegebene Datei zum Lesen. Dann wird in einer Schleife die Datei zeilenweise eingelesen. Nach dem Einlesen jeder Zeile wird diese gleich ausgewertet. Dabei soll jede Klasse nur seine eigenen Eigenschaften einlesen, d.h. (fast) jede Klasse erhält eine `load`-Methode, die jeweils den geöffneten Datenstrom als Parameter erhält; bei manchen Klassen muss zusätzlich eine Referenz auf das `CBookings`-Objekt übergeben werden, damit auf die `find`-Methoden der bereits eingelesenen und in den Vektoren gespeicherten Objekte zugegriffen werden kann.

Beispiel:

Eingelesene Zeile: `<Student>`

Es wird ein neues Objekt der Klasse `CStudent` erstellt (per `new`) und dann die `load`-Methode der Klasse `CStudent` aufgerufen. Diese liest wieder in einer Schleife die nächste Zeile ein (z.B. `<name>Max</name>`) und setzt die Eigenschaft `Name` der Klasse `CStudent`. Dies wird wiederholt, bis das Ende-Tag `</Student>` gelesen wurde. Damit ist die `load`-

Methode der Klasse `CStudent` fertig und wird beendet. Wieder zurück im Konstruktor der Klasse `CBookings` wird nun der Zeiger des fertig eingelesenen Studenten dem Personen-Vektor hinzugefügt.

Die Destruktoren müssen sich um das Freigeben der reservierten Speicherbereiche kümmern. Zur besseren Kontrolle sollen dabei die Destruktoren der einzelnen Klassen jeweils auf dem Bildschirm ausgeben, welches Objekt gerade vernichtet wird (siehe Beispiel-Ausgabe).

Zum Testen der Klassen soll das vorgegebene (und überraschend kurze) Hauptprogramm verwendet werden, das Objekte der angegebenen Klassen erzeugt, auf verschiedene Werte setzt und wieder auf dem Bildschirm ausgibt.

vorgegebenes Hauptprogramm:

```
#include <iostream>

using namespace std;

#include "cbookings.h"

int main()
{
    CBookings Belegungen("belegungen.xml");

    Belegungen.print();

    return 0;
}
```

Beispiel-Ausgabe:

Datei wurde erfolgreich eingelesen!

Belegungen:

```
Belegung Nr. 1:
Am 04.10.2022 um 05:17:29
hat Anna Albrecht (* 15.09.1989; MatrNr. 191278; ID 1)
das Fach 'OOP (31101; Technische Informatik)' belegt.

Belegung Nr. 2:
Am 04.10.2022 um 05:19:43
hat Anna Albrecht (* 15.09.1989; MatrNr. 191278; ID 1)
das Fach 'Systemprogrammierung (33102; Technische Informatik)' belegt.

Belegung Nr. 3:
Am 04.10.2022 um 05:23
hat Anna Albrecht (* 15.09.1989; MatrNr. 191278; ID 1)
das Fach 'Mathe III (31103; Technische Informatik)' belegt.

Belegung Nr. 4:
Am 05.10.2022 um 19:21:03
hat Detlef Dicht (* 27.05.1991; MatrNr. 191115; ID 2)
das Fach 'Mathe III (31103; Technische Informatik)' belegt.

Belegung Nr. 5:
Am 04.10.2022 um 11:03:31
hat Frank Faust (* 03.11.1987; MatrNr. 191492; ID 3)
das Fach 'Mathe III (31103; Technische Informatik)' belegt.

Belegung Nr. 6:
Am 05.10.2022 um 01:52:11
hat Max Muster (* 19.07.1985; MatrNr. 191368; ID 4)
das Fach 'Mathe III (31103; Technische Informatik)' belegt.
```

Bloেকে freigegeben ...
Block Nr. 1 wird vernichtet.
Block Nr. 2 wird vernichtet.
Block Nr. 3 wird vernichtet.
Block Nr. 4 wird vernichtet.
Block Nr. 5 wird vernichtet.
Block Nr. 6 wird vernichtet.
Block Nr. 7 wird vernichtet.
Bloেকে freigegeben ok
Räume freigegeben ...
Raum D 113 wird vernichtet.
Raum D 114 wird vernichtet.
Raum D 117 (H3) wird vernichtet.
Raum D 136 (H5) wird vernichtet.
Raum D 209 wird vernichtet.
Raum D 419 wird vernichtet.
Raum B 301 wird vernichtet.
Räume freigegeben ok
Studiengänge freigegeben ...
Studiengang Technische Informatik wird vernichtet.
Studiengang Physik wird vernichtet.
Studiengang Elektrotechnik wird vernichtet.
Studiengänge freigegeben ok
Studenten freigegeben ...
Student*in Anna Albrecht wird vernichtet.
Person Anna Albrecht wird vernichtet.
Student*in Detlef Dicht wird vernichtet.
Person Detlef Dicht wird vernichtet.
Student*in Frank Faust wird vernichtet.
Person Frank Faust wird vernichtet.
Student*in Max Muster wird vernichtet.
Person Max Muster wird vernichtet.
Studenten freigegeben ok
Dozenten freigegeben ...
Lehrer*in Kevin Kaufmann wird vernichtet.
Person Kevin Kaufmann wird vernichtet.
Lehrer*in Nathan Neuling wird vernichtet.
Person Nathan Neuling wird vernichtet.
Lehrer*in Sabine Sauber wird vernichtet.
Person Sabine Sauber wird vernichtet.
Lehrer*in Willi Witzig wird vernichtet.
Person Willi Witzig wird vernichtet.
Dozenten freigegeben ok
Fächer freigegeben ...
Studienfach OOP wird vernichtet.
Studienfach Mathe III wird vernichtet.
Studienfach Englisch wird vernichtet.
Studienfach Systemprogrammierung wird vernichtet.
Fächer freigegeben ok
Belegungen freigegeben ...
Belegung Nr. 1 wird vernichtet.
Belegung Nr. 2 wird vernichtet.
Belegung Nr. 3 wird vernichtet.
Belegung Nr. 4 wird vernichtet.
Belegung Nr. 5 wird vernichtet.
Belegung Nr. 6 wird vernichtet.
Belegungen freigegeben ok