

TEIL I Modellierung und Realisierung einer Datenbank

Aufgabenbeschreibung

Im Datenbankpraktikum wollen wir uns mit der Vorlesungsplanung der FH beschäftigen.

Ihnen ist der Sachverhalt aus studentischer Sicht (Vorlesungspläne) bestens bekannt, so dass auf eine detaillierte Beschreibung der Ist-Situation verzichtet werden kann. Dabei soll nicht nur die studentische Sicht, sondern auch das Deputat der Dozenten, die Raumplanung und -belegung berücksichtigt werden.

Wir wollen zunächst verschiedene konzeptionelle Datenmodelle dazu entwickeln und diese anschließend in Form einer datenbankgestützten Anwendung umsetzen.

Aufgabe 1 (Datenmodellierung)

- a) Erstellen Sie ein Entity-Relationship-Modell für die o. g. Vorlesungsplanung. Gehen Sie systematisch vor, indem Sie zuerst die Entitäten, dann ihre Attribute und zuletzt die Beziehungen (Relationships) definieren. Als Ergebnis sind ein ER-Diagramm und eine Attributliste mit Konsistenzbedingungen verlangt.
- b) Entwerfen Sie ein objektorientiertes Modell der Vorlesungsplanung.
Erstellen Sie zuerst mit Hilfe von „Together“ (das UML-Java-Entwicklungswerkzeug von TogetherSoft) ein Hierarchiediagramm für die Fachklassen, dann Kollaborations- und Sequenz- (Transaktions-)diagramme für wenigstens eine nichttriviale Methode (z.B. Belegung eines Raumes vornehmen). Achten Sie darauf, dass Ihr Modell die notwendige Präzision aufweist und die Navigationsmöglichkeiten bietet, welche Ihre Anwendung benötigt, z.B. wenn Ihr Programm die Vorlesungen eines Dozenten anzeigen soll, benötigen Sie eine Objektreferenz zu den Vorlesungen, wenn Sie auch den Dozenten einer Vorlesung ermitteln wollen, muss auch ein Rückverweis angelegt werden.

Aufgabe 2 (relationaler Datenbankentwurf)

Überblick: Setzen Sie das ER-Modell von Aufgabe 1a in einen relationalen Datenbankentwurf um und implementieren Sie ihn in Access.

Aufgaben:

- a) Transformieren Sie das ER-Modell aus Aufg. 1a in ein relationales.
Hinweis: Komplexe Objekte/Entitäten/Beziehungen sind zu normalisieren.
- b) Erstellen Sie eine Access-Datenbank.
Hinweis: Benützen Sie die Entwurfsansicht (Tabellen/Neu/Entwurfsansicht) zur Definition der Tabellen und ihrer Attribute. Konsistenzbedingungen können in das Feld Gültigkeitsregel eingetragen werden. Der Ausdruckseditor (Schaltfläche [...] rechts neben dem Eingabefeld) hilft Ihnen bei der Erstellung syntaktisch korrekter Bedingungen. Die Beziehungen können am anschaulichsten in der Beziehungsansicht (Schaltfläche: Beziehungen), d.h. im Bachman-Diagramm erstellt werden.
- c) Fügen Sie in der Tabellenansicht so viele Testdaten ein, dass Sie wenigstens einen Belegungsplan für Ihr Semester erstellen können.
Hinweise: Sie können auch Masken mit dem Assistenten (Formulare/Neu/Formular-Assistent) entwerfen, die Daten aus mehreren Tabellen enthalten. Damit ist es möglich, über Fremdschlüssel verbundene Tabelleninhalte hierarchisch mit einer Kopf-Zeile und mehreren Detail-Zeilen darzustellen.

Dies kann dazu verwendet werden, für einen Raum (Kopf-Zeile), die Belegung in Listenform (Detailzeilen, 1. Zeile pro Zeitblock) darzustellen. Als Eingabehilfe können Felder aus der über einen Fremdschlüssel verbundenen Stammdatentabelle in einem Listen- oder Kombinationsfeld ausgewählt werden.

- d) Erstellen Sie ein Formular für die Raumbellegung bei dem die zulässigen Werte aus einem Auswahlfeld ausgewählt werden können.
- e) Erzeugen Sie ein Ausgabeformular für die Vorlesungspläne der Studenten in Matrixform (Wochentage horizontal, Zeit vertikal, Matrixelemente enthalten Vorlesung, Dozent und Raum)
Hinweis: Benützen Sie eine Pivottabelle.

Aufgabe 3 (objektorientierter Datenbankentwurf)

Implementieren Sie das Modell von Aufgabe 1b in der objektorientierten Datenbank von Poet (ObjectServer).

- a) Ergänzen Sie das oo-Modell von 1b um technische Belange (z.B. Containerklassen).

Hinweise: Ausgehend vom Modell in 1b können Sie die generierten Klassendefinitionen weiterverwenden. Die Klassendefinitionen sind mit einem Editor oder einem der integrierten Entwicklungswerkzeuge (IDEs) „Together“ oder „Eclipse“ um Zugriffsmethoden (setXX, getXX) und um Objekt-Beziehungen zu ergänzen. Für Mehrfachbeziehungen (1:n, n:m) verwenden Sie die persistenten Container-Klassen (SetOfObject, BagOfObject, etc.) von Poet. Dazu sind in „Together“ oder „Eclipse“ die Klassenbibliotheken POET6ODMG3JC_SDK.jar und POET6ODMG3JC_Runtime.jar von Poet einzutragen, damit die Klassen gefunden werden.

- b) Erstellen Sie das Objektschema für Poet.

Hinweise: In der Dateivorlage „*ptj.opt*“ müssen die persistenten Klassen eingetragen werden. Der Poet Enhancer „*ptj.exe*“ verwendet diese Steuerdatei, um die Klassen ‚persistent zu machen‘ und das Object Schema für die Objektdatenbank zu generieren. Die Datenbankstruktur kann mit dem Poet „*Developer*“ untersucht werden.

- c) Fügen Sie so viele Testdaten ein, dass Sie wenigstens einen Belegungsplan für Ihr Semester erstellen können.

Hinweise: Wenn die Struktur der OODB angelegt ist, können Sie mit dem Poet *Developer*-Tool auch Testobjekte für Ihren Semesterstundenplan anlegen. Alternativ kann die DB mit Hilfe eines kleinen Testprogramms mit Objekten geladen werden. Dazu erstellen Sie einfache ‚Ladeprogramme‘ in Java, um Objekte zu erzeugen und damit die Datenbank zu füllen. Das programmatische Laden der Datenbank ist deshalb vorzuziehen, da die Mehrfachbeziehungen zwischen den Objekten nur so eingetragen werden können.