



[Gymnasium Odenkirchen]

Datenbanken in der 11.1

- Thorsten Reinsch
[Gymnasium Odenkirchen]
- Einführungsvortrag zum Workshop "Datenbanken in der 11.1" auf dem Informatiktag NRW am 29. März 2004 in Siegen.
- Der Workshop stellt eine Möglichkeit vor Datenbanken bereits in der 11.1 vorzustellen.



[Gymnasium Odenkirchen]

Wieso Datenbanken in der 11.1?

- Lernen im Kontext
- viele Schüler wählen nach der 11 ab
- besseres Bild der Informatik als die Algorithmik in einem Schuljahr vermitteln kann
- zwei verschiedene Paradigmen
- Anknüpfungspunkte zum Themengebiet „Informatik und Gesellschaft“
- Stärkung der Allgemeinbildung



[Gymnasium Odenkirchen]

Bisherige Themen streichen

- Einführung in die IDE
 - „Hello World“
 - Schleifen, Abfragen
 - Array, Record
 - Sortieren
 - ~~Algorithmen~~
 - ~~Pointer~~
 - ~~Units~~
 - ~~Lineare Liste~~
- Datenbanken, aber ...



[Gymnasium Odenkirchen]

Nicht überfrachten!

- Keine Datenbankvorlesung!
- Blackbox (Bibliothek) benutzen
- Konzept des Spiralcurriculums nutzen
- z. B.:
 - keine Tabellen erzeugen oder Löschen
 - keine Transaktion
- statt dessen:
 - einfügen, ändern und löschen von Datensätzen



[Gymnasium Odenkirchen]

Bsp.: Buchungssystem

- Tabelle mit Flügen
- Tabelle mit Kundendaten
- Tabelle die Kunden Flüge zuordnet

- kleines Programm, dass Flüge anzeigt und Kunden Flüge buchen lässt.



[Gymnasium Odenkirchen]

Neu aber doch alt...

- statt Lineare Liste selbst zu entwickeln Datenbankschnittstelle nutzen

- Vorteile:
 - realitätsnahe Beispiele
 - Softwareentwicklung: vorhandene Bibliotheken benutzen
 - Chancen und Risiken von Informatiksystemen erkennen



[Gymnasium Odenkirchen]

Umsetzung: Server

- Datenbankserver:
 - z.B. Linux mit Apache, Mysql, PHP (LAMP)
 - alle aktuellen Distributionen lassen dies per Knopfdruck installieren

- Funktionierendes Netzwerk

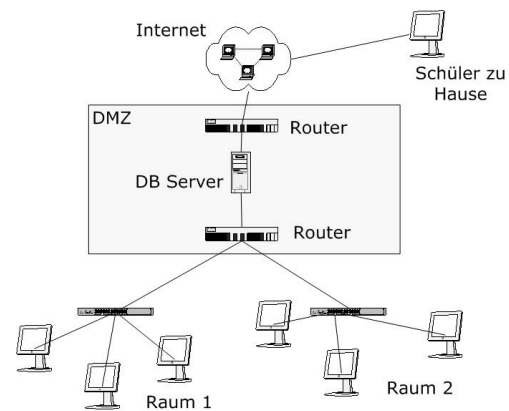


[Gymnasium Odenkirchen]

Umsetzung: Client

- Für Mysql gibt es viele Schnittstellen:
 - Delphi
 - Free Pascal
 - Java
 - ODBC

Bonus: DynDNS



Structured Query Language

- normierte Abfragesprache
- Industriestandard
- DDL (Data Definition Language)
- DML (Data Manipulation Language)

Auswählen: *SELECT*

```
SELECT spalte1 [,spalte2] FROM tabelle1 [,tabelle2]
WHERE bedingung
ORDER BY spalte1 [,spalte2]
```

```
SELECT COUNT(*) FROM tabelle
```

Einfügen: *Insert*

```
INSERT INTO tabelle VALUES(spalte1,...,'spalteN')
```



Löschen: *delete*

DELETE FROM tabelle WHERE bedingung

Achtung: Ohne Bedingung wird die ganze Tabelle gelöscht!



Ändern: *update*

UPDATE tabelle SET spalte1=wert WHERE bedingung

Achtung: Ohne Bedingung wird jede Zeile der Spalte auf den neuen Wert gesetzt!