

# PHP & MySQL

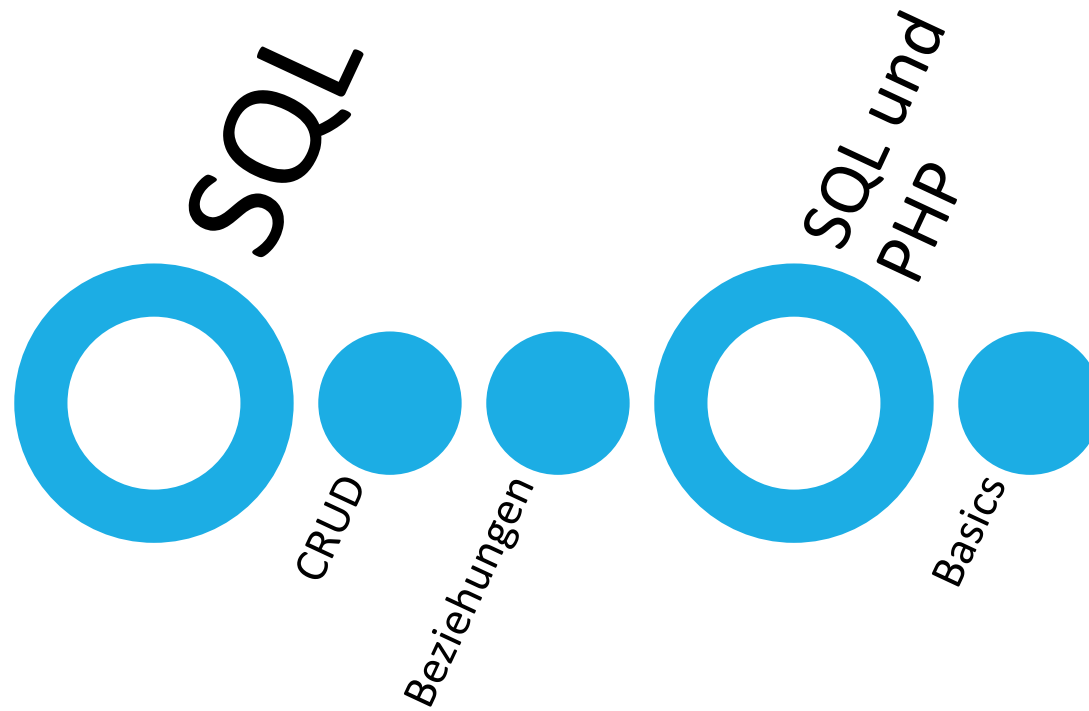
---

DYNAMISCHE WEBSITES ERSTELLEN

TERRY HARKER, [TERRY.HARKER@BYTEKULTUR.NET](mailto:TERRY.HARKER@BYTEKULTUR.NET)

# Programm diese Woche

---



# Kompetenz-Ziel

---

Arbeiten mit Datenstrukturen unter Verwendung von MySQL  
Datenmanipulations Statements

## **Kenntnisse und Fertigkeiten**

- Struktur und verwalten und exportieren
- Beziehungen zwischen Tabellen / Spalten
- MySQL Statements Syntax und CRUD

***"Bad programmers worry about the code.  
Good programmers worry about data  
structures and their relationships."***

**- Linus Torvalds**



# MySQL Datenbanken

---

AUFBAU UND VERWALTUNG

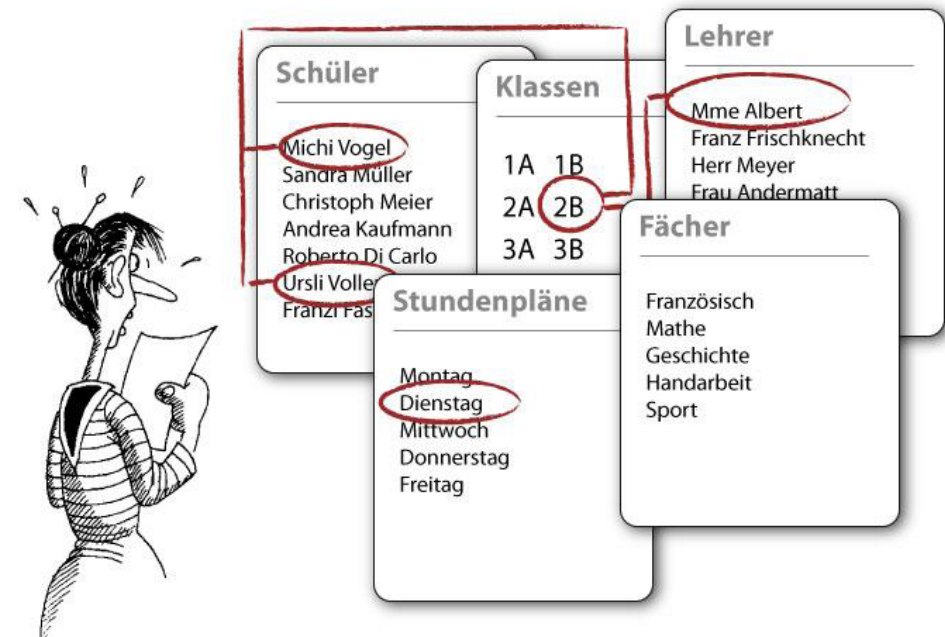
# MySQL – ein relationales DB-System

## Gruppierung der Daten

- Grosse Datenmengen effizient ablegen

## Beziehung statt Duplikation

- bessere Performance
- weniger Fehler

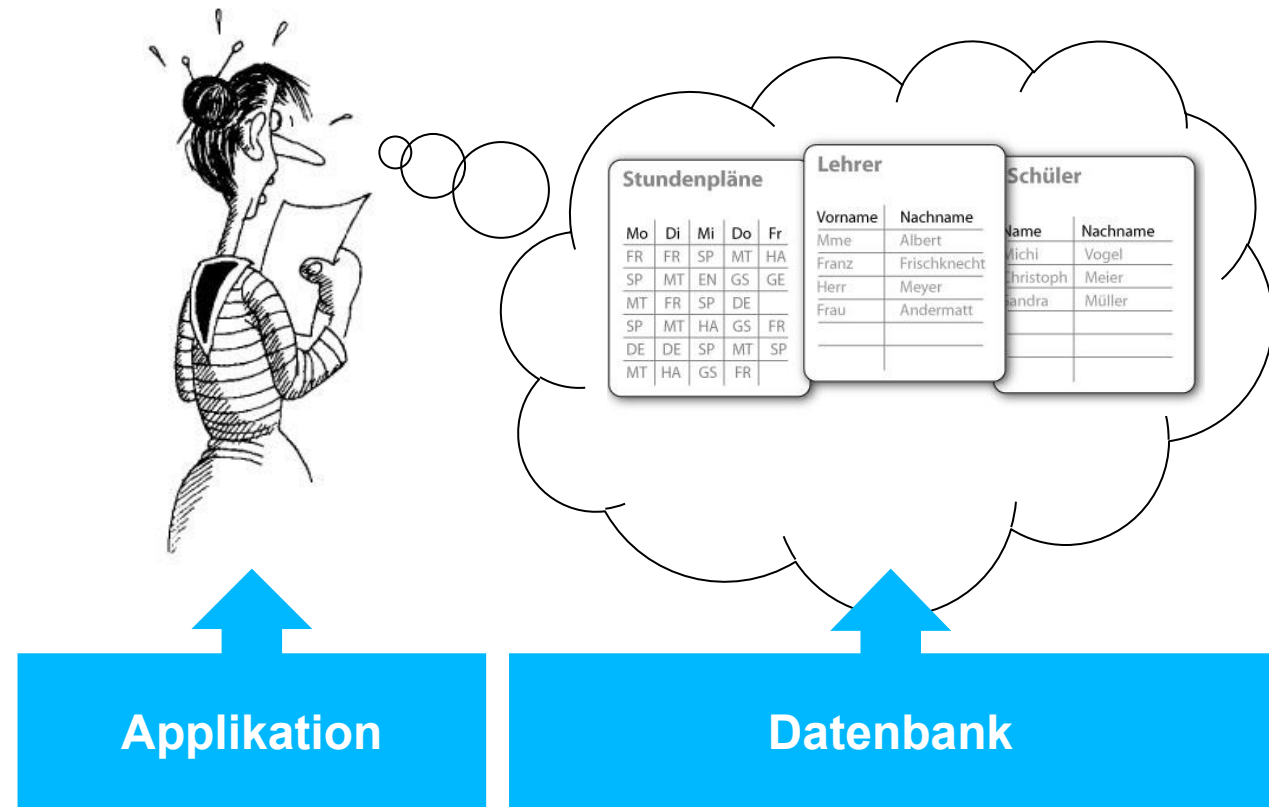


# Konzept DB Basierte Applikation

## Auslagerung der Daten

- Unabhängigkeit vom Layout
- Unabhängigkeit von der Programm-logik
- Erfassen durch das Programm / den Besucher (user generated Content)

*Datenbank = «Gedächtnis» der Applikation*



# «Tabellen haben immer eine Index-Spalte»

---

SIE ENTHALTEN ANZAHL DATENSPALTEN + 1



# Beziehungen

---

WIE EIN RELATIONALES DATENBANKSYSTEM GENUTZT WERDEN  
KANN, UM REDUNDANZ ZU VERMEIDEN

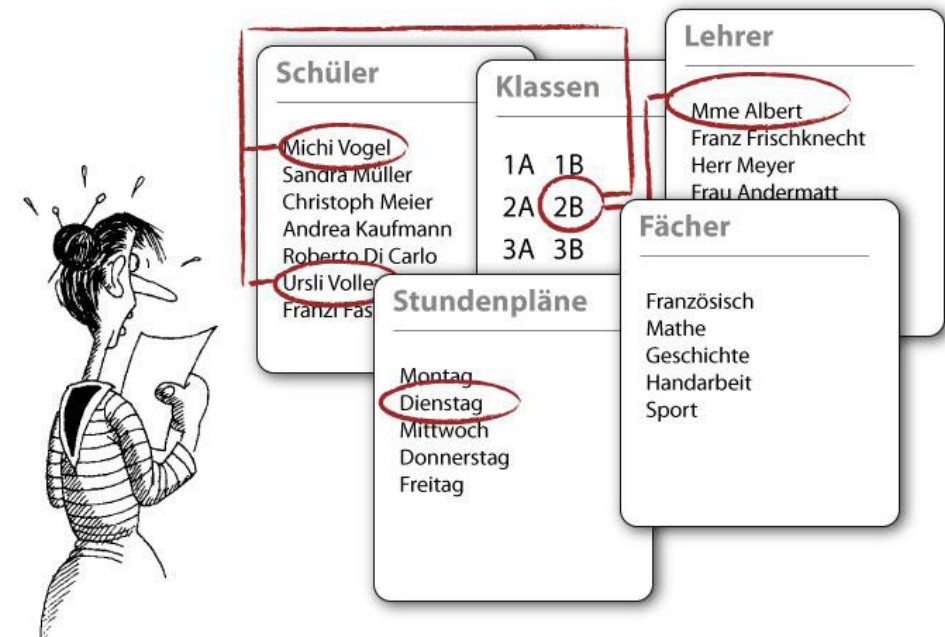
# Zur Erinnerung: relationales DB-System

## Gruppierung der Daten

- Grosse Datenmengen effizient ablegen

## Beziehung statt Duplikation

- bessere Performance
- weniger Fehler



# Verbindung speichern

---

- Statt alles in einer Tabelle zu speichern, Daten aufteilen und Beziehung festlegen
- Verschiedene Beziehungen zwischen Daten möglich.

**TabelleA**

ID	name	vorname
1	Vogel	Michi
2	Müller	Sandra

**TabelleB**

ID	klasse
7	WDD313
8	WDD919
9	WDD320

# 1:1 Beziehungen (one-to-one)

- 1 Eintrag in Tabelle A ist genau mit 1 Eintrag aus Tabelle B verbunden
- 1 Eintrag in Tabelle B ist genau mit 1 Eintrag aus Tabelle A verbunden

ID	name	adresse_id
7	Fritzli Müller	8
8	Hans Mustermann	
9	Maria Muster	

**Primärschlüssel**  
= *primary key (pk)*

**Fremdschlüssel**  
= *foreign key (fk)*

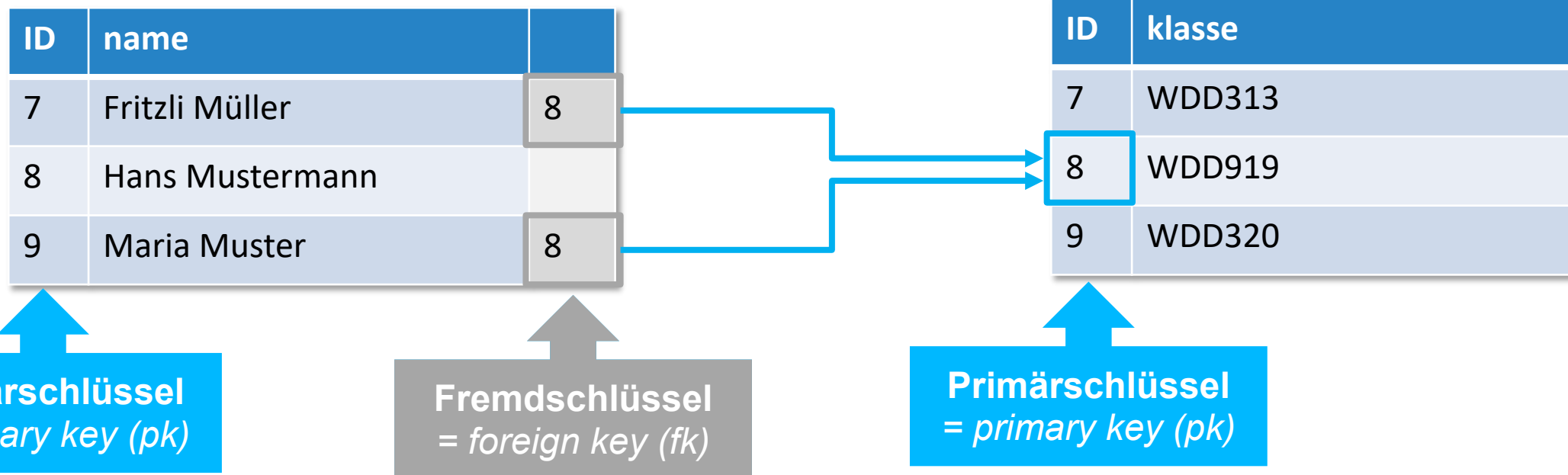
ID	adresse
7	Musterstrasse 5
8	Hauptstrasse 1
9	Bahnhofstrasse 10

**Primärschlüssel**  
= *primary key (pk)*

# 1:Nbeziehungen (one-to-many)

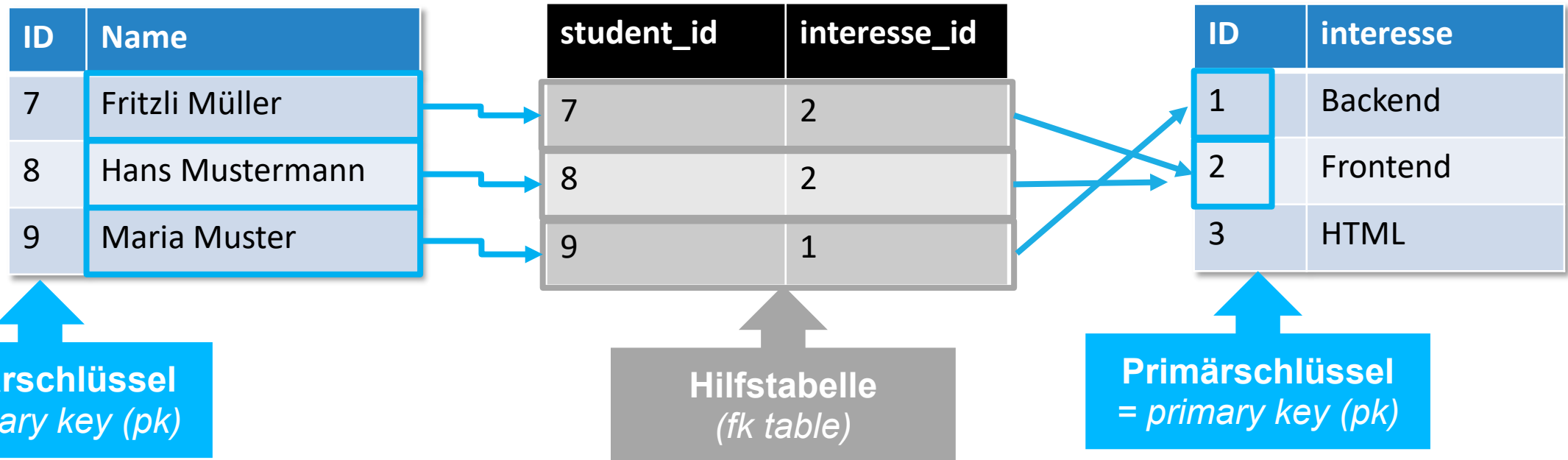
1 Eintrag in Tabelle A ist genau mit 1 Eintrag aus Tabelle B verbunden

1 Eintrag in Tabelle B kann mit **N** (variable zahl) Einträgen aus Tabelle A verbunden sein



# N:M Beziehungen (many-to-many)

- 1 Eintrag in Tabelle A kann mit **N** (variable zahl) Einträgen aus Tabelle B verbunden sein
- 1 Eintrag in Tabelle B kann mit **M** (variable zahl) Einträgen aus Tabelle A verbunden sein



# Fragestellung beim Aufbau

---

1. Wie viele Einträge aus A müssen B zugewiesen werden können?
2. Wie viele Einträge aus B müssen A zugewiesen werden können?
3. **Skalierbarkeit:** könnte sich dies mit wachsendem Projekt ändern?

# Links / Weitere Infos

---

## MySQL Referenz:

- MySQL Datentypen: <https://dev.mysql.com/doc/refman/5.7/en/data-types.html>

## GUI Tools für MySQL:

- Online meist vorhanden: <https://www.phpmyadmin.net>
- Workbench (div. OS): <https://dev.mysql.com/downloads/workbench>
- Sequel Pro (Mac): <http://www.sequelpro.com>
- DB Diagram.io (Online): <https://dbdiagram.io>