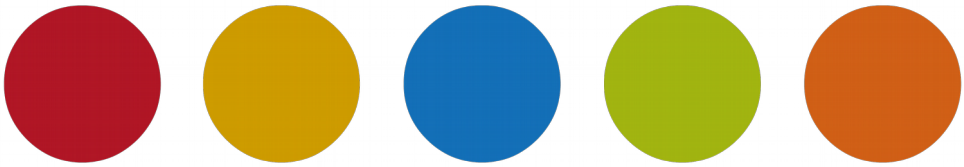
Materialsammlung Relationale Datenbanken (RDB)

Skript

Arbeitsmaterial

Schulung:	Informatik und Wirtschaftsinformatik

Stand: 19. Apr 2020



© Christine Janischek

Christine Janischek
Berufliches Schulzentrum Wangen

https://edublog.emotionalspirit.de/

Inhaltsverzeichnis

1 Grundlagen....3
2 Modellierung Relationaler Datenbanken....9
3 Begriffe im Kontext Relationaler Datenbanken....18
4 Entwicklung Relationaler Datenbanken....19
5 Structured Query Language (SQL)....22
6 Normalisierung....41

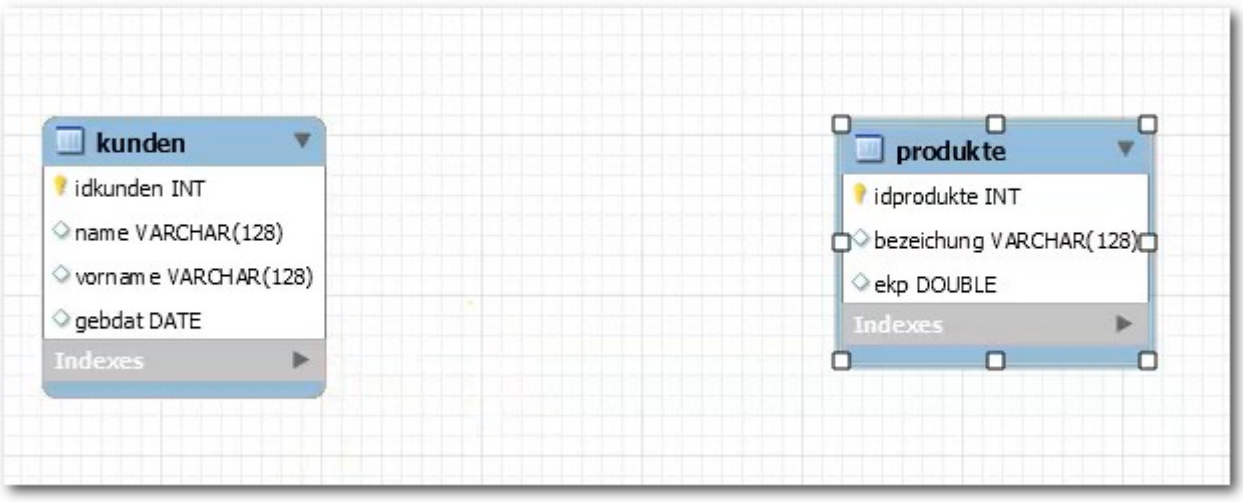
CC BY NC SA

1 Grundlagen

Grundlagen

CC BY NC SA

Thema:	Relationale Datenbanken Arbeitsauftrag – Grundbegriffe Relationale Datenbanken Quelle: Christine Janischek



webshop_db

Arbeitsauftrag:

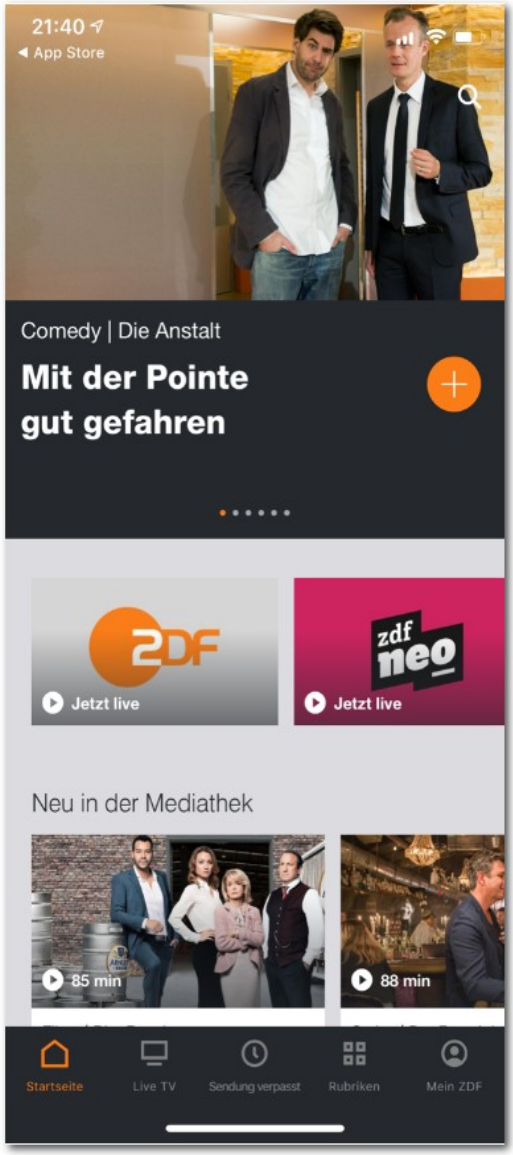
1. Wie erfolgt Heutzutage die Verwaltung vielen Daten?
2. Welche Daten verwaltet ein Webshop noch?
3. In welchem Zusammenhang stehen diese Daten?
4. Klären Sie Ihnen unbekannte Begrifflichkeiten!

Thema:	Einführung in die Softwareentwicklung Urquelle: Christine Janischek Webprogrammierung und native Anwendungsentwicklung

Web Apps

ZDF-Mediathek Web-App

Native Mobile-Apps



ZDF-Mediathek Mobile-App

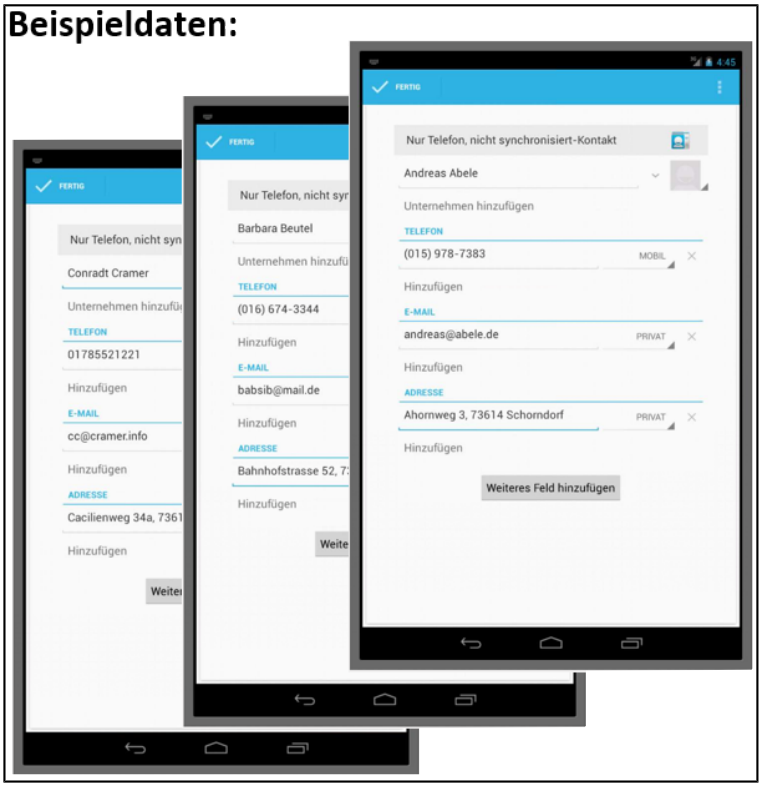
Arbeitsauftrag:

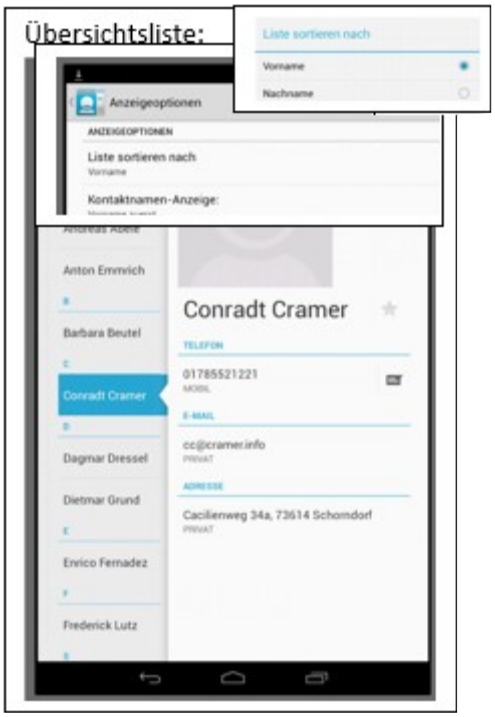
5. Welche Daten verwaltet das ZDF in der Mediathek?
6. In welchem Zusammenhang stehen diese Daten?
7. Wie erfolgt Heutzutage die Verwaltung vielen Daten?
8. Klären Sie Ihnen unbekannte Begrifflichkeiten!

Thema:	Relationale Datenbanken Arbeitsauftrag – Einführung Relationale Datenbanken Quelle: L1 1 Entwurf einer Tabelle erstellen

L1_1 Entwurf einer Tabelle erstellen – Aufgaben

Heiner Blechle hat sich seinen beruflichen Traum erfüllt: Er hat seine Fahrlehrerprüfung bestanden und macht sich nun mit einem gebrauchten Fahrschulauto selbstständig. Über seinen Sportverein hat er zu Jugendlichen und jungen Erwachsenen bereits Kontakte geknüpft, die bei ihm die Fahrschule besuchen möchten. Deren Namen und Kontaktdaten hat er vorläufig auf seinem Smartphone gespeichert:





https://edublog.emotionalspirit.de/

Heiner Blechle stellt bald fest, dass die Speicherung der Daten seiner Kunden allein auf dem Handy unpraktisch ist. So braucht er diese Informationen regelmäßig für Meldungen an die Führerscheinstelle bei der Kreisverwaltung, den TÜV für Fahrprüfungen aber auch für die Erstellung von Rechnungen. Außerdem plant er, zu Werbezwecken entsprechende Infopost zu verschicken.

Alexander, ein Freund von Heiner aus dem Sportverein, studiert Wirtschaftsinformatik.

„Wenn du die Daten deiner Kunden nicht nur zum Telefonieren nutzen willst, brauchst du eine Datenbank. Das ist auch gar nicht so schwierig. Du musst nur eine Datenbanktabelle erstellen und die Daten, die du im Smartphone gespeichert hast, dort erfassen. Für den Entwurf einer Datenbanktabelle orientierst du dich am besten an folgendem Entwurfsraster."

Entwurfsraster einer Tabelle

Tabellenname:								
Attributname								
Datentyp								
max. Zeichenanzahl								

Aufgaben:

Beachten Sie zur Bearbeitung der nachfolgenden Aufgabenstellungen das Informationsmaterial *L1_1 Information Tabellenentwurf.docx.*

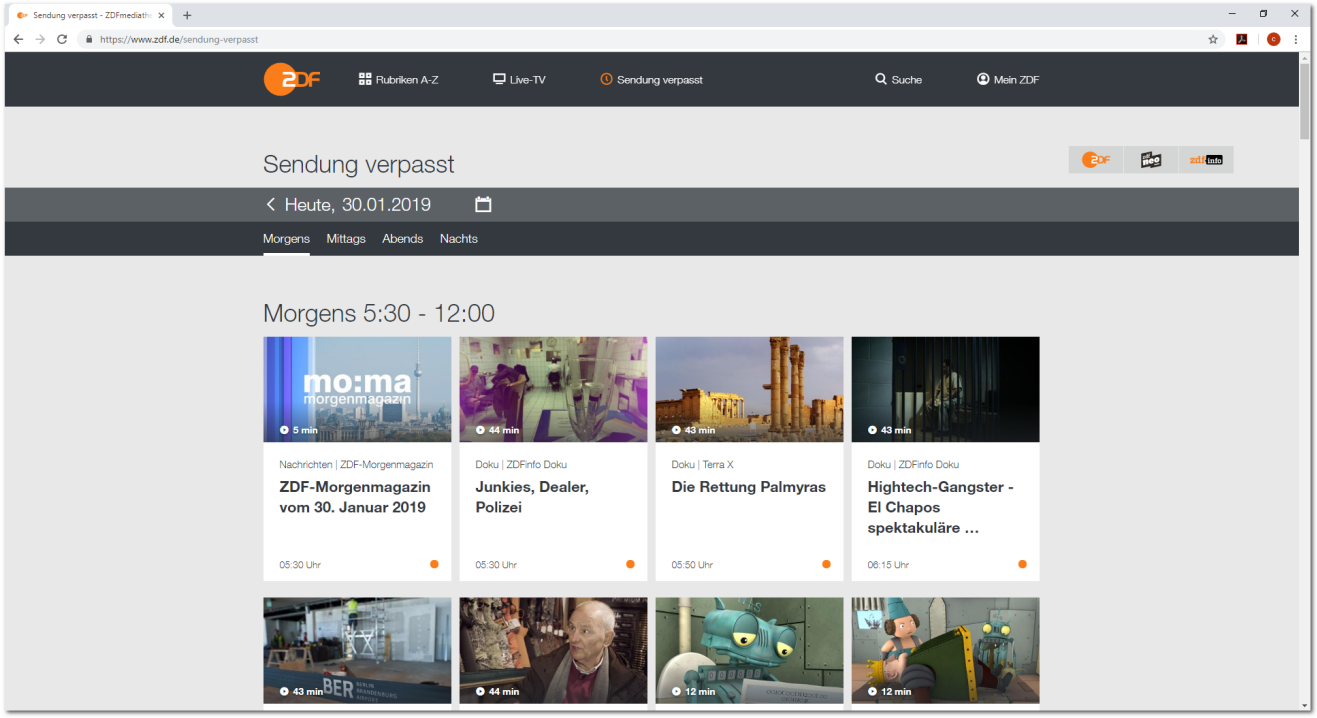
1 Formulieren Sie jeweils eine Definition für folgende Begriffe:

- Datensatz
- Primärschlüssel
- Attribut
- Attributwert

2 Erstellen Sie für Heiner Blechle nach dem abgebildeten Entwurfsraster einen Tabellenentwurf für die Datenbanktabelle *fahrschueler*. Verwenden Sie die Entwurfsvorlage *L1_1 Vorlage Tabellenentwurf.docx*.

Thema:	Einführung in Relationale Datenbanken Urquelle: Christine Janischek ZDF Mediathek - Strukturierung von Daten und deren Zusammenhänge

Responsive Webdesign



ZDF-Mediathek online

Arbeitsauftrag:

1. Nennen Sie alle wesentlichen Bestandteile (Daten), die das System ausmachen!
2. Welche Eigenschaften weisen die einzelnen Bestandteile auf?
3. Welcher mengenmäßige Zusammenhang ergibt sich für die Daten?
4. Fassen Sie die Erkenntnisse der Klasse in der Dokumentation zusammen.

2 Modellierung Relationaler Datenbanken

Modellierung
Relationaler Datenbanken

CC BY NC SA

Thema:	Relationale Datenbanken Arbeitsauftrag Quelle: L1 2 Datenbank mit einer Tabelle modellieren

L1_2 Datenbank mit einer Tabelle modellieren – Aufgaben

Heiner Blechle zeigt seinem Freund Alexander den von Ihnen entwickelten Tabellenentwurf. Er empfiehlt ihm, den Entwurf um zwei weitere Attribute zu erweitern, um das Geburtsdatum der Fahrschüler (geburtsdatum) und die Anzahl der Fahrstunden (fahrstundenzahl) erfassen zu können.

Alexander erklärt, dass der erweiterte Entwurf nun mit Hilfe einer Software modelliert und in eine Datenbank überführt werden muss. Er schlägt vor, die Software MySQL Workbench zu verwenden. Hierbei handelt es sich um ein grafisches Entwicklungssystem zur Erstellung, Bearbeitung und Verwaltung von Datenbanken.

Beachten Sie zur Bearbeitung der folgenden Aufgabenstellungen das Informationsmaterial L1_2.1 Information Datenbank modellieren.docx.

1 Welche Aufgaben hat ein Entity-Relationship-Diagramm?

2 Erstellen Sie ein Entity-Relationship-Diagramm für die Daten der Fahrschüler.

3 Welche Aufgaben erfüllt ein Relationenmodell?

4 Erstellen Sie auf der Grundlage des entwickelten Entity-Relationship-Diagramms ein entsprechendes Relationenmodell.

Beachten Sie zur Bearbeitung der folgenden Aufgabenstellung das Informationsmaterial L1_2.2 Information Datenbank softwaregestützt modellieren.docx.

5 Erstellen Sie ein Datenmodell für die Datenbank fahrschule mit der Tabelle fahr schueler softwaregestützt mit Hilfe der MySQL Workbench.

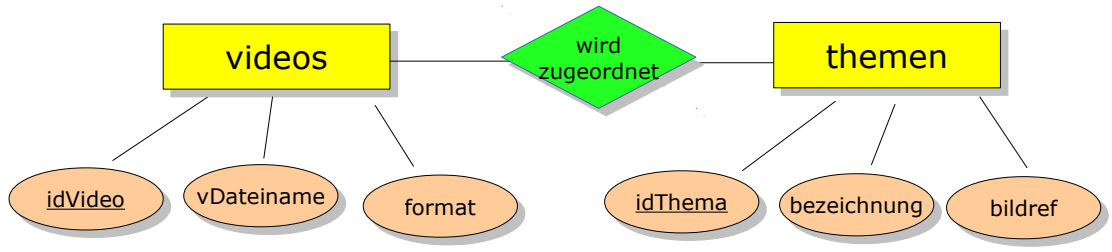
Speichern Sie Ihr Modell unter dem Namen 'L1_2 Lösung fahrschule.mwb'.

https://edublog.emotionalspirit.de/

Thema:	Einführung in die Modellierung einer relationalen Datenbank Urquelle: Christine Janischek Modellierung: ZDF Mediathek - Datenbanktabellen und Beziehungen

In jeder Datenbanktabelle werden Daten (konkrete Werte) gespeichert. Damit Ordnung herrscht werden die Daten in Spalten organisiert. Jede Spalte erhält eine genaue Bezeichnung und einen Wertebereich. **Die Architektur einer Anwendung muss stimmen!**

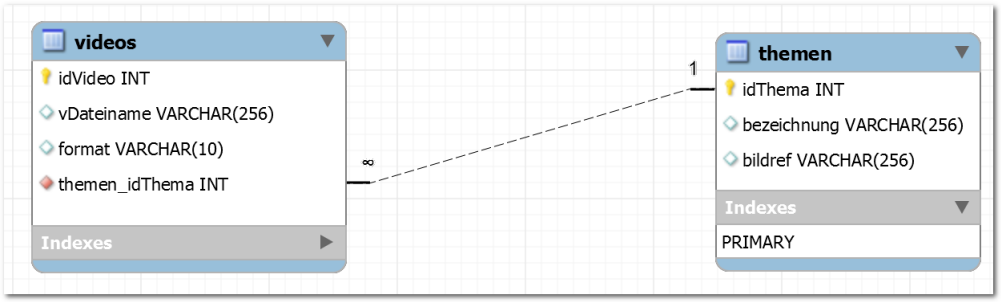
Entity Relatioship Modell (ERM)



Mengenmäßiger Zusammenhang (EINE-Sätze):

1. Ein Video wird genau **einem** Thema zugeordnet **[1]**.
2. Zu einem Thema werden ein oder **mehrere** Videos zugeordnet **[N]**.

Enhanced Entity Relatioship Modell (EERM)



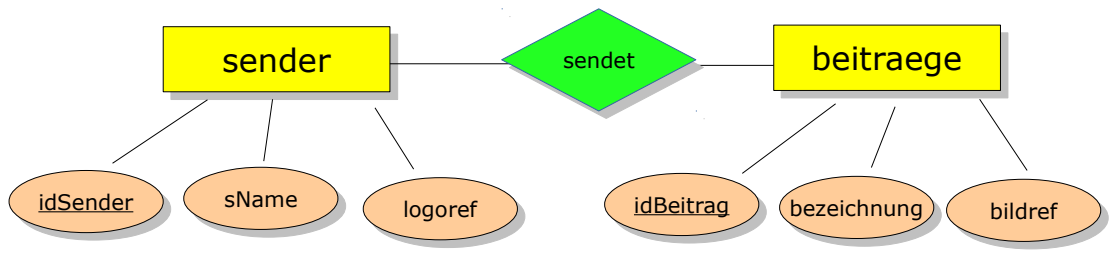
Arbeitsauftrag:

1. Welche Eigenschaften können wir den Datenbanktabellen (Entitätstypen) Videos und Temen eindeutig zuordnen?
2. Welche Art von Werten sollen gespeichert werden?
3. Welcher mengenmäßige Zusammenhang besteht zwischen den Datensätzen in den beiden Tabellen?
4. Modellieren Sie diesen Zusammenhang mit Hilfe mit einer geeigneten Entwicklungsumgebung.
5. Fassen Sie die Erkenntnisse der Klasse in der Dokumentation zusammen.

Thema:	Einführung in die Modellierung einer relationalen Datenbank Urquelle: Christine Janischek Modellierung: ZDF Mediathek - Datenbanktabellen und Beziehungen

Wie wir haben festgestellt, dass der mengenmäßige Zusammenhang zwischen den Datensätzen der Datenbanktabellen eine wesentliche Rolle spielt. Ohne die verbindenden Attribute können Such-, Änderungs- und Löschoperationen auf unserem System nicht erfolgreich ausgeführt werden. Im Kontext Relationaler Datenbanken heißt der mengenmäßige Zusammenhang zwischen den Datensätzen der Tabellen Kardinalität. Wir unterscheiden drei mögliche Hauptvarianten. Finden Sie heraus, welche Variante im unten genannten Fall vorliegt und welche Konsequenzen sich für die Struktur ergeben. **Die Architektur einer Anwendung muss stimmen!**

Entity Relatioship Modell (ERM)



Mengenmäßiger Zusammenhang (EINE-Sätze):

1. Ein Sender _____ [ ].
2. Ein Beitrag _____ [ ].

Enhanced Entity Relatioship Modell (EERM)

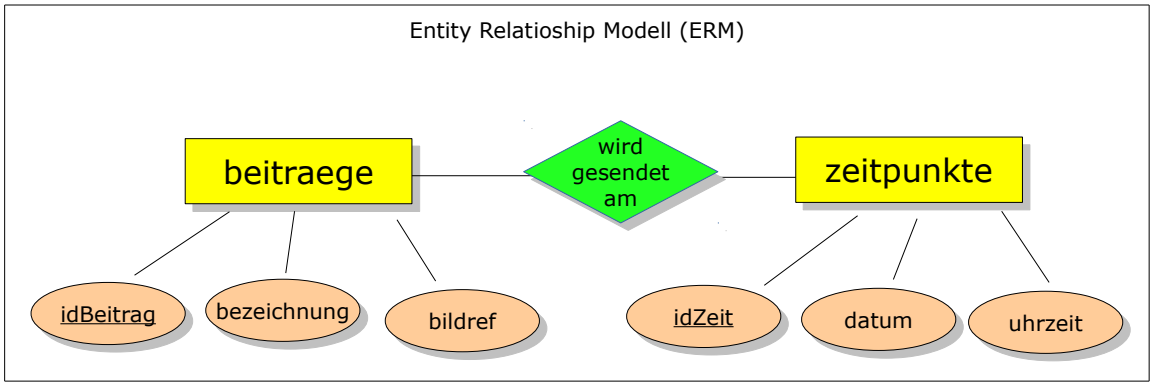
?

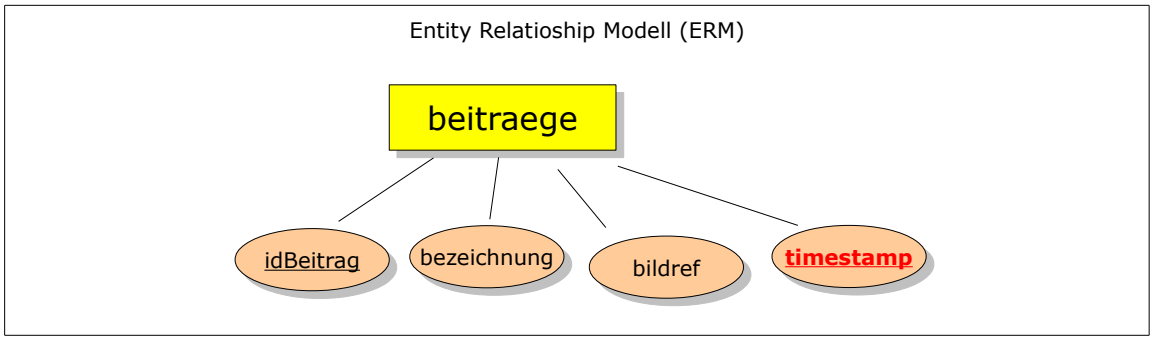
Arbeitsauftrag:

1. Welche Eigenschaften können wir den Datenbanktabellen (Entitätstypen) Sender und Beitrag eindeutig zuordnen?
2. Welche Art von Werten sollen gespeichert werden?
3. Welcher mengenmäßige Zusammenhang besteht zwischen den Datensätzen in den beiden Tabellen?
4. Modellieren Sie diesen Zusammenhang mit Hilfe mit einer geeigneten Entwicklungsumgebung.
5. Fassen Sie die Erkenntnisse der Klasse in der Dokumentation zusammen.

Thema:	Einführung in die Modellierung einer relationalen Datenbank Urquelle: Christine Janischek Modellierung: ZDF Mediathek - Datenbanktabellen und Beziehungen

Die meisten Datenbanken sind Heutzutage immer noch → Relationale Datenbanken. Das hat hauptsächlich zwei Gründe. Zum Einen rückt die Beziehung (engl. Relation) zwischen Daten ins Zentrum der Betrachtung, damit werden **Mehrfachnennungen** und **Widersprüche** vermieden. Zum Anderen werden aktuell bereits riesige Datenmengen in relationalen Systemen verwaltet. Das konvertieren (strukturelle und sprachliche Anpassung) bestehender Datenmengen an neuartige Systeme wäre sehr kostenintensiv, da die Lösungen in der Regel sehr spezifisch und Datenbankspezialisten sehr teuer sind. **Die Architektur einer Anwendung muss stimmen!**





Arbeitsauftrag:

1. Welche Eigenschaften können wir den Datenbanktabellen (Entitätstypen) Beitrag und Zeitpunkt eindeutig zuordnen?
2. Welche Art von Werten sollen gespeichert werden?
3. Welcher mengenmäßige Zusammenhang besteht zwischen den Datensätzen in den beiden Tabellen?
4. Modellieren Sie diesen Zusammenhang mit Hilfe mit einer geeigneten Entwicklungsumgebung.
5. Fassen Sie die Erkenntnisse der Klasse in der Dokumentation zusammen.

Thema:	Einführung in die Modellierung einer relationalen Datenbank Urquelle: Handreichung: J1, BPE 6 Relationale Datenbanken 2018/2019 Information: Schlüsselattribute, Atomare Attributwerte und Redundanz

Ein funktionierendes relationales Datenbankmodell verlangt u.a., dass folgende Kriterien beachtet werden:

Primärschlüsselattribut: Ein Primärschlüssel wird zur eindeutigen Identifizierung eines Datensatzes verwendet. Der Wert eines Primärschlüssels muss in einer Tabelle einmalig sein, da er jeden Datensatz eindeutig kennzeichnet.

Atomare Attributwerte: Einem Attribut dürfen nicht mehrere Werte zugewiesen werden, d.h. in einem Datenfeld darf nur ein Wert enthalten sein.

Redundanzfreiheit: Unter Redundanz versteht man eine doppelte bzw. mehrfache Speicherung gleicher Daten. Redundante Daten führen zu einem unnötigen Zeitaufwand bei der Erfassung, zur Verschwendung von Speicherplatz und zu erhöhten Verarbeitungszeiten. Außerdem können sie zu widersprüchlichen Daten führen, wenn eine Änderung einer Information nicht an allen Stellen der Datenbank vorgenommen wird. In diesem Fall spricht man von einer Änderungs-Anomalie, die zur Inkonsistenz der Daten führt.

Ziel der Datenbankmodellierung muss es sein, solche Redundanzen zu vermeiden.

Redundanzfreiheit liegt dann vor, wenn sämtliche Daten nur einmal erfasst und gespeichert werden, ohne dass ein Informationsverlust stattfindet.

Redundanzfreiheit kann erreicht werden, indem die redundanten Daten in eigene Tabellen ausgelagert werden.

CC BY NC SA

Thema:	Übung: Modellierung einer Relationalen Datenbank Urquelle: Christine Janischek Kardinalitäten, Entity Relationship Modell (ERM) und Enhanced Entity Relationship Modell (EERM)

Anwendungsfall: Online-Shop für Mobiltelefone

In einem Onlineshop für Mobiltelefone beraten Mitarbeiter jeden Tag viele Kunden über eine Telefonhotlein. Die Kunden werden dazu mit dem nächsten freien Mitarbeiter verbunden.

Die Mobiltelefone sind immer mit einem Betriebssytem ausgestattet.

Kunden können die Mobiltelefone bestellen.

Die Mobiltelefone werden direkt beim Hersteller bestellt.

Nutzen Sie die folgenden Angaben um die Aufgabenstellung zu lösen.

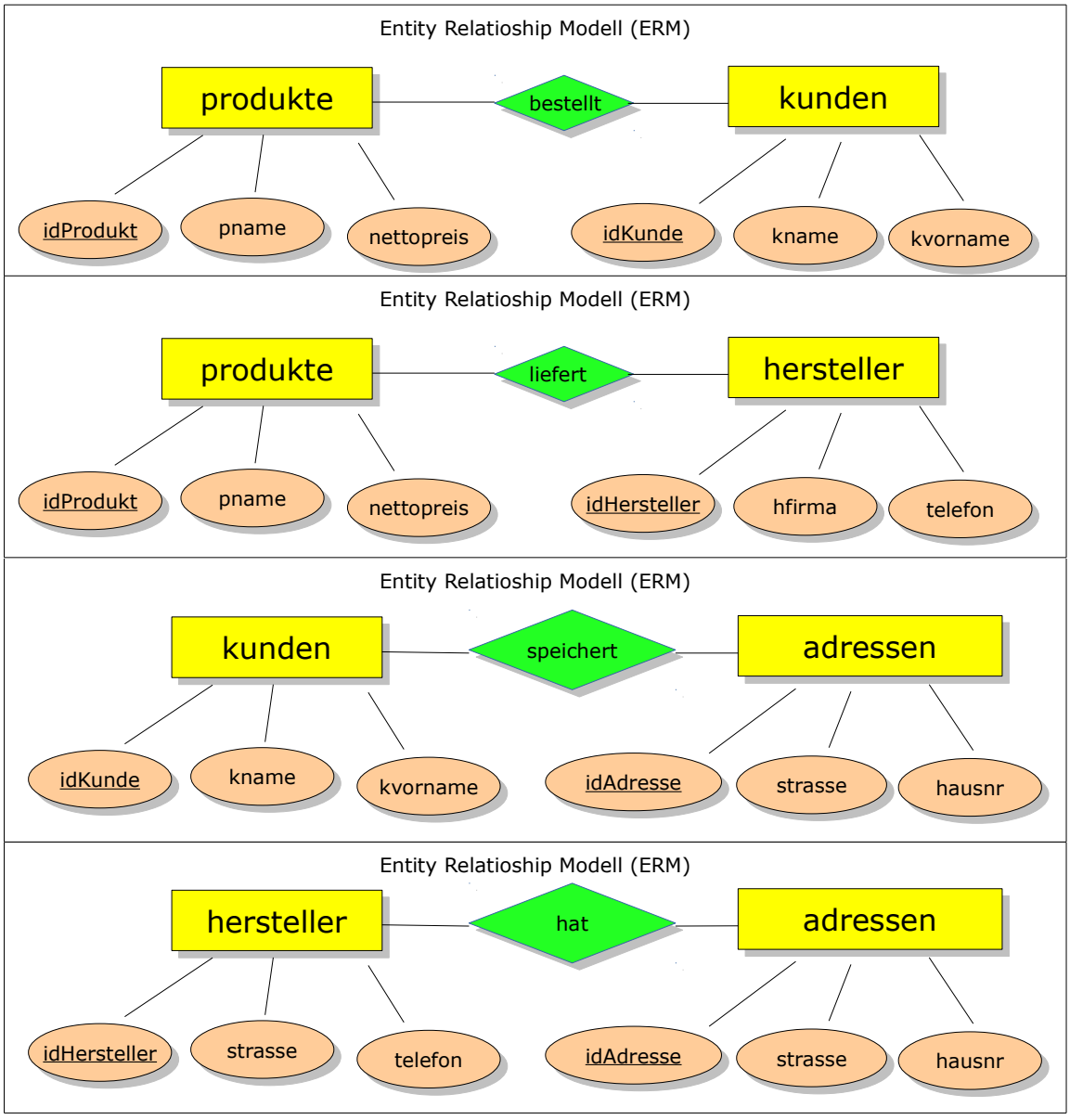
Ermitteln Sie

1. den mengenmäßigen Zusammenhang der gespeicherten Objekte.
2. Beachten Sie, dass die Textarbeit und der Lösungsweg benotet wird. → Relationen paarweise betrachten!
3. Treffen Sie Annahmen wenn Sie unsicher sind!
4. Erzeugen Sie anhand der Erkenntnisse ein optimiertes Gesamtmodells (ERM **und** Relationenmodell) der Datenbank und
5. berücksichtigen Sie darin alle benötigten Schlüsselattribute.

CC BY NC SA

Thema:	Übung: Modellierung einer Relationalen Datenbank Urquelle: Christine Janischek Kardinalitäten, Entity Relationship Modell (ERM) und Enhanced Entity Relationship Modell (EERM)

Sie möchten einen Online-Shop für Mobiltelefone etablieren.



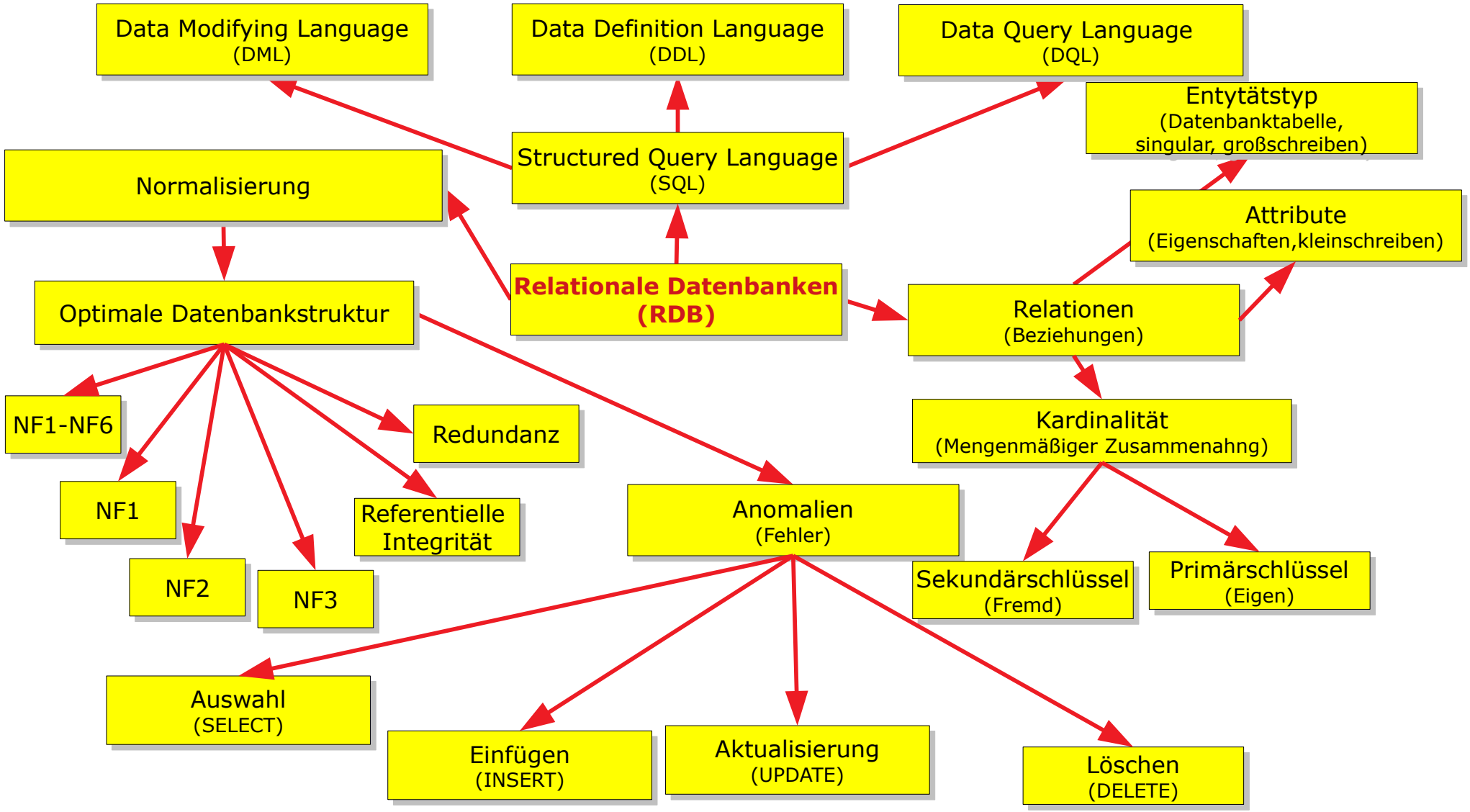
Arbeitsauftrag:

1. Welcher mengenmäßige Zusammenhang besteht zwischen den Datensätzen in den beiden Tabellen?
2. Modellieren Sie diesen Zusammenhang in einem normalisierten Gesamtmodell mit Hilfe mit einer geeigneten Entwicklungsumgebung.

3. Wo würden Sie die Attribute für die Adresse (Anschrift) unterbringen?

CC BY NC SA

3 Begriffe im Kontext Relationaler Datenbanken



4 Entwicklung Relationaler Datenbanken

Entwicklung
Relationaler Datenbanken

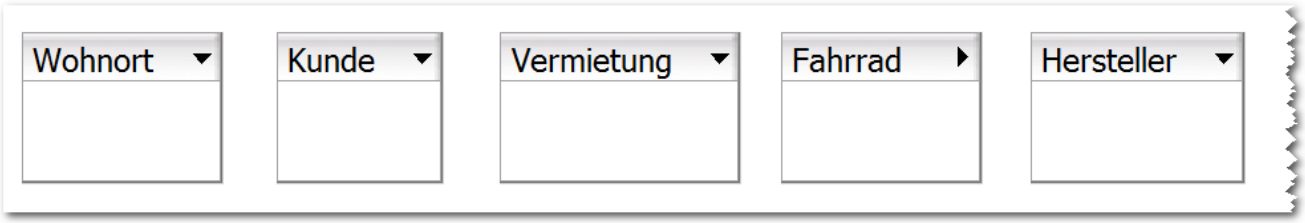
CC BY NC SA

Thema:	Übung: Entwicklung Relationaler Datenbanken Urquelle: Christine Janischek DDL: Data Definition Language

Das Unternehmen **Rent-A-Bike** plant einen Online-Shop. Zur Modellierung der zugehörigen Datenbank konnten bisher folgende Zusammenhänge formuliert werden.

1. Vermietungen gehören zu Kunden.
2. Kunden speichern den Wohnort.
3. Hersteller liefern Fahrräder.
4. Fährräder gehören zu Vermietungen.
5. Kunden haben einen Namen, Vornamen und leben in einer Straße.
6. Wohnorte haben einen Ortsnamen und Postleitzahl.
7. Vermietungen werden mit dem Mietdatum und dem Zeitraum (von,bis) erfasst.

Entitätstypen von Rent-A-Bike



Arbeitsauftrag:

1. Identifizieren Sie die Teilprobleme?
2. Modellieren Sie die Entitätstypen, Attribute und Datentypen.
3. Bilden Sie die EINE-Sätze und modellieren Sie die Relationen (Kardinalitäten).
4. Führen Sie die Anpassung durch und exportieren Sie das SQL-Create-Script.
5. Fassen Sie die Erkenntnisse der Klasse in der Dokumentation zusammen.

Das **SQL-Create-Script** enthält die Struktur der Datenbank. Die Struktur und Daten müssen in eine Datenbank importiert werden. Die Entwicklungsumgebungen im Kontext Relationaler Datenbanken:

Thema:	Relationale Datenbanken Arbeitsauftrag Quelle: L1 4 Daten in eine Datenbank importieren

Heiner Blechle möchte in der neuen Datenbank *fahrschueler* die Daten seiner Fahrschüler einfügen. Sein Freund Alexander hat die Kontaktdaten von Heiners Smartphone in eine Skriptdatei überführt und um die fehlenden Informationen ergänzt.

Die Datei *L1_4.1_fahrschule_1_Tabelle_Daten_einfügen.sql* steht Ihnen im Ordner '*Scripte*' in digitaler Form zur Verfügung.

Importieren Sie die Daten in die Tabelle *fahrschueler* der Datenbank *fahrschule*.

Beachten Sie das Informationsmaterial *L1_4 Information Daten importieren.docx*.

Hinweis: Sofern Sie das Datenbankmodell der Fahrschule nicht in eine Datenbank überführt haben bzw. beim Generieren der Datenbank Fehler auftraten, importieren Sie die komplette Datenbank mit der Datei *L1_4.2_fahrschule_1_Tabelle_komplett.sql.*

CC BY NC SA

5 Structured Query Language (SQL)

SQL
Structured Query Language

CC BY NC SA

Thema:	Relationale Datenbanken Arbeitsauftrag Quelle: L1 5.1 Daten einer Datenbank mit SQL abfragen - Projektion

Beachten Sie zur Bearbeitung der nachfolgenden Aufgabenstellungen das Informationsmaterial *L1_5.1 Information Datenbankabfrage Projektion.docx.*

1. Erläutern Sie den Begriff 'Projektion'.

2. Heiner Blechle möchte aus seiner Datenbank *fahrschule* die Informationen Schülernummer sowie Vorname und Nachname aufgelistet haben.
   Die Liste soll nach den Nachnamen alphabetisch geordnet sein.

3. Formulieren Sie einen entsprechenden SQL-Befehl.

4. Dokumentieren Sie Ihre Lösung.

5. Alle Fahrschüler mit Schülernummer, Vor- und Nachnamen sowie Wohnort sollen nach Wohnort absteigend geordnet (sortiert) aufgelistet werden.
6. Formulieren Sie einen entsprechenden SQL-Befehl.

7. Dokumentieren Sie Ihre Lösung.

8. Alle Fahrschüler mit Schülernummer sowie Vor- und Nachnamen sollen aufgelistet werden. Die Liste soll alphabetisch nach den Nachname geordnet (sortiert) sein. Bei gleichen Nachnamen soll nach den Vornamen sortiert werden.
9. Formulieren Sie einen entsprechenden SQL-Befehl.

Dokumentieren Sie Ihre Lösung.

CC BY NC SA

Thema:	Relationale Datenbanken Arbeitsauftrag Quelle: L1 5.1 Projektion – Vertiefungsaufgaben

Die Fahrradvermietung 'Rent A Bike e.K.', Meersburg, verwaltet die Daten ihrer Fahrräder in der Datenbank fahrradvermietung.

Die Datenbank steht Ihnen im Ordner 'Scripte' als Skriptdatei zum Import zur Verfügung (L1_5.2_fahrradvermietung_1_Tabelle_komplett.sql)

Importieren Sie die Datenbank in die MySQL Workbench und selektieren Sie die importierte Datenbank im Object Browser der MySQL Workbench (Doppelklick).

Die folgende Abbildung zeigt einen Auszug aus der Tabelle *fahrraeder* der Datenbank *fahrradvermietung*:

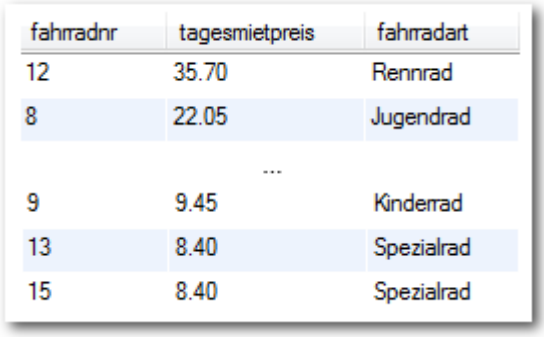
fahrradnr	rahmennr	tagesmietpreis	anschaffungswert	kaufdatum	fahrradart
1	55-88-333	15.75	700.00	2018-02-02	Trekkingrad
2	55-88-334	15.75	770.00	2014-06-27	Trekkingrad
3	U/H2345	19.95	1285.00	2017-10-11	Jugendrad
4	56/32	10.50	889.00	2018-05-05	Trekkingrad
5	56/33	10.50	819.00	2017-09-23	Trekkingrad
6	MTB/B34	19.95	998.00	2018-07-23	Mountainbike
7	0002	12.60	234.00	2015-03-11	Kinderrad
8	88/07	22.05	1550.00	2017-03-10	Jugendrad
9	340/90089	9.45	545.00	2017-01-17	Kinderrad
10	298H46	17.85	810.00	2017-05-27	Jugendrad
11	MTB/G11	21.00	1098.00	2018-05-15	Mountainbike
12	MTB/Z68	35.70	689.00	2018-05-05	Rennrad
13	4590/H2	8.40	56.00	2018-05-23	Spezialrad
14	198H47	9.45	310.00	2015-09-21	Kinderrad
15	4890/H2	8.40	118.00	2018-01-02	Spezialrad
16	CB/098	11.55	1780.00	2018-05-16	Trekkingrad
17	76/67654e	12.60	646.00	2017-09-05	Jugendrad

Formulieren Sie für die folgenden Aufgabenstellungen die entsprechenden SQL-Befehle und dokumentieren Sie Ihre Lösungen.

Aufgaben:

	Aufgabe	Gewünschte Ausgabe
1	Alle Fahrräder mit Fahrradnummer, Tagesmietpreis und Fahrradart.	

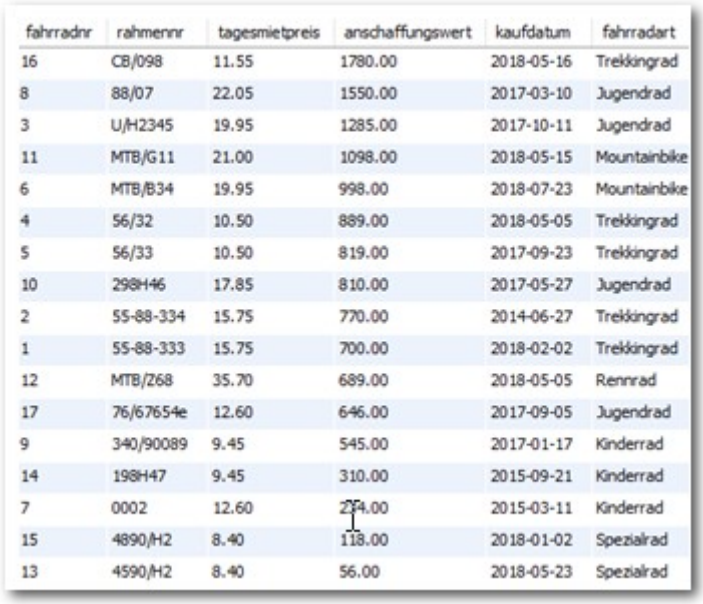
fahrradnr	tagesmietpreis	fahrradart
1	15.75	Trekkingrad
2	15.75	Trekkingrad
3	19.95	Jugendrad
...		
16	11.55	Trekkingrad
17	12.60	Jugendrad

	Aufgabe	Gewünschte Ausgabe
2	… wie Aufgabe 1, zusätzlich nach dem Tagesmietpreis absteigend sortiert.	

fahrradnr	tagesmietpreis	fahrradart
12	35.70	Rennrad
8	22.05	Jugendrad
...		
9	9.45	Kinderrad
13	8.40	Spezialrad
15	8.40	Spezialrad

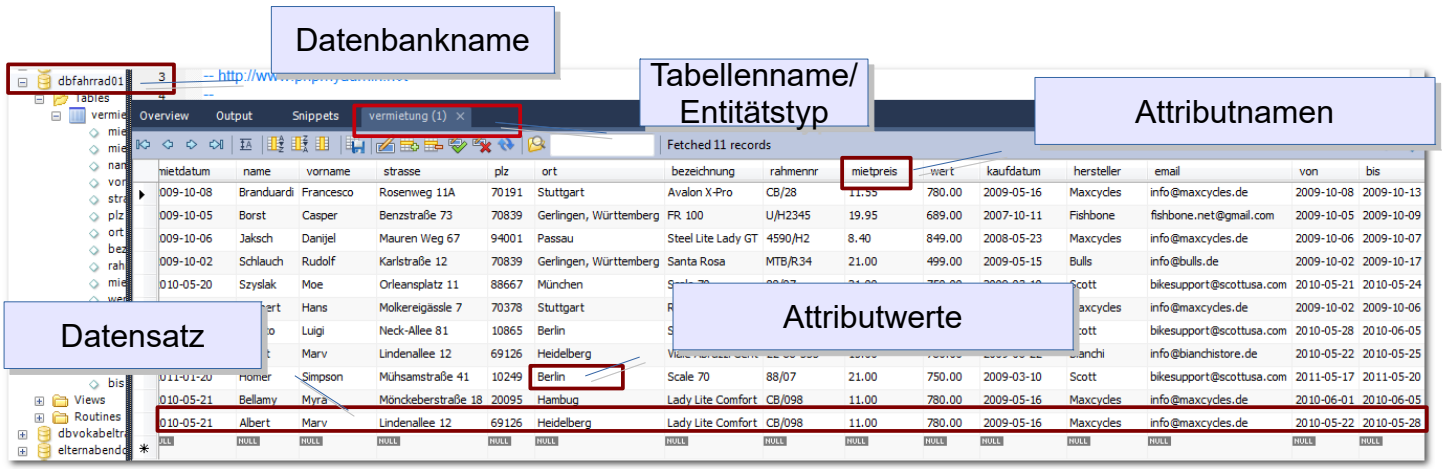
	Aufgabe	Gewünschte Ausgabe
3	Alle Fahrräder mit Fahrradnummer, Fahrradart und Tagesmietpreis, sortiert nach Fahrradart und Tagesmietpreis.	

fahrradnr	fahrradart	tagesmietpreis
17	Jugendrad	12.60
10	Jugendrad	17.85
3	Jugendrad	19.95
8	Jugendrad	22.05
14	Kinderrad	9.45
9	Kinderrad	9.45
7	Kinderrad	12.60
...		
16	Trekkingrad	11.55
2	Trekkingrad	15.75
1	Trekkingrad	15.75

	Aufgabe	**Gewünschte Ausgabe**
4	Alle Fahrräder, wobei das Rad mit dem höchsten Anschaffungswert oben stehen soll. Bei gleichem Anschaffungswert soll nach dem Kaufdatum aufsteigend sortiert werden.	

fahrradnr	rahmennr	tagesmietpreis	anschaffungswert	kaufdatum	fahrradart
16	CB/098	11.55	1780.00	2018-05-16	Trekkingrad
8	88/07	22.05	1550.00	2017-03-10	Jugendrad
3	U/H2345	19.95	1285.00	2017-10-11	Jugendrad
11	MTB/G11	21.00	1098.00	2018-05-15	Mountainbike
6	MTB/B34	19.95	998.00	2018-07-23	Mountainbike
4	56/32	10.50	889.00	2018-05-05	Trekkingrad
5	56/33	10.50	819.00	2017-09-23	Trekkingrad
10	298H46	17.85	810.00	2017-05-27	Jugendrad
2	55-88-334	15.75	770.00	2014-06-27	Trekkingrad
1	55-88-333	15.75	700.00	2018-02-02	Trekkingrad
12	MTB/Z68	35.70	689.00	2018-05-05	Rennrad
17	76/67654e	12.60	646.00	2017-09-05	Jugendrad
9	340/90089	9.45	545.00	2017-01-17	Kinderrad
14	198H47	9.45	310.00	2015-09-21	Kinderrad
7	0002	12.60	274.00	2015-03-11	Kinderrad
15	4890/H2	8.40	118.00	2018-01-02	Spezialrad
13	4590/H2	8.40	56.00	2018-05-23	Spezialrad

Thema:	Grundgerüst einer SQL-Abfrage Urquelle: Christine Janischek Auswahlabfragen: Einführung in SQL



[Abb. 1: Begriffe im Kontext von Datenbanken]

Anwendungsfall	SQL-Abfrage

1. Der Name, Vorname und Fahrradbezeichnung aller im Jahr 2010 durchgeführten Vermietungen.	`SELECT name,vorname,bezeichnung`
	`FROM Vermietung`
	`WHERE YEAR(mietdatum) = 2010;`

2. Der Zeitraum, Fahrradbezeichnung und Hersteller der Vermietungen an Marv Albert.	`SELECT von,bis,bezeichnung,hersteller`
	`FROM Vermietung`
	`WHERE name = 'Albert' AND vorname = 'Marv';`

3. Der Name, Vorname, Fahrradbezeichnung und Hersteller aller Vermietungen deren Tagesmietpreis höher ist als 11 €.	`SELECT name,vorname,bezeichnung,hersteller`
	`FROM Vermietung`
	`WHERE mietpreis > 11.00;`

4. Der Name, Vorname, Fahrradbezeichnung und Hersteller aller Vermietungen der Marke „Maxcycles“	`SELECT name,vorname,bezeichnung,hersteller`
	`FROM Vermietung`
	`WHERE hersteller = 'Maxcycles';`

5. Der Name, Vorname und Hersteller aller Vermietungen des Kunden „Albert“	`SELECT name,vorname,hersteller`
	`FROM Vermietung`
	`WHERE name = 'Albert';`

Arbeitsauftrag:

Die Aufgaben beziehen sich auf die Datenbank-Struktur und Daten (dbfahrrad01)

1. Importieren Sie die Struktur und Daten.
2. Testen Sie die gegebenen Abfragen und dokumentieren Sie die Ergebnisse.
3. Dokumentieren Sie den Importvorgang und das Grundgerüst einer SQL-Abfrage.

Thema:	Auswahlabfragen in SQL Urquelle: Christine Janischek Data Query Language: WHERE, ORDER BY, GROUP BY, HAVING

```
SELECT <attributsbezeichnung1>, <attributsbezeichnung2> ...   ( * = alle Felder )
FROM <tabelle> [, <tabelle>, <tabelle>....]
[WHERE <bedingung> ]
[ORDER BY <attributsbezeichnung> [ASC|DESC]..]
[GROUP BY <attributsbezeichnung> ]
[HAVING <bedingung> ];
```

Für den Online-Fahhrad-Vermietungs-Shop werden neue Datenbankabfragen benötigt. Um diese im System zu integrieren müssen die Abfragen erstellt und getestet werden.
Sie übernehmen diesen Job!

Arbeitsauftrag:

Die Aufgaben beziehen sich auf die Datenbank-Struktur und Daten (dbfahrrad02)

1. Importieren Sie die Struktur und Daten.
2. Testen Sie die gegebenen Abfragen und dokumentieren Sie die Ergebnisse.
3. Dokumentieren Sie die neuen SQL-Klauseln WHERE, ORDER BY, GROUP BY, HAVING.

Anwendungsfall	**Lösungsschema**
Selektion mit einer Bedingung (einfacher Textvergleich) und einer Ordnung	

1. Aufzulisten sind die Attribute Fahrradnr, Bezeichnung, Wert und Kaufdatum des Fahrrads mit der Bezeichnung „Scott Scale 70“.
 	
Erweiterung:
Alle Fahrräder, deren Bezeichnung mit ‚Scott‘ beginnt, soll aufgelistet werden. Die Liste soll nach dem Kaufdatum absteigend sortiert sein.
	`SELECT <Tabellenname>.<Attributname>,...`
	`FROM <Tabellenname>`
	`WHERE <Bedingung>`
	`ORDER BY <Attributname> DESC;`
	
	`*DESC für Absteigend, ASC für Aufsteigend`

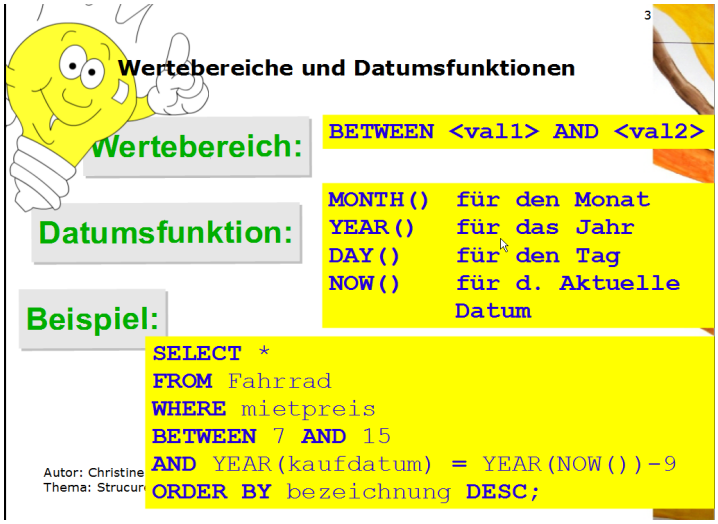
Selektion mit einer Bedingung (numerischer Vergleich)	

2. Gewünscht wird eine Auflistung aller Fahrräder, deren Tagesmietpreis unter 15.00 € liegt (Attribute: Fahrradnr, Bezeichnung, Tagesmietpreis). Die Liste soll aufsteigend sortiert sein.	`SELECT <Tabellenname>.<Attributname>,...`
	`FROM<Tabellenname>`
	`WHERE <Attributname> <Vergleischsoperator>`
	`<Attributname/-wert>;`
	
	`*Beispiele Vergleichsoperatoren:`
	• `> für größer als`
	• `< für kleiner als`
	• `= für gleich`

Selektion mit einer Bedingung (Vergleich mit Datumswert)	

3. Alle Fahrräder, die vor dem 01.01.2009 angeschafft wurden, sind nach dem Kaufdatum absteigend sortiert auszugeben. (Fahrradnr, Bezeichnung, Rahmennummer, Kaufdatum).	`SELECT <Tabellenname>.<Attributname>,...`
	`FROM<Tabellenname>`
	`WHERE <Attributname> <Vergleischsoperator>`
	`<Attributwert>`
	`ORDER BY <Attributname>;`
	
	`*Datumsangaben:`
	• `"2009-01-01"`

Thema:	Auswahlabfragen in SQL Urquelle: Christine Janischek Data Query Language: Datumsabfragen, Intervalle



Für den Online-Fahhrad-Vermietungs-Shop werden neue Datenbankabfragen benötigt. Um diese im System zu integrieren müssen die Abfragen erstellt und getestet werden. Sie übernehmen diesen Job!

Arbeitsauftrag: **Die Aufgaben beziehen sich auf die Datenbank-Struktur und Daten (dbfahrrad03)**

1. Importieren Sie die Struktur und Daten.
2. Testen Sie die gegebenen Abfragen und dokumentieren Sie die Ergebnisse.
3. Dokumentieren Sie die neuen SQL-Klauseln für Datumsabfragen und Intervalle

Anwendungsfall	Lösungsschema
Selektion mit Wertebereichen	
1. Aufzulisten sind alle Fahrräder, deren Tagesmietpreis zwischen 15,00€ und 20,00€ liegt (Erwartete Ausgabe: Fahrradnr., Bezeichnung und Tagesmietpreis).	SELECT <*Tabellenname*>.<Attributname>,... FROM <Tabellenname> WHERE <*Tabellenname*>.<Attributname> BETWEEN <value1> AND <value2>;
Selektion mit einfachen Datumsfunktionen	
2. Ermitteln Sie, welche Fahrräder im Jahre 2007 gekauft wurden.	SELECT <Alle Felder> FROM<Tabellenname> WHERE YEAR(<Attributname>) = <Attributwert>; *Datumsabfragen: • YEAR() für das Jahr • MONTH() für den Monat • DAY() für den Tag • NOW() für das aktuelle Datum
3. Alle Fahrräder, die im Monat Mai gekauft wurden, sollen nach ihrer Bezeichnung absteigend aufgelistet werden.	SELECT <Alle Felder> FROM <Tabellenname> WHERE Month(<Attributname>) = <Attributwert> ORDER BY <Attributname> DESC;
Selektion mit berechnenden Datumsfunktionen	
4. Welche Fahrräder werden im laufenden Jahr fünf Jahre alt?	SELECT <Alle Felder> FROM <Tabellenname> WHERE YEAR(<Attributname>) =YEAR(NOW()) - <value> ;

Thema:	Auswahlabfragen in SQL Urquelle: Handreichung Relationale Datenbanken SQL Data Query Language: Gruppierungsfunktionen (Aggregatsfunktionen)

AVG(Spalte) = Durchschnittswert | **COUNT**(Spalte) = Anzahl aller Einträge
MAX(Spalte) = Maximalwert | **MIN**(Spalte) = Minimalwert
SUM(Spalte) = Summe aller Einträge in einer Spalte

Gruppierungsfunktionen können nur anstelle eines Spaltennamens direkt hinter der SELECT Anweisung stehen.Sie liefern genau einen Wert, beziehen auf mehrere Tabellenzeilen.

Selektion über mehrere Tabellen

1. Von allen Mitarbeitern sind die gesamten Monatsgehälter, das durchschnittliche sowie das höchste und das niedrigste Gehalt aufzulisten (siehe Abb.).

Gehaltssumme	durchschnittliches Gehalt	hoechstes Gehalt	geringtes_Gehalt
222537.50	2853.044872	6150.00	1800.00

2. Mit Hilfe einer Abfrage ist die Anzahl der Mitarbeiter, die in der Abteilung mit der Kostenstellennummer 04 beschäftigt sind, zu ermitteln.

kostnr	Anzahl_Mitarbeiter
04	12

Arbeitsauftrag:

Die Aufgaben beziehen sich auf die Datenbank-Struktur und Daten (muecke_1)

1. Importieren Sie die Struktur und Daten.
2. Testen Sie die gegebenen Abfragen und dokumentieren Sie die Ergebnisse.
3. Dokumentieren Sie die neuen SQL-Klauseln.

Thema:	Auswahlabfragen in SQL Urquelle: Handreichung Relationale Datenbanken SQL Data Query Language: Auswahlabfragen über mehrere Tabellen

kundennr	firma	strasse	ortnr	ortnr	ort
270033	Einrichtungshaus Blessing	Bissinger Str. 5	34976	34976	Augsburg, Bay
270036	Meinschneider Einrichtungen	Brunnenweg 88	29303	29303	Esslingen am Neckar
270017	NimmMit Mitnahmemöbel GmbH	Brühlstr. 50	29306	29306	Esslingen am Neckar
270053	Einrichtungshaus Bluthardt	Kastellstr. 14	29303	29303	Esslingen am Neckar
270040	Schober Vollholzmöbel KG	Heilbronner Str. 3	29675	29675	Stuttgart
270002	FIRENCE Möbel & Leuchten GmbH	Königsstraße 12	29521	29521	Stuttgart
270018	Einrichtungshaus Bull e.K.	Hauptstraße	29654	29654	Stuttgart
270008	Bär-Möbel GmbH	Luisenstr. 55	36023	36023	Ulm, Donau

```
SELECT kundennr, firma, strasse, kunden, plz, ort ← (1)
FROM kunden, orte ← (2)
WHERE kunden.ortnr = orte.ortnr ← (3)
AND (ort LIKE "Ulm%" OR ort LIKE "Stutt%") ← (4)
ORDER BY ort; ← (5)
```

(1) Zusammenstellen der anzuzeigenden Attribute (Projektion).
(2) Auswahl der benötigten Tabellen.
(3) Bedingung für die Verbindung der beiden Tabellen. Für das gemeinsame Attribut ortnr müssen in beiden Tabellen übereinstimmende Attributswerte vorhanden sein. Attribute, die in beiden Tabellen vorkommen, müssen mit dem Tabellennamen als „Vorsilbe“ angesprochen werden, Beispiel *kunden.ortnr.*
(4) Bedingung für die Auswahl der Datensätze (= Selektion). (Die in der WHERE-Klausel verwendeten Platzhalter sollen sicher stellen, dass auch eine Ortsbezeichnung wie beispielsweise „Ulm, Donau“ selektiert wird.)
(5) Sortierung.

Selektion über mehrere Tabellen

1. Alle Kunden mit einem Rabattsatz von 10% sollen mit den Feldern firma, plz, ort und rabatt aufgelistet werden.

2. Die Auftragskopfdaten des Auftrages mit der Auftragsnummer 20370 sind wie in der nachfolgenden Anzeige dargestellt, anzuzeigen.

auftragsnr	auftragsdat	kundennr	firma	strasse	plz	ort
20370	2014-08-20	270015	Schlenz Möbel KG	Bachrain 71	89250	Senden, Iller

3. Die Vertriebsleiterin Frau Angelika Schlauch-Köpf wünscht von der Produktgruppe I der Fertigerzeugnisse eine Auflistung der Attribute erzeugnisnr, bezeichnung, herstellkosten sowie Zuschlagssatznummer (Attribut zuschlagsnr) und Kalkulationszuschlagssatz (Attribut zuschlagssatz)

fertigernr	bezeichnung	herstellkosten	zuschlagsnummer	zuschlagssatz
200030	Stuhl EIFEL	275.50	Z1	16
200032	Stuhl TAUNUS	197.50	Z1	16
200012	Sideboard INN	1125.50	Z2	20
200017	Vertiko BLEIBTREU	995.75	Z2	20
200018	Vertiko SEHNSUCHT	1250.75	Z2	20
200024	Esstisch INNSBRUCK	915.50	Z2	20
200001	Kleiderschrank PARIS	1675.50	Z3	23
200003	Schrankwand ALLGÄU	955.50	Z3	23
200007	Schrankwand TIROL	883.75	Z3	23
200010	Sideboard DONAU	695.50	Z3	23
200013	Sideboard ISAR	755.00	Z3	23
200026	Couchtisch LECH	712.50	Z3	23
200004	Schrankwand LINZ	995.00	Z4	25
200006	Schrankwand OSTSEE	825.75	Z4	25
200002	Kleiderschrank LONDON	1175.75	Z5	30

4. Die Verkaufsleitung wünscht eine Aufstellung der Herstellkosten, Zuschlagssätze und der Netto-Verkaufspreise aller Fertigerzeugnisse. (Netto-Verkaufspreis = Herstellkosten * Zuschlagssatz/100). Die Aufstellung soll nach den Verkaufspreisen aufsteigend sortiert sein und folgendes Aussehen haben (auszugsweise):

fertigerznr	bezeichnung	herstellkosten	zuschlagssatz	Verkaufspreis
200028	Tisch ALPHA	98.50	16	114.260000
200023	Wandregal BETA	175.00	16	203.000000
200034	Rollcontainer UNIT	175.50	20	210.600000

5. Welche Fertigerzeugnisse hat der Kunde City-Möbel GmbH (kundennr = 270042) im Juni 2009 in Auftrag gegeben? Die Aufstellung soll folgendes Aussehen haben (auszugsweise):

kundennummer	firma	plz	ort	auftragsdat	fertigungsnr	menge	bezeichnung
270042	City-Möbel GmbH	73312	Geislingen an der Steige	2009-06-28	200014	4	Sideboard MENOS

Arbeitsauftrag:

Die Aufgaben beziehen sich auf die Datenbank-Struktur und Daten (muecke_2)

1. Importieren Sie die Struktur und Daten.
2. Testen Sie die gegebenen Abfragen und dokumentieren Sie die Ergebnisse.
3. Dokumentieren Sie die neuen SQL-Klauseln.

CC BY NC SA

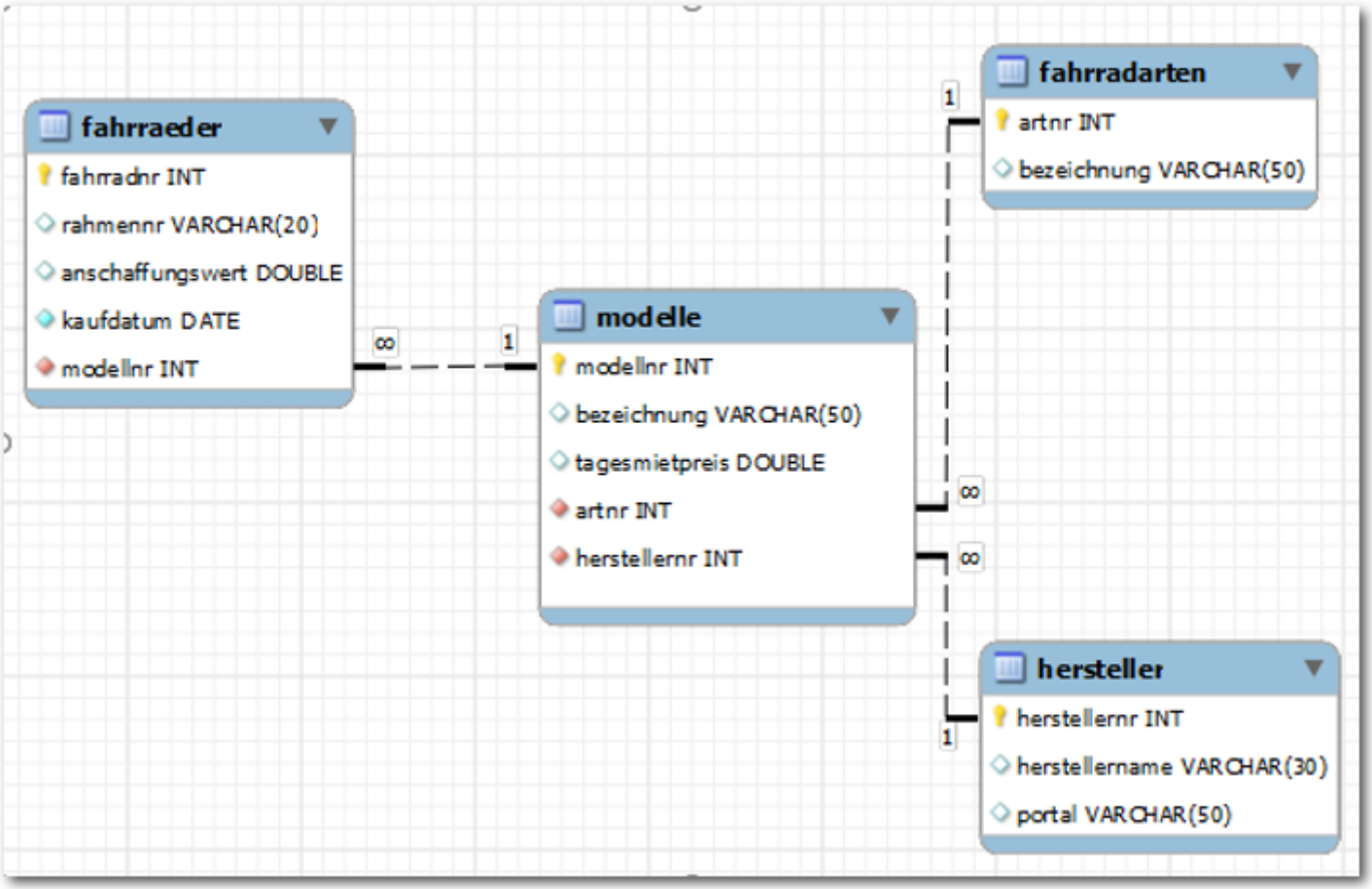
Thema:	Auswahlabfragen in SQL Urquelle: Handreichung Relationale Datenbanken SQL Data Query Language: Auswahlabfragen über mehrere Tabellen Quelle: L2_5.2.2 Datenbankabfragen Fahrradvermietung – Vertiefungsaufgabe

Die Datenbank *fahrradvermietung* wurde weiterentwickelt. Zu jedem Fahrrad können nun Informationen zum jeweiligen Modell, zum Hersteller sowie zur jeweiligen Fahrradart gespeichert werden.

Die Skriptdatei zum Importieren der Datenbank *fahrradvermietung_1* steht Ihnen im Unterordner *'Scripte'* zur Verfügung (*L2_5.2_fahrradvermietung_4_Tabellen.sql*)

Importieren Sie die Datenbank in die MySQL Workbench und selektieren Sie die importierte Datenbank im Object Browser der MySQL Workbench.

Der Datenbank *fahrradvermietung* liegt folgendes Entity-Realtionship-Diagramm zugrunde:



https://edublog.emotionalspirit.de/

Entwickeln Sie für die folgenden Aufgabenstellungen die jeweiligen SQL-Anweisungen und dokumentieren Sie Ihre Ergebnisse.

	Aufgabe	**Gewünschte Ausgabe**

	Aufgabe	**Gewünschte Ausgabe**

	Aufgabe	**Gewünschte Ausgabe**

	Aufgabe	**Gewünschte Ausgabe**

	Aufgabe	**Gewünschte Ausgabe**

	Aufgabe	**Gewünschte Ausgabe**

CC BY NC SA

	Aufgabe	**Gewünschte Ausgabe**

	Aufgabe	**Gewünschte Ausgabe**

	Aufgabe	**Gewünschte Ausgabe**

	Aufgabe	**Gewünschte Ausgabe**

	Aufgabe	**Gewünschte Ausgabe**

	Aufgabe	**Gewünschte Ausgabe**

CC BY NC SA

	Aufgabe	**Gewünschte Ausgabe**

	Aufgabe	**Gewünschte Ausgabe**

	Aufgabe	**Gewünschte Ausgabe**

	Aufgabe	**Gewünschte Ausgabe**

	Aufgabe	**Gewünschte Ausgabe**

CC BY NC SA

Thema:	Structured Query Language Urquelle: Handreichung Relationale Datenbanken SQL Data Manipulation Language: INSERT

Um einen Widerspruch zu den Anforderungen der *referentiellen Integrität* beim Erfassen der Daten zu vermeiden, müssen also zunächst die Daten der Parent-Tabellen und danach die Daten der Child-Tabellen eingegeben werden.

	Parent-Tabelle	Primärschlüssel	Child-Tabelle	Fremdschlüssel
1.	orte	ortnr		
2.	kunden	kundennr	kunden	ortnr
3.	auftraege	auftragsnr	auftraege	kundennr
5.	fertigerzeugnisse	fertigerznr		
6.	auftragspositionen	auftragsposnr	auftragspositionen	auftragsnr fertigerznr

Syntax: INSERT INTO <*tabellenname*>(<*attributbezeichnung1*>, <*attributsbezeichung2*>, …) VALUES (<*attributwert1*>, <*attributwert2*>, …);

Die Firma Möbel - Kaiser aus 81241 München, Theresienwiese 12, Telefon-nummer 089-223520, wird als Kunde geworben und erhält die Kundennummer 270056. In der Tabelle *orte* hat der Eintrag des Ortes 81241 München die *ortnummer* 33011. Diese Daten sollen in der Tabelle *kunden* erfasst werden.

Vorgehensweise:

1. Verschaffen Sie sich einen Überblick zu den Abhängigkeiten der Datenbanktabellen (Reverse Engineering).
2. Prüfen, ob die Referenzdaten in der Parent-Tabelle orte vorhanden sind!
3. Daten in die Tabelle kunden erfassen!

Vorausgesetzt der Ort ist vorhanden:

```
INSERT INTO kunden(kundennr, firma, strasse, ortnr, telefon)
VALUES (270056, 'Möbel Kaiser', 'Theresienwiese 12', 33019, '089-223520');
```

Einfügen von Daten (Referentielle Integrität)

Herr Klaus Meinhardt, Inhaber der Firma "Meinhardt - Möbel nach Maß", 70599 Stuttgart, Neckarstraße 55, Telefon 0711-453423 bestellt am 05.10.d.J. auf der Hausmesse der Mücke & Partner GmbH spontan 9 Sideboards Donau (fertigerznr 200010) und 2 Sideboards Ultimo (fertigerznr 200016). Erfassen Sie den Auftrag!

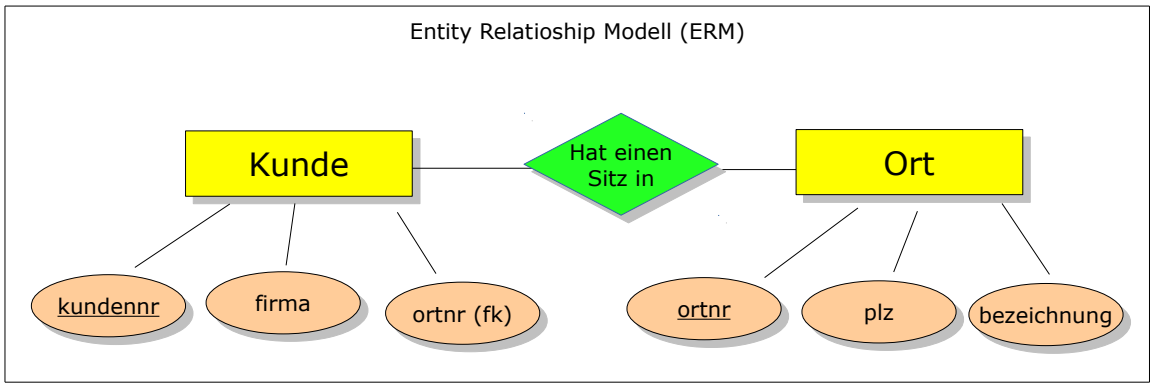
Arbeitsauftrag:

Die Aufgaben beziehen sich auf die Datenbank-Struktur und Daten (muecke_0)

1. Importieren Sie die Struktur und Daten.
2. Testen Sie die gegebenen Abfragen und dokumentieren Sie die Ergebnisse.
3. Dokumentieren Sie die neuen SQL-Klauseln und Ihre Vorgehensweise.

Thema:	Structured Query Language Urquelle: Handreichung Relationale Datenbanken SQL Information: Referentielle Integrität

Damit Datenbanken keine unkorrekten Daten enthalten, müssen sogenannte Integritätsbedingungen erfüllt sein. Bei der referentiellen Integrität geht es um die Beziehungen zwischen Tabellen einer Datenbank. Der Primärschlüssel einer Tabelle kann in einer anderen Tabelle als Fremdschlüssel verwendet werden. Über diese Fremdschlüssel werden Tabellen verknüpft und Abhängigkeiten zwischen diesen Tabellen aufgebaut.



Die **referentielle Integrität** stellt sicher, dass eine Datenbank frei von Widersprüchen (konsistent) ist und keine verwaisten Datensätze enthält.

Bei der referentiellen Integrität können Datensätze, die einen Fremdschlüssel enthalten (Child-Tabelle) nur dann gespeichert werden, wenn ein entsprechender Wert im Primärschlüsselattribut der in Beziehung stehenden Tabelle (Parent-Tabelle) vorhanden ist.

Konkret:

Die Daten eines Kunden können nur dann gespeichert werden, wenn die zu erfassende Ortsnummer bereits in der Tabelle *orte* als Primärschlüsselwert existiert.

Umgekehrt kann ein Datensatz aus einer Parent-Tabelle nicht gelöscht werden, solange der Wert seines Primärschlüssels als Eintrag im Fremdschlüsselattribut der in Beziehung stehenden Tabelle (Child-Tabelle) vorhanden ist.

Konkret:

Die Daten eines Ortes können nur dann gelöscht werden, wenn kein Kunde den Wert der Ortsnummer in seinem Fremdschlüsselattribut aufweist.

Prinzipiell kann die referentielle Integrität beim Einfügen, Ändern und Löschen von Datensätzen verletzt werden. Solche Verletzungen führen zu Anomalien (Unregelmäßigkeiten) in der Datenbank.

Thema:	Structured Query Language Urquelle: Handreichung Relationale Datenbanken SQL Data Manipulation Language: Daten aktualisieren (UPDATE)

Wegen gestiegener Kosten sollen bei allen Fertigerzeugnissen die Herstellkosten um 5 % erhöht werden. In der Tabelle fertigerzeugnisse sind die Herstellkosten (Attribut herstellkosten) durch die erhöhten Herstellkosten zu ersetzen.

Vorgehensweise:

1. Die zu aktualisierenden Datensätze auflisten!
2. Aktualisierungsabfrage!
3. Die aktualisierten Datensätze auflisten!

Vorausgesetzt es sind zu aktualisierende Datensätze vorhanden:

```
UPDATE fertigerzeugnisse
SET herstellkosten = herstellkosten * 1.05;
```

Aktualisieren von Daten (UPDATE-Anomalien)

Der Kunde Möbel Bison Heck (Kundennummer 270007) ist umgezogen. Sein Geschäftssitz ist jetzt in der Weidenstraße 55, 70950 Freiburg (ortnr = 32131). Außerdem hat sich die Telefonnummer des Kunden geändert. Neue Nummer: "0761-122334". Die Datenbank soll um diesen Sachverhalt ergänzt werden.

Arbeitsauftrag:

Die Aufgaben beziehen sich auf die Datenbank-Struktur und Daten (muecke_0)

1. Importieren Sie die Struktur und Daten.
2. Testen Sie die gegebenen Abfragen und dokumentieren Sie die Ergebnisse.
3. Dokumentieren Sie die neuen SQL-Klauseln und Ihre Vorgehensweise.

CC BY NC SA

Thema:	Structured Query Language Urquelle: Handreichung Relationale Datenbanken SQL Data Manipulation Language: Daten löschen (DELETE)

Syntax: DELETE FROM <*tabellenname*> WHERE <*bedingung*>;

Der Kunde "Wohnwelt GmbH" (Kundennr. 270016) zeigt kein Interesse an einer weiteren Geschäftsbeziehung zur Firma Mücke & Partner OHG. Die Daten dieses Kunden sollen gelöscht werden.

Vorgehensweise:

1. Zuerst wird also überprüft, ob es in der Tabelle auftraege noch eine Verbindung zum Kunden Wohnwelt GmbH (kundnummer = 20016) gibt.
2. Löschen des Datensatzes!

Vorausgesetzt es sind keine Aufträge vorhanden:

```
DELETE FROM kunden
WHERE kundennr = '20016';
```

Löschen von Daten (Referentielle Integrität)

Aus der Tabelle fertigerzeugnisse soll der Artikel "Sideboard Ultimo" 14 (fertigerznr = ‘200016‘) gelöscht werden. Dieses Produkt passt nicht mehr in die Produktstrategie des Unternehmens.

Arbeitsauftrag:

Die Aufgaben beziehen sich auf die Datenbank-Struktur und Daten (muecke_0)

1. Importieren Sie die Struktur und Daten.
2. Testen Sie die gegebenen Abfragen und dokumentieren Sie die Ergebnisse.
3. Dokumentieren Sie die neuen SQL-Klauseln, Ihre Vorgehensweise und die Problematik.

6 Normalisierung

Normalisierung
Relationaler Datenbanken

CC BY NC SA

Thema:	Normalisierung im Kontext Relationaler Datenbanken Handreichung: Relationale Datenbanken, LPE03, 20131105 Einführung in die Aufgabenstellungen zur Normalisierung

Aufgabenstellung

Eine Datenbank zur Verwaltung der **Stundenplandatenverwaltung** des Beruflichen Schulzentrums Wangen soll entwickelt werden. Im Vorfeld wurden folgende Anforderungen ermittelt und bereits strukturiert aufbereitet.

Datensammlung:

lehrer_name_vorname	lehrerkuerzel	uhrzeit	fach_nr	unterricht_wochentag	unterricht_datum	fach_gebaeude_raum	kalenderwoche	klasse
Janischek Christine	Janc	07:45:00	9	Freitag	2015-03-19	WINF-S-118	KW12	BK1W
Janischek Christine	Janc	09:30:00	11	Freitag	2015-03-19	DV11-S-118	KW12	12
Janischek Christine	Janc	11:15:00	10	Freitag	2015-03-19	DV-S-213	KW12	11G
Janischek Christine	Janc	07:45:00	12	Donnerstag	2015-03-18	WINF-S-118	KW12	BK2W
Janischek Christine	Janc	09:30:00	9	Donnerstag	2015-03-18	WINF-S-118	KW12	BK1W
Janischek Christine	Janc	11:15:00	13	Donnerstag	2015-03-18	DV03-S-118	KW12	13
Janischek Christine	Janc	13:35:00	10	Donnerstag	2015-03-18	INF-N-100	KW12	TG11

Zusammenhänge:

- Lehrer unterrichten Klassen.
- Lehrer unterrichtet Fächer.
- Unterricht erfolgt zum Zeitpunkt.
- Zeitpunkt bezieht sich auf Kalenderwoche
- Unterricht findet in Räumen statt.
- Räume befinden sich in Gebäuden.

Wann befindet sich eine Datenbanktabelle in 2.Normalform?

Beschreiben die Risiken die bei fehlender Normalisierung auftreten können?

In welcher Normalform befindet sich die vorliegende Datensammlung. Begründen Sie Ihre Aussage.

Normalisieren Sie die Datensammlung bis zur dritten Normalform (gefordert sind alle drei Stufen). Ermitteln Sie die Kardinalitäten auf korrekte Weise, um Ihre Zwischenergebnisse und das Endergebnis (Gesamtmodell) ausreichend zu begründen.

Thema:	Normalisierung im Kontext Relationaler Datenbanken Handreichung: J1, BPE 6 Relationale Datenbanken 2018/2019 Informationen: Normalformen

Die Erste Normalform (1NF):

Eine Tabelle ist in der Ersten Normalform, wenn

- alle Attribute atomare Werte beinhalten
  (atomar = nicht mehr teilbare Werte, keine Mehrfachwerte)
- die Tabelle einen eindeutigen Primärschlüssel besitzt

Die Zweite Normalform (2NF):

Eine Tabelle ist in der Zweiten Normalform, wenn

- sie die Bedingungen der Ersten Normalform erfüllt,
- jedes Nichtschlüssel – Attribut nur vom gesamten Primärschlüssel, nicht aber von einem Teil des Primärschlüssels abhängig ist.

Hinweis:

Die Definition der 2NF bringt mit sich, dass die 2NF nur dann verletzt sein kann, wenn die Tabelle einen zusammengesetzten Primärschlüssel hat. Ist das nicht der Fall, braucht die 2NF nicht überprüft zu werden.

Die Dritte Normalform (3NF):

Eine Tabelle ist in der Dritten Normalform, wenn

- sie die Bedingungen der Zweiten Normalform erfüllt,
- kein Nichtschlüsselattribut von einem anderen Nichtschlüsselattribut abhängig ist.

Thema:	Normalisierung im Kontext Relationaler Datenbanken Handreichung: J1, BPE 6 Relationale Datenbanken 2018/2019 Informationen: 3. Normalform

Bevor ein Datenmodell mit Hilfe eines relationalen Datenbankmanagementsystems implementiert wird, muss überprüft werden, ob sich das Modell in der Dritten Normalform (3NF) befindet.

Ein Datenmodell befindet sich in Dritten Normalform, wenn für alle Relationen folgende Bedingungen (Regeln) erfüllt sind:

Regel 1
Alle Attribute einer Relation weisen einfache Attributwerte auf.

Regel 2
Jedes Nicht-Schlüsselattribut einer Relation ist vom Primärschlüssel voll funktional abhängig und nicht von Schlüsselteilen.Wenn ein Attribut von einem Teil des Schlüssels eindeutig identifiziert werden kann, liegt ein Verstoß gegen die Regel vor

Regel 3
Kein Nicht-Schlüsselattribut darf von einem weiteren Nicht-Schlüsselattribut abhängig sein (transitive Abhängigkeit).Wenn ein Nicht-Schlüsselattribut von einem anderen Nicht-Schlüsselattribut abhängig ist, liegt ein Verstoß gegen die Regel vor.

Die Einhaltung dieser Regeln erhält ihre Bedeutung durch die Forderung, Redundanzen innerhalb der zu erstellenden Datenbank zu verhindern. (Von Redundanz spricht man dann, wenn gleiche Informationen mehrfach gespeichert werden.)
Darüber hinaus ist die Gewährleistung der genannten Bedingungen Voraussetzung dafür, dass in der zu erstellenden Datenbank die referentielle Integrität sichergestellt werden kann.

Die Normalisierung bzw. die Überprüfung der Dritten Normalform dient somit der Sicherung folgender Ziele:

1. Erzeugen einer flexiblen und stabilen Datenbankstruktur, bei der Datenänderungen leicht durchzuführen sind.
2. Vermeidung von Redundanzen
3. Vermeidung von Anomalien (Regelwidrigkeiten) beim
   - Ändern von Daten → Änderungsanomalie,
   - Löschen von Daten → Löschanomalie und
   - Einfügen von Daten → Einfügeanomalie.

Thema:	Normalisierung im Kontext Relationaler Datenbanken Handreichung: Relationale Datenbanken, LPE03, 20131105 Exkurs: 1. Normalform

Eine Relation befindet sich in 1. Normalform,
wenn jedes Attribut atomar (einwertig) ist.

Beispiel für eine Datensammlung:

CD_Nr	Album	Jahr der Gründung	Song
1711	Coldplay - Mylo Xyloto	1996	1. Mylo Xyloto, 2. Hurts Like Heaven, 3. Paradise
1712	Nickelback - Here And Now	1995	1. This Means War, 2. Bottoms Up

Die Attribute in der Datensammlung sind teilweise nicht atomar!

Verstoß gegen 1NF, da das Attribut ‘Song‘ eine komma-separierte Liste von Liedern und das Attribut ‘Album‘ die beiden Attributwertebereiche ‘Interpret‘ und ‘Albumtitel‘ beinhaltet.

Veränderungen:

1. Das Attribut ‘Song‘ wird in die Attribute ‘Song_Nr‘ und ‘Titel‘ aufgespalten und auf mehrere Zeilen aufgeteilt.
2. Das Attribut ‘Album‘ wird in die Attribute ‘Interpret‘ und ‘Titel‘ aufgelöst.
3. CD_Nr und Song_Nr bilden gemeinsam den Primärschlüssel (kombinierter Schlüssel).

Lösung:

Relation CD Sammlung					
CD_Nr	**Interpret**	**Album**	**Jahr der Gründung**	**Song_Nr**	**Titel**
1711	Coldplay	Mylo Xyloto	1996	1	Mylo Xyloto
1711	Coldplay	Mylo Xyloto	1996	2	Hurts Like Heaven
1711	Coldplay	Mylo Xyloto	1996	3	Paradise
1712	Nickelback	Here And Now	1995	1	This Means War
1712	Nickelback	Here And Now	1995	2	Bottoms Up

Arbeitsauftrag: Die Aufgaben beziehen sich auf die Datenbank-Struktur und Daten (Personal_NF1_Start)

1. Importieren Sie die Struktur und Daten.
2. Testen Sie die folgende Abfrage.
3. Prüfen Sie das Ergebnis, welches Risiko besteht?
4. Welche Veränderungen sind notwendig?
5. Was sagt Ihr Gefühl sind die behandelten Strukturen nun optimal?

idMitarbeiter	name	abt_nr	abt_bez	projekt_nr	projekt_bez
2001	Lehmann	101	Personal	11	Installation Intranet
2002	Berger	102	Fertigung	12,11	Aufbau Web-Site, Installation Intranet
2003	Hofele	101	Personal	12,11	Aufbau Web-Site, Installation Intranet
2006	Niemand	102	Fertigung	13,11	Entlohnungsverfahren, Installation Intranet
2007	Blümchen	103	Konstruktion	12,13,11	Aufbau Web-Site, Entlohnungsverfahren, Installation Intranet

```
SELECT idMitarbeiter,name, projekt_nr, projekt_bez
FROM mitarbeiter
WHERE projekt_bez = "Installation Intranet";
```

Thema:	Normalisierung im Kontext Relationaler Datenbanken Handreichung: Relationale Datenbanken, LPE03, 20131105 Exkurs: 2. Normalform

Eine Relation befindet sich in 2. Normalform, wenn sie sich in 1NF befindet und jedes Nicht-Schlüsselattribut funktional abhängig vom gesamten Primärschlüssel ist (volle funktionale Abhängigkeit).

Ein Attribut *B* einer Relation *R* ist dann von einem weiteren Attribut *A* dieser Relation funktional abhängig, wenn *B* eindeutig von *A* bestimmt wird. (Schreibweise: *A* ☐ *B*)

Anders ausgedrückt heißt das: Ist der Wert des Attributs *A* bekannt, kann damit eindeutig auf den Wert des Attributs *B* geschlossen werden.

Volle funktionale Abhängigkeit liegt dann vor, wenn jedes Nicht-Schlüsselattribut vom gesamten Primärschlüssel abhängt und nicht nur von einem Teil dieses Schlüssels.

Sofern die 1. Normalform erfüllt ist, kann ein Verstoß gegen die 2. Normalform nur bei zusammengesetzten Primärschlüsseln auftreten.

Beispiel für eine Datensammlung:

Relation CD Sammlung

CD_Nr	**Interpret**	**Album**	**Jahr der Gründung**	**Song_Nr**	**Titel**
1711	Coldplay	Mylo Xyloto	1996	1	Mylo Xyloto
1711	Coldplay	Mylo Xyloto	1996	2	Hurts Like Heaven
1711	Coldplay	Mylo Xyloto	1996	3	Paradise
1712	Nickelback	Here And Now	1995	1	This Means War
1712	Nickelback	Here And Now	1995	2	Bottoms Up

Die Relation ‘CD_Sammlung‘ wird dieser Bedingung nicht gerecht.

Verstoß gegen 2NF, da die Attribute Interpret, Titel und Gründung ausschließlich vom Schlüsselteil CD_Nr eindeutig identifiziert werden können und nicht vom gesamten Primärschlüssel CD_Nr und Song_Nr.

Lösung:
Trennung nach Sachverhalten

- Ermittlung der Kardinalitäten
- Identifizierung der Schlüsselattribute.
- N:M auflösen → wird zu 1:N und N:1

Hier:
Aufteilung in die Relationen ‘Song‘ und ‘CD‘.

Relation Song

CD_Nr	**Song_Nr**	**Titel**
1711	1	Mylo Xyloto
1711	2	Hurts Like Heaven
1711	3	Paradise
1712	1	This Means War
1712	2	Bottoms Up

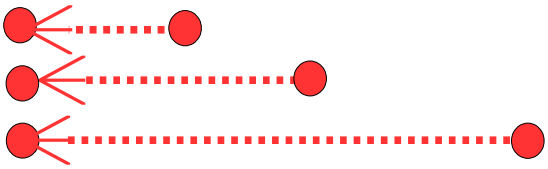
Arbeitsauftrag: Die Aufgaben beziehen sich auf die Datenbank-Struktur und Daten (Personal_NF2_Start)

1. Importieren Sie die Struktur und Daten.
2. Testen Sie die folgende Abfragen.
3. Prüfen Sie das Ergebnis, welches Risiko besteht?
4. Welche Veränderungen sind notwendig?
5. Was sagt Ihr Gefühl sind die behandelten Strukturen nun optimal?

Relation CD			
CD_Nr	**Interpret**	**Album**	**Jahr der Gründung**
1711	Coldplay	Mylo Xyloto	1996
1712	Nickelback	Here And Now	1995

```
UPDATE Mitarbeiter
SET abt_bez = 'Human Ressources'
WHERE abt_bez ='Personal';
```

```
SELECT  idMitarbeiter, name, abt_nr, abt_bez
        FROM    Mitarbeiter
        WHERE   projekt_bez1 = "Installation Intranet"
        OR      projekt_bez2 = "Installation Intranet"
        OR      projekt_bez3 = "Installation Intranet";
```



Thema:	Normalisierung im Kontext Relationaler Datenbanken Handreichung: Relationale Datenbanken, LPE03, 20131105 Exkurs: 3. Normalform

Eine Relation befindet sich in 3. Normalform, wenn sie sich in 2NF befindet und kein Nicht-Schlüsselattribut in funktionaler Abhängigkeit zu einem anderen Nicht-Schlüsselattribut steht.

Im Gegensatz zur 2. Normalform, bei der die Abhängigkeit der Nicht-Schlüsselattribute von den Schlüsselattributen untersucht wird, wird hier geprüft, dass keine zusätzlichen Beziehungen zwischen zwei Nicht-Schlüsselattributen bestehen.

Relation CD			
CD_Nr	**Interpret**	**Album**	**Jahr der Gründung**
1711	Coldplay	Mylo Xyloto	1996
1712	Nickelback	Here And Now	1995

Beispiel:
Es existieren zwei Attribute B und C, die gemäß 2NF funktional vom Primärschlüssel A abhängig sind. Ist das Attribut C nun zusätzlich vom Attribut B abhängig, entsteht eine indirekte (transitive) Abhängigkeit des Nicht-Schlüsselattributs C vom Primärschlüssel A. (Schreibweise: *A* ☐ *B* ☐ C)

Die 3. Normalform verbietet solche transitiven Abhängigkeiten. Die Relation ‘CD‘ wird daher den Regeln der 3. Normalform nicht gerecht:
Verstoß gegen 3NF, da das Gründungsjahr vom Interpreten und damit transitiv von der CD_Nr abhängig sind

Lösung:

Relation Song		
CD_Nr	**Song_Nr**	**Titel**
1711	1	Mylo Xyloto
1711	2	Hurts Like Heaven
1711	3	Paradise
1712	1	This Means War
1712	2	Bottoms Up

Erweiterung der Tabelle (N) in den 1:N-Kardinalitäten um das Fremdschlüsselattribut. Hier:Aufteilung in die Relationen ‘Kuenstler‘ und ‘CD‘.

Relation Kuenstler		
K_Nr	**Interpret**	**Gründung**
1000	Coldplay	1996
1001	Nickelback	1995

Relation CD		
CD_Nr	**Album**	**K_Nr_fk**
1711	Mylo Xyloto	1000
1712	Here And Now	1001

Arbeitsauftrag: Die Aufgaben beziehen sich auf die Datenbank-Struktur und Daten (Personal_NF3_Start)

1. Analysieren Sie die Personaldaten und erzeugen Sie die Datenbankstruktur (EERM) der Personaldaten in der 3.NF.
2. Erzeugen Sie die Datenbasis (INSERT INTO-Befehle) für die Struktur.
3. Importieren Sie Struktur und Daten.
4. Worauf müssen wir achten?
5. Dokumentieren Sie abschließend die Ziele der Normalisierung und die Risiken für den Fall der Missachtung (Anomalien).