MySQL – Informationsgewinnung
Teil 2

Stefan Maihack Dipl. Ing. (FH)
Datum: 28.4.2013

Die JOIN-Anweisung

Einführung

- JOINS dienen dazu Daten aus mehreren Tabellen gleichzeitig abzurufen
- WICHTIG: Die Tabellen müssen durch gemeinsame Daten in Beziehung stehen.

Beispiel: Autorenname aus Tabelle **autor** und Buchtitel aus Tabelle **buch**.
→ Tabelle **autor** hat Primärschlüssel **autor_nummer**.
→ Tabelle **buch** hat Fremdschlüssel **autor_nummer**.

Abarbeitung:
(1) MySQL kombiniert alle Zeilen von **buch** mit allen Zeilen von **autor**.
(2) aus der Ergebnismenge alle Zeilen aussuchen, die die Bedingung erfüllen.

Die JOIN-Anweisung

Varianten

- Es gibt fünf prinzipielle Varianten:

  1. INNER JOIN (Equi Join): Zeilen mit übereinstimmenden Werten
  SELECT p.besteller, p.menge, p.artikelnr a.bezeichnung FROM posten p, artikel a
  WHERE p.artikelnr = a.artikelnr;

  2. LEFT JOIN (Linksverknüpfung): Gibt alle Zeilen der Tabelle auf der linken Seite der Join-Anweisung.
  SELECT p.bestellnr, p.menge, p.artikelnr, a.bezeichnung FROM **posten** p
  LEFT JOIN artikel a ON p.artikelnr = a.artikelnr;

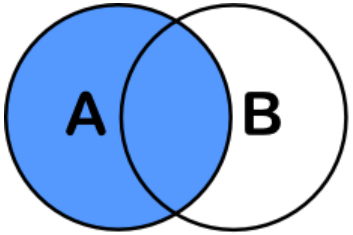
  3. NATURAL JOIN (natürliche Verknüpfung): Kurzform von Left Join; nur Übereinstimmungen werden ausgegeben.
  SELECT * FROM kunden NATURAL JOIN rechnung;

  4. RIGHT JOIN (Rechtsverknüpfung): Funktioniert analog LEFT JOIN; aus Kompatibilitätsgründen LEFT JOIN statt RIGHT JOIN nutzen.

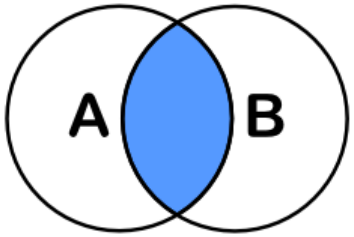
  5. CROSS JOIN: Kreuzverknüpfung: Jede Zeile der Tabelle wird mit jeder Zeile einer anderen Tabelle verknüpft. Sehr große Datenmenge; meistens nicht sinnvoll.
  SELECT * FROM kunden, rechnung;

Die JOIN-Anweisung

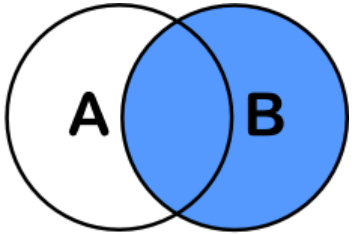
Varianten



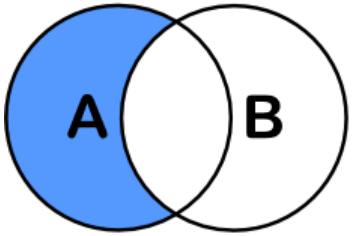

```
SELECT <auswahl>
FROM tabelleA A
LEFT JOIN tabelleB B
ON A.key = B.key
```



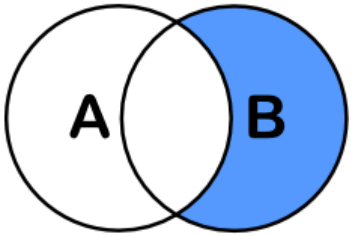

```
SELECT <auswahl>
FROM tabelleA A
INNER JOIN tabelleB B
ON A.key = B.key
```



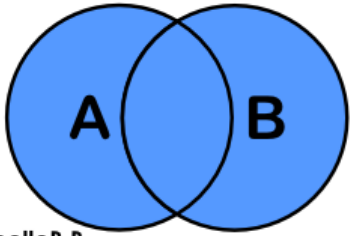

```
SELECT <auswahl>
FROM tabelleA A
RIGHT JOIN tabelleB B
ON A.key = B.key
```



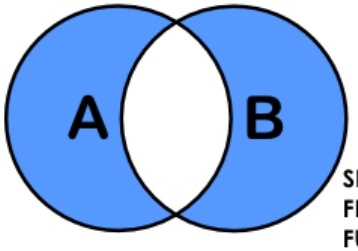

```
SELECT <auswahl>
FROM tabelleA A
LEFT JOIN tabelleB B
ON A.key = B.key
WHERE B.key IS NULL
```




```
SELECT <auswahl>
FROM tabelleA A
RIGHT JOIN tabelleB B
ON A.key = B.key
WHERE A.key IS NULL
```




```
SELECT <auswahl>
FROM tabelleA A
FULL OUTER JOIN tabelleB B
ON A.key = B.key
```




```
SELECT <auswahl>
FROM tabelleA A
FULL OUTER JOIN tabelleB B
ON A.key = B.key
WHERE A.key IS NULL
OR B.key IS NULL
```

Die JOIN-Anweisung

Inner Join oder Equi Join

- Der am häufigsten verwendete Join-Typ.

- In diesem Beispiel werden Details zu den Rechnungen der Kunden ausgegeben:
  SELECT * FROM rechnung r, kunden k WHERE r.kdnr = k.kdnr;

```
mysql> SELECT * FROM rechnung r, kunden k WHERE r.kdnr = k.kdnr;
+--------+------+------------+------------+------+----------+----------+------+
| rechnr | kdnr | rechdat    | rechbetrag | kdnr | kdname   | ort      | vnr  |
+--------+------+------------+------------+------+----------+----------+------+
|     81 | 2133 | 2001-07-20 |   10600.00 | 2133 | Meier    | Bayreuth |  224 |
|     83 | 2133 | 2004-08-20 |    1000.00 | 2133 | Meier    | Bayreuth |  224 |
|     82 | 7533 | 2023-07-20 |   28400.00 | 7533 | Schmitz  | K├ôln    |  115 |
|     84 | 8511 | 2005-08-20 |   20900.00 | 8511 | Schneider | Bonn     |  115 |
+--------+------+------------+------------+------+----------+----------+------+
4 rows in set (0.01 sec)

mysql>
```

- Oder wenn nur bestimmte Spalten auszugeben sind:
  SELECT r.rechnr, r.rechbetrag, k.kdnr FROM rechnung r, kunden k WHERE r.kdnr = k.kdnr;

```
mysql> SELECT r.rechnr, r.rechbetrag, k.kdnr FROM rechnung r, kunden k WHERE r.kdnr = k.kdnr;
+--------+------------+------+
| rechnr | rechbetrag | kdnr |
+--------+------------+------+
|     81 |   10600.00 | 2133 |
|     82 |   28400.00 | 7533 |
|     83 |    1000.00 | 2133 |
|     84 |   20900.00 | 8511 |
+--------+------------+------+
4 rows in set (0.00 sec)

mysql>
```

Die JOIN-Anweisung

Left Join

- Der Left Join ergänzt die Ergebnismenge eines Equi-Join, um die Datensätze aus der ersten Tabelle, die keinen einzigen passenden Eintrag in der zweiten Tabelle haben.

  SELECT r.rechnr, r.rechbetrag, k.kdnr FROM kunden k LEFT JOIN rechnung r ON r.kdnr = k.kdnr;

```
mysql> SELECT r.rechnr, r.rechbetrag, k.kdnr FROM kunden k LEFT JOIN rechnung r ON r.kdnr = k.kdnr;
+--------+------------+------+
| rechnr | rechbetrag | kdnr |
+--------+------------+------+
|     81 |   10600.00 | 2133 |
|     83 |    1000.00 | 2133 |
|   NULL |       NULL | 3557 |
|     82 |   28400.00 | 7533 |
|     84 |   20900.00 | 8511 |
+--------+------------+------+
5 rows in set (0.01 sec)

mysql>
```

- Der Wert „NULL“ steht in MySQL für NICHT BESETZT. Er hat nichts mit der Zahl „0“ zu tun.
- Den NULL-Wert kann man für die Auswertung des Problems verwenden. Bei einer Equi-Anweisung werden diese Zeilen einfach unterdrückt.
- SELECT r.rechnr, r.rechbetrag, k.kdnr FROM kunden k LEFT JOIN rechnung r ON r.kdnr = k.kdnr WHERE r.rechnr IS NULL;

```
+--------+------------+------+
| rechnr | rechbetrag | kdnr |
+--------+------------+------+
|   NULL |       NULL | 3557 |
+--------+------------+------+
1 row in set (0.00 sec)
```

Die JOIN-Anweisung

weitere Beispiele

- Folgende Tabellen der Datenbank „kunde_rechnung.sql“ liegen vor:

Relation: Rechnung

Rechnr	KdNr	Rechdat	RechBetrag
0081	2133	1.7.07	10600
0082	7533	23.7.07	28400
0083	2133	4.8.07	1000
0084	8511	5.8.07	20900

Relation: Kunden

KdNr	KdName	Ort	VNr
2133	Meier	Bayreuth	224
3557	Müller	Bonn	115
7533	Schmitz	Köln	115
8511	Schneider	Bonn	115

Relation: Lieferprogramm

Liefnr	Teilenr
1	0009
1	0010
2	0009
2	0010
2	0011
3	0003
3	0004
4	0012

Relation: Vertreter

Vnr	Vname	Bezirk	UmsSum
211	Berger	Nord	74800
115	Adam	Mitte	127900
224	Zeiss	Süd	91750
315	Beck	Ost	0
452	Weih	West	10000

Relation: Lieferant

Liefnr	Liefname
1	Alu-GmbH
2	Bike-GmbH
3	Cross-GmbH
4	Wichtek&Sachs
5	Rad AG

Die JOIN-Anweisung

weitere Beispiele – Inner Join / Equi Join

```
mysql> SELECT * FROM lieferant;
+--------+---------------+
| liefnr | liefname      |
+--------+---------------+
|      1 | Alu-GmbH      |
|      2 | Bike-GmbH     |
|      3 | Cross-GmbH    |
|      4 | Wichtel&Sachs |
|      5 | Rad AG        |
+--------+---------------+
5 rows in set (0.07 sec)

mysql>
```

```
mysql> SELECT * FROM lieferprogramm;;
+--------+---------+
| liefnr | teilenr |
+--------+---------+
|      1 |       9 |
|      1 |      10 |
|      2 |       9 |
|      2 |      10 |
|      2 |      11 |
|      3 |       3 |
|      3 |       4 |
|      4 |      12 |
+--------+---------+
8 rows in set (0.01 sec)

ERROR:
No query specified

mysql>
```

→ Inner Join über die Tabellen „lieferant“ und „lieferprogramm“.

→ Der Lieferant „5 – Rad AG“ wurde hier nicht ausgegeben, da er bisher nichts geliefert hat.

```
mysql>
mysql> SELECT lieferant.liefnr, lieferant.liefname, lieferprogramm.teilenr FROM lieferant, lieferprogramm WHERE lieferprogramm.liefnr = lieferant.liefnr;
+--------+---------------+---------+
| liefnr | liefname      | teilenr |
+--------+---------------+---------+
|      1 | Alu-GmbH      |       9 |
|      1 | Alu-GmbH      |      10 |
|      2 | Bike-GmbH     |       9 |
|      2 | Bike-GmbH     |      10 |
|      2 | Bike-GmbH     |      11 |
|      3 | Cross-GmbH    |       3 |
|      3 | Cross-GmbH    |       4 |
|      4 | Wichtel&Sachs |      12 |
+--------+---------------+---------+
8 rows in set (0.01 sec)

mysql>
```

Die JOIN-Anweisung

weitere Beispiele – Right outer Join

```
mysql>
mysql> SELECT lieferant.liefnr, lieferant.liefname, lieferprogramm.teilenr FROM lieferant
    -> RIGHT OUTER JOIN lieferprogramm ON lieferant.liefnr = lieferprogramm.liefnr;
+--------+--------------+---------+
| liefnr | liefname     | teilenr |
+--------+--------------+---------+
|      1 | Alu-GmbH     |       9 |
|      1 | Alu-GmbH     |      10 |
|      2 | Bike-GmbH    |       9 |
|      2 | Bike-GmbH    |      10 |
|      2 | Bike-GmbH    |      11 |
|      3 | Cross-GmbH   |       3 |
|      3 | Cross-GmbH   |       4 |
|      4 | Wichtel&Sachs|      12 |
+--------+--------------+---------+
8 rows in set (0.00 sec)

mysql>
```

→ Gleiches Ergebnis wie beim Inner Join.

→ Warum wurde der 5. Lieferant nicht ausgegeben und wann würde er ausgegeben?

Die JOIN-Anweisung

weitere Beispiele – Left outer Join

```
mysql> SELECT lieferant.liefnr, lieferant.liefname, lieferprogramm.teilenr FROM lieferant
    -> LEFT OUTER JOIN lieferprogramm ON lieferant.liefnr = lieferprogramm.liefnr;
+--------+---------------+---------+
| liefnr | liefname      | teilenr |
+--------+---------------+---------+
|      1 | Alu-GmbH      |       9 |
|      1 | Alu-GmbH      |      10 |
|      2 | Bike-GmbH     |       9 |
|      2 | Bike-GmbH     |      10 |
|      2 | Bike-GmbH     |      11 |
|      3 | Cross-GmbH    |       3 |
|      3 | Cross-GmbH    |       4 |
|      4 | Wichtel&Sachs |      12 |
|      5 | Rad AG        |    NULL |
+--------+---------------+---------+
9 rows in set (0.00 sec)

mysql>
```

NULL-Eintrag, da es keine Beziehung hierzu gibt.

Die SELECT-Anweisung

Übungsaufgabe

- Folgende Tabellen der Datenbank „world.sql“ liegen vor:

Relation: City

ID	Name	CountryCode	District	Population
1	Kabul	AFG	Kabul	1780000
2	Quandahar	AFG	Quandahar	237500
3	Herat	AFG	Herat	186800
…	…	…	…	…

Relation: CountryLanguage

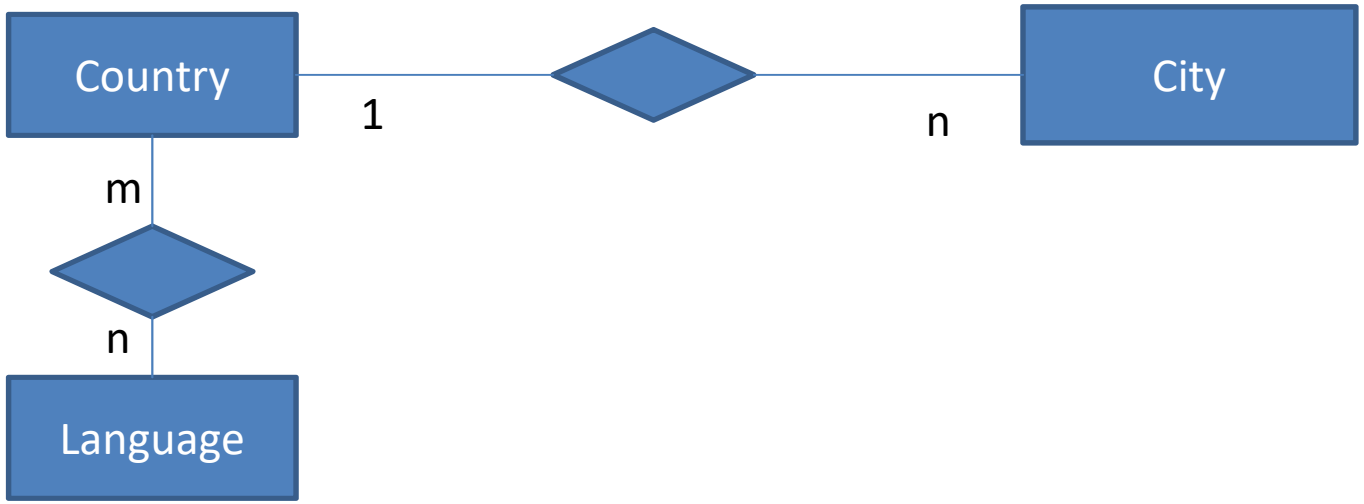
CountryCode	Language	IsOfficial	Percentage
ABW	Dutch	T	5.3
ABW	English	F	9.5
ABW	Papiamento	F	76.7
…	…	…	…

Relation: Country

Code	Name	Continent	Region	…
ABW	Anuba	North Amerika	Caribbean	…
AFG	Afganistan	Asia	Southern and Central Asia	…
AGO	Angola	Africa	Central Afrika	…
…	…	…	…	…

Die SELECT-Anweisung

Übungsaufgabe. ER-Darstellung



Die JOIN-Anweisung

Übungsaufgaben – Equi Join

- Lösen sie folgende Aufgaben bezogen auf die Datenbank „world“

- Geben sie alle Städte und deren Länder jeweils in einer Zeile aus. Es sollen nur die Spalten „CountryCode“, „Name“ und „Continent ausgegeben werden. Achten sie darauf, dass nur die ersten 20 Zeilen ausgegeben werden.
- SELECT y.CountryCode, y.Name, c.continent FROM city y , country c WHERE y.countrycode = c.code limit 20;

CountryCode	Name	continent
AFG	Kabul	Asia
AFG	Qandahar	Asia
AFG	Herat	Asia
AFG	Mazar-e-Sharif	Asia
NLD	Amsterdam	Europe
NLD	Rotterdam	Europe
NLD	Haag	Europe
NLD	Utrecht	Europe
NLD	Eindhoven	Europe
NLD	Tilburg	Europe
NLD	Groningen	Europe
NLD	Breda	Europe
NLD	Apeldoorn	Europe
NLD	Nijmegen	Europe
NLD	Enschede	Europe
NLD	Haarlem	Europe
NLD	Almere	Europe
NLD	Arnhem	Europe
NLD	Zaanstad	Europe
NLD	's-Hertogenbosch	Europe

20 rows in set (0.00 sec)

Die JOIN-Anweisung

Übungsaufgaben – Equi Join

- Geben Sie nun in einer Joinbedingung die Spalten „CountryCode“, „Stadtname“, „Continent“ und „language“ aus. Equi Join.

- SELECT y.CountryCode, y.Name, c.continent, l.language FROM city y , country c, countrylanguage l WHERE y.countrycode = c.code AND y.countrycode = l.countrycode LIMIT 20;

CountryCode	Name	continent	language
AFG	Kabul	Asia	Balochi
AFG	Kabul	Asia	Dari
AFG	Kabul	Asia	Pashto
AFG	Kabul	Asia	Turkmenian
AFG	Kabul	Asia	Uzbek
AFG	Qandahar	Asia	Balochi
AFG	Qandahar	Asia	Dari
AFG	Qandahar	Asia	Pashto
AFG	Qandahar	Asia	Turkmenian
AFG	Qandahar	Asia	Uzbek
AFG	Herat	Asia	Balochi
AFG	Herat	Asia	Dari
AFG	Herat	Asia	Pashto
AFG	Herat	Asia	Turkmenian
AFG	Herat	Asia	Uzbek
AFG	Mazar-e-Sharif	Asia	Balochi
AFG	Mazar-e-Sharif	Asia	Dari
AFG	Mazar-e-Sharif	Asia	Pashto
AFG	Mazar-e-Sharif	Asia	Turkmenian
AFG	Mazar-e-Sharif	Asia	Uzbek

20 rows in set (0.00 sec)

Die JOIN-Anweisung

Übungsaufgaben – Equi Join

- Geben Sie nun die Städte aus , bei denen die Population < 10.000 ist.

- SELECT y.CountryCode, y.Name, c.continent, l.language, c.population FROM city y , country c, countrylanguage l WHERE y.countrycode = c.code AND y.countrycode = l.countrycode AND c.population < 10000;

CountryCode	Name	continent	language	population
AIA	South Hill	North America	English	8000
AIA	The Valley	North America	English	8000
FLK	Stanley	South America	English	2000
SJM	Longyearbyen	Europe	Norwegian	3200
SJM	Longyearbyen	Europe	Russian	3200
CXR	Flying Fish Cove	Oceania	Chinese	2500
CXR	Flying Fish Cove	Oceania	English	2500
CCK	Bantam	Oceania	English	600
CCK	Bantam	Oceania	Malay	600
CCK	West Island	Oceania	English	600
CCK	West Island	Oceania	Malay	600
NIU	Alofi	Oceania	English	2000
NIU	Alofi	Oceania	Niue	2000
NFK	Kingston	Oceania	English	2000
PCN	Adamstown	Oceania	Pitcairnese	50
SHN	Jamestown	Africa	English	6000
SPM	Saint-Pierre	North America	French	7000
TKL	Fakaofo	Oceania	English	2000
TKL	Fakaofo	Oceania	Tokelau	2000
VAT	Città del Vaticano	Europe	Italian	1000

20 rows in set (0.09 sec)