

JOIN-Operationen

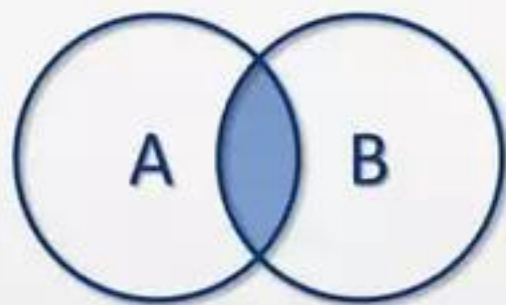
Inner und Outer Joins

Definition

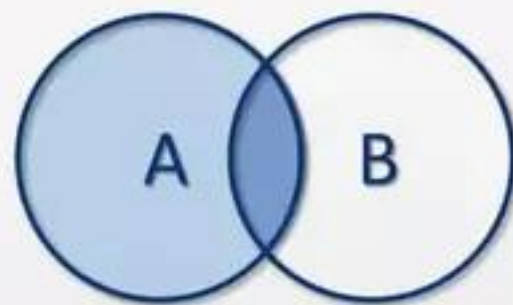
JOIN-Operationen in SQL ermöglichen es, **Daten aus mehreren Tabellen zu kombinieren**, um zusammengehörige Informationen abzurufen. Diese Operationen sind nützlich, wenn Daten in verschiedenen Tabellen verteilt sind und miteinander in Beziehung stehen.

Es gibt verschiedene Arten von JOINS, darunter **INNER JOIN**, **LEFT JOIN** (auch **LEFT OUTER JOIN** genannt), **RIGHT JOIN** (auch **RIGHT OUTER JOIN** genannt) und **FULL JOIN** (auch **FULL OUTER JOIN** genannt).

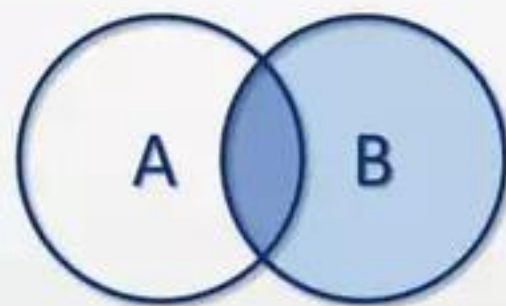
INNER JOIN



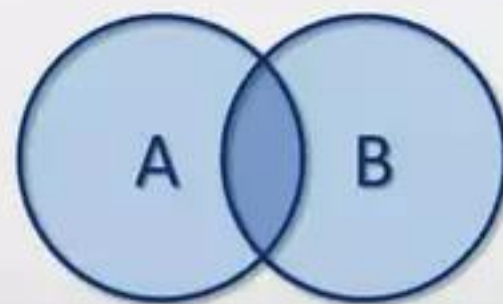
OUTER JOIN



LEFT



RIGHT



FULL

Tabellen

Tabelle Kunde

<u>id</u>	vorname	nachname
1	John	von Neumann
2	Grace	Hopper
3	Donald	Knuth
5	Dennis	Richie

Tabelle Bestellung

<u>id</u>	Produkt	kunden_id	Datum
1	Teekanne	1	12.12.2022
2	Snickers XXL	1	01.10.2022
3	Lightbulp	2	10.03.2011
4	CPU	1	10.05.2012
5	Hut	2	10.04.2023
6	TV	3	10.04.2023

Inner Join

Ein **INNER JOIN** gibt nur die Zeilen zurück, für die eine Übereinstimmung in beiden verknüpften Tabellen vorhanden ist. Es schließt nicht übereinstimmende Zeilen aus. Gibt nur die Datensätze zurück, die in beiden Tabellen übereinstimmen.

INNER JOIN kombiniert Zeilen aus beiden Tabellen, wenn die Bedingung im ON-Abschnitt erfüllt ist.

```
SELECT Kunde.vorname, Bestellung.datum  
FROM Kunde  
INNER JOIN Bestellung ON Kunde.id = Bestellung.kunden_id;
```

Diese Abfrage gibt eine Liste der Kunden und ihrer Bestelldaten zurück, wobei nur Kunden angezeigt werden, die Bestellungen getätigt haben.

LEFT (OUTER) JOIN

Ein LEFT JOIN gibt **alle Zeilen aus der linken** (ersten) Tabelle und die **übereinstimmenden Zeilen aus der rechten** (zweiten) Tabelle zurück. Wenn keine Übereinstimmung in der rechten Tabelle gefunden wird, werden **NULL-Werte** angezeigt.

```
SELECT Kunde.vorname, Bestellung.datum  
FROM Kunde  
LEFT JOIN Bestellung ON Kunde.id = Bestellung.kunden_id;
```

Diese Abfrage gibt eine Liste aller Kunden zurück, einschließlich derer, die keine Bestellungen getätigt haben. **Die Spalte "datum" zeigt NULL-Werte für Kunden ohne Bestellungen.**

RIGHT (OUTER) JOIN

Ein **RIGHT JOIN** ist das Gegenstück zum LEFT JOIN. Es gibt alle Zeilen aus der rechten Tabelle und die übereinstimmenden Zeilen aus der linken Tabelle zurück. Wenn keine Übereinstimmung in der linken Tabelle gefunden wird, werden NULL-Werte angezeigt.

```
SELECT Kunde.vorname, Bestellung.datum  
FROM Kunde  
RIGHT JOIN Bestellung ON Kunde.id = Bestellung.kunden_id;
```

Diese Abfrage gibt eine Liste aller Bestellungen zurück, einschließlich derer, die keinem Kunden zugeordnet sind. Die Spalte "Vorname" zeigt NULL-Werte für Bestellungen ohne zugehörigen Kunden.

FULL (OUTER) JOIN

Ein FULL JOIN gibt alle Zeilen aus beiden Tabellen zurück, einschließlich nicht übereinstimmender Zeilen. Wenn keine Übereinstimmung in einer Tabelle gefunden wird, werden NULL-Werte angezeigt.

```
SELECT Kunde.Name, Bestellung.datum  
FROM Kunde  
FULL JOIN Bestellung ON Kunde.id = Bestellung.kunden_id;
```

Diese Abfrage gibt eine Liste aller Kunden und Bestellungen zurück, einschließlich derer, die keine Übereinstimmung haben. Die Spalten "Name" und "Bestelldatum" zeigen NULL-Werte für nicht übereinstimmende Zeilen. **MySQL unterstützt keinen FULL JOIN.**

FULL JOIN in Mysql mit Union

Mit UNION sowie einem LEFT und RIGHT JOIN kann ein Full Join in MySQL erreicht werden:

```
SELECT Kunde.vorname, Bestellung.datum  
FROM Kunde  
LEFT JOIN Bestellung ON Kunde.id = Bestellung.kunden_id
```

UNION

```
SELECT Kunde.vorname, Bestellung.datum  
FROM Kunde  
RIGHT JOIN Bestellung ON Kunde.id = Bestellung.kunden_id;
```