



---

## RELATIONEN

**Fragen?**

Quiz zu Relation.

Gegeben sei die Menge  $M = \{1, 2, 3\}$  und die Relation  $R$  auf  $M$  mit  $R = \{(1, 2), (2, 1), (2, 3), (1, 3)\}$ . Kreuze alle wahren Aussagen an!

☐  $R$  ist asymmetrisch.

☐  $R$  ist antisymmetrisch.

☐  $R$  ist transitiv.

☐  $R$  ist irreflexiv.

Lösung überprüfen

Lösung.

Eigener Lösungsversuch.

**\* Relationen auf  $\mathbb{Z}$ .** Betrachten Sie die Relationen auf  $\mathbb{Z}$  in der linken Spalte der folgenden Tabelle. Füllen Sie dann folgende Tabelle aus:

	reflexiv?	irrefl.?	symm.?	asymm.?	antisymm.?	trans.?	Äqu. Rel.?	Ordnung?
$<$								
$\leq$								
$=$								
$\equiv$								
$\mid$								

**Eigener Lösungsversuch.**

	reflexiv?	irrefl.?	symm.?	asymm.?	antisymm.?	trans.?	Äqu. Rel.?	Ordnung?
$<$								
$\leq$								
$=$								
$\equiv$								
$\mid$								

## Zusammenhang: Funktion und Relation

Von einer Funktion

$$\begin{aligned} f : A &\rightarrow B \\ x &\mapsto f(x) \end{aligned}$$

bildet der Graph

$$R_f := \{(x, f(x)) \mid x \in A\} \subseteq A \times B$$

eine Relation, die sogenannte **von f induzierte Relation**. Beispiel:

**Graph der Parabel als Relation.** Geben Sie die von  $f : \mathbb{R} \rightarrow \mathbb{R}, x \mapsto x^2$  induzierte Relation an und skizzieren Sie diese.

**Lösung.**

Dagegen kann man aus einer Relation i.A. keine Funktion konstruieren. Beispiel:

$$R = \{(-2, 4), (0, 5), (-2, 6), (1, 8)\} \subseteq \{-2, 0, 1\} \times \{4, 5, 6, 8\}$$

Ein weiteres Beispiel wird in den Übungen besprochen.

**Anwendung: relationale Datenbanken.** Eine Relation kann man als auch eine Tabelle interpretieren, also:

Beispiel: Tabellen **product** und **manuf** als Relationen

id	name	price	id_manuf
1	iPhone	600.59	
2	PC	499.0	
3	Server AIX	9999.00	
4	Drucker	95.0	

id	name	city
1	Apple	Cupertino
2	IBM	NY
3	HP	Palo Alto

Verschiedene Operationen ( $\rightarrow$  relationale Algebra und SQL) liefern daraus neue Tabellen, z.B. ein JOIN der beiden Tabellen liefert eine neue Tabelle/Relation:

SQL: `SELECT * FROM product JOIN manuf ON product.id_manuf=manuf.id`

liefert das Ergebnis

product.id	name	price	id_manuf	manuf.id	name	city
1	iPhone	600.59	1	1	Apple	Cupertino
2	PC	499.0	2	2	IBM	NY
3	Server AIX	9999.00	2	2	IBM	NY
4	Drucker	95.0	3	3	HP	Palo Alto

## Demo Datenbank-Skript.

```
-----
-- DDL --
-----
CREATE TABLE manuf (
    id INTEGER NOT NULL,
    name varchar(45) DEFAULT NULL,
    city varchar(45) DEFAULT NULL,
    PRIMARY KEY (id)
);
CREATE TABLE product (
    id INTEGER NOT NULL,
    name varchar(45) DEFAULT NULL,
    price float DEFAULT NULL,
    id_manuf INTEGER DEFAULT NULL,
    PRIMARY KEY (id),
    CONSTRAINT fk_manuf FOREIGN KEY (id_manuf) REFERENCES manuf (id)
);
-----
-- DATA --
-----
INSERT INTO manuf (id,name,city) VALUES (1,'Apple','Cupertino');
INSERT INTO manuf (id,name,city) VALUES (2,'IBM','NY');
INSERT INTO manuf (id,name,city) VALUES (3,'HP','Palo Alto');
INSERT INTO product (id,name,price,id_manuf) VALUES (1,'iPhone',600,1);
INSERT INTO product (id,name,price,id_manuf) VALUES (2,'PC',500,2);
INSERT INTO product (id,name,price,id_manuf) VALUES (3,'Server',10000,2);
INSERT INTO product (id,name,price,id_manuf) VALUES (4,'Drucker',400,3);
COMMIT;
-----
-- SQL --
-----
SELECT * FROM product; -- Eine ganze Tabelle lesen
SELECT * FROM product WHERE price < 5000; -- Einen Teil einer Tabelle lesen
SELECT city FROM manuf; -- Projektion auf eine Spalte

-- JOIN zweier Tabellen:
SELECT * FROM product
    LEFT JOIN manuf
        ON product.id_manuf = manuf.id;

-- LEFT JOIN zweier Tabellen:
SELECT * FROM product
    LEFT JOIN manuf
        ON product.id_manuf = manuf.id;
```