

**Datenbankabfragesprachen
Übungsserie 02**

1.) Was versteht man unter einer Relation?

2.) Sei $A := \{1, 2\}$, $B := \{\text{eins}, \text{zwei}\}$

Geben Sie sämtliche Elemente von $A \times B$ an. (Kreuzprodukt)

3.) Wozu dient ein Primärschlüssel?

4.) Wie kann man Fremdschlüssel einsetzen, um referenzielle Integrität zu erreichen?

5a) Entwerfen Sie eine Datenbankstruktur mit folgenden Tabellen:

Film:

ID: Primärschlüssel
Drehort_ID: Fremdschlüssel
Regisseur_ID: Fremdschlüssel
Titel
Länge

Regisseur:

ID: Primärschlüssel
Name
Wohnort

Drehort:

ID: Primärschlüssel
Name

Kino:

ID: Primärschlüssel
Name
Ort
Straße

DVD:

ID: Primärschlüssel
Publisher

Veröffentlichung

Film_ID: Fremdschlüssel
Kino_ID: Fremdschlüssel
DVD_ID: Fremdschlüssel
Jahr

5b) Füllen Sie die Tabellen wie folgt mit Daten

Film

ID	Titel	Drehort_ID	Regisseur_ID	Länge
1	Spaceballs2	A	23	100
2	Faust3	C	11	1024
3	3 Katzen und ein Hund	A	23	90
4	Polizeihund 110	K	10	93

Regisseur

ID	Name	Wohnort
10	Sven Angstrom	Stockholm
11	Pierre Macron	Paris
15	Donald Obama	Austin
23	Melania Gates	NYC
30	Michael Putin	St. Petersburg

Drehort

ID	Name
A	Babelsberg
B	Bielefeld
C	NYC
K	Boston

Kino

ID	Name	Ort	Straße
100	Filmu Pasaule	Riga	Negu Iela 7
200	Filmpalast Lehmann	Berlin	Alexanderplatz 1
300	Palais de Cinema	Paris	129 Rue de Grenelle

DVD

ID	Publisher
1000	Lohengrin Pictures
2000	Laser Land
3000	A.Y.B.A.B.T.U

Veröffentlichung

Film_ID	Kino_ID	DVD_ID	Jahr
1	100	NULL	2023
1	NULL	1000	2023
2	300	NULL	1998
4	NULL	3000	2003

5c) Welche Filmtitel wurden auf DVD veröffentlicht?

5d) In welchem Kino / Kinos wurden Filme von Melania Gates gezeigt?

Hilfe:

Ergebnis für Aufgabe 3:

Titel
Spaceballs 2
Polizeihund 110

Ergebnis für Aufgabe 4:

Name	Regisseur_ID
Filmu Pausale	23

Lösungshinweise:

- 1.) Eine Relation ist eine Menge aus Tupeln, deren Tupelelemente in einer gewissen Beziehung zueinander stehen.
- 2.) $M = \{ (1, \text{eins}), (1, \text{zwei}), (2, \text{eins}), (2, \text{zwei}) \}$
- 3.) Ein Primärschlüssel dient zum eindeutigen Kennzeichnen eines Datensatzes.
- 4.) Mittels eines Fremdschlüssels kann man sicherstellen, dass Verweise auf Datensätze anderer Tabellen nur auf existierende Schlüssel zeigen.

5a)

```
create table Regisseur (  
  ID int(10) primary key,  
  Name varchar(255),  
  Wohnort varchar(255)  
);
```

```
create table Drehort(  
  ID varchar(1) primary key,  
  Name varchar(255)  
);
```

```
create table Film(  
  ID int(10) primary key,  
  Titel varchar(255),  
  Drehort_ID varchar(1),  
  Regisseur_ID int(10),  
  Länge int(10),  
  foreign key(Drehort_ID) references Drehort(ID),  
  foreign key(Regisseur_ID) references Regisseur(ID)  
);
```

```
create table DVD(  
  ID int(10) primary key,  
  Publisher varchar(255)  
);
```

```
create table Kino(  
  ID int(10) primary key,  
  Name varchar(255),  
  Ort varchar(255),  
  Straße varchar(255)  
);
```

```
create table Veröffentlichung(  
  Film_ID int(10),  
  Kino_ID int(10),  
  DVD_ID int(10),  
  Jahr int(10),  
  foreign key(Film_ID) references Film(ID),  
  foreign key(Kino_ID) references Kino(ID),  
  foreign key(DVD_ID) references DVD(ID)  
);
```

```
INSERT INTO Film (ID, Titel, Drehort_ID, Regisseur_ID, Länge) VALUES
```

```
(1, 'Spaceballs2', 'A', '23', '100'),
(2, 'Faust3', 'C', '11', '1024'),
(3, '3 Katzen und ein Hund', 'A', '23', '90'),
(4, 'Polizeihund 110', 'K', '10', '93');
```

```
INSERT INTO Regisseur (ID, Name, Wohnort) VALUES
(10, 'Sven Angstrom', 'Stockholm'),
(11, 'Pierre Macron', 'Paris'),
(15, 'Donald Obama', 'Austin'),
(23, 'Melania Gates', 'NYC'),
(30, 'Michael Putin', 'St. Petersburg');
```

```
INSERT INTO Drehort (ID, Name) VALUES
(A, 'Babelsberg'),
(B, 'Bielefeld'),
(C, 'NYC'),
(K, 'Boston');
```

```
INSERT INTO Kino (ID, Name, Ort, Straße) VALUES
(100, 'Filmu Pasaule', 'Riga', 'Negu Iela 7'),
(200, 'Filmpalast Lehmann', 'Berlin', 'Alexanderplatz 1'),
(300, 'Palais de Cinema', 'Paris', '129 Rue de Grenelle');
```

```
INSERT INTO DVD (ID, Publisher) VALUES
(1000, 'Lohengrin Pictures'),
(2000, 'Laser Land'),
(3000, 'A.Y.B.A.B.T.U');
```

```
INSERT INTO Veröffentlichung (Film_ID, Kino_ID, DVD_ID, Jahr) VALUES
(1, '100', NULL, '2023'),
(1, NULL, '1000', '2023'),
(2, '300', NULL, '1998'),
(4, NULL, '3000', '2003');
```

5c)

```
select f.titel from Film f, Veröffentlichung v where v.Film_ID = f.ID and v.DVD_ID is not NULL;
```

5d)

```
select distinct k.Name, f.Regisseur_ID from Veröffentlichung v, Film f, Kino k
where v.Film_ID = f.ID and v.Kino_ID=k.ID and Regisseur_ID = 23;
```