

Datenbanksysteme: Übung 4

Tutor: Miao Wang

Lisa Dohrmann (4130066), Adrian Neumann (4140810), Naja v. Schmude (4127652)

19. Mai 2008

Aufgabe 1

- „Ich hätte gerne Name und Vorname aller Personen, die vor dem 12.3.85 geboren sind.“
- „Ich hätte gerne Name und Vorname aller Spielfiguren, die älter als 120 sind und die in der Welt DSA spielen.“
- „Ich hätte gerne den Vornamen aller Spielfiguren, die in einer Welt spielen, in der es das Abenteuer „Ein Gott greift ein“ gibt sowie das Genre dieser Welt.“

Aufgabe 2

•

$$\Pi_{Title} \underbrace{Movie \bowtie Tape}_{\text{alle}} \setminus \underbrace{\left(\underbrace{Movie \bowtie Tape}_{\text{alle}} \setminus \underbrace{\left(\underbrace{Movie \bowtie Tape}_{\text{alle, die ausgeliehen}} \right)}_{\text{alle, von denen Tape übrig}} \right)}_{\text{alle, die ausgeliehen}} \right)$$

•

$$\Pi_{Title}(Movie \bowtie (Tape \setminus (Tape \overset{T.ID=TAPE.ID}{\bowtie} \sigma_{from_date \geq today-6months} Rental)))$$

•

$$\Pi_{name,city} \left(Customer \setminus \left(\Pi_{name,first_name,zip,city,email} \left(Costumer \overset{MEM_NO = CUSTOMER}{\bowtie} Phones \right) \right) \right)$$

•

$$\Pi_{Title}(Movie \setminus (Movie \bowtie (Tape \bowtie (\sigma_{Extrach= Tape} Format))))$$

•

$$\Pi_{name}(\sigma_{city=Berlin}(Customer))$$

Aufgabe 3

- a)
- ok
 - Nicht einfügbar, da der Sportkurs „Springen für Doofe“ noch nicht als Sportkurs eingetragen ist, und er somit nicht per Fremdschlüssel referenziert werden kann.
 - ok
 - Nicht einfügbar, da der Nachname mit 11 Zeichen um eins zu lang ist.

- Nicht einfügbar, da `Dauer_in_min` vom Typ `number(3)` ist, hier aber ein String angegeben wird.
 - Nicht einfügbar, da sonst die `Sportler_ID` 23 doppelt vergeben wäre und damit keine eindeutige Identifizierung es Sportlers mehr möglich ist. Die Schlüsseleigenschaft von `Sportler_ID` wäre dadurch nicht mehr gegeben.
 - ok
 - ok
 - Nicht einfügbar, da hier die Bedingung verletzt wird, dass blau keine Haarfarbe sein darf.
- b) Es lässt sich auf eine N:M-Beziehung schließen, da verschiedene Sportler den selben Sportkurs belegen können und damit den selben Fremdschlüssel haben dürfen. Jeder Sportler darf auch mehrere Sportkurse belegen, da er theoretisch mit verschiedenen Sportler_Ide auftreten kann. Ob das in der Praxis Sinn macht, wenn der selbe Sportler für verschieden Sportkurse verschiedene IDs erhält, ist eine andere Frage.
- c) Gefordert ist nun eine 1:1 Beziehung oder noch etwas genauer eine (0,1):(1,1) Beziehung zwischen Sportkurs und Sportler. Dafür verwenden wir die Schlüsselwörter `UNIQUE` und `NOT NULL`. Jeder Sportkurs erhält als Fremdschlüssel die ID seines einzigen Teilnehmers. Der Fremdschlüssel darf nicht `NULL` sein, sodass dadurch genau ein Teilnehmer pro Sportkurs garantiert ist. Außerdem legen wir fest das der Fremdschlüssel `Teilnehmer` für die ganze Tabelle `Sportkurs` einzigartig ist, was garantiert, dass sich ein Sportler für höchstens einen Kurs eintragen kann.

Außerdem legen wir für jeden Sportler noch eine `CHECK` Klausel fest, die nur IDs aus zwischen 1 und 100 zulässt.

```
CREATE TABLE Sportkurs(
    Titel varchar2(20),
    Trainer varchar2(10),
    Dauer_in_min number(3),
    Teilnehmer number(3) NOT NULL,
    CONSTRAINT Titel_pk PRIMARY KEY (Titel),
    CONSTRAINT Dauer_in_min_ck CHECK (Dauer_in_min <= 90)
    CONSTRAINT Teilnehmer_unique UNIQUE (Teilnehmer)
    CONSTRAINT Teilnehmer_fk FOREIGN KEY (Teilnehmer) REFERENCES Sportler(Sportler_ID);
);

CREATE TABLE Sportler(
    Vorname varchar2(10),
    Nachname varchar2(10),
    Geburtsdatum date,
    Haarfarbe varchar2(10),
    Sportler_ID number(3),
    Sportkurs varchar2(20),
    CONSTRAINT Sportler_ID_pk PRIMARY KEY (Sportler_ID),
    CONSTRAINT Haarfarbe_ck CHECK (Haarfarbe != 'blau'),
    CONSTRAINT Sportkurs_fk FOREIGN KEY (Sportkurs) REFERENCES Sportkurs(Titel)
    CONSTRAINT Sportler_ID_ck CHECK (Sportler_ID <= 100 AND Sportler_ID >= 1)
);
```