

2. Anfragen

Zu einer Website, auf der Besucher im Browser Online-Spiele spielen können, liegt das folgende relationale Schema einer Datenbank vor:

Team : {[TNr, Name, Teamfarbe]}

Spieler : {[SNr, Name, Icon, TNr, EMail]}

Minispiel : {[MNr, Name, Kategorie, Schwierigkeit]}

Wettkampf : {[WNr, Sieger, Geschlagener, MNr, Dauer]}

Auf der Website treten jeweils zwei Spieler gegeneinander in Minispielen an. In diesen ist es das Ziel, den gegnerischen Spieler in möglichst kurzer Zeit zu besiegen. Minispiele gibt es dabei in verschiedenen Schwierigkeitsstufen („leicht“, „mittel“, „schwer“, „sehr schwer“) und verschiedenen Kategorien („Denkspiel“, „Geschicklichkeitsspiel“, usw.). Die Attribute *Sieger* und *Geschlagener* sind jeweils Fremdschlüsselattribute, die auf das Attribut *SNr* der Relation *Spieler* verweisen. Beachten Sie, dass das Dauer-Attribut der Wettkampf-Relation die Dauer eines Wettkampfes in der Einheit Sekunden speichert.

- (a) Formulieren Sie geeignete Anfragen in relationaler Algebra für die folgenden Teilaufgaben:

- (i) Geben Sie die *Namen* der *Minispiele* zurück, die zur *Kategorie* „Denkspiele“ zählen.

$$\pi_{\text{Name}}(\sigma_{\text{Kategorie}='Denkspiele'}(\text{Minispiele}))$$

- (ii) Geben Sie die *Namen* und *E-Mail-Adressen* aller Spieler zurück, die in einem Minispiel des Typs „Geschicklichkeitsspiel“ gewonnen haben.

$$\pi_{\text{Spieler.Name}, \text{Spieler.Email}}(\sigma_{\text{Kategorie}='Geschicklichkeitsspiel'}(\text{Minispiele}) \bowtie_{\text{Minispiel.MNr}=\text{Wettkampf.MNr}} \text{Wettkampf} \bowtie_{\text{Wettkampf.Sieger}=\text{Spieler.SNr}} \text{Spieler})$$

- (b) Formulieren Sie geeignete SQL-Anfragen für die folgenden Teilaufgaben. Beachten Sie dabei, dass Ihre Ergebnisrelationen keine Duplikate enthalten sollen.

- (i) Geben Sie jede Spielekategorie aus, für die ein Minispiel der Schwierigkeitsstufe sehr schwer vorhanden ist.

```
1 SELECT DISTINCT Kategorie
2 FROM Minispiel
3 WHERE Schwierigkeit = 'sehr schwer';
```

- (ii) Geben Sie die Wettkämpfe aus, deren Dauer unter der durchschnittlichen Dauer der Wettkämpfe liegt.

```

1  SELECT WNr
2  FROM Wettkampf
3  WHERE Dauer < (
4      SELECT AVG(Dauer) FROM Wettkampf
5  );

```

- (iii) Geben Sie für jeden *Spieler* seine *SNr*, seinen *Namen*, die Anzahl seiner Siege, die durchschnittliche Dauer seiner siegreichen Wettkämpfe und die Anzahl der Teams, aus denen er bereits mindestens einen Spieler besiegt hat, zurück.

```

1  SELECT
2      SNr,
3      Name,
4      (
5          SELECT COUNT(*)
6          FROM Wettkampf
7          WHERE Wettkampf.Sieger = SNr
8      ) AS Anzahl_Siege,
9      (
10         SELECT AVG(Dauer)
11         FROM Wettkampf
12         WHERE Wettkampf.Sieger = SNr
13     ),
14     (
15         SELECT COUNT(DISTINCT Team.TNr)
16         FROM Team, Spieler, Wettkampf
17         WHERE
18             Team.TNr = Spieler.TNr AND
19             Wettkampf.Geschlagener = Spieler.SNr AND
20             Wettkampf.Sieger = SNr
21     )
22 FROM Spieler;

```

```

1  SELECT
2      s.SNr,
3      s.Name,
4      COUNT(*) AS AnzahlSiege,
5      AVG(w.Dauer) AS DurchschnittlicheWettkampfdauer,
6      COUNT(DISTINCT g.TNr) AS TeamsBesiegt
7  FROM Spieler s, Wettkampf w, Spieler g
8  WHERE
9      s.SNr = w.Sieger AND
10     g.SNr = w.Geschlagener
11 GROUP BY s.SNr, s.Name;

```

- (c) Verwenden Sie den relationalen Tupelkalkül, um die folgenden Anfragen zu formulieren:
- (i) Finden Sie die Namen der Spieler des Teams Dream Team.
 - (ii) Geben Sie die Namen der Minispiele zurück, bei denen bereits Spieler gegeneinander angetreten sind, deren Teams dieselbe Teamfarbe haben.

Hinweis: Die Anfragen im relationalen Tupelkalkül dürfen auch nicht sicher sein.