Assignment 1

Business Intelligence 2024W

Bosse Behrens, st.ID: 12347333 e12347333@student.tuwien.ac.at

1 Exercise 1

Creating an appropriate set of relations for the given ER diagram. Example notation:

• **RelationName**{[PrimaryKey, Attribute1, Attribute 2, ..., ForeignKey → Relation-Referenced]}

The relations for the given ER diagram:

- $driver{[driverID, name, address, hasPartner <math>\longrightarrow driver]}$
- logisticsCompany{[companyName, address, phoneNumber]}
- **vehicle**{[licensePlateNumber, ownedBy \longrightarrow logisticsCompany]}
- transport{ $[departure, driverID \longrightarrow driver, CarriedOutBy \longrightarrow vehicle, arrival]}$
- $\mathbf{worksFor}\{[driverID \longrightarrow driver, companyName \longrightarrow logisticsCompany, salary]\}$

2 Exercise 2

We have given following three functional dependencies:

- 1. $A \longrightarrow B$
- 2. $B \longrightarrow AC$
- 3. $C \longrightarrow D$

Using Armstrongs Axioms and its derived rules for union and decomposition for functional dependencies we obtain:

- ullet B \longrightarrow AC \Longrightarrow B \longrightarrow A and B \longrightarrow C
- ullet B \longrightarrow C and C \longrightarrow D \Longrightarrow B \longrightarrow D
- ullet A \longrightarrow B , B \longrightarrow C and B \longrightarrow C \Longrightarrow A \longrightarrow C and A \longrightarrow D

Therefore both A and B are candidate and superkeys. Now we want to check which normal forms are fulfilled.

• 2NF is fulfilled if 1NF is fullied (already assumed) and all non-prime attributes are fully functional dependent of each candidate key. The non-prime attributes are the one snot part of a candidate key: $R \setminus A$, B = C, D. Since we already established that C and D are fully functional dependent on both candidate keys A and B, 2NF is fulfilled.

- 3NF is fulfilled if for every functional dependency $\alpha \longrightarrow \beta$ one of the following three holds true:
 - 1. $\beta \in \alpha$, e.g. trivial
 - 2. α is a superkey of R
 - 3. β is a prime attribute

If we look at the functional dependency $C \longrightarrow D$, we notice that $D \notin C$, C is not a superkey and D is not prime. Therefore 3NF is not fulfilled.

- BCNF is fulfilled if for every functional dependency $\alpha \longrightarrow \beta$ one of the following two holds true:
 - 1. $\beta \in \alpha$, e.g. trivial
 - 2. α is a superkey of R

If we look at the functional dependency $C \longrightarrow D$, we notice that $D \notin C$ and C is not a superkey. Therefore BCNF is not fulfilled.

2.1 Exercise 3

We formulate the appropriate SQL statements to create the 6 specified tables from the relational schema.

• student

```
CREATE TABLE student (
sid INT PRIMARY KEY,
firstname VARCHAR(30),
lastname VARCHAR(30),
semester INT,
birthdate DATE
);
```

• tutor

```
CREATE TABLE tutor (
tid INT PRIMARY KEY,
firstname VARCHAR(30),
lastname VARCHAR(30),
issenior BOOLEAN
);
```

• studygroup

```
CREATE TABLE studygroup (
          gid INT PRIMARY KEY,
          tid INT,
          weekday VARCHAR(30),
          room VARCHAR(30),
          starttime DATE,
          FOREIGN KEY (tid) REFERENCES tutor(tid)
      );

    exercisesheet

      CREATE TABLE exercisesheet (
          eid INT PRIMARY KEY,
          maxpoints INT
      );

    handsin

      CREATE TABLE handsin (
          sid INT,
          eid INT,
          achievedpoints INT,
          PRIMARY KEY (sid, eid),
          FOREIGN KEY (sid) REFERENCES student(sid),
          FOREIGN KEY (eid) REFERENCES exercisesheet(eid)
      );
• member
      CREATE TABLE member (
          sid INT,
          gid INT,
          PRIMARY KEY (sid, gid),
          FOREIGN KEY (sid) REFERENCES student(sid),
          FOREIGN KEY (gid) REFERENCES studygroup(gid)
      );
```

2.2 Exercise 4

We will now formulate the 4 given statements into SQL.

1. Find the IDs of all students who have achieved at least 10 points for exercise sheet 1.

```
SELECT sid FROM handsin
WHERE eid = 1 AND achievedpoints >= 10;
```

2. Determine the ID and the last name of all students whose study group meets on Mondays.

```
SELECT s.sid, s.lastname FROM student s

JOIN member m ON s.sid = m.sid

JOIN studygroup g ON m.gid = g.gid

WHERE g.weekday = 'Monday';
```

3. For each exercise sheet, determine its ID and the on average achieved number of points.

```
SELECT eid, AVG(achievedpoints) FROM handsin GROUP BY eid;
```

4. Determine the IDs of all students who achieved between 1 and 5 points (1 \leq achieved \leq 5) for at least 3 exercise sheets.

```
SELECT sid FROM handsin
WHERE achievedpoints >= 1 AND achievedpoints <= 5
GROUP BY sid
HAVING COUNT(eid) >= 3;
```

I hereby confirm that I did the exercise	es myself without support by any type of AI, LLM or similar tools.
	Page 5