

Individuell uppgift 3, H1030



Yasir Riyadh Jabbar (TIDAA KTH)

Cleaning Table

cleaningObject	date	cleanerId	cleanerName	cleaningTime	foremanId	foremanName
SO001	19-10-23	S001	Kalle Karlsson	5,5	F01	Arne Boss
SO001	19-10-23	S002	Anna Svensson	5,5	F01	Arne Boss
SO002	19-10-23	S001	Kalle Karlsson	2,7	F01	Arne Boss
SO002	19-10-24	S001	Kalle Karlsson	7,7	F01	Arne Boss
SO003	19-10-24	S003	Per Persson	8,5	F02	Anna Bossig

Universal Relation:

Cleaning (cleaningObject, date, cleanerId, cleanerName, cleaningTime, foremanId, foremanName)

Task-1: List all complete functional dependencies. Also state which of these are partial and transitive respectively.

partial (non-fully) dependency: cleanerId \rightarrow {cleanerName, foremanId, foremanName}

transitive dependency: foremanId \rightarrow foremanName

fully dependency: {cleaningObject, cleanerId} \rightarrow {date, cleaningTime}

The primary key is the composed {cleaningObject, cleanerId}

Task-2: State which normal form the relationship is in? Justify the answer

For composite key {cleaningObject, cleanerId} each value in the table is single-valued, so the relation is in 1NF.

Task-3: Give three examples of problems that can arise, so-called anomalies, one on insert, one on update and one on delete of tuples (rows).

1. Insertion anomaly:

If we want to insert new cleaningObject row with values for cleaningObject, date, and cleaningTime. We cannot record this info, even though we have values for these attributes. Because the key is {cleaningObject, cleanerId}, we are not allowed to insert row with null value for cleanerId.

2. Update anomaly:

It is data inconsistency that results from partial update. If foreman of Kalle Karlsson is incorrect and needs to update, then it must be updated at least 3 times (in 3 rows) or there will be inconsistent data in the table.

3. Deletion anomaly:

If cleaner S001 dropped out of cleaningObject SO001, we would delete only row with this object info. It is desirable to keep the object info, but we cannot do so without a corresponding cleanerId.

Task-4: Normalize the relation until it is in 3NF. Take one step at a time, normalize relation first to 2NF and then to 3NF. Also explain how you carry out the normalization

1) Normalization to 2NF

{cleaningObject, cleanerId} → {date, cleanerName, cleaningTime, foremanId, foremanName}

Since it is in 1NF and primary key is composed key then taking the non-fully dependencies in the relation according to projection steps:

1. Identifying each of non-fully dependencies attributes,
2. Removing attributes that depend on determinants that found in step1,
3. Placing determinants in separate tables with their dependent attributes,
4. Old table will contain only the remainder attributes.

cleanerId → {cleanerName, foremanId, foremanName}

New relation: **Cleaner** (cleanerId, cleanerName, foremanId, foremanName)

New relation: **Cleaning2** (cleaningObject, cleanerId, date, cleaningTime)

2) Normalization to 3NF

Since **foremanId → foremanName** is transitive dependency (foremanId is not super key, and foremanName is not part of candidate key). So, Cleaner relation needs to rewrite to:

New relation: **Cleaner** (cleanerId, cleanerName, foremanId)

New relation: **Foreman** (foremanId, foremanName)

New relation: **NewCleaning** (cleaningObject, cleanerId, date, cleaningTime)

Cleaning2 Table

<u>cleaningObject</u>	<u>cleanerId</u>	date	cleaningTime
SO001	S001	19-10-23	5,5
SO001	S002	19-10-23	5,5
SO002	S001	19-10-23	2,7
SO002	S001	19-10-24	7,7
SO003	S003	19-10-24	8,5

Cleaner Table

<u>cleanerId</u>	cleanerName	foremanId
S001	Kalle Karlsson	F01
S002	Anna Svensson	F01
S003	Per Persson	F02

Foreman Table

<u>foremanId</u>	foremanName
F01	Arne Boss
F02	Anna Bossig