Lab 5

1. **Will the conversion to BCNF be dependency preserving in any case? Proof**

**your statement and give a reasoning for choosing BCNF design.**

It is not always possible to achieve both BCNF and

dependency preservation

Consider a schema:

*dept\_advisor(s\_ID, i\_ID, department\_name*)

With function dependencies:

*i\_ID* → *dept\_name*

*s\_ID, dept\_name* → *i\_ID*

*dept\_advisor* is not in BCNF

• *i\_ID* is not a superkey.

Any decomposition of *dept\_advisor* will not include all the

attributes in

*s\_ID, dept\_name → i\_ID*

Thus, the composition is NOT be dependency preserving

Task 2

Given table in 1NF, convert to 3NF if PK is UnitID:

Изображение выглядит как стол

Автоматически созданное описание

|  |  |
| --- | --- |
| Tutor\_ID | TutEmail |
| Tut1 | Tut1@fhbb.ch |
| Tut3 | Tut3@fhbb.ch |
| Tut5 | Tut5@fhbb.ch |

|  |  |
| --- | --- |
| Topic | Book |
| GMT | Deumlich |
| Gin | Zehnder |
| PhF | Dümmlers |
| AVQ | Swiss Topo |

|  |  |  |
| --- | --- | --- |
| UnitID | StudentID | Grade |
| U1 | St1 | 4.7 |
| U2 | St1 | 5.1 |
| U1 | St4 | 4.3 |
| U5 | St2 | 4.9 |
| U4 | St2 | 5.0 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| UnitID | Date | Tutor\_ID | Topic | Room |
| U1 | 23.02.03 | Tut1 | GMT | 629 |
| U2 | 18.11.02 | Tut3 | Gin | 631 |
| U5 | 05.05.03 | Tut3 | PhF | 632 |
| U4 | 04.07.03 | Tut5 | AVQ | 621 |

Task 3. Given table in 1NF, convert to 2NF if PK is {ProjectName, ProjectManager}, use decomposition:

Изображение выглядит как стол

Автоматически созданное описание

|  |  |
| --- | --- |
| ProjectManager | Position |
| Manager1 | CTO |
| Manager2 | CTO2 |

|  |  |  |
| --- | --- | --- |
| ProjectName | Budget | TeamSize |
| Project1 | 1 kk $ | 15 |
| Project2 | 1.5 kk $ | 12 |

|  |  |
| --- | --- |
| ProjectName | ProjectManager |
| Project1 | Manager1 |
| Project2 | Manager2 |

**Task 4. Given table, convert to 3NF if PK is Group, use decomposition:**

Изображение выглядит как стол

Автоматически созданное описание

|  |  |
| --- | --- |
| Group | Speciality |
| g1 | s1 |
| g2 | s2 |

|  |  |
| --- | --- |
| Speciality | Faculty |
| s1 | f1 |
| s2 | f2 |

Изображение выглядит как стол

Автоматически созданное описание

ProjectID -> {Curator,TeamSize}

TeamSize->ProjectGroupsNumber

Department->{Curator,TeamSize}

|  |  |  |
| --- | --- | --- |
| ProjectID | Curator | TeamSize |
| p1 | e1 | 100 |
| p2 | e2 | 120 |

|  |  |
| --- | --- |
| TeamSize | ProjectGroupsNumber |
| 100 | 5 |
| 120 | 6 |

|  |  |
| --- | --- |
| ProjectID | Department |
| p1 | d1 |
| p2 | d2 |

**Task 6. List the three design goals for relational databases, and explain why each is**

**desirable. Give an example of both desirable and undesirable types of**

**decompositions.**

Goal for a relational database design is:

• BCNF.

Boyce–Codd normal form (or BCNF or 3.5NF) is a normal form used in database normalization. It that every non-trivial determinant is a candidate key. It is even stricter that all the functional dependencies must have the candidate keys on the left hand side. Though 3NF insist on removing non-key dependencies, it accepts a transitive FD where a key attribute determined by non-key attribute. BCNF eliminates this FD too to become a stricter normal form.

• Lossless join.

Decomposition is lossless if it is feasible to reconstruct relation R from decomposed tables using Joins. This is the preferred choice. The information will not lose from the relation when decomposed. The join would result in the same original relation.

• Dependency preservation.

The dependency preservation decomposition is another property of decomposed relational database schema D in which each functional dependency X -> Y specified in F either appeared directly in one of the relation schemas Ri in the decomposed D or could be inferred from the dependencies that appear in some Ri.

Desirable type: Lossless Decomposition

Undesirable type: Lossy Decomposition