**Task 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 be a dependency preserving when you converse relation to BSNF design, but still BSNF design is convenient and correct for many cases. For example table of Student(id, name, tot_credit, dept_name, building, budget). We can decompose it to Student(id, name, tot_credit, dept_name) and Department(dept_name, building, budget).

I think it is very convenient to use BCNF, because if some attribute depends on another attribute, and it is not a PK, then it is very inconvenient to store data in this table.

**Task 2. Given table in 1NF, convert to 3NF if PK is UnitID:**

PK(UnitId, StudentID)-> TutorID, Date, Room, Grade

TutorID->TutEmail

UnitID->Book,Topic

| UnitID | StudentID | Date | Tutor ID | Room | Grade |
|---|---|---|---|---|---|
| U1 | St1 | 23.02.03 | Tut1 | 629 | 4.7 |
| U2 | St1 | 18.11.02 | Tut3 | 631 | 5.1 |
| U1 | St4 | 23.02.03 | Tut1 | 629 | 4.3 |
| U5 | St2 | 05.05.03 | Tut3 | 632 | 4.9 |
| U4 | St2 | 04.07.03 | Tut5 | 621 | 5.0 |

| Tutor ID | TutEmail |
|---|---|
| Tut1 | tut1@fhbb.ch |
| Tut3 | tut3@fhbb.ch |
| Tut1 | tut1@fhbb.ch |
| Tut3 | tut3@fhbb.ch |
| Tut5 | tut5@fhbb.ch |

| UnitID | Topic | Book |
|---|---|---|
| U1 | GMT | Deumlich |
| U2 | GIn | Zehnder |
| U1 | GMT | Deumlich |
| U5 | PhF | Dümmlers |
| U4 | AVQ | SwissTopo |

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

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

| ProjectName | ProjectManager | Position |
|---|---|---|
| Project1 | Manager1 | CTO |
| Project2 | Manager2 | CTO2 |

**Task 4. Given table, convert to 3NF if PK is Group, use decomposition: (Faculties have a number of specialities, each speciality consists of a set of particular groups**)

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

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

**Task 5. Given table, convert to BCNF if PK is {ProjectID, Department}, use decomposition: (Curator depends on projectID and related departments, teamSize directly relates to project and related departments, ProjectGroupsNumber depends on TeamSize)**

| ProjectID | Department | Curator | TeamSize |
|---|---|---|---|
| p1 | d1 | e1 | 100 |
| p2 | d2 | e2 | 120 |

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

**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.**

**Desirable-**

1)BCNF: name and city depend on StudentID and StudentID is PK

| StudentID | Name | City |
|---|---|---|
| 1 | Zhandos | Almaty |

2)Lossless join(decomposition), 3)Dependency preservation: there is if we join we don't loss a data and we have dependency preservation(studentID->name,city and city->country)

| StudentID | Name | City |
|---|---|---|
| 1 | Zhandos | Almaty |
| 2 | Almas | Almaty |

| City | Country |
|---|---|
| Almaty | Kazakhstan |

**Undesirable-**

1)There is address depends on city. But city is not a primary key

| StudentID | Name | Address | City |
|---|---|---|---|
| 1 | Zhandos | Tole bi | Almaty |

2)Lossy join(decomposition): there we will loss data if want to join

| StudentID | Name |
|---|---|
| 1 | Zhandos |
| 2 | Zhandos |

| Name | City | Country |
|---|---|---|
| Zhandos | Almaty | Kazakhstan |
| Zhandos | Aktobe | Kazakhstan |

3)there is name and city depend on studentID, so no dependency between name and city. When we decompose table(StudentID,name,city) we don't preserve the dependency. So after join there will be only (studentID->name) dependency

| StudentID | Name |
|---|---|
| 1 | Zhandos |

| Name | City |
|---|---|
| Zhandos | Almaty |