Laboratory work 5 DB

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 achieve both BCNF and dependency preservation

Proof:
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

Reason for choosing BCNF:

- BCNF is an extension of 3NF and it is has more strict rules than 3NF. Also, it is comparatively more stronger than 3NF.
- In BCNF the functional dependencies are already in 1NF, 2NF and 3NF.
- The redundancy is comparatively low in BCNF.
- There are no non-trivial functional dependencies and therefore the relation is in BCNF

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

StudentID	Date
St1	23.02.03
St2	18.11.02
St4	23.02.03
St2	05.05.03
St2	04.07.03

Topic	Room	Book	TutorID
GMT	629	Deumlich	Tut1
Gln	631	Zehnder	Tut3
GMT	629	Deumlich	Tut1
PhF	632	Dummlers	Tut3
AVQ	621	SwissTopo	Tut5

TutorID	TutEmail
Tut1	tut1@fhbb.ch
Tut3	tut3@fhbb.ch
Tut1	tut1@fhbb.ch
Tut3	tut3@fhbb.ch
Tut5	tut5@fhbb.ch

UnitID(PK)	StudentID	Topic	Grade
U1	St1	GMT	4.7
U2	St2	Gln	5.1
U1	St4	GMT	4.3
U5	St2	PhF	4.9
U4	St2	AVQ	5.0

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

ProjectManager	Position
Manager1	CTO1
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:

Faculties have a number of specialities, each speciality consists of a set of particular groups.

Speciality	Faculty
s1	f1
s2	f2

Group	Speciality
g1	s1
g2	s2

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	Curator
p1	e1
p2	e2

Department	TeamSize
d1	100
d2	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 a example of both desirable and undesirable types of decompositions.

- Minimization of repetition of information (maintain an accurate database)
- Dependency preserving decomposition (check correctness of updates quickly)
- Lossless join decomposition (se the smallest amount of space possible)

1) Consider the following relation R(A , B , C)-

A	B	C
1	2	1
2	5	3
3	3	3

Consider this relation is decomposed into two sub relations R1(A , B) and R2(B , C)

A	B
1	2
2	5
3	3

B	C
2	1
5	3
3	3

Now, let us check whether this decomposition is lossless or not. For lossless decomposition, we must have- R1 ⋈ R2 = R Now, if we perform the natural join (⋈) of the sub relations R1 and R2 , we get relation is same as the original relation R. Thus, we conclude that the above decomposition is lossless join decomposition.

2) Consider the above relation R(A , B , C)- Consider this relation is decomposed into two sub relations as R1(A , C) and R2(B , C)-

A	C
1	1
2	3
3	3

B	C
2	1
5	3
3	3

Now, let us check whether this decomposition is lossy or not. For lossy decomposition, we must haveR1 ⋈ R2 ⊃ R Now, if we perform the natural join (⋈) of the sub relations R1 and R2 we get

A	B	C
1	2	1
2	5	3
2	3	3
3	5	3
3	3	3

This relation is not same as the original relation R and contains some extraneous tuples. Clearly, R1 ⋈ R2 ⊃ R. Thus, we conclude that the above decomposition is lossy join decomposition.