

Task 1.

Answer: No

For example:

We have a schema: student(id, dept_name, a_id)

a_id -> dept_name

id, dept_name -> a_id

student is not BCNF (a_id is not superkey).

Any decomposition of student won't include all attributes in (id, dept_name -> a_id)

If (s_id, a_id) and (a_id, dept-name) then it is not be dependency preserving.

Task 2.

UnitID	Date	TutorID	Topic	Room
U1	Tut1	GMT	629	23.02.03
U2	Tut3	Gln	631	18.11.02
U5	Tut3	PhF	632	05.05.03
U4	Tut5	AVQ	621	04.07.03

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

Topic	Book
GMT	Deumlich
Gln	Zehnder
PhF	Dümmlers
AVQ	SwissTopo

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

Task 3.

ProjectName	ProjectManager	Budget	TeamSize
Project1	Manager1	1 kk \$	15
Project2	Manager2	1.5 kk \$	12

ProjectManager	Position
Manager1	CTO
Manager2	CTO2

Task 4.

Speciality	Faculty
S1	F1
S2	F2

Group	Speciality
G1	S1
G2	S2

Task 5.

ProjectID	Department
P1	D1
P2	D2

TeamSize	ProjectGroupsNumber
100	6
120	5

ProjectID	Curator	TeamSize
P1	E1	100
P2	E2	120

Task 6.

1. Lossless-join decomposition

We can maintain exact relationships in database.

2. Dependency preserving decomposition

This allows you to check the validity of the update without having to compute the join of relations in the decomposition.

3. Minimization of repetition information

The minimum possible space is used to store information.