Task 1

We can prove it by giving a counter example: Consider the following schema;
a b c and c->b
Clearly the above schema is in 3NF, because ab->c is a superkey dependency and, from c->b we can see that b-c=b, which is a subset of the primary key .
But, the above schema is not in BCNF because c->b is neither super-key nor trivial dependency.
So we decompose above schema , keeping it lossless.
Only possible lossless decomposition is: ac and cb. (because, their intersection c is primary key for the 2nd table).
But clearly the dependency ab->c is lost.

TASK 2

UnitID	Topic	Room	Book	Date
U1	GMT	629	Deumlich	23.02.03
U2	Gin	631	Zehnder	18.11.02
U1	GMT	629	Deumlich	23.02.03
U5	Phf	632	Dummlers	05.05.03
U4	AVQ	621	SwissTopo	04.07.03

StudentID	UnitID	Grade	TutorID
St1	U1	4.7	Tut1
St1	U2	5.1	Tut3
St4	U1	4.3	Tut1
St2	U5	4.9	Tut2
St2	U4	5.0	Tut5

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

TASK 3

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

Speciality	Faculty
s1	f1
s2	f2

Group	Speciality
g1	s1
g2	s2

TASK 5

ProjectID	Departament	Curator	TeamID
p1	d1	e1	t1
p2	d2	e2	t2

TeamID	TeamSize	ProjectGroupsNumber
t1	100	5
t2	120	6

Task 6

1)Minimization of information repetition.
When we perform update in such relation, it will be checked easily whether it suits to our relation.
2)Dependency preserving decomposition.
By this way we can maintain an accurate relations in our database.
3)Lossless join decomposition
The smallest possible amount of space is used for storing the information.

Types of decomposition:
1)A Lossy Decomposition
In this type of decomposition we may lose some information from initial table.
2)Lossless Decomposition
In this type of decomposition there is no loss of information when we replace relation R by two relations R1 and R2.