Изображение выглядит как стол

Автоматически созданное описание

MY SOLUTION:

|  |  |  |  |
| --- | --- | --- | --- |
| StudentID | Date | UnitID | Grade |
| St1 | 23.02.03 | U1 | 4.7 |
| St1 | 18.11.02 | U2 | 5.1 |
| St4 | 23.02.03 | U1 | 4.3 |
| St2 | 05.05.03 | U5 | 4.9 |
| St2 | 04.07.03 | U4 | 5.0 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| unitID | Topic | Book | TutorID | Room |
| U1 | GMT | Deumlich | Tut1 | 629 |
| U2 | Gln | Zehnder | Tut3 | 631 |
| U1 | GMT | Deumlich | Tut1 | 629 |
| U5 | PhF | Dummlers | Tut3 | 632 |
| U4 | AVQ | SwiisTopo | Tut5 | 621 |

|  |  |
| --- | --- |
| tutorID | tutEmail |
| Tut1 | Tut1@fhbb.ch |
| Tut3 | Tut3@fhbb.ch |
| Tut1 | Tut1@fhbb.ch |
| Tut3 | Tut3@fhbb.ch |
| Tut5 | Tut5@fhbb.ch |
| \ |  |

Изображение выглядит как стол

Автоматически созданное описание

|  |  |
| --- | --- |
| projectName | budget |
| Project1 | 1kk $ |
| Project2 | 1.5 kk $ |

|  |  |
| --- | --- |
| projectManager | teamSize |
| Manager1 | 15 |
| Manager2 | 12 |

Изображение выглядит как стол

Автоматически созданное описание

|  |  |
| --- | --- |
| Group | Speciality |
| G1 | S1 |
| G2 | S2 |

Изображение выглядит как стол

Автоматически созданное описание

|  |  |
| --- | --- |
| Curator | Department |
| E1 | D1 |
| E2 | D2 |

|  |  |  |  |
| --- | --- | --- | --- |
| projectID | projectGroupNumber | teamSize | Curator |
| P1 | 5 | 100 | E1 |
| P2 | 6 | 120 | E2 |



We only need to give a counter example: Consider the following schema;

a b c and c->b

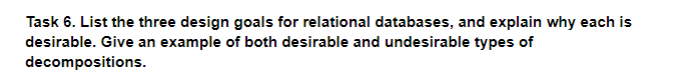
Clearly the above schema is in 3NF, because ab->c is a superkey dependency and, from c->b

we can see that b-c=b, which is a subset of the primary key (such dependency is also allowed in 3NF). 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.



Lossless Join Decomposition -

* Union of Attributes of R1 and R2 must be equal to attribute of R. Each attribute of R must be either in R1 or R2
* Intersection of Attributes of R1 and R2 must not be NULL.
* Common attribute must be a key for at least one relation (R1 or R2).

Dependency Preserving Decomposition –

* For dependency preserving decomposition, A->B can be ensured in R1(AB) and C->D can be ensured in R2(CD). Hence it is dependency preserving decomposition.
* If a relation R is decomposed into relation R1 and R2, then the dependencies of R either must be a part of R1 or R2 or must be derivable from the combination of functional dependencies of R1 and R2.

Repetition of information –

* condition in database, where the values of one attribute are determined by the values of another attribute in the same relation, and both values are repeated throughout the relation, and both values are repeated throughout the relation.