1. RR 1 and 2 are said to be equivalent if and only if they define the same set of databases. That is, they are equivalent if all the databases that satisfy RR1 also satisfy RR2, and all that satisfy RR2 also satisfy RR1. For tables that are fully normalized (that is, tables that are in 4NF), are they equivalent? Prove your answer.

Ans: RR1 states that one table should describe only one entity type. RR2 states that each fact should be represented only once in database. i.e the subject and value of that fact should be different.

Lets consider the following database:

Employee

|  |  |  |
| --- | --- | --- |
| E ID | E Name | D ID |
| E001 | Alex | D002 |
| E002 | Amy | D003 |
| E003 | Ajay | D001 |

Department

|  |  |  |
| --- | --- | --- |
| D ID | D Name | D Location |
| D001 | IT | Funger Hall |
| D002 | Marketing | Monroe Hall |
| D003 | Business | Corcoran Hall |

Manager

|  |  |  |
| --- | --- | --- |
| M ID | M Name | D ID |
| M101 | Jim Halpert | D002 |
| M102 | Dwight | D001 |
| M102 | Kevin | D003 |

The tables above are fully normalized. Hence, RR1 and RR2 are equivalent if and only if they define the same set of databases and the tables are fully normalized.

1. First, show a simple table of sample data-that does not meet the definitions for 1NF, 2NF, 3NF, 4NF and BCNF. That is, this database at once does not meet the specific requirements to be in that normal form. For example, for 2NF, it is not enough that the table is not in 1NF to not be in 2NF, it also must not meet the specific requirement for 2NF that differentiates it from 1 NF.

Ans: Lest build a table that is not in any normal form.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Emp ID** | **Emp Name** | **Dept Name** | **Skill** | **Dept Location** | **DID** |
| E001 | Jon Snow | Sales, Marketing | Communication, Marketing | Block 1, Block 2 | D101,D102 |
| E002 | Tyrion | Engineering, IT | CAD, Python | Block 3, Block 4 | D103, D104 |
| E003 | Cersei | IT, Marketing | Python, Marketing | Block 4, Block 2 | D104, D102 |
| E004 | Jaime | Finance, HR | Business, Management | Block 5, Block 6 | D105, D106 |
| E005 | Daenerys | IT | Python | Block 4 | D104 |
| E006 | Arya | DB | DBMS | Block 7 | D107 |

1NF states that all the attribute values must be atomic. But if we look at Dept Name, Skill, Dept Location and DID, they have multiple values separated by comma. Hence this table is not in any normal from.

1. Now change that database-keeping the content the same-so that it is in 1NF but not any of the other higher normal forms. Show FDDs. Explain why it is in 1NF and not in any of the other normal forms.

Ans: For the table to be in 1NF, all the attributes must be atomic. Hence lets modify the table here.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Emp ID** | **Emp Name** | **Dept Name** | **Skill** | **Dept Location** | **DID** |
| E001 | Jon Snow | Sales | Communication | Block 1 | D101 |
| E001 | Jon Snow | Marketing | Marketing | Block 2 | D102 |
| E002 | Tyrion | Engineering | CAD | Block 3 | D103 |
| E002 | Tyrion | IT | Python | Block 4 | D104 |
| E003 | Cersei | IT | python | Block 4 | D104 |
| E003 | Cersei | Marketing | Marketing | Block 2 | D107 |
| E004 | Jaime | Finance | Business | Block 5 | D105 |
| E004 | Jaime | HR | Management | Block 6 | D107 |
| E005 | Daenerys | IT | Python | Block 4 | D104 |
| E006 | Arya | DB | DBMS | Block 7 | D107 |

Now here all the values are atomic. Now the table is in first normal form. But we can notice that the Skill attribute depends only on the Department Name but not on the Emp ID. Hence this is not in 2nd normal form.

1. Now change that database-keeping its content the same-so that it is in 2NF but not any other higher normal forms. Show FDDs. Explain why it is in 2NF and not in any of the other normal forms.

Ans: For the table to be in 2nd NF, the table must be in 1st NF and all non-prime attributes must be fully functionally dependent on whole candidate key. Lets make the necessary changes.

|  |  |
| --- | --- |
| **Emp ID** | **Emp Name** |
| E001 | Jon Snow |
| E002 | Tyrion |
| E003 | Cersei |
| E004 | Jaime |
| E005 | Daenerys |
| E006 | Arya |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Emp ID** | **Dept Name** | **Skill** | **Dept Location** | **DID** |
| E001 | Sales | Communication | Block 1 | D101 |
| E001 | Marketing | Marketing | Block 2 | D102 |
| E002 | Engineering | CAD | Block 3 | D103 |
| E002 | IT | Python | Block 4 | D104 |
| E003 | IT | python | Block 4 | D104 |
| E003 | Marketing | Marketing | Block 2 | D107 |
| E004 | Finance | Business | Block 5 | D105 |
| E004 | HR | Management | Block 6 | D107 |
| E005 | IT | Python | Block 4 | D104 |
| E006 | DB | DBMS | Block 7 | D107 |

Now the Table is split into two parts. Each table has its own unique purpose and the Skill is dependent only on the Dname and DID, meeting the requirements of the second normal form. But there are transitive dependencies present, hence not in 3rd NF.

1. Now change that database-keeping its content the same-to convert it to 3NF but not any other higher normal form. Show FDDs. Explain why it is in 3NF and not in 4NF and BCNF.

Ans: For a table to be in 3rd NF, it should be in 2nd NF and no transitive dependencies should exist. Lets make the required changes.

|  |  |
| --- | --- |
| **Emp ID** | **Emp Name** |
| E001 | Jon Snow |
| E002 | Tyrion |
| E003 | Cersei |
| E004 | Jaime |
| E005 | Daenerys |
| E006 | Arya |

|  |  |  |
| --- | --- | --- |
| **D ID** | **Dept Name** | **Dept Location** |
| D101 | Sales | Block 1 |
| D102 | Marketing | Block 2 |
| D103 | Engineering | Block 3 |
| D104 | IT | Block 4 |
| D105 | Finance | Block 5 |
| D106 | HR | Block 6 |
| D107 | DB | Block 7 |

|  |  |  |
| --- | --- | --- |
| **Emp ID** | **Dept Name** | **Skill** |
| E001 | Sales | Communication |
| E001 | Marketing | Marketing |
| E002 | Engineering | CAD |
| E002 | IT | Python |
| E003 | IT | python |
| E003 | Marketing | Marketing |
| E004 | Finance | Business |
| E004 | HR | Management |
| E005 | IT | Python |
| E006 | DB | DBMS |

Now we have removed all the transitive functional dependencies from the table. Hence this is in 3rd NF.

The above table is also in BCNF, so for our requirement for question continuation, we will add an attribute called Skill description, now non-trivial functional dependencies where the determinant is not a superkey. Hence this is not in BCNF

|  |  |  |  |
| --- | --- | --- | --- |
| **Emp ID** | **Dept Name** | **Skill** | **Skill Description** |
| E001 | Sales | Communication | Excellent |
| E001 | Marketing | Marketing | Good |
| E002 | Engineering | CAD | Excellent |
| E002 | IT | Python | Good |
| E003 | IT | python | Good |
| E003 | Marketing | Marketing | Good |
| E004 | Finance | Business | Excellent |
| E004 | HR | Management | Excellent |
| E005 | IT | Python | Good |
| E006 | DB | DBMS | Excellent |

1. Now change that database-keeping its content the same-to convert it to BCNF. Show FDDs. Explain why is it in BCNF and not in 4NF.

Ans: For a table to be in BCNF, the table should be in 3rd NF and every determinant is a key. Lets make the required changes.

|  |  |
| --- | --- |
| **Emp ID** | **Emp Name** |
| E001 | Jon Snow |
| E002 | Tyrion |
| E003 | Cersei |
| E004 | Jaime |
| E005 | Daenerys |
| E006 | Arya |

|  |  |  |
| --- | --- | --- |
| **D ID** | **Dept Name** | **Dept Location** |
| D101 | Sales | Block 1 |
| D102 | Marketing | Block 2 |
| D103 | Engineering | Block 3 |
| D104 | IT | Block 4 |
| D105 | Finance | Block 5 |
| D106 | HR | Block 6 |
| D107 | DB | Block 7 |

|  |  |
| --- | --- |
| **Skill** | **Skill Description** |
| Communication | Excellent |
| Marketing | Good |
| CAD | Excellent |
| Python | Good |
| python | Good |
| Marketing | Good |
| Business | Excellent |
| Management | Excellent |
| Python | Good |
| DBMS | Excellent |

Here for the dependency X -> Y, X is the super key and Y is a candidate key. Hence this is in BCNF. For 4th NF, there should be no multi valued dependencies.

The above table has no multi valued table, hence we can alter to add one and solve that in next section.

|  |  |  |  |
| --- | --- | --- | --- |
| **Emp ID** | **DID** | **Skill** | **Training** |
| E001 | D101 | Communication | Public speking |
| E001 | D102 | Marketing | Design |
| E002 | D103 | CAD | CAD classes |
| E002 | D104 | Python | Programming |
| E003 | D104 | python | Programming |
| E003 | D107 | Marketing | Design |
| E004 | D105 | Business | BA |
| E004 | D107 | Management | Leadership |
| E005 | D104 | Python | Programming |
| E006 | D107 | DBMS | SQL |

1. Now change that database-keeping its content the same-to convert it to 4NF. Show FDDs. Explain why it is in 4NF.

Ans: For a table to be in 4th NF, it should be in BCNF and no presence of multi valued dependencies. We make the required changes.

|  |  |  |
| --- | --- | --- |
| **Emp ID** | **DID** | **Skill** |
| E001 | D101 | Communication |
| E001 | D102 | Marketing |
| E002 | D103 | CAD |
| E002 | D104 | Python |
| E003 | D104 | python |
| E003 | D107 | Marketing |
| E004 | D105 | Business |
| E004 | D107 | Management |
| E005 | D104 | Python |
| E006 | D107 | DBMS |

|  |  |
| --- | --- |
| **Skill** | **Training** |
| Communication | Public speking |
| Marketing | Design |
| CAD | CAD classes |
| Python | Programming |
| python | Programming |
| Marketing | Design |
| Business | BA |
| Management | Leadership |
| Python | Programming |
| DBMS | SQL |

Here there are no multivalued dependency present. Hence We can say that these tables are in 4th NF.

1. What is the purpose of functional dependency diagrams? In the steps taken to develop a data model, when are FDDs used? That is, are FDDs used before or after an ERD is constructed?

Ans: After we've used an Entity-Relationship Diagram (ERD) to establish the general structure of our data, Functional Dependency Diagrams (FDDs) become relevant. We get a broad overview of the entities via the ERD. After getting a rough notion, we use functional dependency diagrams (FDDs) to further explore the functional relationships between these entities and their components.