**Normalization Exercise:**

**Q1: Consider the following:**

1. **Production Company database schema, which keeps records of production machines, its operator and produced components.**

Production ( MachineNo, MachineType, OperatorName, { ComponenetNo, ComponentType } )

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Borrowing No:**  245163  **Borrowing Date:** 07/02/05  **Return Date:** 28/03/05 | | | | **Borrower Code:** A562  **Borrower Name:** J Phillips  **Borrower Address:** Belfast | | |
| **BookID** | **BookName** | **BookType** | **Author** | | **PublisherName** | **PublisherLocation** |
| 76283 | Gormenghast | Fantasy | Peake | | Penguin | London |
| 56152 | Maskerade | SciFi | Pratchett | | MacMillan | Basingstoke |
| 48293 | Moonstone | Adventure | Collins | | Penguin | London |
| 67123 | Puckoon | Comedy | Milligan | | OUP | Oxford |

**Normalize the above relations to the 3NF, showing appropriate dependency diagrams to justify decomposition.**

**First Normal Form (1NF)**

* A relation is in first normal form if it meets the [definition of a relation](http://holowczak.com/database-normalization/#relationdef):
  1. Each attribute (column) value must be a single value only.
  2. All values for a given attribute (column ) must be of the same type.
  3. Each attribute (column) name must be unique.
  4. The order of attributes (columns) is insignificant
  5. No two tuples (rows) in a relation can be identical.
  6. The order of the tuples (rows) is insignificant.
* If you have a *key* defined for the relation, then you can meet the *unique row* requirement.
* Example relation in 1NF (note that key attributes are underlined):
* STOCKS (Company, Symbol, Headquarters, Date, Close\_Price)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Company** | **Symbol** | **Headquarters** | **Date** | **Close Price** |
| Microsoft | MSFT | Redmond, WA | 09/07/2013 | 23.96 |
| Microsoft | MSFT | Redmond, WA | 09/08/2013 | 23.93 |
| Microsoft | MSFT | Redmond, WA | 09/09/2013 | 24.01 |
| Oracle | ORCL | Redwood Shores, CA | 09/07/2013 | 24.27 |
| Oracle | ORCL | Redwood Shores, CA | 09/08/2013 | 24.14 |
| Oracle | ORCL | Redwood Shores, CA | 09/09/2013 | 24.33 |

Note that the key (which consists of the Symbol and the Date) can uniquely determine the Company, headquarters and Close Price of the stock. Here was assume that Symbol must be unique but Company, Headquarters, Date and Price are not unique

**Second Normal Form (2NF)**

* A relation is in second normal form (2NF) if all of its non-key attributes are dependent on all of the *key*.
* Relations that have a single attribute for a key are automatically in 2NF.
* This is one reason why we often use artificial identifiers as keys.
* In the example below, Close Price is dependent on Company, Date
* The following example relation *is not* in 2NF:
* STOCKS (Company, Symbol, Headquarters, Date, Close\_Price)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Company** | **Symbol** | **Headquarters** | **Date** | **Close Price** |
| Microsoft | MSFT | Redmond, WA | 09/07/2013 | 23.96 |
| Microsoft | MSFT | Redmond, WA | 09/08/2013 | 23.93 |
| Microsoft | MSFT | Redmond, WA | 09/09/2013 | 24.01 |
| Oracle | ORCL | Redwood Shores, CA | 09/07/2013 | 24.27 |
| Oracle | ORCL | Redwood Shores, CA | 09/08/2013 | 24.14 |
| Oracle | ORCL | Redwood Shores, CA | 09/09/2013 | 24.33 |

* To start the normalization process, list the functional dependencies (FD):

* FD1: Symbol, Date → Company, Headquarters, Close Price
* FD2: Symbol → Company, Headquarters
* Consider that Symbol, Date → Close Price.

So we might use Symbol, Date as our key.

* However: Symbol → Headquarters

This violates the rule for 2NF. Also, consider the insertion and deletion anomalies.

* Another name for this is a *Partial key dependency*. Symbol is only a “part” of the key and it determines a non-key attribute.
* **One Solution:** Split this up into two new relations:
* COMPANY (Company, Symbol, Headquarters)
* STOCK\_PRICES (Symbol, Date, Close\_Price)
* At this point we have two new relations in our relational model. The original “STOCKS” relation we started with is removed form the model.
* Sample data and functional dependencies for the two new relations:
* COMPANY Relation:

|  |  |  |
| --- | --- | --- |
| **Company** | **Symbol** | **Headquarters** |
| Microsoft | MSFT | Redmond, WA |
| Oracle | ORCL | Redwood Shores, CA |

* FD1: Symbol → Company, Headquarters
* STOCK\_PRICES relation:

|  |  |  |
| --- | --- | --- |
| **Symbol** | **Date** | **Close Price** |
| MSFT | 09/07/2013 | 23.96 |
| MSFT | 09/08/2013 | 23.93 |
| MSFT | 09/09/2013 | 24.01 |
| ORCL | 09/07/2013 | 24.27 |
| ORCL | 09/08/2013 | 24.14 |
| ORCL | 09/09/2013 | 24.33 |

* FD1: Symbol, Date → Close Price
* In checking these new relations we can confirm that they meet the definition of 1NF (each one has well defined unique keys) and 2NF (no partial key dependencies).

**Third Normal Form (3NF)**

* A relation is in third normal form (3NF) if it is in [second normal form](http://holowczak.com/database-normalization/3/#2NF) and it contains no *transitive dependencies*.
* Consider relation R containing attributes A, B and C. R(A, B, C)
* If A → B and B → C then A → C
* **Transitive Dependency**: Three attributes with the above dependencies.
* Example: At CUNY:
* Course\_Code → Course\_Number, Section
* Course\_Number, Section → Classroom, Professor
* Consider one of the new relations we created in the STOCKS example for 2nd normal form:

|  |  |  |
| --- | --- | --- |
| **Company** | **Symbol** | **Headquarters** |
| Microsoft | MSFT | Redmond, WA |
| Oracle | ORCL | Redwood Shores, CA |

* The functional dependencies we can see are:
* FD1: Symbol → Company
* FD2: Company → Headquarters
* so therefore:
* Symbol → Headquarters
* This is a transitive dependency.
* What happens if we remove Oracle?

We loose information about 2 different facts.

* The solution again is to split this relation up into two new relations:
* STOCK\_SYMBOLS(Company, Symbol)
* COMPANY\_HEADQUARTERS(Company, Headquarters)
* This gives us the following sample data and FD for the new relations

|  |  |
| --- | --- |
| **Company** | **Symbol** |
| Microsoft | MSFT |
| Oracle | ORCL |

* FD1: Symbol → Company

|  |  |  |
| --- | --- | --- |
|  | **Company** | **Headquarters** |
|  | Microsoft | Redmond, WA |
|  | Oracle | Redwood Shores, CA |

* FD1: Company → Headquarters
* Again, each of these new relations should be checked to ensure they meet the definition of 1NF, 2NF and now 3NF.