**Study and check how we put the below table in to the normalized form.**

OID = Order ID, O\_Date = Order Date,

CID = Customer ID, C\_Name = Customer Name, C\_State = Customer’s State,

PID = project id, P\_Desc =Project Name, P\_Price = Product Price, Qty = Quantity Purchased Note: 7, 5, 4 means three Product IDs. Similarly, 1, 1, 5 means three Quantities.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **OID** | **O\_Date** | **CID** | **C\_Name** | **C\_State** | **PID** | **P\_Desc** | **P\_Price** | **Qty** |
| 1006 | 10/24/09 | 2 | Apex | NC | 7, 5, 4 | Table,  Desk, Chair | 800,  325,  200 | 1, 1, 5 |
| 1007 | 10/25/09 | 6 | Acme | GA | 11, 4 | Dresser, Chair | 500,  200 | 4, 6 |

Put the table in 1NF Tables

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **OID** | **O\_Date** | **CID** | **C\_Name** | **C\_State** | **PID** | **P\_Desc** | **P\_Price** | **Qty** |
| 1006 | 10/24/09 | 2 | Apex | NC | 7 | Table | 800 | 1 |
| 1006 | 10/24/09 | 2 | Apex | NC | 5 | Desk | 325 | 1 |
| 1006 | 10/24/09 | 2 | Apex | NC | 4 | Chair | 200 | 5 |
| 1007 | 10/25/09 | 6 | Acme | GA | 11 | Dresser | 500 | 4 |
| 1007 | 10/25/09 | 6 | Acme | GA | 4 | Chair | 200 | 6 |

Total # of Tables: 1

Now, put the table in 2NF

|  |  |  |
| --- | --- | --- |
| **OID** | **PID** | **Qty** |
| 1006 | 7 | 1 |
| 1006 | 5 | 1 |
| 1006 | 4 | 5 |
| 1007 | 11 | 4 |
| 1007 | 4 | 6 |

|  |  |  |
| --- | --- | --- |
| **PID** | **P\_Desc** | **P\_Price** |
| 7 | Table | 800 |
| 5 | Desk | 325 |
| 4 | Chair | 200 |
| 11 | Dresser | 500 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **OID** | **O\_Date** | **CID** | **C\_Name** | **C\_State** |
| 1006 | 10/24/09 | 2 | Apex | NC |
| 1006 | 10/24/09 | 2 | Apex | NC |
| 1006 | 10/24/09 | 2 | Apex | NC |
| 1007 | 10/25/09 | 6 | Acme | GA |
| 1007 | 10/25/09 | 6 | Acme | GA |

All three tables make up the second normal form. Total # of Tables: 3

Now, put the tables in 3NF Tables

|  |  |  |
| --- | --- | --- |
| **OID** | **PID** | **Qty** |
| 1006 | 7 | 1 |
| 1006 | 5 | 1 |
| 1006 | 4 | 5 |
| 1007 | 11 | 4 |
| 1007 | 4 | 6 |

|  |  |  |
| --- | --- | --- |
| **PID** | **P\_Desc** | **P\_Price** |
| 7 | Table | 800 |
| 5 | Desk | 325 |
| 4 | Chair | 200 |
| 11 | Dresser | 500 |

|  |  |  |
| --- | --- | --- |
| **OID** | **O\_Date** | **CID** |
| 1006 | 10/24/09 | 2 |
| 1007 | 10/25/09 | 6 |

|  |  |  |
| --- | --- | --- |
| **CID** | **C\_Name** | **C\_State** |
| 2 | Apex | NC |
| 6 | Acme | GA |

Please note that in all four of these tables, the rows are unique and none of the primary keys are repeated.

Total # of Tables: 4

Final set of Tables with meaningful names and PKs and FKs

In this step, you must name the tables that were created and finalized in 3NF. Table names must have a meaningful name such that some third party looking at your design will know immediately what that table is used for. In the case where the table has a single primary key, you can call the table by the key such as Order, Product, or Customer. The other strategy is to call it Order\_Info, Product\_Info, or Customer\_Info. The choice is entirely up to you and both are correct. The first table does not have a single primary key; instead it has a composite key and so the naming convention is somewhat different. One strategy is to call it by one of its owning entities and adding Detail to the end such as Order\_Detail. Another strategy is to name the table by all of its owning entities such as Order- Product.

Better practice is to underline the primary keys and put an asterisk (\*) at the end of a foreign key. For a composite key, underline all attributes of the key and put an asterisk after each one. The results are the following four table names:

Order-Product (OID\*, PID\*, Qty) Product\_Info (PID, P\_Desc, P\_Price) Order\_Info (OID, O\_Date, CID\*) Customer\_Info (CID, C\_Name, C\_State)