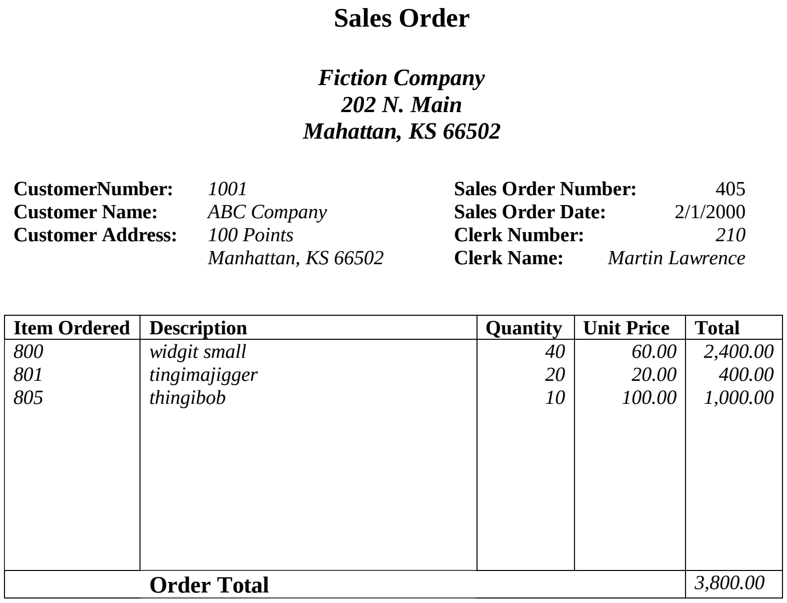
See Sales Order from below:



Sales Order

Fiction Company
202 N. Main
Mahattan, KS 66502

CustomerNumber:	*1001*	**Sales Order Number:**	405
Customer Name:	*ABC Company*	**Sales Order Date:**	2/1/2000
Customer Address:	*100 Points*	**Clerk Number:**	*210*
	Manhattan, KS 66502	**Clerk Name:**	*Martin Lawrence*

Item Ordered	Description	Quantity	Unit Price	Total
800	*widgit small*	*40*	*60.00*	*2,400.00*
801	*tingimajigger*	*20*	*20.00*	*400.00*
805	*thingibob*	*10*	*100.00*	*1,000.00*
	Order Total			*3,800.00*

Fields in the original data table will be as follows:

SalesOrderNo, Date, CustomerNo, CustomerName, CustomerAdd, ClerkNo, ClerkName, ItemNo, Description, Qty, UnitPrice

Think of this as the baseline – one large table

Normalization: First Normal Form

- Separate Repeating Groups into New Tables.
- ***Repeating Groups*** Fields that may be repeated several times for one document/entity
- Create a new table containing the repeating data
- The primary key of the new table (repeating group) is always a composite key; Usually document number and a field uniquely describing the repeating line, like an item number.

First Normal Form Example

The new table is as follows:

SalesOrderNo, ItemNo, Description, Qty, UnitPrice

The repeating fields will be removed from the original data table, leaving the following.

SalesOrderNo, Date, CustomerNo, CustomerName, CustomerAdd, ClerkNo, ClerkName

These two tables are a database in first normal form

Normalization: Second Normal Form

- Remove Partial Dependencies.
- ***Functional Dependency*** The value of one attribute in a table is determined entirely by the value of another.
- ***Partial Dependency*** A type of functional dependency where an attribute is functionally dependent on only part of the primary key (primary key must be a composite key).
- Create separate table with the functionally dependent data and the part of the key on which it depends. Tables created at this step will usually contain descriptions of resources.

Second Normal Form Example

The new table will contain the following fields:
ItemNo, Description

All of these fields except the primary key will be removed from the original table. The primary key will be left in the original table to allow linking of data:
SalesOrderNo, ItemNo, Qty, UnitPrice

Never treat price as dependent on item. Price may be different for different sales orders (discounts, special customers, etc.)
Along with the unchanged table below, these tables make up a database in second normal form:
SalesOrderNo, Date, CustomerNo, CustomerName, CustomerAdd, ClerkNo, ClerkName

Normalization: Third Normal Form

- Remove transitive dependencies.
- ***Transitive Dependency*** A type of functional dependency where an attribute is functionally dependent on an attribute other than the primary key. Thus its value is only indirectly determined by the primary key.
- Create a separate table containing the attribute and the fields that are functionally dependent on it. Tables created at this step will usually contain descriptions of either resources or agents. Keep a copy of the key attribute in the original file.

Third Normal Form Example

The new tables would be:
CustomerNo, CustomerName, CustomerAdd
ClerkNo, ClerkName

All of these fields except the primary key will be removed from the original table. The primary key will be left in the original table to allow linking of data as follows:
SalesOrderNo, Date, CustomerNo, ClerkNo

Together with the unchanged tables below, these tables make up the database in third normal form.
ItemNo, Description
SalesOrderNo, ItemNo, Qty, UnitPrice

Completed Tables in Third Normal Form

Customers: CustomerNo, CustomerName, CustomerAdd
Clerks: ClerkNo, ClerkName
Inventory Items: *ItemNo*, Description
Sales Orders: SalesOrderNo, Date, CustomerNo, ClerkNo
SalesOrderDetail: SalesOrderNo, ItemNo, Qty, UnitPrice