**Business Rules**

1. ProductID must be unique for every product and require to have six digits.
2. Phone number must have 11 digits, no more than or less than it.
3. Email addresses are required to follow the format of @domain\_name.
4. Customers can be eligible for only five promo codes based on the promo code’s criteria.
5. Customers can get 10% discounts on products using DISCOUNT10, not for other promo codes.
6. Customers can get 5% discounts on products using the STAFFDISC promo.
7. Customers can get 15% discounts on products using the SPRING promo.
8. STAFFDISC, SPRING15, DISCOUNT10 must come under the DiscountAppliedPC category which means the discount of 5,15,10 percentage respectively should be applied on the TotalSalePrice and updated on NewTotalSalePrice.
9. PROMO10, FIVEOFF must come under DiscountAppliedabs category which means the deduction of 10 and 5 respectively on TotalSalePrice and updated on NewTotalSalePrice .
10. After applying the promo code, it must require that the NewTotalSales price is less than compared to TotalSalePrice.
11. The description must have only words for the product.
12. SaleDate contains the date and time of order of customers, and it must be 12 digits.
13. Attribute customerAddress5 only has the postal code of the customer’s address which is not greater than 7 letters and not less than 6 letters.
14. Customer name attributes are built by combining customer first name and customer’s last name, and other things are not accepted.
15. A repeat customer has only two values, No and Yes; when one particular customer is repeated, then it must show Yes otherwise, it is needed to guide No.
16. Product Id must range between 4-6 digits.
17. Customer Address1 should have Apartment or street Number and its Name.
18. Except @ not any special characters should be used in Email.
19. Products whose TotalSalePrice is NULL should not eligible for any sort of DISCOUTN or PROMOCDE.
20. Customer’s whose CustomerEmail Field is NULL is not be eligible for DISCOUNT PromoCode.
21. Customer address one must be combined with street number and street address, nothing else.
22. Customer addresses three contains the only area of their address, and it must be in one word
23. Postal code is separated by space from total character’s median
24. Each buyer must assign one and only one discount to an order.
25. Each estimate should include an amount.
26. System must provide supplier return address.
27. Discount Code cannot be applied on the products with a clearance tag.
28. SaleDate should store the data for over a year which is from 2015 to 2016.
29. TotalSalesprice should not be less than NewTotalSalesPrice more so it should be equal.

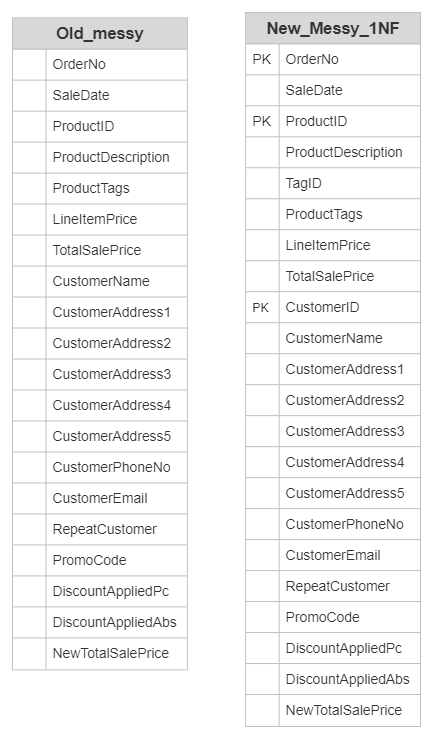
**Normalization**

The table **Messy** is not in 1NF How do we know that? Let’s discuss the rules of 1NF, 2NF, 3NF.

1. 1NF: Given below are the established rules to achieve the 1NF.

* Each cell to be single valued (Atomic Values).
* Entries in a column are same type.
* Rows uniquely identified- Add unique ID or Add more columns to make unique ID.
* Table stores data be in rows and columns.
* ⦁In Messy table, there were repeated values in ProductTags column. It was showing three tags in one column. To achieve the first rule of 1NF, we added one more column TagID to table, which referred to individual tag.
* There was no Primary Key in table. According to the 3rd rule of 1NF there should be one or more unique ID to identify the rows uniquely. So, we altered the Messy table and make the OrderNo, ProductID as Primary key and we added one more column CustomerID as primary key for customer data.

**Note:** Check below diagram for detail understanding**.**



**1NF**

1. 2NF: Given below are the established rules to achieve the 1NF

* The table must be in 1NF.
* All attributes (Non key columns) depend on primary key.
* In simple word, we must check each column of table is dependent on any primary key or not. If column is dependent on any of one primary key, then keep that column in table as it is but what if not? Then create another table and move that column into that table. Messy table is already in 1NF. So, Let’s check this condition on Messy table.
* In Messy table the below columns are not dependent on any primary key. Thus, we createdseparate table name as **Product details** and added these below columns to this new table with **ProductID** as **primary key**.

1. ProductID
2. ProductDescription
3. TagID
4. ProductTag
5. LineItemPrice

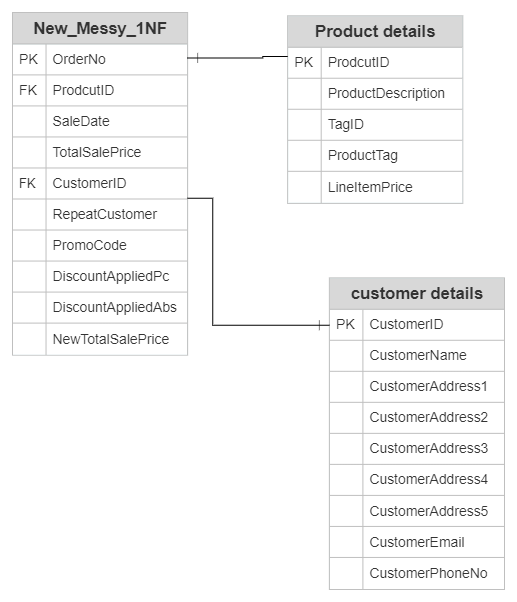
* Similarly, below columns are not dependent on any primary key. So, we created a new table name as **Customer Details** and added these below columns to this new table with **CustomerID** as **primary key**.

1. CustomerID
2. CustomerName
3. CustomerAddress1
4. CustomerAddress2
5. CustomerAddress3
6. CustomerAddress4
7. CustomerAddress5
8. CustomerEmail
9. CustomerPhoneNo

* However, other remaining columns are dependent on **OrderNo** primary key, so we kept that column as it is in **New\_Messy\_1NF** table. There are two foreign key columns also **ProductID** and **CustomerID** in that table**.**

Now we can say that tables are in 2NF

**Note:** Check the diagram on next page for detail understanding of 2NF.



**2NF**

1. 3NF: Given below are the established rules to achieve the 1NF.

* All the columns in table can be determined by the primary key in the table and no other column (non-Transitive dependency).
* Table must be in 2NF
* First check the **Product Details** table, is every column dependent on the primary key or not with non-Transitive dependency?

Observe **ProductTag** column is not uniquely dependent on **ProductID** primary key. Thus, again we must create new table **Tag\_Details** and move the above column into **Tag\_Details** table.

Add one more column in same (**Tag\_Details**) table **TagID** as primary key to the column **ProductTag** and keep **Product\_details** table with three columns **ProductID, ProductDescription, and LineItemPrice** only**.**

* Create new table **Product\_tag\_join** as intersection table for **Product Details** and **Tag\_Details** table with two foreign key columns in that **ProductID** and **TagID**.
* Now rename the **new\_Messy\_1NF** table by **Order\_Details**.
* Check every column of **Order\_Details** are uniquely dependent on primary key **OrderNo** or not.

Here the **Promocode** column is dependent on **OrderNo** but other two columns **DiscountAppliedAbs, DiscountAppliedPc** are transitively dependent on **Promocode.**

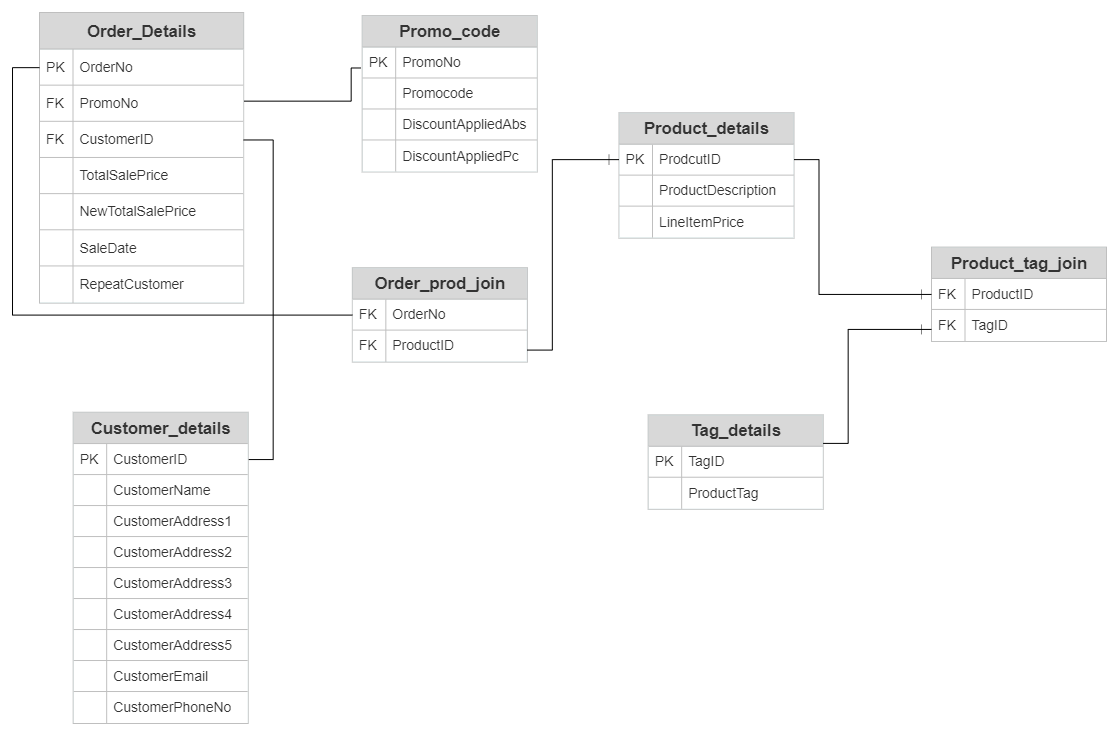
Thus, we must create a new table **Promo\_code** in which move columns **Promocode DiscountAppliedAbs, DiscountAppliedPc.**

Next add new column **PromoNo** as primary key in **Promo\_code** table and as foreign key in **Order\_Details** because **Promo\_code** has some Null values also, so, it can’t be a primary key and foreign key too.

* Create a new table **Order\_prod\_join** as intersection table for **Order\_details** and **Product\_details**. Add **ProductID** and **OrderNo** as foreign key in that table.

Now we can say that tables are in 3NF

**Note:** Check the diagram on next page for detail understanding of 3NF.



**3NF**

**Normalized Table queries:**

USE NF  
 ---------===========================  
 -- Create Promo\_code table  
 ---------==========================

CREATE TABLE Promo\_code(  
 PromoNo int NOT NULL IDENTITY(1,1),  
 Promocode VARCHAR(100),  
 DiscountAppliedPc int,  
 DiscountAppliedAbs NUMERIC(16,2),  
 CONSTRAINT PK\_Promocode PRIMARY KEY (PromoNo)

);

--++++++++++++++++++++  
 -- INSERT data  
 --++++++++++++++++++++

INSERT INTO Promo\_code(Promocode,DiscountAppliedAbs,DiscountAppliedPc) VALUES('STAFFDISC',0,5)  
 INSERT INTO Promo\_code(Promocode,DiscountAppliedAbs,DiscountAppliedPc) VALUES('SPRING15',0,15)  
 INSERT INTO Promo\_code(Promocode,DiscountAppliedAbs,DiscountAppliedPc) VALUES('PROMO10',10.00,0)  
 INSERT INTO Promo\_code(Promocode,DiscountAppliedAbs,DiscountAppliedPc) VALUES('FIVEOFF)',5.00,0)  
 INSERT INTO Promo\_code(Promocode,DiscountAppliedAbs,DiscountAppliedPc) VALUES('DISCOUNT10',0,10)  
 INSERT INTO Promo\_code(Promocode,DiscountAppliedAbs,DiscountAppliedPc) VALUES('No Discount',0,0)

---------===========================  
 -- Create Customer\_details table  
 ---------==========================

CREATE TABLE Customer\_details(  
 CustomerID int NOT NULL IDENTITY(1,1),  
 CustomerName VARCHAR(100),  
 CustomerAddress1 VARCHAR(100),  
 CustomerAddress2 VARCHAR(100),  
 CustomerAddress3 VARCHAR(100),  
 CustomerAddress4 VARCHAR(100),  
 CustomerAddress5 VARCHAR(100),  
 CustomerPhoneNo VARCHAR(20),  
 CustomerEmail VARCHAR(100),

CONSTRAINT PK\_Customer\_details PRIMARY KEY (CustomerID)  
 );  
 --++++++++++++++++++++  
 -- INSERT data  
 --++++++++++++++++++++

INSERT INTO Customer\_details(CustomerName,CustomerAddress1,CustomerAddress2,CustomerAddress3,CustomerAddress4,CustomerAddress5,CustomerPhoneNo,CustomerEmail)  
 VALUES('ALEX STOLTCHNEV','423 Fabien Freeway','St. Bees','Selkirkshire','Derbyshire','CV81 1KK','07717 [472383','rjjxjzf.qcvumrmq@vimois.org](mailto:472383','rjjxjzf.qcvumrmq@vimois.org)')  
 INSERT INTO Customer\_details(CustomerName,CustomerAddress1,CustomerAddress2,CustomerAddress3,CustomerAddress4,CustomerAddress5,CustomerPhoneNo,CustomerEmail)  
 VALUES('Aaron Bradford','27 Green Oak Road','Presteigne','Essex',' '[,'TA447ZN','02771611652','hgwotqj.uyhfwcm@elmdht.org](mailto:,'TA447ZN','02771611652','hgwotqj.uyhfwcm@elmdht.org)')  
 INSERT INTO Customer\_details(CustomerName,CustomerAddress1,CustomerAddress2,CustomerAddress3,CustomerAddress4,CustomerAddress5,CustomerPhoneNo,CustomerEmail)  
 VALUES('Elisabeth Koch','274 Clarendon Avenue','Barnetby','Nottinghamshire',' ','SP41 7EM','02121 [912333','fvvebagi442@vupazt.gaejrr.com](mailto:912333','fvvebagi442@vupazt.gaejrr.com)')  
 INSERT INTO Customer\_details(CustomerName,CustomerAddress1,CustomerAddress2,CustomerAddress3,CustomerAddress4,CustomerAddress5,CustomerPhoneNo,CustomerEmail)  
 VALUES('Elisabeth Mendez','878 Milton Street','Ilminster','Dunbartonshire','Gwent','EC6 9UT','01717 [152024','jclhxjlt.ahye@ehgnqk.com](mailto:152024','jclhxjlt.ahye@ehgnqk.com)')  
 INSERT INTO Customer\_details(CustomerName,CustomerAddress1,CustomerAddress2,CustomerAddress3,CustomerAddress4,CustomerAddress5,CustomerPhoneNo,CustomerEmail)  
 VALUES('Elizabeth Mc Gee','931 Hague Blvd.','High Wycombe','Buckinghamshire','Suffolk','TA8 [4YX','01721580737','eoumdfa@kbspvg.net](mailto:4YX','01721580737','eoumdfa@kbspvg.net)')  
 INSERT INTO Customer\_details(CustomerName,CustomerAddress1,CustomerAddress2,CustomerAddress3,CustomerAddress4,CustomerAddress5,CustomerPhoneNo,CustomerEmail)  
 VALUES('Emma Lamb','64 South Green Milton Blvd.','Feltham','Rutland','Stirlingshire','DH1 3DX','01111 [111332','qipl681@ptajdszv.owxc-v.org](mailto:111332','qipl681@ptajdszv.owxc-v.org)')  
 INSERT INTO Customer\_details(CustomerName,CustomerAddress1,CustomerAddress2,CustomerAddress3,CustomerAddress4,CustomerAddress5,CustomerPhoneNo,CustomerEmail)  
 VALUES('Enrique Mccann','624 New Freeway','New Cross','Tyrone',' ','KT9 [9TX','02127375261','xzww@tzsbpdeq.tmwzgb.com](mailto:9TX','02127375261','xzww@tzsbpdeq.tmwzgb.com)')  
 INSERT INTO Customer\_details(CustomerName,CustomerAddress1,CustomerAddress2,CustomerAddress3,CustomerAddress4,CustomerAddress5,CustomerPhoneNo,CustomerEmail)  
 VALUES('Erin Rhodes','77 Rocky Clarendon Street','Shildon','Tyrone','Suffolk','CF25 [7GD','02172063464','xfsr63@ruaznomce.ldnbbi.net](mailto:7GD','02172063464','xfsr63@ruaznomce.ldnbbi.net)')  
 INSERT INTO Customer\_details(CustomerName,CustomerAddress1,CustomerAddress2,CustomerAddress3,CustomerAddress4,CustomerAddress5,CustomerPhoneNo,CustomerEmail)  
 VALUES('Esteban Andrade','349 Cowley Blvd.','Criccieth','Hertfordshire',' ','H61 1SK','02771 [295455','cbzvki.rfyof@hghmjvbzw.-otmvu.org](mailto:295455','cbzvki.rfyof@hghmjvbzw.-otmvu.org)')  
 INSERT INTO Customer\_details(CustomerName,CustomerAddress1,CustomerAddress2,CustomerAddress3,CustomerAddress4,CustomerAddress5,CustomerPhoneNo,CustomerEmail)  
 VALUES('Francis Sexton','417 Clarendon Drive','Ulceby','Lancashire','Norfolk','LL5 2KX','07771 [462614','diarcgbf598@hvpovzepg.obgre-.org](mailto:462614','diarcgbf598@hvpovzepg.obgre-.org)')

---------===========================  
 -- Create Orders\_Details table  
 ---------==========================

CREATE TABLE Orders\_Details(  
 OrderNo VARCHAR(40) NOT NULL,  
 SaleDate VARCHAR(14),  
 PromoNo int NOT NULL,  
 CustomerID int NOT NULL,  
 RepeatCustomer VARCHAR(3),  
 TotalSalePrice NUMERIC(16,2),  
 NewTotalSalePrice NUMERIC(16,2),

CONSTRAINT PK\_Orders\_Details PRIMARY KEY (OrderNo),  
   
);

ALTER TABLE dbo.Orders\_Details   
ADD CONSTRAINT FK\_order\_details\_\_promo\_code FOREIGN KEY(PromoNo)  
 REFERENCES dbo.Promo\_code (PromoNo)  
 ON UPDATE NO ACTION ON DELETE NO ACTION;

ALTER TABLE dbo.Orders\_Details   
ADD CONSTRAINT FK\_order\_details\_\_customer\_details FOREIGN KEY(CustomerID)  
 REFERENCES dbo.Customer\_details (CustomerID)  
 ON UPDATE NO ACTION ON DELETE NO ACTION;

--++++++++++++++++++++  
 -- INSERT data  
 --++++++++++++++++++++

INSERT INTO Orders\_Details(OrderNo,SaleDate,PromoNo,CustomerID,RepeatCustomer,TotalSalePrice,NewTotalSalePrice)  
 VALUES ('00005AEE-4C5F-4744-80ED-A69840F9989A','20151207072003',6,2,'No',2063.73,2063.73)  
 INSERT INTO Orders\_Details(OrderNo,SaleDate,PromoNo,CustomerID,RepeatCustomer,TotalSalePrice,NewTotalSalePrice)  
 VALUES ('00A950D7-1F73-4416-80CE-F0EF58959759','20150528160400',5,10,'No',1703.21,1532.89)  
 INSERT INTO Orders\_Details(OrderNo,SaleDate,PromoNo,CustomerID,RepeatCustomer,TotalSalePrice,NewTotalSalePrice)  
 VALUES ('01ACBCF4-C977-4B6C-855C-0B28973AF556','20150720095424',2,10,'Yes',611.66,519.91)  
 INSERT INTO Orders\_Details(OrderNo,SaleDate,PromoNo,CustomerID,RepeatCustomer,TotalSalePrice,NewTotalSalePrice)  
 VALUES ('02EE4DF4-0D1F-43FE-A06C-1A7A0D7FEA5D','20151105081628',3,7,'No',1492.19,1482.19)  
 INSERT INTO Orders\_Details(OrderNo,SaleDate,PromoNo,CustomerID,RepeatCustomer,TotalSalePrice,NewTotalSalePrice)  
 VALUES ('03D6C048-5833-4B27-8E8D-C18257DE5F9F','20160115044842',6,2,'Yes',674.80,674.80)  
 INSERT INTO Orders\_Details(OrderNo,SaleDate,PromoNo,CustomerID,RepeatCustomer,TotalSalePrice,NewTotalSalePrice)  
 VALUES ('04BE6D61-3C65-4523-B260-9609FA091F10','20160117151548',2,1,'No',1152.84,979.91)  
 INSERT INTO Orders\_Details(OrderNo,SaleDate,PromoNo,CustomerID,RepeatCustomer,TotalSalePrice,NewTotalSalePrice)  
 VALUES ('05A49FC7-B938-43CC-9E3D-B5F40C5F107C','20150628030821',4,3,'No',1521.23,1516.23)  
 INSERT INTO Orders\_Details(OrderNo,SaleDate,PromoNo,CustomerID,RepeatCustomer,TotalSalePrice,NewTotalSalePrice)  
 VALUES ('08800E16-6CC8-4156-8C39-768375F06473','20151229064709',1,7,'No',964.74,916.50)  
 INSERT INTO Orders\_Details(OrderNo,SaleDate,PromoNo,CustomerID,RepeatCustomer,TotalSalePrice,NewTotalSalePrice)  
 VALUES ('093B3888-45DF-47C6-B723-27BCC10FDDD0','20151211001655',6,8,'No',1400.59,1400.59)

--------------------drop the constraint-----------------

--ALTER TABLE dbo.Orders\_Details DROP CONSTRAINT FK\_order\_details\_promo\_code;  
 --ALTER TABLE dbo.Orders\_Details DROP CONSTRAINT FK\_order\_details\_customer\_details;  
 -------------------add new constraint-------------------

---------===========================  
 -- Create Product\_details table  
 ---------==========================

CREATE TABLE Product\_details (  
 ProductID int NOT NULL,  
 ProductDescription varchar(255),  
 LineItemPrice NUMERIC(16,2),  
 CONSTRAINT PK\_Product\_details PRIMARY KEY (ProductID)  
 );

--++++++++++++++++++++  
 -- INSERT data  
 --++++++++++++++++++++  
 INSERT INTO Product\_details(ProductID,ProductDescription,LineItemPrice) VALUES(233431,'Inwerpentor',263.28)  
 INSERT INTO Product\_details(ProductID,ProductDescription,LineItemPrice) VALUES(915439,'Adtanin',361.18)  
 INSERT INTO Product\_details(ProductID,ProductDescription,LineItemPrice) VALUES(903053,'Surglibantor',77.50)  
 INSERT INTO Product\_details(ProductID,ProductDescription,LineItemPrice) VALUES(644239,'Tupvenefantor',284.68)  
 INSERT INTO Product\_details(ProductID,ProductDescription,LineItemPrice) VALUES(516914,'Puroban',59.82)  
 INSERT INTO Product\_details(ProductID,ProductDescription,LineItemPrice) VALUES(523403,'Groquestover',237.58)  
 INSERT INTO Product\_details(ProductID,ProductDescription,LineItemPrice) VALUES(436404,'Mepebilentor',186.69)  
 INSERT INTO Product\_details(ProductID,ProductDescription,LineItemPrice) VALUES(640789,'Punepentor',222.98)  
 INSERT INTO Product\_details(ProductID,ProductDescription,LineItemPrice) VALUES(974918,'Winvenentor',181.77)  
 INSERT INTO Product\_details(ProductID,ProductDescription,LineItemPrice) VALUES(806360,'Qwizapplazz',124.23)

---------===========================  
 -- Create Order\_prod\_join table  
 ---------==========================

CREATE TABLE Order\_prod\_join (  
 OrderNo varchar(40),  
 ProductID int,

);

ALTER TABLE dbo.Order\_prod\_join   
ADD CONSTRAINT FK\_Order\_prod\_join\_\_Orders\_Details FOREIGN KEY(OrderNo)  
 REFERENCES dbo.Orders\_Details (OrderNo)  
 ON UPDATE NO ACTION ON DELETE NO ACTION;

ALTER TABLE dbo.Order\_prod\_join   
ADD CONSTRAINT FK\_order\_prod\_join\_\_product\_details   
FOREIGN KEY (ProductID) REFERENCES Product\_details(ProductID)  
 ON UPDATE NO ACTION ON DELETE NO ACTION;

--++++++++++++++++++++  
 -- INSERT data  
 --++++++++++++++++++++

INSERT INTO Order\_prod\_join(OrderNo,ProductID) VALUES('00005AEE-4C5F-4744-80ED-A69840F9989A',233431)  
 INSERT INTO Order\_prod\_join(OrderNo,ProductID) VALUES('00005AEE-4C5F-4744-80ED-A69840F9989A',903053)  
 INSERT INTO Order\_prod\_join(OrderNo,ProductID) VALUES('00005AEE-4C5F-4744-80ED-A69840F9989A',644239)  
 INSERT INTO Order\_prod\_join(OrderNo,ProductID) VALUES('03D6C048-5833-4B27-8E8D-C18257DE5F9F',436404)  
 INSERT INTO Order\_prod\_join(OrderNo,ProductID) VALUES('03D6C048-5833-4B27-8E8D-C18257DE5F9F',523403)  
 INSERT INTO Order\_prod\_join(OrderNo,ProductID) VALUES('00A950D7-1F73-4416-80CE-F0EF58959759',644239)  
 INSERT INTO Order\_prod\_join(OrderNo,ProductID) VALUES('00A950D7-1F73-4416-80CE-F0EF58959759',640789)  
 INSERT INTO Order\_prod\_join(OrderNo,ProductID) VALUES('02EE4DF4-0D1F-43FE-A06C-1A7A0D7FEA5D',806360)  
 INSERT INTO Order\_prod\_join(OrderNo,ProductID) VALUES('02EE4DF4-0D1F-43FE-A06C-1A7A0D7FEA5D',516914)  
 INSERT INTO Order\_prod\_join(OrderNo,ProductID) VALUES('02EE4DF4-0D1F-43FE-A06C-1A7A0D7FEA5D',915439)

---------===========================  
 -- Create Tag\_details table  
 ---------==========================

CREATE TABLE Tag\_details (  
 TagID int NOT NULL IDENTITY(1,1),  
 ProductTag varchar(255),  
 CONSTRAINT PK\_Tag\_details PRIMARY KEY (TagID)  
 );

--++++++++++++++++++++  
 -- INSERT data  
 --++++++++++++++++++++

INSERT INTO Tag\_details(ProductTag) VALUES('Tools')  
 INSERT INTO Tag\_details(ProductTag) VALUES('Garden')  
 INSERT INTO Tag\_details(ProductTag) VALUES('Clearance')  
 INSERT INTO Tag\_details(ProductTag) VALUES('Home')  
 INSERT INTO Tag\_details(ProductTag) VALUES('DIY')

---------===========================  
 -- Create Product\_tag\_join table  
 ---------==========================

CREATE TABLE Product\_tag\_join (  
 ProductID int,  
 TagID int

);

ALTER TABLE dbo.Product\_tag\_join   
ADD CONSTRAINT FK\_product\_tag\_join\_\_product\_details   
FOREIGN KEY (ProductID) REFERENCES Product\_details(ProductID)  
 ON UPDATE NO ACTION ON DELETE NO ACTION;

ALTER TABLE dbo.Product\_tag\_join   
ADD CONSTRAINT FK\_product\_tag\_join\_\_tag\_details   
FOREIGN KEY (TagID) REFERENCES Tag\_details(TagID)  
 ON UPDATE NO ACTION ON DELETE NO ACTION;

--++++++++++++++++++++  
 -- INSERT data  
 --++++++++++++++++++++

INSERT INTO Product\_tag\_join(ProductID,TagID) VALUES(233431,4)  
 INSERT INTO Product\_tag\_join(ProductID,TagID) VALUES(233431,3)  
 INSERT INTO Product\_tag\_join(ProductID,TagID) VALUES(915439,4)  
 INSERT INTO Product\_tag\_join(ProductID,TagID) VALUES(915439,2)  
 INSERT INTO Product\_tag\_join(ProductID,TagID) VALUES(915439,1)  
 INSERT INTO Product\_tag\_join(ProductID,TagID) VALUES(903053,2)  
 INSERT INTO Product\_tag\_join(ProductID,TagID) VALUES(644239,3)  
 INSERT INTO Product\_tag\_join(ProductID,TagID) VALUES(644239,4)  
 INSERT INTO Product\_tag\_join(ProductID,TagID) VALUES(516914,3)  
 INSERT INTO Product\_tag\_join(ProductID,TagID) VALUES(806360,3)