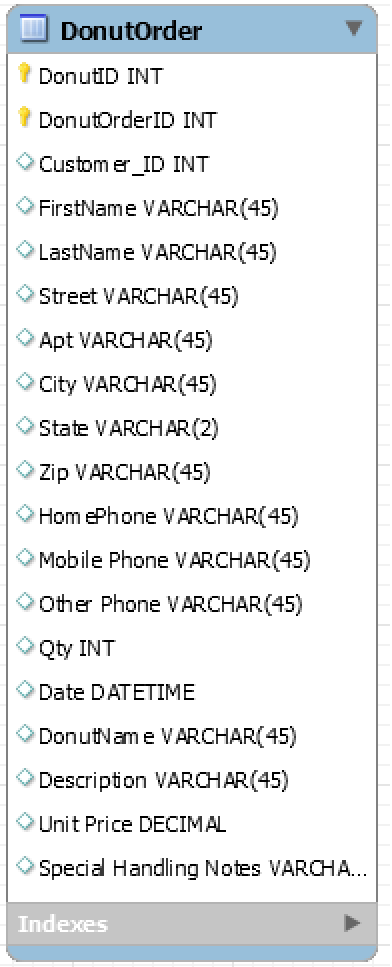
A.1.

a.

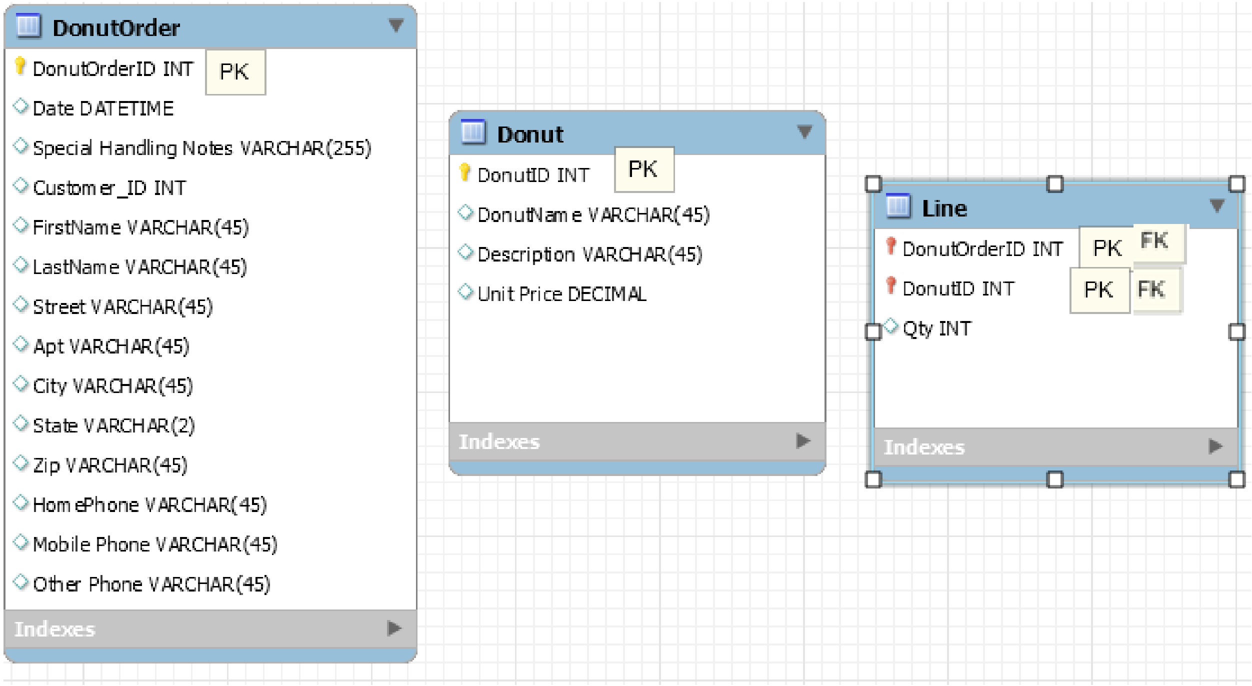


DonutOrder

- DonutID INT
- DonutOrderID INT
- Customer_ID INT
- FirstName VARCHAR(45)
- LastName VARCHAR(45)
- Street VARCHAR(45)
- Apt VARCHAR(45)
- City VARCHAR(45)
- State VARCHAR(2)
- Zip VARCHAR(45)
- HomePhone VARCHAR(45)
- Mobile Phone VARCHAR(45)
- Other Phone VARCHAR(45)
- Qty INT
- Date DATETIME
- DonutName VARCHAR(45)
- Description VARCHAR(45)
- Unit Price DECIMAL
- Special Handling Notes VARCHA...

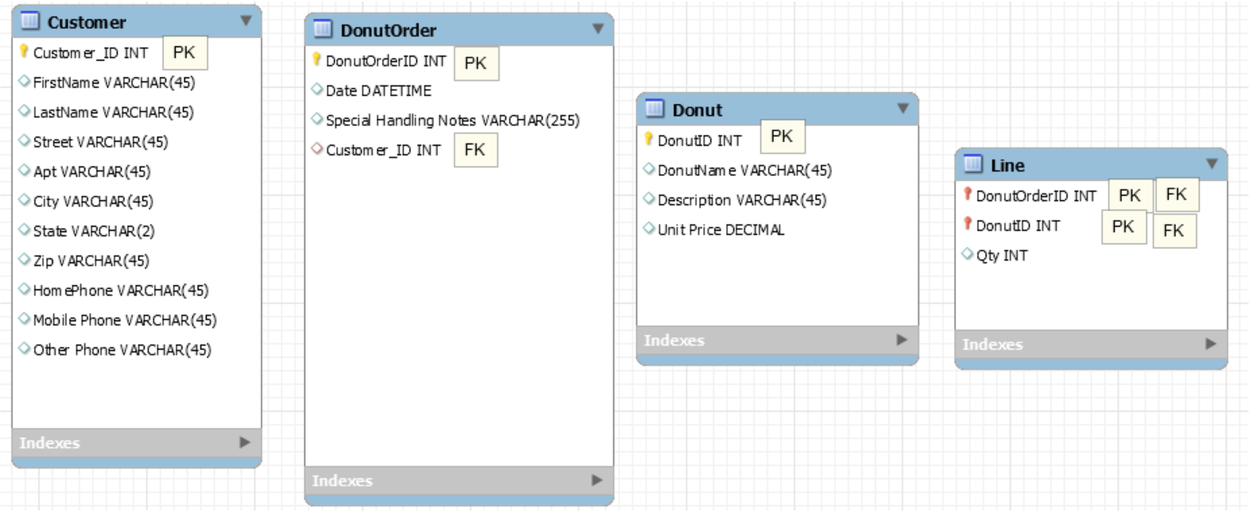
i) I designed this 1st Normal Form table diagram by first opening the attached “Sales Order Form” as instructed. After looking through the Sales Order Form, I picked out all the fields that appeared necessary to be included in the database system. This included the Donut Order ID, Date, Customer ID, FirstName and LastName (of the customer), Street, Apt, City, State, Zip, HomePhone, MobilePhone, Other Phone, Qty, DonutID, DonutName, Description, Unit Price and Special Handling Notes. I chose to use the DonutOrderID and DonutID as the composite primary key, since these should uniquely identify all the attributes in the table.

b.



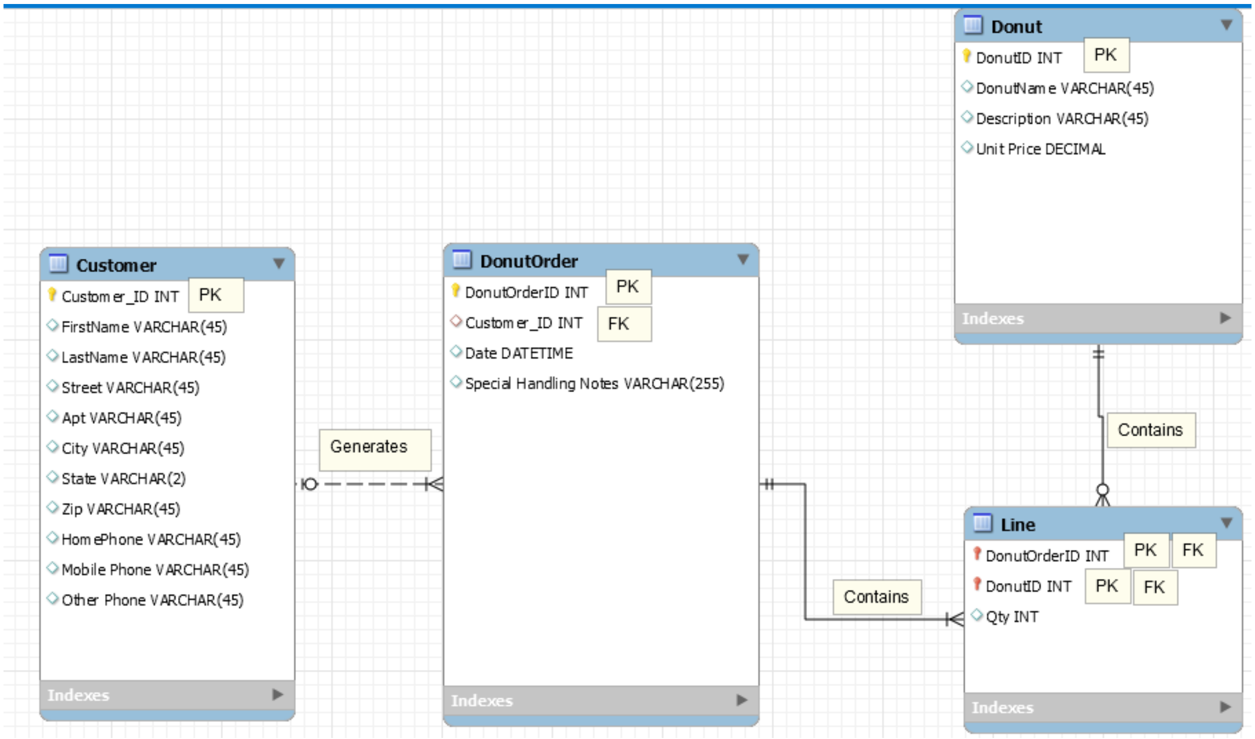
i) I designed this 2nd Normal Form diagram by using the column values in the 1NF to create two new tables, Donut and Line. The appropriate fields that apply to the Donut table were then cut and pasted from the DonutOrder table to the Donut table. The same was done for the Line table by moving the Qty attribute to this table. DonutOrderID and DonutID now make up a composite primary key for the Line table All partial dependencies are now eliminated, putting this into 2nd Normal Form.

c.



i) I designed this 3rd Normal Form Diagram by adding the Customer table and moving the Customer related fields to it to remove the transitive dependency that was occurring inside the DountOrder table . I made the Custyomer_ID a foreign key in the DonutOrder table to reference the Customer_ID Primary key in the newly created Customer table. This eliminates all dependencies, therefore qualifying this as the 3rd Normal Form.

B.



4.a. I chose to use these entities because they all contribute to normalizing the tables to the 3rd normal form, which is the highest normal form that is necessary. In order to eliminate partial depencies after the 1st Normal form was created, the Donut and Line tables were created so that the attributes currently shown in them would no longer be partially dependent on the composite primary key. This put it into 2nd Normal Form. In order to convert to 3rd Normal Form, the Customer table was created, since there were multiple transitive dependencies existing in the DonutOrder table. By creating the Customer table and moving the appropriate attributes affected by the transitive dependencies to it, it is now is 3rd Normal Form.

b. I determined the relationship between these entities by initally reasoning through the process of ordering a donut, as well as using the attached Sales Order Form document. I then reasoned that a Customer would generate a DonutOrder, which would contain a Line or Lines, and those Lines would each contain a Donut(information about the Donut being ordered).

c. The Customer and DonutOrder relationship is a one(optional) – to – many relationship, because ONE Customer can generate MANY DonutOrder's. A customer does not HAVE to generate a DonutOrder, but a DonutOrder MUST be generated IF a Customer places an order. The DonutOrder and Line relationship is one – to- many because ONE DonutOrder can contain MANY lines. In other words, there can be many lines that pertain to one DonutOrder. The Line and Donut relationship is a many – to – one relationship, because there may be MANY lines, but only ONE Donut will appear on any given line.

C.1.

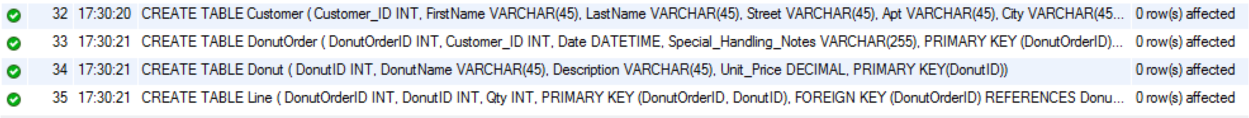
```
CREATE TABLE Customer (

Customer_ID INT UNIQUE, FirstName VARCHAR(45), LastName VARCHAR(45), Street
VARCHAR(45), Apt VARCHAR(45), City VARCHAR(45), State VARCHAR(2), Zip VARCHAR(45),
HomePhone VARCHAR(45), MobilePhone VARCHAR(45), OtherPhone VARCHAR(45),

PRIMARY KEY (Customer_ID));
```

```
CREATE TABLE DonutOrder (

DonutOrderID INT, Customer_ID INT, Date DATETIME, Special_Handling_Notes VARCHAR(255),

PRIMARY KEY (DonutOrderID),

FOREIGN KEY (Customer_ID) REFERENCES Customer(Customer_ID));
```

```
CREATE TABLE Donut (

DonutID INT UNIQUE, DonutName VARCHAR(45), Description VARCHAR(45), Unit_Price
DECIMAL,

PRIMARY KEY(DonutID));
```

```
CREATE TABLE Line (

DonutOrderID INT, DonutID INT, Qty INT,

PRIMARY KEY (DonutOrderID, DonutID),

FOREIGN KEY (DonutOrderID) REFERENCES DonutOrder(DonutOrderID),

FOREIGN KEY (DonutID) REFERENCES Donut(DonutID));
```

a. *Screenshot of results (hopefully this is sufficient)



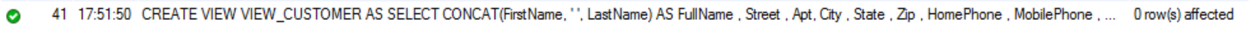
```
32  17:30:20  CREATE TABLE Customer ( Customer_ID INT, FirstName VARCHAR(45), LastName VARCHAR(45), Street VARCHAR(45), Apt VARCHAR(45), City VARCHAR(45...  0 row(s) affected
33  17:30:21  CREATE TABLE DonutOrder ( DonutOrderID INT, Customer_ID INT, Date DATETIME, Special_Handling_Notes VARCHAR(255), PRIMARY KEY (DonutOrderID)...  0 row(s) affected
34  17:30:21  CREATE TABLE Donut ( DonutID INT, DonutName VARCHAR(45), Description VARCHAR(45), Unit_Price DECIMAL, PRIMARY KEY(DonutID))  0 row(s) affected
35  17:30:21  CREATE TABLE Line ( DonutOrderID INT, DonutID INT, Qty INT, PRIMARY KEY (DonutOrderID, DonutID), FOREIGN KEY (DonutOrderID) REFERENCES Donu...  0 row(s) affected
```

D.

1.

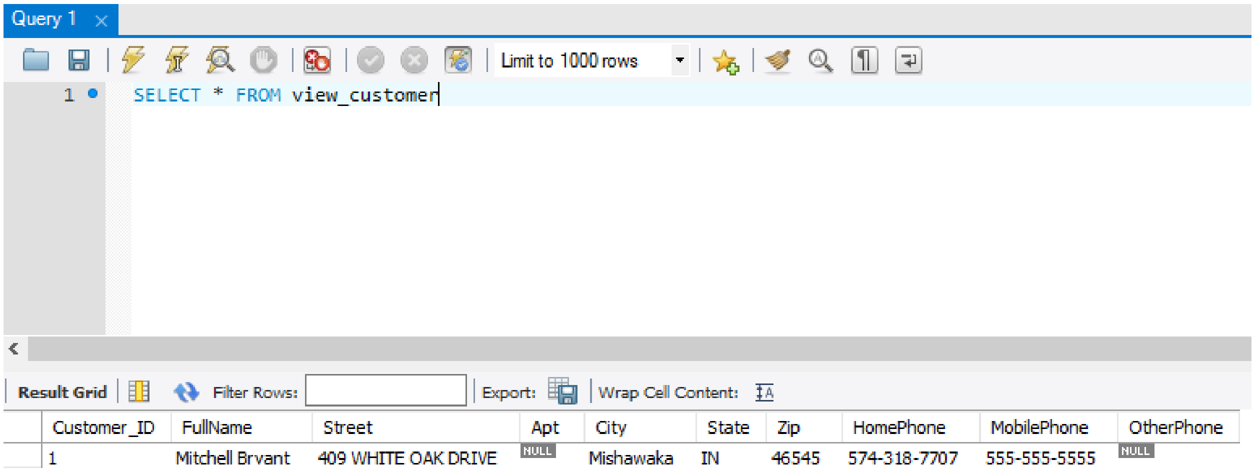
CREATE VIEW VIEW_CUSTOMER AS SELECT Customer_ID, CONCAT(FirstName, ' ', LastName) AS FullName , Street , Apt, City , State , Zip , HomePhone , MobilePhone , OtherPhone

FROM Customer;

a. 

```
41  17:51:50  CREATE VIEW VIEW_CUSTOMER AS SELECT CONCAT(FirstName, ' ', LastName) AS FullName , Street , Apt, City , State , Zip , HomePhone , MobilePhone , ...  0 row(s) affected
```

I also ran ‘SELECT * FROM VIEW_CUSTOMER ‘ to test, and here is the screenshot



```
SELECT * FROM view_customer
```

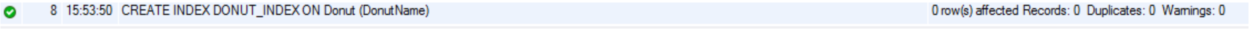
Customer_ID	FullName	Street	Apt	City	State	Zip	HomePhone	MobilePhone	OtherPhone
1	Mitchell Bryant	409 WHITE OAK DRIVE	NULL	Mishawaka	IN	46545	574-318-7707	555-555-5555	NULL

E.

1.

CREATE INDEX DONUT_INDEX

ON Donut (DonutName);

a. 

```
8  15:53:50  CREATE INDEX DONUT_INDEX ON Donut (DonutName)    0 row(s) affected Records: 0  Duplicates: 0  Warnings: 0
```

F.

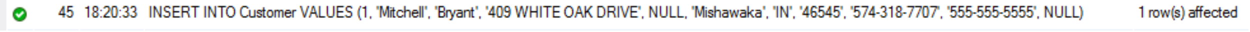
1.

INSERT INTO Customer VALUES (1, 'Mitchell', 'Bryant', '409 WHITE OAK DRIVE', NULL, 'Mishawaka', 'IN', '46545', '574-318-7707', '555-555-5555', NULL);

INSERT INTO Donut VALUES (1, 'Jelly', 'Delicious jelly-filled donut', 2.25);

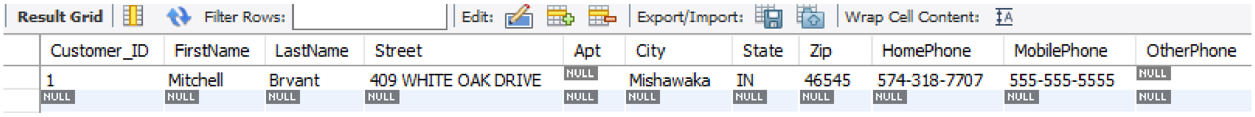
INSERT INTO DonutOrder VALUES(1,1, '2017-07-05', 'handle with care');

INSERT INTO Line VALUES(1,1, 3);

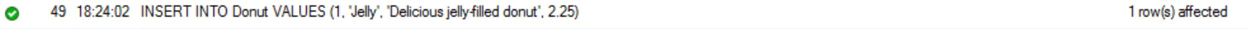
a. 

```
45  18:20:33  INSERT INTO Customer VALUES (1, 'Mitchell', 'Bryant', '409 WHITE OAK DRIVE', NULL, 'Mishawaka', 'IN', '46545', '574-318-7707', '555-555-5555', NULL)    1 row(s) affected
```

*I then ran SELECT * FROM Customer and received the below result

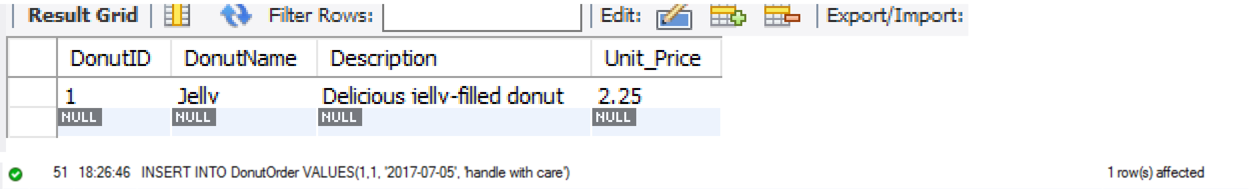


Customer_ID	FirstName	LastName	Street	Apt	City	State	Zip	HomePhone	MobilePhone	OtherPhone
1	Mitchell	Bryant	409 WHITE OAK DRIVE	NULL	Mishawaka	IN	46545	574-318-7707	555-555-5555	NULL
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL



```
49  18:24:02  INSERT INTO Donut VALUES (1, 'Jelly', 'Delicious jelly-filled donut', 2.25)    1 row(s) affected
```

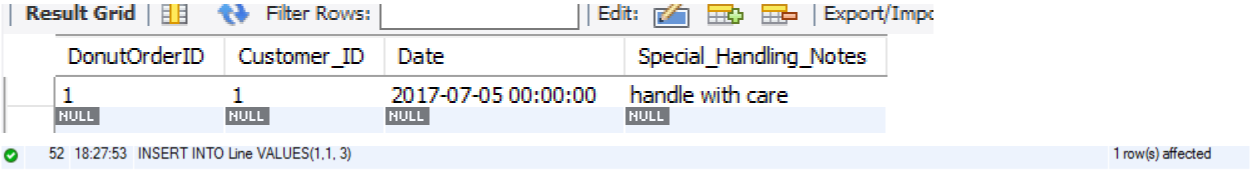
*I then ran SELECT * FROM Donut and received the below result



DonutID	DonutName	Description	Unit_Price
1	Jellv	Delicious iellv-filled donut	2.25
NULL	NULL	NULL	NULL

```
51  18:26:46  INSERT INTO DonutOrder VALUES(1,1, '2017-07-05', 'handle with care')    1 row(s) affected
```

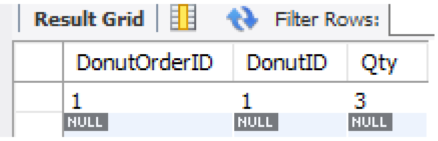
*I then ran SELECT * FROM DonutOrder and received the below result



DonutOrderID	Customer_ID	Date	Special_Handling_Notes
1	1	2017-07-05 00:00:00	handle with care
NULL	NULL	NULL	NULL

```
52  18:27:53  INSERT INTO Line VALUES(1,1, 3)    1 row(s) affected
```

*I then ran SELECT * FROM Line and received the below result

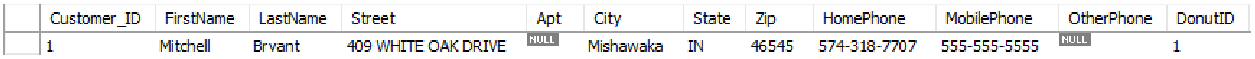


DonutOrderID	DonutID	Qty
1	1	3
NULL	NULL	NULL

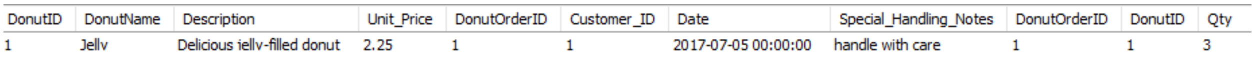
G.

1. SELECT * FROM Customer, Donut, DonutOrder, Line ;

a. The result is one long row so I am forced to cut into two screenshots



Customer_ID	FirstName	LastName	Street	Apt	City	State	Zip	HomePhone	MobilePhone	OtherPhone	DonutID
1	Mitchell	Brvant	409 WHITE OAK DRIVE	NULL	Mishawaka	IN	46545	574-318-7707	555-555-5555	NULL	1



DonutID	DonutName	Description	Unit_Price	DonutOrderID	Customer_ID	Date	Special_Handling_Notes	DonutOrderID	DonutID	Qty
1	Jellv	Delicious iellv-filled donut	2.25	1	1	2017-07-05 00:00:00	handle with care	1	1	3

2.

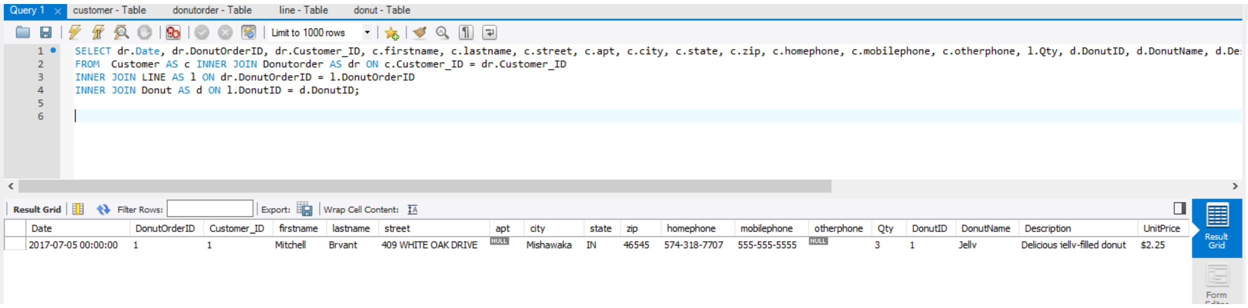
SELECT dr.Date, dr.DonutOrderID, dr.Customer_ID, c.firstname, c.lastname, c.street, c.apt, c.city, c.state, c.zip, c.homephone, c.mobilephone, c.otherphone, l.Qty, d.DonutID, d.DonutName, d.Description, CONCAT('$', d.Unit_Price) AS UnitPrice, dr.Special_Handling_Notes

FROM Customer AS c INNER JOIN Donutorder AS dr ON c.Customer_ID = dr.Customer_ID

INNER JOIN LINE AS l ON dr.DonutOrderID = l.DonutOrderID

INNER JOIN Donut AS d ON l.DonutID = d.DonutID;

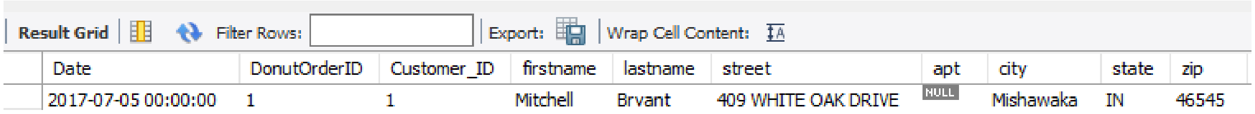
a.



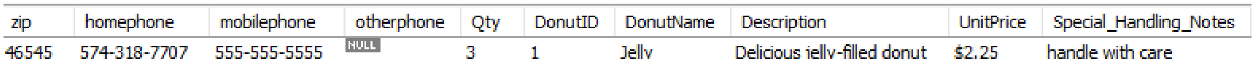
```
SELECT dr.Date, dr.DonutOrderID, dr.Customer_ID, c.firstname, c.lastname, c.street, c.apt, c.city, c.state, c.zip, c.homephone, c.mobilephone, c.otherphone, l.Qty, d.DonutID, d.DonutName, d.De
FROM  Customer AS c INNER JOIN Donutorder AS dr ON c.Customer_ID = dr.Customer_ID
INNER JOIN LINE AS l ON dr.DonutOrderID = l.DonutOrderID
INNER JOIN Donut AS d ON l.DonutID = d.DonutID;
```

Date	DonutOrderID	Customer_ID	firstname	lastname	street	apt	city	state	zip	homephone	mobilephone	otherphone	Qty	DonutID	DonutName	Description	UnitPrice
2017-07-05 00:00:00	1	1	Mitchell	Brvant	409 WHITE OAK DRIVE	NULL	Mishawaka	IN	46545	574-318-7707	555-555-5555	NULL	3	1	Jellv	Delicious iellv-filled donut	$2.25

Since this is screenshot is pretty small I will cut into two images and re-paste below



Date	DonutOrderID	Customer_ID	firstname	lastname	street	apt	city	state	zip
2017-07-05 00:00:00	1	1	Mitchell	Brvant	409 WHITE OAK DRIVE	NULL	Mishawaka	IN	46545



zip	homephone	mobilephone	otherphone	Qty	DonutID	DonutName	Description	UnitPrice	Special_Handling_Notes
46545	574-318-7707	555-555-5555	NULL	3	1	Jellv	Delicious iellv-filled donut	$2.25	handle with care