Edith Cowan University
CSG1207
Systems & Database Design
Assignment 1

Martin Ponce
Student 10371381

Tutor: Greg Baatard

March 24, 2015

Contents

1 Task 1: Normalisation

Figure 1 below shows part of a spreadsheet used by a tavern which allows customers to book rooms for events and functions. Each row represents a booking.

Figure 1: Tavern Bookings

Booking #	Booking Date	Duration	Room #	Room Name	Room Capacity	Customer Phone	Customer Name
1241	12-08-21 18:30	4	3	Side Bar	15	0432514658	Sam Crocker
1242	12-08-21 18:30	4	1	Function Room 1	30	0432514658	Sam Crocker
1243	12-08-23 16:00	8	2	Function Room 2	50	0425748641	Joe Pardy
1244	12-08-24 17:00	5	2	Function Room 2	50	0485475265	Cameron West
1245	12-08-26 15:00	3	1	Function Room 1	30	0428654854	Jimbo Lawkins
1246	12-08-26 19:30	4	1	Function Room 1	30	0438924565	Pattie Forbes
1247	12-08-27 17:30	3	4	Garden Area	25	0425748641	Joe Pardy

1.1 Assumptions

- A room cannot have multiple bookings at the same time

- Auto-incrementing Customer# has been created, replacing CustomerPhone as customer identifier

    - Auto-incrementing identifier avoids user input error which may result in multiple customers with the same phone number
    - Allows CustomerPhone to be updated without having to update foreign keys if CustomerPhone remained as identifier

1.2 0NF: Unnormalised form

R1 = (Customer#, CustomerPhone, CustomerName, Booking#, BookingDate, Duration, Room#, RoomName, RoomCapacity)

1.3 1NF: First normal form

~~R1 = (**Customer#**, CustomerPhone, CustomerName, {**Booking#**, BookingDate, Duration, Room#,RoomName, RoomCapacity})~~

R11 = (**Customer#**, CustomerPhone, CustomerName)

R12 = (**Booking#**, BookingDate, Duration, Room#, RoomName, RoomCapacity, *Customer#*)

1.4 2NF: Second normal form

No partial dependencies, already 2NF.

R11 = (**Customer#**, CustomerPhone, CustomerName)

R12 = (**Booking#**, BookingDate, Duration, Room#, RoomName, RoomCapacity, *Customer#*)

1.5   3NF: Third normal form

R11 = (**Customer#**, CustomerPhone, CustomerName)

~~R12 = (**Booking#**, BookingDate, Duration, Room#, RoomName, RoomCapacity, *Customer#*)~~

R121 = (**Booking#**, BookingDate, Duration, *Room#*, *Customer#*)

R122 = (**Room#**, RoomName, RoomCapacity)

1.6   Named relations

Customer = (**Customer#**, CustomerPhone, CustomerName)

Booking = (**Booking#**, BookingDate, Duration, *Room#*, *Customer#*)

Room = (**Room#**, RoomName, RoomCapacity)

1.7   Physical E-R diagram

2   Task 2: Advanced normalisation

Figure 2 below depicts an invoice for an order from a store.

Figure 2: Pakoko Tax Invoice

Tax Invoice   **Pakoko**   *Tax Invoice*

112 St. Georges Terrace, Perth, WA 6000
Ph: 9325 2458 • ABN: 658475896

Invoice #:   24130
Invoice Date: 23-04-2012
Delivery Address:
         52 Brook Street, Noranda, 6062, WA
Delivery Instructions:
         Knock on side door not front door

Email: p.ford@gmail.com
Name: Patrick Ford
Phone: 0425874569

Item Code	Item Name	Cat. Code	Cat. Name	Cost (each)	Qty	Subtotal
SKU8789	Hunter x Hunter, volume 31	CMGN	Comics & Graphic Novels	$9.99	1	$9.99
SKU6927	Watchmen (Hard Cover)	CMGN	Comics & Graphic Novels	$29.99	1	$29.99
SKU3305	Final Fantasy Master Creatures - Kefka	AFIG	Action Figures	$34.99	1	$34.99
SKU6421	Serenity Movie Poster	PSTR	Posters	$9.80	2	$19.60
SKU3312	Final Fantasy Master Creatures - Ifrit	AFIG	Action Figures	$34.99	1	$34.99
SKU7899	Angry Birds 9" Plushies (Birds)	PLSH	Plush Toys	$35.00	2	$70.00
SKU7898	Angry Birds 9" Plushies (Pigs)	PLSH	Plush Toys	$25.00	1	$25.00
					Grand Total	$214.57

Thank you for shopping with Pakoko! Please see our return policy at www.pakoko.com.au/returns for any missing, incorrect or damaged items.

2.1   Assumptions

- Auto-incrementing Cust# has been created, replacing CustEmail as customer identifier
    - Auto-incrementing identifier avoids user input error which may result in multiple customers with the same email address
    - Allows CustEmail to be updated without having to update foreign keys if CustEmail remained as identifier
- Each item is only in one category
- Item codes are unique per item, even if the items are in different categories
- Invoice header and footer is static and is not stored in the database
    - Includes Pakoko business details header and thank you / return policy URL footer
- Derived attributes are not stored in the database
    - Includes Item Subtotal and Invoice Grand Total

2.2   0NF: Unnormalised form

R1 = (Cust#, CustEmail, CustName, CustPhone, DeliveryAddress, DeliveryInstructions, {Invoice#, InvoiceDate, {ItemCode, ItemName, CatCode, CatName, Cost, Qty}})

2.3   1NF: First normal form

~~R1 = (**Cust#**, CustEmail, CustName, CustPhone, DeliveryAddress, DeliveryInstructions, {**Invoice#**, InvoiceDate, {**ItemCode**, ItemName, CatCode, CatName, Cost, Qty}})~~

R11 = (**Cust#**, CustEmail, CustName, CustPhone, DeliveryAddress, DeliveryInstructions)

R12 = (**Invoice#**, InvoiceDate, *Cust#*)

R13 = (***Invoice#***, **ItemCode**, ItemName, CatCode, CatName, Cost, Qty)

2.4   2NF: Second normal form

R11 = (**Cust#**, CustEmail, CustName, CustPhone, DeliveryAddress, DeliveryInstructions)

R12 = (**Invoice#**, InvoiceDate, *Cust#*)

~~R13 = (***Invoice#***, **ItemCode**, ItemName, CatCode, CatName, Cost, Qty)~~

R131 = (***Invoice#***, ***ItemCode***, Qty)

R132 = (**ItemCode**, ItemName, CatCode, CatName, Cost)

2.5   3NF: Third normal form

R11 = (**Cust#**, CustEmail, CustName, CustPhone, DeliveryAddress, DeliveryInstructions)

R12 = (**Invoice#**, InvoiceDate, *Cust#*)

R131 = (***Invoice#***, ***ItemCode***, Qty)

~~R132 = (**ItemCode**, ItemName, CatCode, CatName, Cost)~~

R1321 = (**ItemCode**, ItemName, *CatCode*)

R1322 = (**CatCode**, CatName)

2.6   Named relations

Customer = (**Cust#**, CustEmail, CustName, CustPhone, DeliveryAddress, DeliveryInstructions)

Invoice = (**Invoice#**, InvoiceDate, *Cust#*)

InvoiceItem = (***Invoice#***, ***ItemCode***, Qty)

Item = (**ItemCode**, ItemName, *CatCode*)

Category = (**CatCode**, CatName)

2.7   Physical E-R diagram
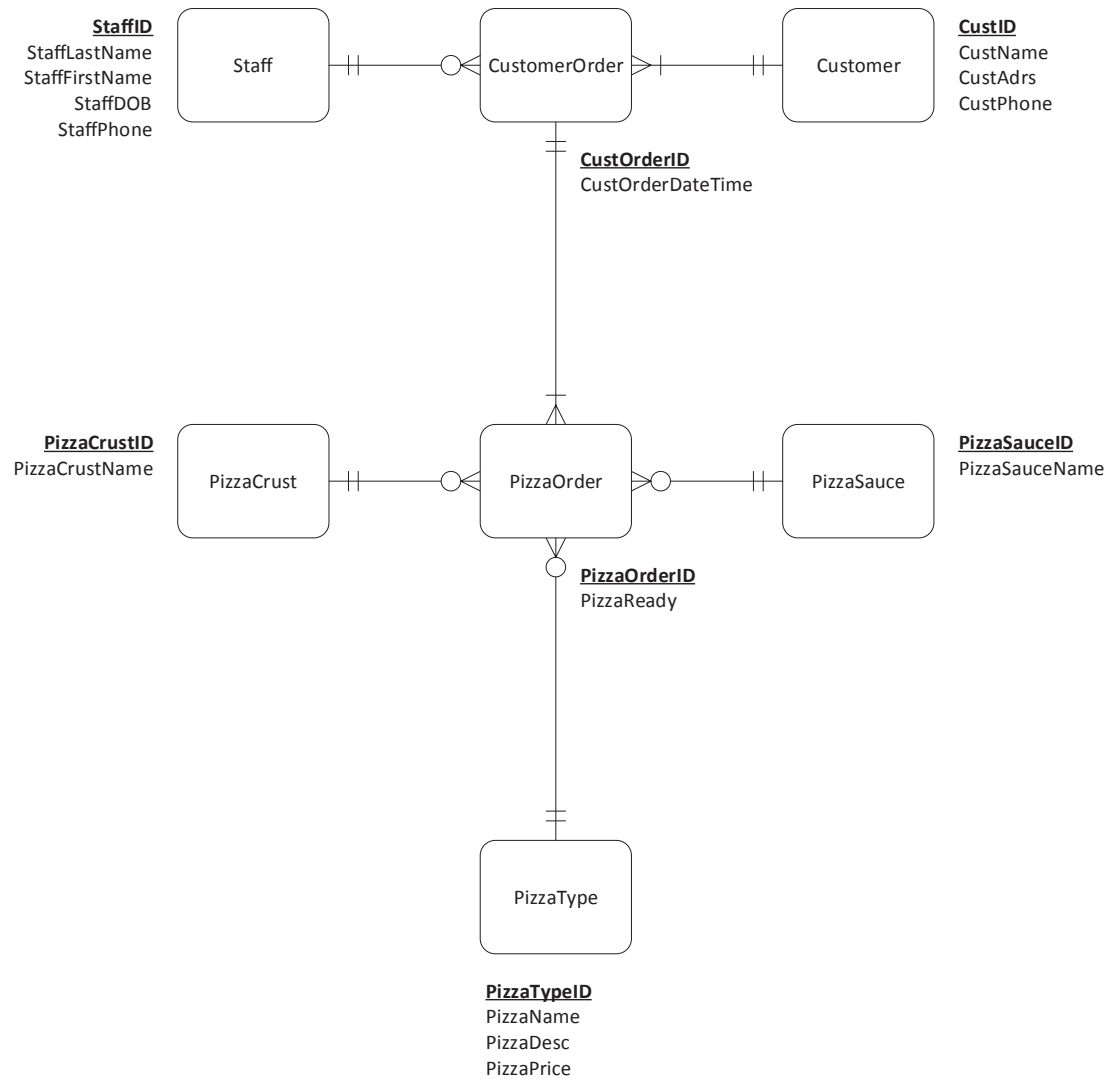
3   Task 3: Entity-Relationship modelling

You have been hired to design a database system for a pizza store. The database must encompass the customers, staff, pizza details and the pizza orders made by customers.

3.1   Assumptions

- A customer must order at least one pizza to exist on database

- Some staff may not take any CustomerOrders

- A CustomerOrder must contain at least one PizzaOrder

- A PizzaOrder must include one PizzaType selection

  - It is possible that a PizzaType may never be selected for a PizzaOrder

- A PizzaOrder must include one PizzaCrust selection

  - It is possible that a PizzaCrust may never be selected for a PizzaOrder

- A PizzaOrder must include one PizzaSauce selection

  - It is possible that a PizzaSauce may never be selected for a PizzaOrder

3.2   Logical E-R diagram

Figure 3: Pizza Store Logical E-R Diagram

StaffID
StaffLastName
StaffFirstName
StaffDOB
StaffPhone

Staff

CustomerOrder

Customer

CustID
CustName
CustAdrs
CustPhone

CustOrderID
CustOrderDateTime

PizzaCrustID
PizzaCrustName

PizzaCrust

PizzaOrder

PizzaSauce

PizzaSauceID
PizzaSauceName

PizzaOrderID
PizzaReady

PizzaType

PizzaTypeID
PizzaName
PizzaDesc
PizzaPrice

3.3   Physical E-R diagram

Figure 4: Pizza Store Physical E-R Diagram

4   Task 4: Advanced Entity-Relationship modelling

The pizza store from Task 3 now requires you to update and expand the previous database you designed for them, in order to record some additional information.

4.1   Assumptions

- A staff member may be a supervisor to many staff members

  - A staff member may only have one supervisor
  - Some staff may not have a supervisor

- A staff member may not deliver any pizzas

- A PizzaType must include one PizzaRange selection

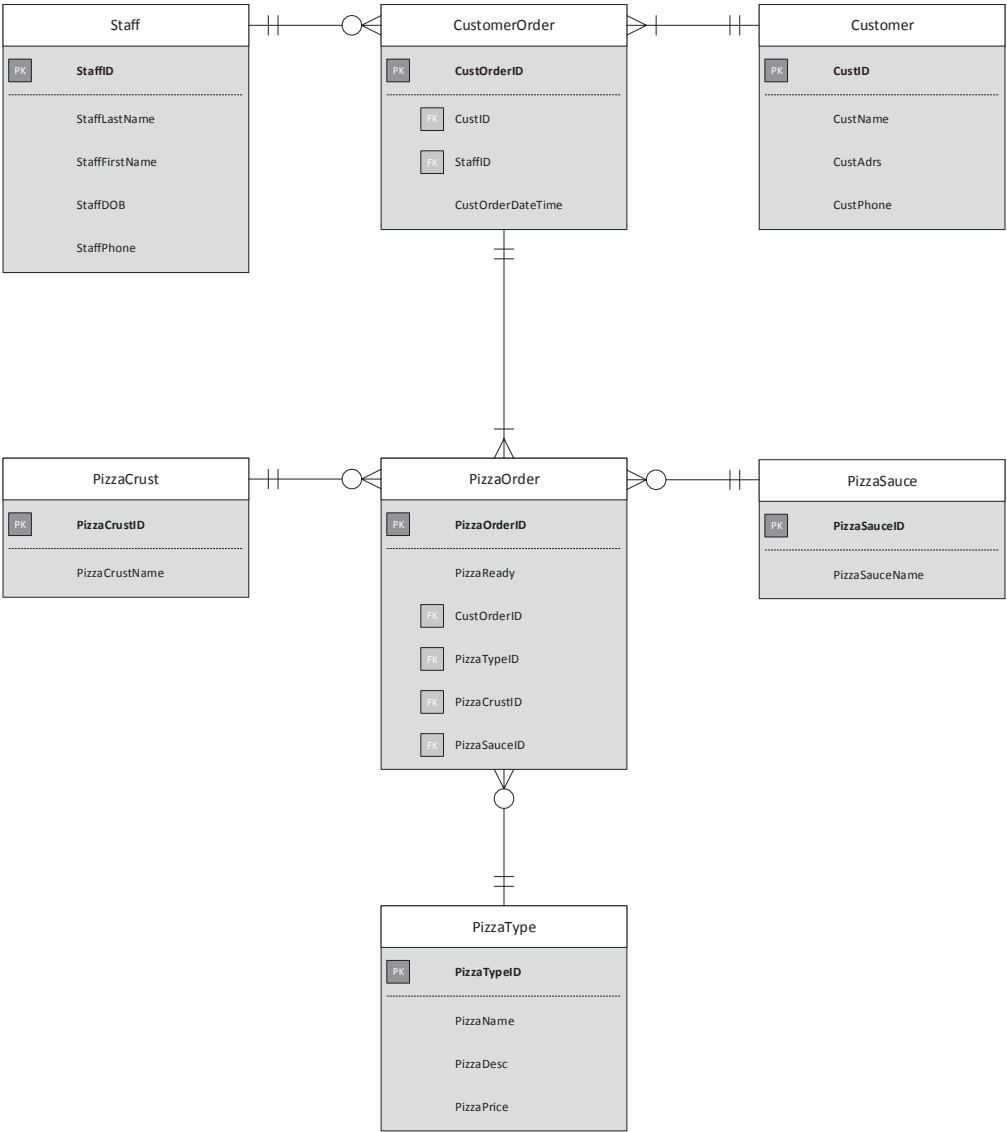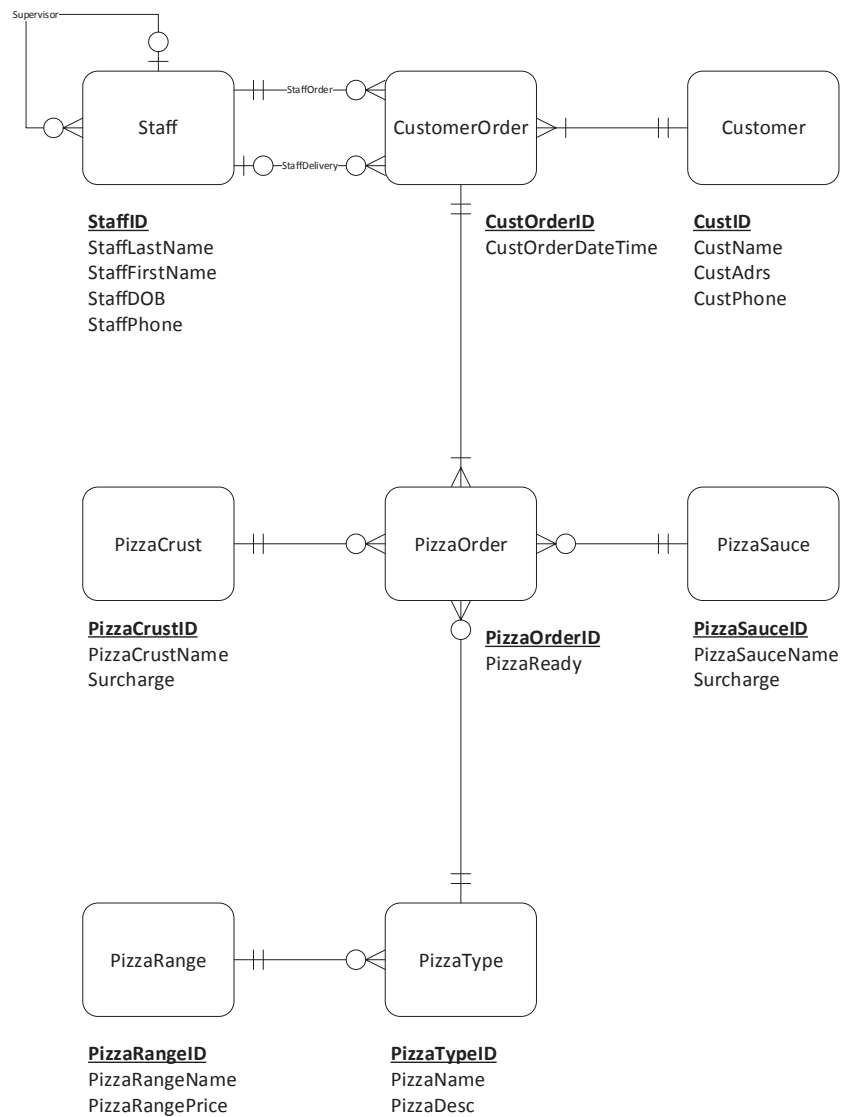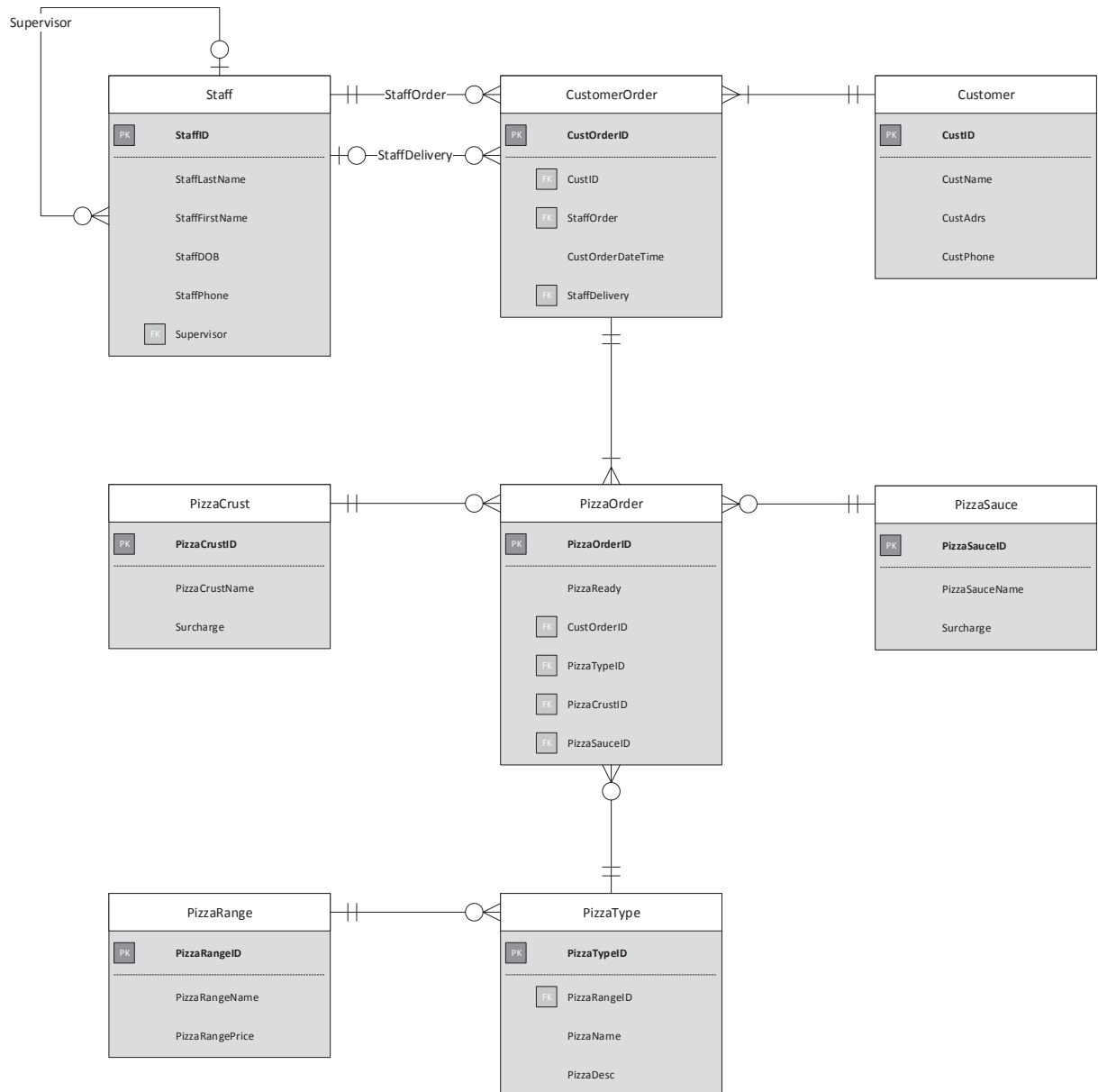 - It is possible that a PizzaRange may never be selected for a PizzaOrder

4.2   Logical E-R diagram

Figure 5: Pizza Store Advanced Logical E-R Diagram

4.3   Physical E-R diagram

Figure 6: Pizza Store Advanced Physical E-R Diagram

4.4   Data Dictionary & Creation Order

Table 1: **"staff"** stores details about staff

Column name	Type/Length	Null	Constraints	Other
staff_id	INT	NOT NULL	PK	IDENTITY
staff_last_name	VARCHAR(20)	NOT NULL		
staff_first_name	VARCHAR(20)	NOT NULL		
staff_dob	DATE	NOT NULL		
staff_phone	VARCHAR(20)	NOT NULL		
supervisor	INT	NULL	FK (staff.staff_id)	

Table 2: **"customer"** stores details about customer

Column name	Type/Length	Null	Constraints	Other
cust_id	INT	NOT NULL	PK	IDENTITY
cust_name	VARCHAR(50)	NOT NULL		
cust_adrs	TEXT	NOT NULL		
cust_phone	VARCHAR(20)	NOT NULL		

Table 3: **"customer_order"** stores details about customer order

Column name	Type/Length	Null	Constraints	Other
cust_order_id	INT	NOT NULL	PK	IDENTITY
cust_id	INT	NOT NULL	FK (customer.cust_id)	
staff_order	INT	NOT NULL	FK (staff.staff_id)	
cust_order_datetime	DATETIME	NOT NULL		
staff_delivery	INT	NULL	FK (staff.staff_id)	

Table 4: **"pizza_crust"** stores details about pizza crust

Column name	Type/Length	Null	Constraints	Other
pizza_crust_id	INT	NOT NULL	PK	IDENTITY
pizza_crust_name	VARCHAR(20)	NOT NULL		
surcharge	MONEY	NOT NULL		

Table 5: **"pizza_sauce"** stores details about pizza sauce

Column name	Type/Length	Null	Constraints	Other
pizza_sauce_id	INT	NOT NULL	PK	IDENTITY
pizza_sauce_name	VARCHAR(20)	NOT NULL		
surcharge	MONEY	NOT NULL		

Table 6: **"pizza_range"** stores details about pizza range

Column name	Type/Length	Null	Constraints	Other
pizza_range_id	INT	NOT NULL	PK	IDENTITY
pizza_range_name	VARCHAR(20)	NOT NULL		
pizza_range_price	MONEY	NOT NULL		

Table 7: **"pizza_type"** stores details about pizza type

Column name	Type/Length	Null	Constraints	Other
pizza_type_id	INT	NOT NULL	PK	IDENTITY
pizza_range_id	INT	NOT NULL	FK (pizza_range.pizza_range_id)	
pizza_name	VARCHAR(20)	NOT NULL		
pizza_desc	TEXT	NOT NULL		

Table 8: **"pizza_order"** stores details about pizza order

Column name	Type/Length	Null	Constraints	Other
pizza_order_id	INT	NOT NULL	PK	IDENTITY
pizza_ready	BIT	NOT NULL		DEFAULT 0
cust_order_id	INT	NOT NULL	FK (customer.cust_id)	
pizza_type_id	INT	NOT NULL	FK (pizza_type.pizza_type_id)	
pizza_crust_id	INT	NOT NULL	FK (pizza_crust.pizza_crust_id)	
pizza_sauce_id	INT	NOT NULL	FK (pizza_sauce.pizza_sauce_id)	