

C170 – DATA MANAGEMENT APPLICATIONS – PERFORMANCE ASSESSMENT

TASK A

1.) 2ND NORMAL FORM

The attributes were divided and assigned to tables by reviewing the provided order form and the names of the tables.

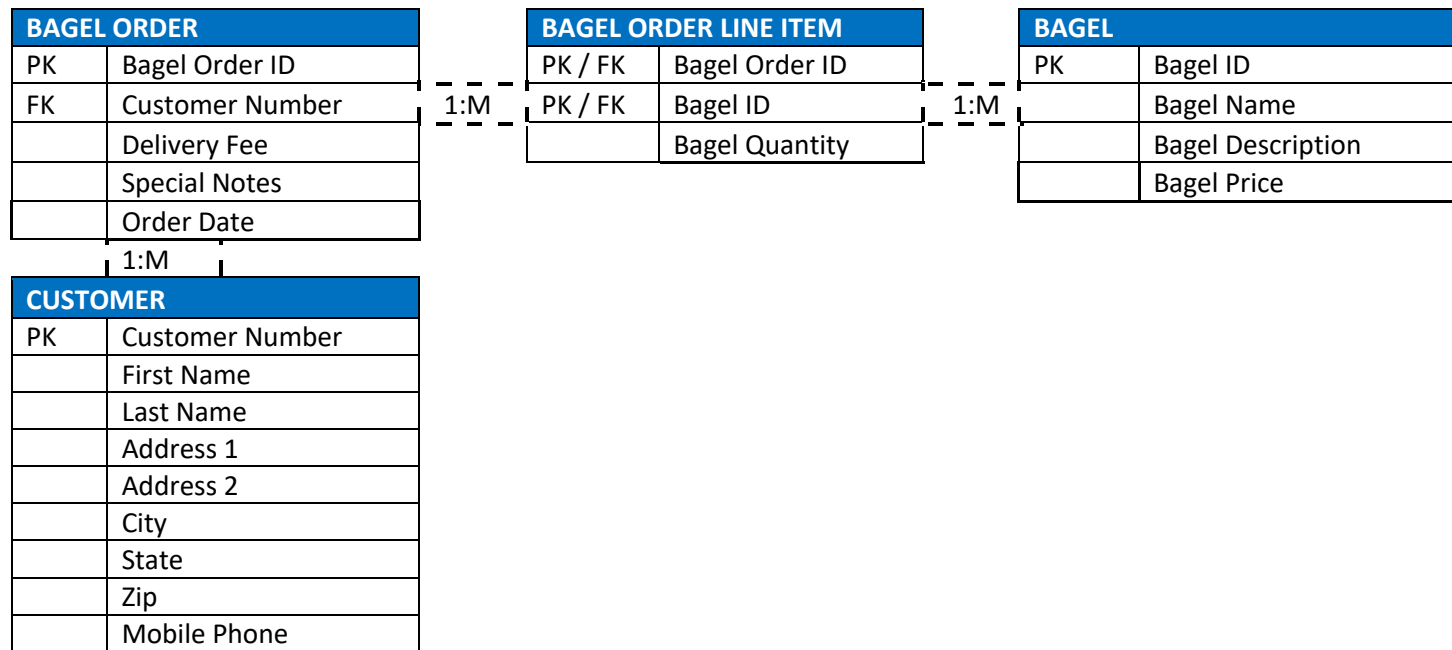
Attributes that pertained specifically to bagels were added to the Bagel table. Bagel quantity was added to the Bagel Order Line Item table because it was the only remaining value that depended on both parts of the original primary key. The attributes in the Bagel Order table were added because they were all functionally dependent on that table's primary key, Bagel Order ID.

One bagel order can have one or many line items. A line item can have one bagel.

BAGEL ORDER			BAGEL ORDER LINE ITEM			BAGEL	
PK	Bagel Order ID		PK / FK	Bagel Order ID		PK	Bagel ID
	Order Date	1:M	PK / FK	Bagel ID	1:M		Bagel Name
	First Name			Bagel Quantity			Bagel Description
	Last Name						Bagel Price
	Address 1						
	Address 2						
	City						
	State						
	Zip						
	Mobile Phone						
	Delivery Fee						
	Special Notes						

2.) 3RD NORMAL FORM

The customer specific attributes in the Bagel Order table have a transitive dependency and were moved from the Bagel Order table to the newly created Customer table. One customer can have one or many bagel orders. One or many bagel orders can have one or many line items. Each line item has one bagel.



3.) Final Physical Database Model

BAGEL ORDER		
PK	bagel_order_ID	INT
FK	customer_number	INT
	delivery_fee	NUMERIC(5,2)
	special_notes	VARCHAR(100)
	order_date	TIMESTAMP

1:M

CUSTOMER		
PK	customer_number	INT
	first_name	VARCHAR(25)
	last_name	VARCHAR(50)
	address_1	VARCHAR(50)
	address_2	VARCHAR(50)
	city	VARCHAR(25)
	state	CHAR(2)
	zip	VARCHAR(10)
	mobile_phone	VARCHAR(15)

BAGEL ORDER LINE ITEM		
PK / FK	bagel_order_id	INT
PK / FK	bagel_id	CHAR(2)
	bagel_quantity	INT

1:M

BAGEL		
PK	bagel_ID	CHAR(2)
	bagel_name	VARCHAR(25)
	bagel_description	VARCHAR(50)
	bagel_price	NUMERIC(5,2)

PART B

- 1.) Develop SQL code to create *each* table as specified in the attached “Jaunty Coffee Co. ERD”

```
29 CREATE TABLE Supplier (
30     supplier_id INTEGER,
31     company_name VARCHAR(50),
32     country VARCHAR(30),
33     sales_contact_name VARCHAR(60),
34     email VARCHAR(50) NOT NULL,
35     PRIMARY KEY (supplier_id)
36 );
```

100% 28:35

Action Output

	Time	Action	Response	Duration / Fetch Time
✓	16 19:55:27	CREATE TABLE Coffee_Shop (shop_id INTEGER, shop_name VARCHAR(50), city V...	0 row(s) affected	0.011 sec
✓	17 20:17:26	CREATE TABLE Coffee (coffee_id INTEGER, shop_id INTEGER, supplier_id INTEGE...	0 row(s) affected	0.011 sec
✓	18 20:18:32	CREATE TABLE Supplier (supplier_id INTEGER, company_name VARCHAR(50), cou...	0 row(s) affected	0.0098 sec

Query Completed

```
18 -- PRIMARY KEY (shop_id)
19 -- );
20 CREATE TABLE Coffee (
21     coffee_id INTEGER,
22     shop_id INTEGER,
23     supplier_id INTEGER,
24     coffee_name VARCHAR(30),
25     price_per_pound DECIMAL(5,2),
26     PRIMARY KEY (coffee_id)
27 );
```

100% 4:27

Action Output

	Time	Action	Response	Duration / Fetch Time
✓	15 19:49:07	CREATE TABLE Employee (employee_id INTEGER, first_name VARCHAR(30), last_...	0 row(s) affected	0.026 sec
✓	16 19:55:27	CREATE TABLE Coffee_Shop (shop_id INTEGER, shop_name VARCHAR(50), city V...	0 row(s) affected	0.011 sec
✓	17 20:17:26	CREATE TABLE Coffee (coffee_id INTEGER, shop_id INTEGER, supplier_id INTEGE...	0 row(s) affected	0.011 sec

Query Completed

```
38 ALTER TABLE Employee
39 ADD FOREIGN KEY (shop_id) REFERENCES Coffee_Shop(shop_id);
```

100% 59:39

Action Output

	Time	Action	Response	Duration / Fetch Time
✓	17 20:17:26	CREATE TABLE Coffee (coffee_id INTEGER, shop_id INTEGER, supplier_id INTEGE...	0 row(s) affected	0.011 sec
✓	18 20:18:32	CREATE TABLE Supplier (supplier_id INTEGER, company_name VARCHAR(50), cou...	0 row(s) affected	0.0098 sec
✓	19 20:19:08	ALTER TABLE Employee ADD FOREIGN KEY (shop_id) REFERENCES Coffee_Shop(sho...	0 row(s) affected Records: 0 Duplicates: 0 Warnings...	0.042 sec

Query Completed

41	ALTER TABLE Coffee		
42	ADD FOREIGN KEY (shop_id) REFERENCES Coffee_Shop(shop_id),		
43	ADD FOREIGN KEY (supplier_id) REFERENCES Supplier(supplier_id);		
100%	64:43		
Action Output			
	Time	Action	Response
✓ 18	20:18:32	CREATE TABLE Supplier (supplier_id INTEGER, company_name VARCHAR(50), cou...	0 row(s) affected
✓ 19	20:19:08	ALTER TABLE Employee ADD FOREIGN KEY (shop_id) REFERENCES Coffee_Shop(sho...	0 row(s) affected Records: 0 Duplicates: 0 Warnings...
✓ 20	20:19:34	ALTER TABLE Coffee ADD FOREIGN KEY (shop_id) REFERENCES Coffee_Shop(shop_id)...	0 row(s) affected Records: 0 Duplicates: 0 Warnings...
Query Completed			

2.) Develop SQL code to populate *each* table in the database design document

1	USE JauntyCoffeeCo;		
2			
3	INSERT INTO Coffee_Shop (shop_id, shop_name, city, state)		
4	VALUES		
5	(60, 'This Town Coffee', 'Las Vegas', 'NV'),		
6	(42, 'Stripes and Stars Coffee', 'Philadelphia', 'PA'),		
7	(38, 'San Fran Coffee', 'Palo Alto', 'CA');		
8			
9	INSERT INTO Employee (employee_id, first_name, last_name, hire_date, job_title, shop_id)		
10	VALUES		
11	(15, 'Frank', 'Sinatra', '2015-01-15', 'Barista', 60),		
12	(3, 'Betsy', 'Ross', '2020-06-03', 'Manager', 42),		
13	(7, 'Lisa', 'Su', '2022-02-27', 'Cashier', 38);		
14			
15	INSERT INTO Supplier (supplier_id, company_name, country, sales_contact_name, email)		
16	VALUES		
17	(1, 'Kimura Coffee Co', 'Brazil', 'Rickson', 'rickson.gracie@kimuracoffeeco.com'),		
18	(2, 'Liberty Coffee Inc', 'United States', 'Ben', 'ben.frank@libertycoffee.com'),		
19	(3, 'Angel Roast', 'United States', 'Angela', 'angela@angelroast.io');		
20			
21	INSERT INTO Coffee (coffee_id, shop_id, supplier_id, coffee_name, price_per_pound)		
22	VALUES		
23	(1, 60, 1, 'Fly With Me Dark Roast', 9.99),		
24	(2, 42, 2, 'Give Me Energy', 8.50),		
25	(3, 38, 3, 'Roasted Silicon', 9.23);		
26			
100%	5:26		
Action Output			
	Time	Action	Response
✓ 25	17:36:23	INSERT INTO Coffee_Shop (shop_id, shop_name, city, state) VALUES (...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
✓ 26	17:37:20	INSERT INTO Employee (employee_id, first_name, last_name, hire_dat...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
✗ 27	17:37:49	INSERT INTO Supplier (supplier_id, company_name, country, sales_con...	Error Code: 1136. Column count doesn't match value count at row 1
✓ 28	17:40:38	INSERT INTO Supplier (supplier_id, company_name, country, sales_con...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
✓ 29	17:40:46	INSERT INTO Coffee (coffee_id, shop_id, supplier_id, coffee_name, pri...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0
Query Completed			

- 3.) Provide the SQL code you wrote to create your view. The view should show *all* of the information from the “Employee” table but concatenate *each* employee’s first and last name, formatted with a space between the first and last name, into a new attribute called employee_full_name.

```
1  -- Create a View
2  CREATE VIEW `Employee_Full` AS
3  SELECT
4  employee_id,
5  CONCAT(first_name, " ", last_name) AS employee_full_name,
6  hire_date,
7  job_title,
8  shop_id
9  FROM Employee;
10
11 SELECT * FROM Employee_Full;
```

100% 29:11

Result Grid Filter Rows: Search Export:

	employee_id	employee_full_na...	hire_date	job_title	shop_id
▶	3	Betsy Ross	2020-06-03	Manager	42
	7	Lisa Su	2022-02-27	Cashier	38
	15	Frank Sinatra	2015-01-15	Barista	60

Employee_Full 1

Action Output

	Time	Action	Response
✓ 1	16:45:22	CREATE VIEW `Employee_Full` AS SELECT employee_id, CONCAT(fir...	0 row(s) affected
✓ 2	16:46:20	SELECT * FROM Employee_Full LIMIT 0, 2000	3 row(s) returned

Query Completed

4.) Develop SQL code to create an index on the coffee_name field from the “Coffee” table.

```
1 CREATE INDEX CoffeeNameIndex
2 ON Coffee (coffee_name);
3
4 SHOW INDEX FROM Coffee;
```

[illegible]

- 5.) Develop SQL code to create an SFW (SELECT-FROM-WHERE) query for *any* of your tables or views.

```
1 SELECT first_name, last_name, hire_date
2 FROM Employee
3 WHERE hire_date > '2019-01-01';
4
```

100% 1:4

Result Grid Filter Rows: Search Export:

	first_name	last_name	hire_date
▶	Betsy	Ross	2020-06-03
	Lisa	Su	2022-02-27

Employee 3

Action Output

	Time	Action	Response
✓ 1	20:08:43	SELECT first_name, last_name, hire_date FROM Employee WHERE hire_date > '2019-01-...	2 row(s) returned

Query Completed

6.) Develop SQL code to create your table joins query. The query should join together **three** different tables and include attributes from *all* three tables in its output.

```
1 SELECT CONCAT(e.first_name, " ", e.last_name) AS Employee_Name, s.shop_name, c.coffee_name
2 FROM Employee e
3     INNER JOIN Coffee_Shop s ON s.shop_id = e.shop_id
4     INNER JOIN Coffee c ON c.shop_id = s.shop_id
5 ORDER BY Employee_Name;
6
```

100% 1:6

Result Grid Filter Rows: Search Export:

Employee_Name	shop_name	coffee_name	
Betsy Ross	Stripes and Stars Coffee	Give Me Energy	
Frank Sinatra	This Town Coffee	Fly With Me Dark Roast	
Lisa Su	San Fran Coffee	Roasted Silicon	

Result 2

Action Output

	Time	Action	Response
✓ 1	20:25:21	SELECT CONCAT(e.first_name, " ", e.last_name) AS Employee_Name, s.shop_name, c.coff...	3 row(s) returned

Query Completed