Cady Wilson
C170 – Database Management -Applications
June 18, 2023

Nora's Bagel Bin

Second Normal Form (2NF)

BAGEL ORDER	
PK	Bagel Order ID
	Order Date
	First Name
	Last Name
	Address 1
	Address 2
	City
	State
	Zip
	Mobile Phone
	Delivery Fee
	Special Notes

1:M

BAGEL ORDER LINE ITEM	
PK / FK	Bagel Order ID
PK / FK	Bagel ID
	Bagel Quantity

M:1

BAGEL	
PK	Bagel ID
	Bagel Name
	Bagel Description
	Bagel Price

The attributes have been assigned based on how an order might progress. The attributes that concern a single bagel type went into the Bagel Table. The Bagel Order table was assigned anything that would be part of a retail order. This includes not only the order itself, but the customer information of the person ordering. This would help in a delivery situation or if clarification was needed.

I assigned the cardinality of Bagel Order to Bagel Order Line item as 1:M, as a single order can contain many different bagels and quantities. I assigned Bagel Order to Bagel as M:1 as there may be several different bagels in an order, but each bagel type and it's attributes are unique.

Third Normal Form (3NF)

Bagel Order	
PK	Bagel Order ID
FK	Customer ID
	Order Date
	Special Notes
	Delivery Fee

M:1

Customer	
PK	Customer ID
	First Name
	Last Name
	Address 1
	Address 2
	City
	State
	Zip
	Mobile Phone

1:M

BAGEL ORDER LINE ITEM	
PK / FK	Bagel Order ID
PK / FK	Bagel ID
	Bagel Quantity

M:1

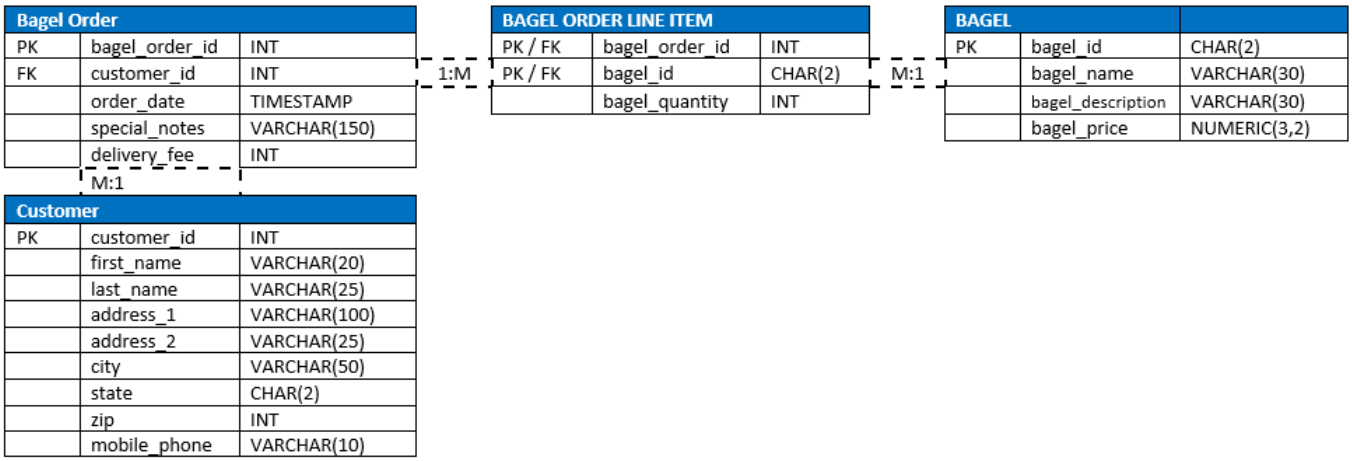
BAGEL	
PK	Bagel ID
	Bagel Name
	Bagel Description
	Bagel Price

To put the table into 3NF I kept the relationships largely the same but separated customer specific data into a new table. This way the customer ID key is the only thing needed when a Bagel Order is placed. I made it a M:1 relationship, many bagels can belong to one order, but only one customer can put in the order.

For the final model I corrected all the attribute names to the correct case and added the variable types for each one.

a. **Bagel Order Table:** Order Date is a TIMESTAMP type as it is the only variable that needs a date associated with it. I made special_notes a VARCHAR() as someone may or may not have any instructions or notes, and if they don't need the whole amount it is saved without the extra space, making retrieval faster. I capped it at 150 because you don't really need more than a sentence or 2 for some notes.
b. **Bagel Order Item:** This is fairly self-explanatory, the number of bagels and the order ID will always be integers and bagel_id according to business rules appears to be capped at 2 characters.
c. **Bagel:** The primary key of bagel_id remained the same, as it should. Bagel_name and bagel_description are VARCHAR(30) for the space saving reasons above, but also because the names have spaces in them. I capped it at 30 because too long of a name or description isn't best business practice. Bagel_price I made NUMERIC(3,2) because all of the prices 3 digits, one for the dollar amount and the second for the cents to the hundredths.
d. **Customer:** As there are varying lengths of names and addresses I made all of those VARCHAR() and capped them at what I determined to be an appropriate length for even the longest of names/addresses. State is CHAR(2) as it's far easier and faster to put the 2 character length state abbreviation. Finally, mobile_phone is VARCHAR(10) for a 10 digit phone number. I didn't make it 11 because the company only operates within the United States, and adding the international digit seemed superfluous.

Final Physical Database Model



Jaunty Coffee

Creating the Tables

```
CREATE TABLE Supplier (
  supplier_id INTEGER PRIMARY KEY,
  company_name VARCHAR(50),
  country VARCHAR(30),
  sales_contact_name VARCHAR(60),
  email  VARCHAR(50)  NOT NULL
  );

CREATE TABLE Coffee_Shop (
  shop_id INTEGER PRIMARY KEY,
  shop_name     VARCHAR(50),
  city VARCHAR(50),
  state   CHAR(2)
);

CREATE TABLE Coffee (
  coffee_id       INTEGER PRIMARY KEY,
  shop_id INTEGER,
  supplier_id INTEGER,
  coffee_name VARCHAR(30),
  price_per_pound NUMERIC(5,2)
);

CREATE TABLE Employee (
  employee_id INTEGER PRIMARY KEY,
  first_name VARCHAR(30),
  last_name       VARCHAR(30),
  hire_date       DATE,
  job_title       VARCHAR(30),
  shop_id INTEGER,
  FOREIGN KEY (shop_id) REFERENCES Coffee_Shop(shop_id)
);
```

Inserting Data into the Tables

INSERT INTO Supplier
(supplier_id, company_name, country, sales_contact_name, email)
VALUES (13566, 'Moonbeam', 'USA', 'Caleb', 'caleb@moonbeam.com'),
(23998, 'Sparky', 'USA', 'Alex', 'alex@sparky.com'),
(47291, 'Pulp-it', 'Canada', 'Collin', 'collin@pulpit.com');

INSERT INTO Coffee_Shop
(shop_id, shop_name, city, state)
VALUES (112, 'Buzzbee', 'Canton', 'OH'),
(003, 'Walkie-Talkie Coffee', 'North Canton', 'OH'),
(096, 'Muggzwigs', 'North Royalton', 'PA');

INSERT INTO Coffee
(coffee_id, shop_id, supplier_id, coffee_name, price_per_pound)
VALUES (122, 003, 47291, 'Suneva', 13.89),
(666, 096, 23998, 'Dark Magic', 12.45),
(946, 096, 13566, 'Odinson', 11.99);

INSERT INTO Employee
(employee_id, first_name, last_name, hire_date, job_title, shop_id)
VALUES (000621, 'Jenny', 'Bierman', '2018-05-03', 'Barista', 112),
(001359, 'Eivor', 'Stjernholm', '2021-09-22', 'Roast Master', 096),
(120610, 'William', 'Watson', '2022-01-30', 'Barista', 112);

```
34 INSERT INTO Supplier
35 (supplier_id, company_name, country, sales_contact_name, email)
36 VALUES (13566, 'Moonbeam', 'USA', 'Caleb', 'caleb@moonbeam.com'),
37 (23998, 'Sparky', 'USA', 'Alex', 'alex@sparky.com'),
38 (47291, 'Pulp-it', 'Canada', 'Collin', 'collin@pulpit.com');
39
40 INSERT INTO Coffee_Shop
41 (shop_id, shop_name, city, state)
42 VALUES (112, 'Buzzbee', 'Canton', 'OH'),
43 (003, 'Walkie-Talkie Coffee', 'North Canton', 'OH'),
44 (096, 'Muggzwigs', 'North Royalton', 'PA');
45
46 INSERT INTO Coffee
47 (coffee_id, shop_id, supplier_id, coffee_name, price_per_pound)
48 VALUES (122, 003, 47291, 'Suneva', 13.89),
49 (666, 096, 23998, 'Dark Magic', 12.45),
50 (946, 096, 13566, 'Odinson', 11.99);
51
52 INSERT INTO Employee
53 (employee_id, first_name, last_name, hire_date, job_title, shop_id)
54 VALUES (000621, 'Jenny', 'Bierman', '2018-05-03', 'Barista', 112),
55 (001359, 'Eivor', 'Stjernholm', '2021-09-22', 'Roast Master', 096),
56 (120610, 'William', 'Watson', '2022-01-30', 'Barista', 112);
57
```

Build Schema | Edit Fullscreen | Browser | [;]

```
1 SELECT *
2 FROM Employee
```

Run SQL | Edit Fullscreen | [;]

employee_id	first_name	last_name	hire_date
621	Jenny	Bierman	2018-05-03
1359	Eivor	Stjernholm	2021-09-22
120610	William	Watson	2022-01-30

Creating a View

SELECT*,
CONCAT(first_name, ' ', last_name) AS employee_full_name
FROM Employee;

```
INSERT INTO Supplier
(supplier_id, company_name, country, sales_contact_name, email)
VALUES (13566, 'Moonbeam', 'USA', 'Caleb', 'caleb@moonbeam.com'),
(23998, 'Sparky', 'USA', 'Alex', 'alex@sparky.com'),
(47291, 'Pulp-it', 'Canada', 'Collin', 'collin@pulpit.com');

INSERT INTO Coffee_Shop
(shop_id, shop_name, city, state)
VALUES (112, 'Buzzbee', 'Canton', 'OH'),
(003, 'Walkie-Talkie Coffee', 'North Canton', 'OH'),
(096, 'Muggzwigs', 'North Royalton', 'PA');

INSERT INTO Coffee
(coffee_id, shop_id, supplier_id, coffee_name, price_per_pound)
VALUES (122, 003, 47291, 'Suneva', 13.89),
(666, 096, 23998, 'Dark Magic', 12.45),
(946, 096, 13566, 'Odinson', 11.99);

INSERT INTO Employee
(employee_id, first_name, last_name, hire_date, job_title, shop_id)
VALUES (000621, 'Jenny', 'Bierman', '2018-05-03', 'Barista', 112),
(001359, 'Eivor', 'Stjernholm', '2021-09-22', 'Roast Master', 096),
(120610, 'William', 'Watson', '2022-01-30', 'Barista', 112);
```

Build Schema | Edit Fullscreen | Browser | [;]

```
SELECT*,
CONCAT(first_name, ' ', last_name) AS employee_full_name
FROM Employee;
```

Run SQL | Edit Fullscreen | [;]

employee_id	first_name	last_name	hire_date	job_title	shop_id	employee_full_name
621	Jenny	Bierman	2018-05-03	Barista	112	Jenny Bierman
1359	Eivor	Stjernholm	2021-09-22	Roast Master	96	Eivor Stjernholm
120610	William	Watson	2022-01-30	Barista	112	William Watson

Record Count: 3; Execution Time: 9ms | View Execution Plan | link

B.4 Indexing Coffee Name

CREATE INDEX coffee_name ON Coffee (coffee_name);

```
(supplier_id, company_name, country, sales_contact_name, email)
VALUES (13566, 'Moonbeam', 'USA', 'Caleb', 'caleb@moonbeam.com'),
(23998, 'Sparky', 'USA', 'Alex', 'alex@sparky.com'),
(47291, 'Pulp-it', 'Canada', 'Collin', 'collin@pulpit.com');

INSERT INTO Coffee_Shop
(shop_id, shop_name, city, state)
VALUES (112, 'Buzzbee', 'Canton', 'OH'),
(003, 'Walkie-Talkie Coffee', 'North Canton', 'OH'),
(096, 'Muggzwigs', 'North Royalton', 'PA');

INSERT INTO Coffee
(coffee_id, shop_id, supplier_id, coffee_name, price_per_pound)
VALUES (122, 003, 47291, 'Suneva', 13.89),
(666, 096, 23998, 'Dark Magic', 12.45),
(946, 096, 13566, 'Odinson', 11.99);

INSERT INTO Employee
(employee_id, first_name, last_name, hire_date, job_title, shop_id)
VALUES (000621, 'Jenny', 'Bierman', '2018-05-03', 'Barista', 112),
(001359, 'Eivor', 'Stjernholm', '2021-09-22', 'Roast Master', 096),
(120610, 'William', 'Watson', '2022-01-30', 'Barista', 112);

CREATE INDEX coffee_name ON Coffee (coffee_name);
```

Build Schema | Edit Fullscreen | Browser | [;]

```
SELECT*
FROM Coffee
```

Run SQL | Edit Fullscreen | [;]

coffee_id	shop_id	supplier_id	coffee_name	price_per_pound
122	3	47291	Suneva	13.89
666	96	23998	Dark Magic	12.45
946	96	13566	Odinson	11.99

Record Count: 3; Execution Time: 20ms | View Execution Plan | link

```sql
CREATE TABLE Supplier (
  supplier_id INTEGER PRIMARY KEY,
  company_name VARCHAR(50),
  country VARCHAR(30),
  sales_contact_name VARCHAR(60),
  email VARCHAR(50) NOT NULL
  );

CREATE TABLE Coffee_Shop (
  shop_id INTEGER PRIMARY KEY,
  shop_name VARCHAR(50),
  city VARCHAR(50),
  state CHAR(2)
);

CREATE TABLE Coffee (
  coffee_id INTEGER PRIMARY KEY,
  shop_id INTEGER,
  supplier_id INTEGER,
  coffee_name VARCHAR(30),
```

Build Schema | Edit Fullscreen | Browser | [;]

```sql
Select *
From Coffee
WHERE shop_id = 096
```

Run SQL | Edit Fullscreen | [;]

coffee_id	shop_id	supplier_id	coffee_name	price_per_pound
666	96	23998	Dark Magic	12.45
946	96	13566	Odinson	11.99

Record Count: 2; Execution Time: 5ms | View Execution Plan | link

Creating a Query

SELECT Employee.first_name, Employee.last_name, Coffee_Shop.shop_name, Coffee.coffee_name
FROM Employee
JOIN Coffee_Shop ON Employee.shop_id = Coffee_Shop.shop_id
JOIN Coffee ON Employee.shop_id = Coffee.shop_id;

```
(supplier_id, company_name, country, sales_contact_name, email)
VALUES (13566, 'Moonbeam', 'USA', 'Caleb', 'caleb@moonbeam.com'),
(23998, 'Sparky', 'USA', 'Alex', 'alex@sparky.com'),
(47291, 'Pulp-it', 'Canada', 'Collin', 'collin@pulpit.com');

INSERT INTO Coffee_Shop
(shop_id, shop_name, city, state)
VALUES (112, 'Buzzbee', 'Canton', 'OH'),
(003, 'Walkie-Talkie Coffee', 'North Canton', 'OH'),
(096, 'Muggzwigs', 'North Royalton', 'PA');

INSERT INTO Coffee
(coffee_id, shop_id, supplier_id, coffee_name, price_per_pound)
VALUES (122, 003, 47291, 'Suneva', 13.89),
(666, 096, 23998, 'Dark Magic', 12.45),
(946, 096, 13566, 'Odinson', 11.99);

INSERT INTO Employee
(employee_id, first_name, last_name, hire_date, job_title, shop_id)
VALUES (000621, 'Jenny', 'Bierman', '2018-05-03', 'Barista', 112),
(001359, 'Eivor', 'Stjernholm', '2021-09-22', 'Roast Master', 096),
(120610, 'William', 'Watson', '2022-01-30', 'Barista', 112);

CREATE INDEX coffee_name ON Coffee (coffee_name);
```

Build Schema | Edit Fullscreen | Browser | [,]

```
SELECT Employee.first_name, Employee.last_name, Coffee_Shop.shop_name, Coffee.coffee_
FROM Employee
JOIN Coffee_Shop ON Employee.shop_id = Coffee_Shop.shop_id
JOIN Coffee ON Employee.shop_id = Coffee.shop_id;
```

Run SQL ▶ | Edit Fullscreen | [,]

first_name	last_name	shop_name	coffee_name
Eivor	Stjernholm	Muggzwigs	Dark Magic
Eivor	Stjernholm	Muggzwigs	Odinson

Record Count: 2; Execution Time: 11ms View Execution Plan link