Data Management – Application C170
VHT2 Task 1: NORMALIZATION AND DATABASE DESIGN
Albreian Joseph

Nora's Bagel Bin Database Blueprints

1. Nora's Bagel Bin: 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

To transition from 1NF to 2NF three tables were created. The tables above are "Bagel Order", "Bagel Order Line Item", and "Bagel". The three tables were created because 2NF does not have any non-attribute keys that is functionally dependent on any candidate key. "Bagel Order" and "Bagel" tables has Primary keys while the "Bagel Order Line Item" has both primary keys that intersects both tables.

The relationship between "Bagel Order" and "Bagel Order Line Item" from left to right is one-to-many. A "Bagel Order "has many orders line items. From right to left, "Bagel Order Line Item" has one bagel order. The "Bagel Order Line Item" has many-to-one relationship with "Bagel". From left to right, "Bagel Order Line Item" has one "Bagel". From right to left, "Bagel" has many "Bagel Order Line Items".

Data Management – Application C170
VHT2 Task 1: NORMALIZATION AND DATABASE DESIGN
Albreian Joseph

2. Third normal form (3NF) "Nora's Bagel Bin Database Blueprints":

BAGEL ORDER

PK	Bagel Order ID
FK	Customer ID
	Order Date
	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

M:1

Customer

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

For a table to be considered in 3NF all transitive dependencies must be removed. The "Customer" table was added to remove the transitive dependencies; making it a foreign key in the "Bagel Order" table.

The table above shows from up-to-down that the "Bagel Order" table has a many-to-one relationship with the "customer" table; "Bagel Order" has one customer; customer has many "Bagel Orders". The "Bagel order" table has a one-to-many relationship with the "Bagel Order Line Item" table; from left-to-right, the "bagel order" has many order line items; from right-to-left, the "Bagel order line items" has one bagel. The "Bagel Order Line Item" has a many-to-one relationship with the "Bagel" table; from left-to-right, "Bagel Order Line Item" has one bagel; read from right-to-left "Bagel" has many "Bagel Order Line Items".

Data Management – Application C170
VHT2 Task 1: NORMALIZATION AND DATABASE DESIGN
Albreian Joseph

3. "Final Physical Database Model": "Nora's Bagel Bin Database Blueprints":

BAGEL ORDER

PK	bagel_order_Id	INT
FK	customer_id	INT
	order_date	TIMESTAMP
	delivery_fee	NUMERIC
	special_notes	VARCHAR (20)

1:M

BAGEL ORDER LINE ITEM

PK/FK	bagel_order_id	INT
PK/FK	bagel_id	CHAR (2)
	bagel_quantily	INT

M:1

BAGEL

PK	bagel_id	CHAR (2)
	bagel_name	VARCHAR (15)
	bagel_description	VARCHAR (100)
	bagel_price	NUMERIC

M:1

CUSTOMER

PK	Customer_id	INT
	first_name	VARCHAR(20)
	last_name	VARCHAR(20)
	address 1	VARCHAR(30)
	address 2	VARCHAR(30)
	city	VARCHAR(30)
	state	CHAR(2)
	zip	VARCHAR(16)
	mobile_phone	VARCHAR(10)

Data Management – Application C170
VHT2 Task 1: NORMALIZATION AND DATABASE DESIGN
Albreian Joseph

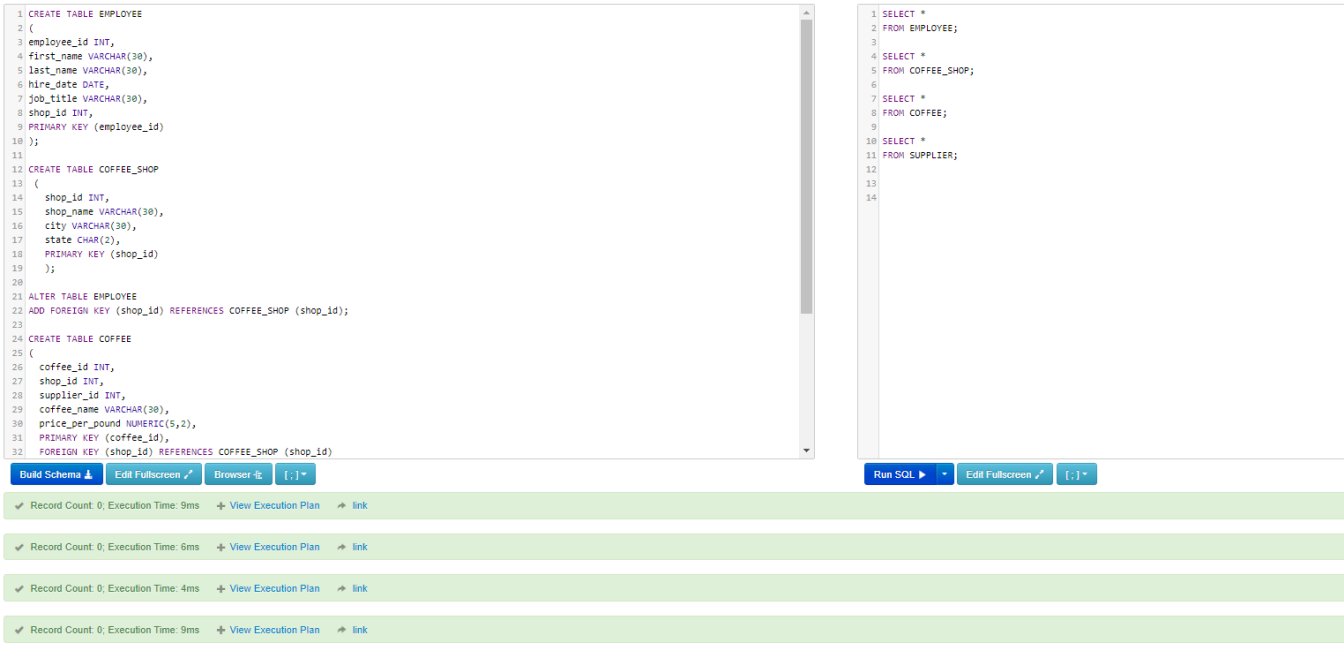
Jaunty Coffee Co. ERD

1. **SQL code "Jaunty Coffee Co. ERD":**

```
CREATE TABLE EMPLOYEE
(
employee_id INT NOT NULL,
first_name VARCHAR(30),
last_name VARCHAR(30),
hire_date DATE,
job_title VARCHAR(30),
shop_id INT,
PRIMARY KEY (employee_id),
);

CREATE TABLE COFFEE_SHOP
 (
  shop_id INT NOT NULL,
  shop_name VARCHAR(30),
  city VARCHAR(30),
  state CHAR(2),
  PRIMARY KEY (shop_id)
  );

CREATE TABLE COFFEE
(
 coffee_id INT NOT NULL,
 shop_id INT,
 supplier_id INT,
 coffee_name VARCHAR(30),
 price_per_pound NUMERIC(5,2),
 PRIMARY KEY (coffee_id),
 FOREIGN KEY (shop_id) REFERENCES COFFEE_SHOP (shop_id)
 );
```

Data Management – Application C170
VHT2 Task 1: NORMALIZATION AND DATABASE DESIGN
Albreian Joseph

```
CREATE TABLE SUPPLIER

(
  supplier_id INT NOT NULL,
  company_name VARCHAR(50),
  country VARCHAR(30),
  sales_contact_name VARCHAR(60),
  email VARCHAR(50) NOT NULL,
  PRIMARY KEY (supplier_id),
  );

  ALTER TABLE COFFEE
  ADD FOREIGN KEY (supplier_id) REFERENCES SUPPLIER (supplier_id);
```



Data Management – Application C170
VHT2 Task 1: NORMALIZATION AND DATABASE DESIGN
Albreian Joseph

2. SQL code to populate *each* table:

```
INSERT INTO COFFEE_SHOP (shop_id, shop_name, city, state) VALUES
 (10 ,'NOLA CAFE', 'NEW ORLEANS', 'LA'),
 (20, 'FRENCH CAFE', 'HARVEY', 'LA'),
 (30, 'CANAL CAFE', 'METAIRIE', 'LA');


INSERT INTO EMPLOYEE (employee_id, first_name, last_name, hire_date, job_title,
shop_id) VALUES
 (100, 'EDWARD', 'GORDON', '2023-01-12', 'waiter', 50),
 (200, 'BRANDON', 'DAVIS', '2023-02-15', 'waiter', 100),
 (300, 'CHRIS', 'HORTON', '2023-03-10','cook', 150);


INSERT INTO SUPPLIER (supplier_id, company_name, country,
sales_contact_name,email) VALUES
 (40, 'HALLOW BEAN', 'USA', 'TOMMY BROWN', 'tbrown@company.com'),
 (50, 'SPOOKY BUCKS', 'CANADA', 'JILL DAVIS', 'jdavis@company.com'),
 (60, 'COCO BEAN', 'PUERTO RICO', 'JOSE SAN', 'jsan@company.com');


INSERT INTO COFFEE (coffee_id, shop_id, supplier_id, coffee_name, price_per_pound)
VALUES
 (5,10, 60, 'HOT BEAN', 1.50),
 (10, 30, 40, 'SPOOKE', 2.00),
 (15, 20, 50, 'MIDNIGHT', 2.50);
```

Data Management – Application C170
VHT2 Task 1: NORMALIZATION AND DATABASE DESIGN
Albreian Joseph

Build Schema | Edit Fullscreen | Browser | Run SQL | Edit Fullscreen

employee_id	first_name	last_name	hire_date	job_title	shop_id
100	EDWARD	GORDON	2023-01-12	waiter	50
200	BRANDON	DAVIS	2023-02-15	waiter	100
300	CHRIS	HORTON	2023-03-10	cook	150

Record Count: 3; Execution Time: 6ms | View Execution Plan | link

shop_id	shop_name	city	state
10	NOLA CAFE	NEW ORLEANS	LA
20	FRENCH CAFE	HARVEY	LA
30	CANAL CAFE	METAIRIE	LA

Record Count: 3; Execution Time: 23ms | View Execution Plan | link

coffee_id	shop_id	supplier_id	coffee_name	price_per_pound
5	10	60	HOT BEAN	1.5
10	30	40	SPOOKE	2
15	20	50	MIDNIGHT	2.5

Record Count: 3; Execution Time: 2ms | View Execution Plan | link

supplier_id	company_name	country	sales_contact_name	email
40	HALLOW BEAN	USA	TOMMY BROWN	tbrown@company.com
50	SPOOKY BUCKS	CANADA	JILL DAVIS	jdavis@company.com
60	COCO BEAN	PUERTO RICO	JOSE SAN	jsan@company.com

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

Data Management – Application C170
VHT2 Task 1: NORMALIZATION AND DATABASE DESIGN
Albreian Joseph

3. Create a view:

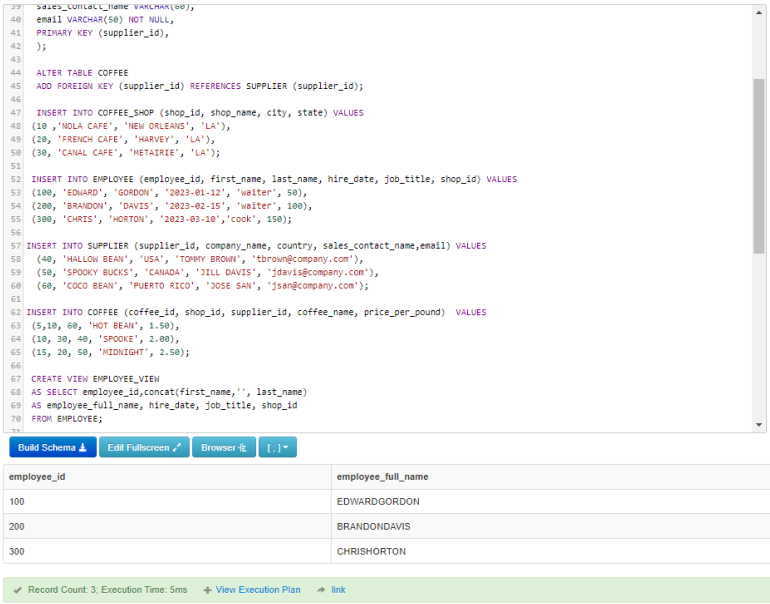
CREATE VIEW EMPLOYEE_VIEW

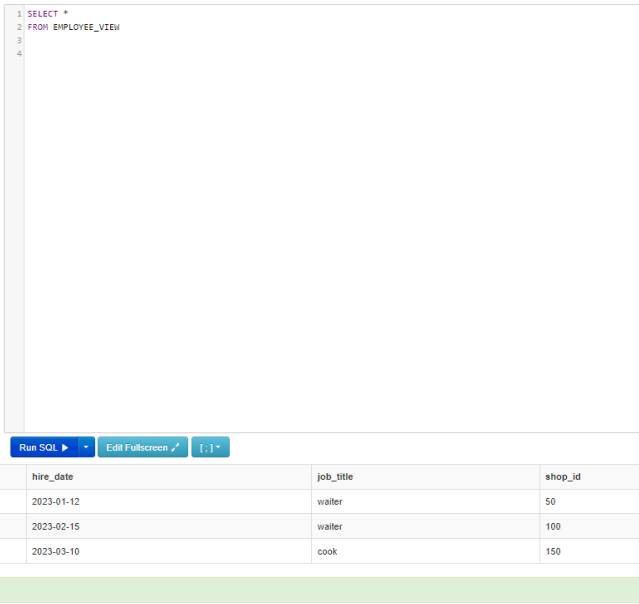
AS SELECT employee_id,

CONCAT (first_name,' ',last_name)

AS employee_full_name, hire_date, job_title, shop_id

FROM EMPLOYEE;





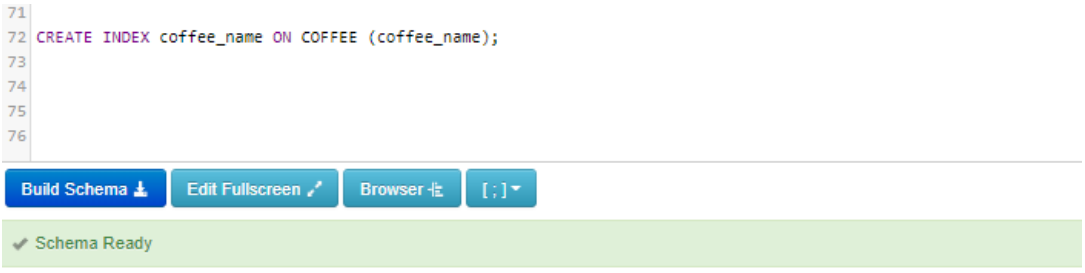
employee_id	employee_full_name	hire_date	job_title	shop_id
100	EDWARDGORDON	2023-01-12	waiter	50
200	BRANDONDAVIS	2023-02-15	waiter	100
300	CHRISHORTON	2023-03-10	cook	150

Record Count: 3, Execution Time: 5ms View Execution Plan link

4. Create an index:

CREATE INDEX coffee_name

ON COFFEE (coffee_name);



Data Management – Application C170
VHT2 Task 1: NORMALIZATION AND DATABASE DESIGN
Albreian Joseph

5. Create an SFW (SELECT–FROM–WHERE):

SELECT supplier_id

FROM SUPPLIER

WHERE suppier_id > = 40;

```
ADD FOREIGN KEY (supplier_id) REFERENCES SUPPLIER (supplier_id);

 INSERT INTO COFFEE_SHOP (shop_id, shop_name, city, state) VALUES
(10 ,'NOLA CAFE', 'NEW ORLEANS', 'LA'),
(20, 'FRENCH CAFE', 'HARVEY', 'LA'),
(30, 'CANAL CAFE', 'METAIRIE', 'LA');

INSERT INTO EMPLOYEE (employee_id, first_name, last_name, hire_date, job_title, shop_id) VALUES
(100, 'EDWARD', 'GORDON', '2023-01-12', 'waiter', 50),
(200, 'BRANDON', 'DAVIS', '2023-02-15', 'waiter', 100),
(300, 'CHRIS', 'HORTON', '2023-03-10','cook', 150);

INSERT INTO SUPPLIER (supplier_id, company_name, country, sales_contact_name,email) VALUES
  (40, 'HALLOW BEAN', 'USA', 'TOMMY BROWN', 'tbrown@company.com'),
  (50, 'SPOOKY BUCKS', 'CANADA', 'JILL DAVIS', 'jdavis@company.com'),
  (60, 'COCO BEAN', 'PUERTO RICO', 'JOSE SAN', 'jsan@company.com');

INSERT INTO COFFEE (coffee_id, shop_id, supplier_id, coffee_name, price_per_pound)  VALUES
 (5,10, 60, 'HOT BEAN', 1.50),
 (10, 30, 40, 'SPOOKE', 2.00),
 (15, 20, 50, 'MIDNIGHT', 2.50);

 CREATE VIEW EMPLOYEE_VIEW
 AS SELECT employee_id,concat(first_name,'', last_name)
 AS employee_full_name, hire_date, job_title, shop_id
 FROM EMPLOYEE;

CREATE INDEX coffee_index ON COFFEE (coffee_name);
```

Build Schema | Edit Fullscreen | Browser | [;]

```
SELECT supplier_id
FROM SUPPLIER
WHERE supplier_id > =40;
```

Run SQL | Edit Fullscreen | [;]

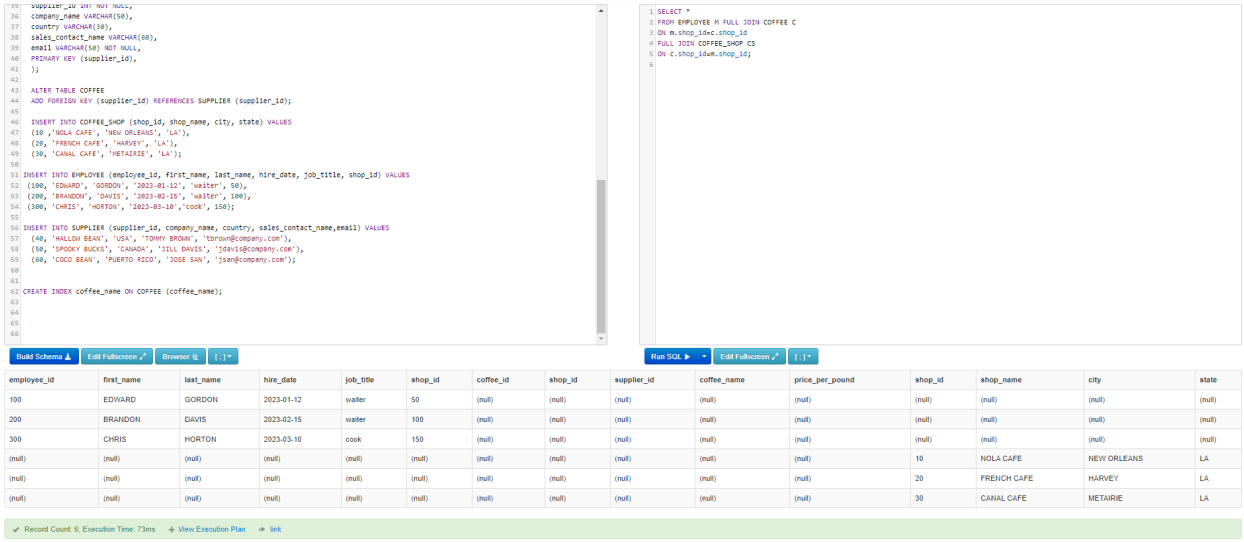
supplier_id
40
50
60

Record Count: 3; Execution Time: 41ms + View Execution Plan link

Data Management – Application C170
VHT2 Task 1: NORMALIZATION AND DATABASE DESIGN
Albreian Joseph

6. Create table joins query:

```
SELECT *
FROM EMPLOYEE M FULL JOIN COFFEE C
ON m.shop_id=c.shop_id
FULL JOIN COFFEE_SHOP CS
ON c.shop_id=m.shop_id;
```



```
  supplier_id INT NOT NULL,
  company_name VARCHAR(50),
  country VARCHAR(30),
  sales_contact_name VARCHAR(60),
  email VARCHAR(50) NOT NULL,
  PRIMARY KEY (supplier_id),
  );

  ALTER TABLE COFFEE
  ADD FOREIGN KEY (supplier_id) REFERENCES SUPPLIER (supplier_id);

  INSERT INTO COFFEE_SHOP (shop_id, shop_name, city, state) VALUES
  (10 ,'NOLA CAFE', 'NEW ORLEANS', 'LA'),
  (20, 'FRENCH CAFE', 'HARVEY', 'LA'),
  (30, 'CANAL CAFE', 'METAIRIE', 'LA');

INSERT INTO EMPLOYEE (employee_id, first_name, last_name, hire_date, job_title, shop_id) VALUES
  (100, 'EDWARD', 'GORDON', '2023-01-12', 'waiter', 50),
  (200, 'BRANDON', 'DAVIS', '2023-02-15', 'waiter', 100),
  (300, 'CHRIS', 'HORTON','2023-03-10','cook', 150);

INSERT INTO SUPPLIER (supplier_id, company_name, country, sales_contact_name,email) VALUES
  (40, 'HALLOW BEAN', 'USA', 'TOMMY BROWN', 'tbrown@company.com'),
  (50, 'SPOOKY BUCKS', 'CANADA', 'JILL DAVIS', 'jdavis@company.com'),
  (60, 'COCO BEAN', 'PUERTO RICO', 'JOSE SAN', 'jsan@company.com');


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

```
SELECT *
FROM EMPLOYEE M FULL JOIN COFFEE C
ON m.shop_id=c.shop_id
FULL JOIN COFFEE_SHOP CS
ON c.shop_id=m.shop_id;
```

Build Schema | Edit Fullscreen | Browser | [ ] | Run SQL | Edit Fullscreen | [ ]

employee_id	first_name	last_name	hire_date	job_title	shop_id	coffee_id	shop_id	supplier_id	coffee_name	price_per_pound	shop_id	shop_name	city	state
100	EDWARD	GORDON	2023-01-12	waiter	50	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)
200	BRANDON	DAVIS	2023-02-15	waiter	100	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)
300	CHRIS	HORTON	2023-03-10	cook	150	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)
(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)	10	NOLA CAFE	NEW ORLEANS	LA
(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)	20	FRENCH CAFE	HARVEY	LA
(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)	(null)	30	CANAL CAFE	METAIRIE	LA

Record Count: 6, Execution Time: 73ms | View Execution Plan | link