**1A – Nora’s Bagels**

**First Normal from Blue Print**

|  |  |
| --- | --- |
| **BAGEL ORDER** | |
| PK | Bagel Order ID |
| PK | Bagel ID |
|  | Order Date |
|  | First Name |
|  | Last Name |
|  | Address 1 |
|  | Address 2 |
|  | City |
|  | State |
|  | Zip |
|  | Mobile Phone |
|  | Delivery Fee |
|  | Bagel Name |
|  | Bagel Description |
|  | Bagel Price |
|  | Bagel Quantity |
|  | Special Notes |

**A1B - Second Normal Form (2NF)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **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 Price |
|  | First Name |  |  | Bagel Quantity |  |  | Bagel Description |
|  | Last Name |  |  |  |  |  | Bagel Name |
|  | Address 1 |  |  |  |  |  |  |
|  | Address 2 |  |  |  |  |  |  |
|  | City |  |  |  |  |  |  |
|  | State |  |  |  |  |  |  |
|  | Zip |  |  |  |  |  |  |
|  | Mobile Phone |  |  |  |  |  |  |
|  | Delivery Fee |  |  |  |  |  |  |
|  | Special Notes |  |  |  |  |  |  |

**Explanation:** 1 bagel order may contain many bagels order line items. A bagel order line item may contain many bagels.

**A2C Third Normal Form (3NF)**

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

**Explanation:** Many Orders can have 1 customer, 1 Order may have many Bagel Order Line Items, A Bagel Order Line Item may have many Bagels.

**Final Physical Database Model**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | |  | **BAGEL ORDER LINE ITEM** | | |  | **BAGEL** | |  |
| PK | BagelOrderID | INT |  | PK / FK | bagelOrderID | INT |  | PK | BagelID | INT |
| FK | CustomerID | INT | 1:M | PK / FK | bagelID | VARCHAR(2) | 1:M |  | BagelPrice | NUMERIC(5,2) |
|  | OrderDate | TIMESTAMP |  |  | BagelQuantity | INT |  |  | BagelDescription | VARCHAR(30) |
|  | DeliveryFee | NUMERIC(5,2) |  |  |  |  |  |  | BagelName | VARCHAR(30) |
|  | SpecialNotes | VARCHAR(60) |  |  |  |  |  |  |  |  |
|  | M:1 |  |  |  |  |  |  |  |  |  |
|  | | |  |  |  |  |  |  |  |  |
| PK | CustomerID | INT |  |  |  |  |  |  |  |  |
|  | FirstName | VARCHAR(30) |  |  |  |  |  |  |  |  |
|  | LastName | VARCHAR(30) |  |  |  |  |  |  |  |  |
|  | Address1 | VARCHAR(30) |  |  |  |  |  |  |  |  |
|  | Address2 | VARCHAR(30) |  |  |  |  |  |  |  |  |
|  | City | VARCHAR(25) |  |  |  |  |  |  |  |  |
|  | State | VARCHAR(2) |  |  |  |  |  |  |  |  |
|  | Zip | INT |  |  |  |  |  |  |  |  |
|  | MobilePhone | VARCHAR(10) |  |  |  |  |  |  |  |  |

**2B – Jaunty Coffee Co.**

**Table Creation**

CREATE TABLE Employee (

employee\_id INT,

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,

shop\_name VARCHAR(50),

city VARCHAR(50),

state CHAR(2),

PRIMARY KEY (shop\_id)

);

CREATE TABLE Coffee (

coffee\_id INT,

shop\_id INT,

supplier\_id INT,

coffee\_name VARCHAR(30),

price\_per\_pound NUMERIC(5,2),

PRIMARY KEY (coffee\_id)

);

CREATE TABLE Supplier (

supplier\_id INT,

company\_name VARCHAR(50),

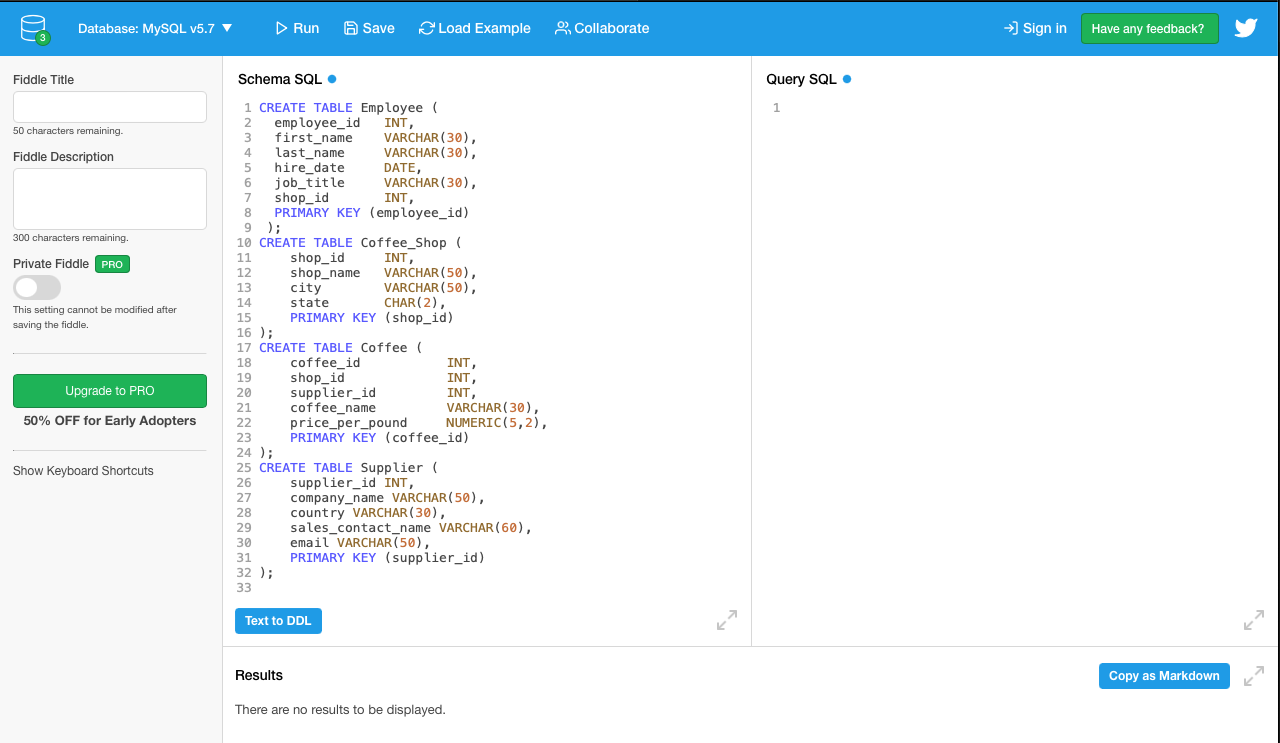
country VARCHAR(30),

sales\_contact\_name VARCHAR(60),

email VARCHAR(50),

PRIMARY KEY (supplier\_id)

);

****

**SHOW TABLES:**

****

**Population of Tables:**

INSERT INTO Employee(employee\_id, first\_name, last\_name, hire\_date, job\_title, shop\_id)

VALUES

(1, 'John', 'Travolta', '1962-01-05', 'Owner', 9001),

(2, 'Samual', 'Jackson', '1969-05-12', 'Manager', 9002),

(3, 'David', 'Hassellhoff', '1955-07-22', 'Temp', 9003);

INSERT INTO Coffee\_Shop(shop\_id, shop\_name, city, state)

VALUES

(9001, 'Caffeine Express', 'Draper', 'UT'),

(9002, 'StarBoys', 'Seattle', 'WA'),

(9003, 'CoffeeExpress', 'Salt Lake City', 'UT');

INSERT INTO Coffee(coffee\_id, shop\_id, supplier\_id, coffee\_name, price\_per\_pound)

VALUES

(2001, 9001, 8, 'Mocha', '5.99'),

(3001, 9002, 9, 'Americano', '6.99'),

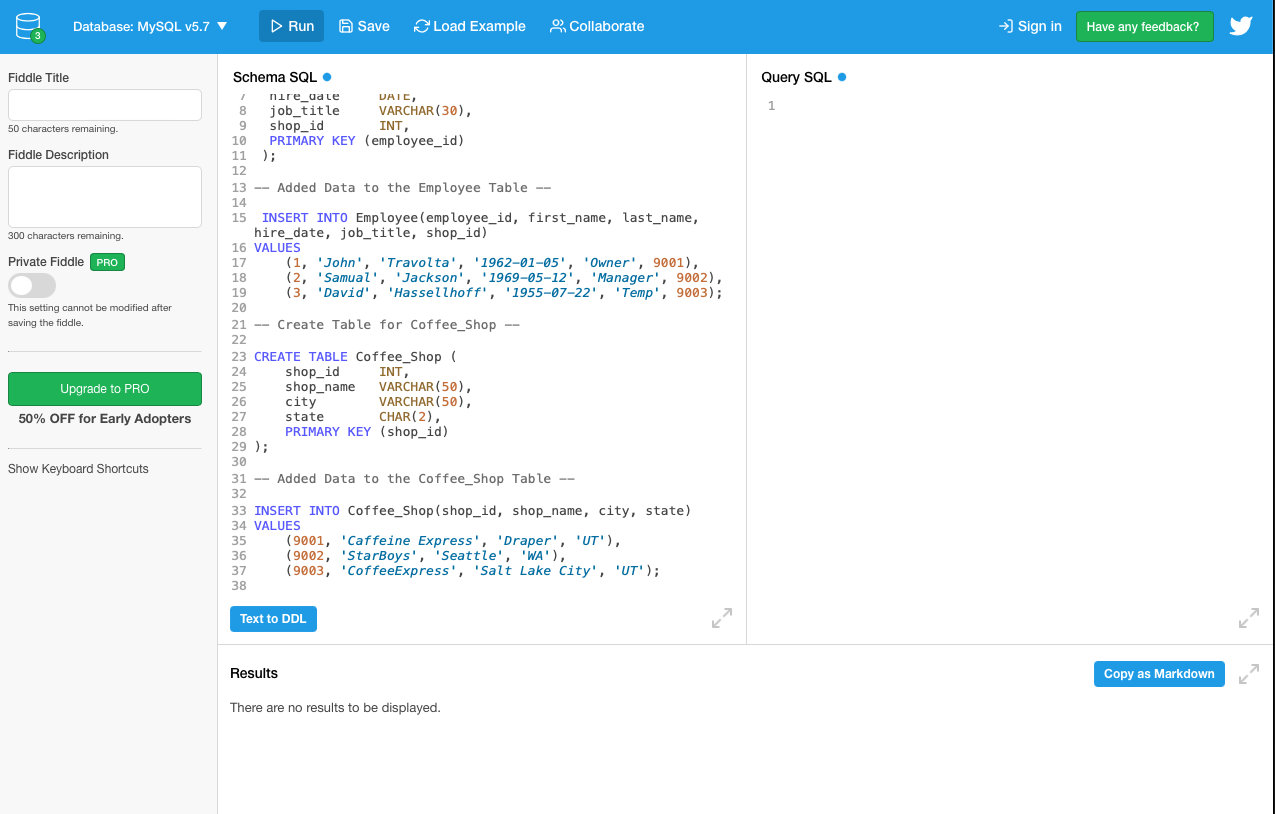
(4001, 9003, 7, 'PineappleExpress', '29.99');

INSERT INTO Supplier(supplier\_id, company\_name, country, sales\_contact\_name, email)

VALUES (8, 'MadHatter', 'USA', 'Tony Stark', 'tony.stark@email.com'),

(9, 'StarStruck', 'USA', 'Steve Rogers', 'steve.rogers@email.com'),

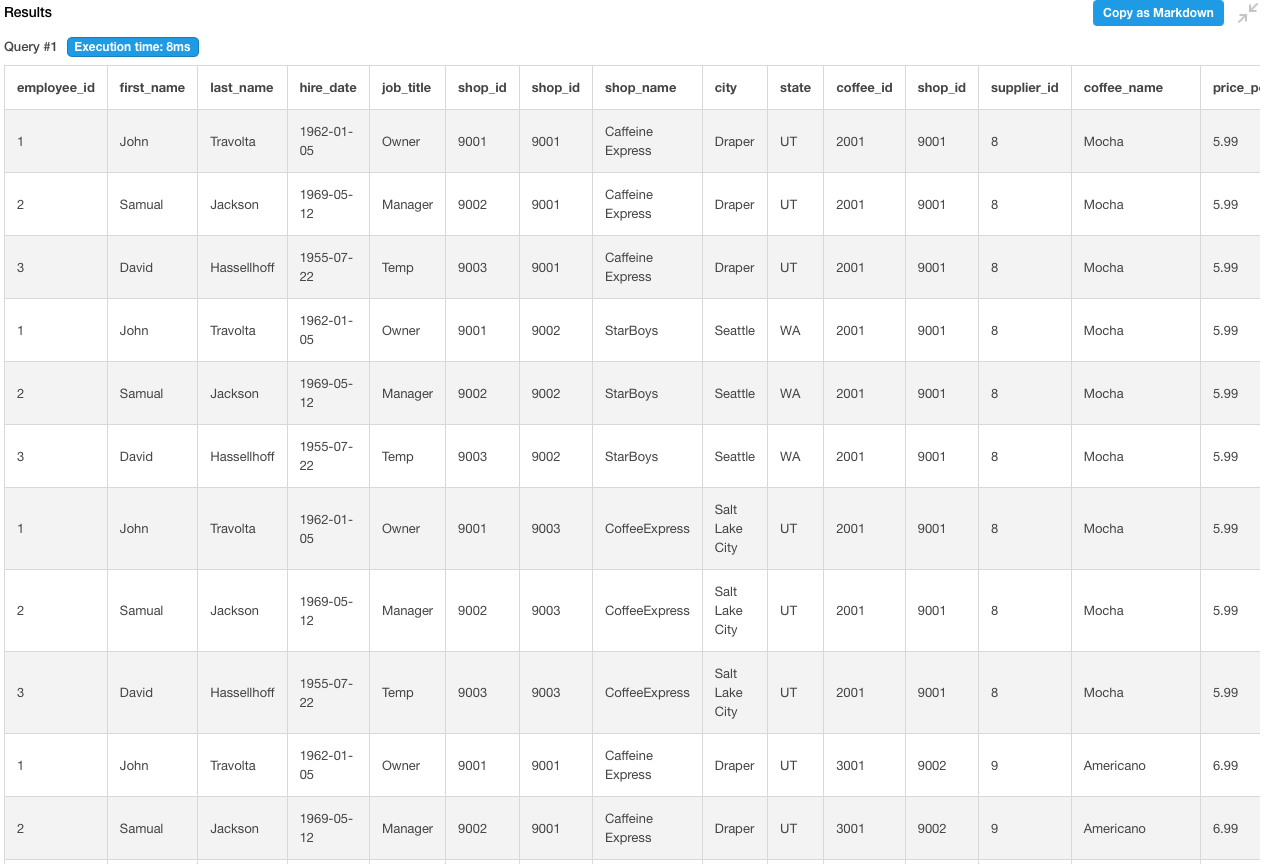
(7, 'StarBoy', 'Guam', 'Bruce Banner', 'brucebanner@gmail.com');

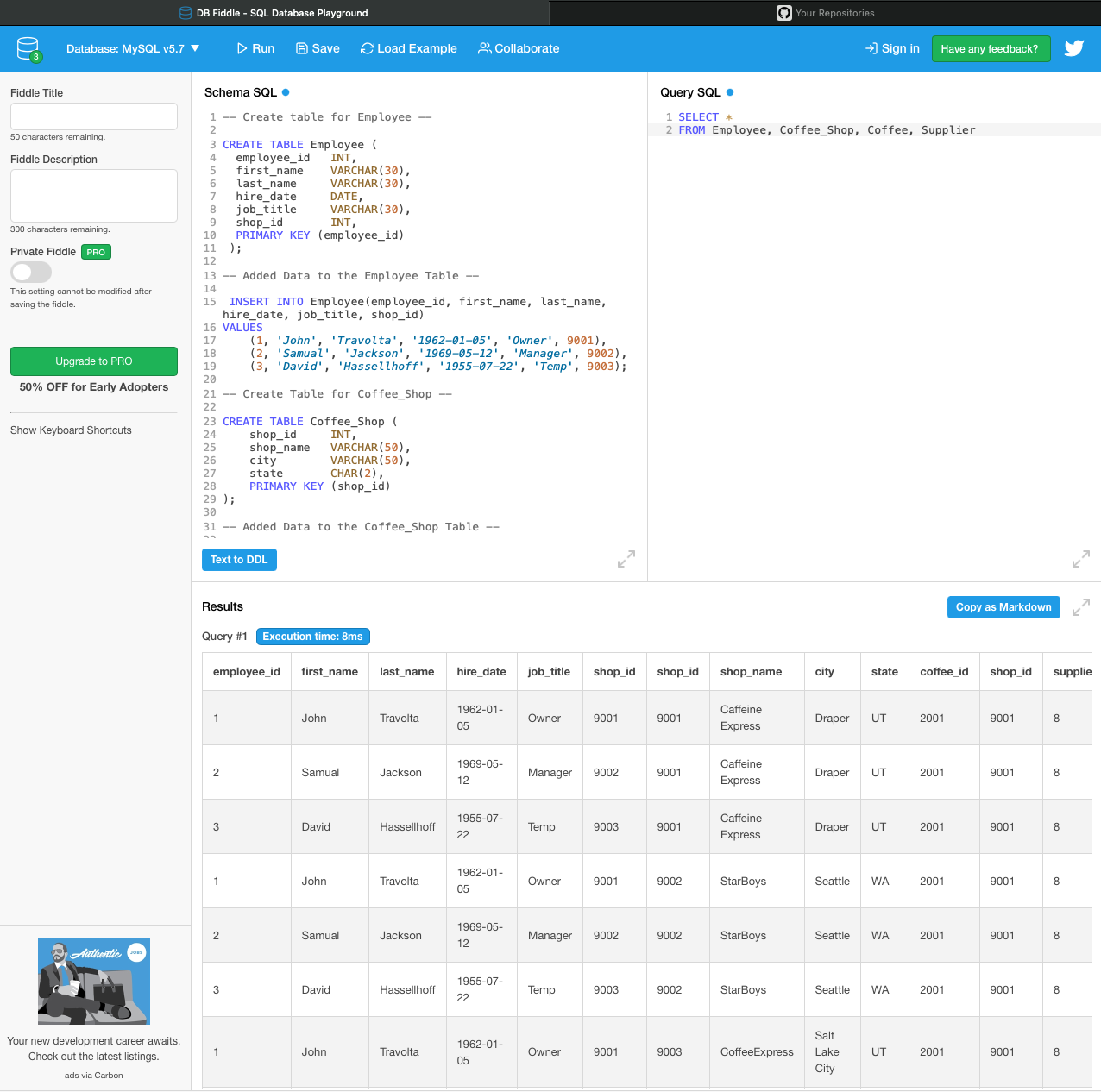
****

**Proof of Working Tables**

SELECT \*

FROM Employee, Coffee\_Shop, Coffee, Supplier

****

****

**CREATE VIEW:** EmployeeView

CREATE VIEW EmployeeView AS

SELECT

employee\_id,

CONCAT (first\_name,

last\_name) AS employee\_full\_name,

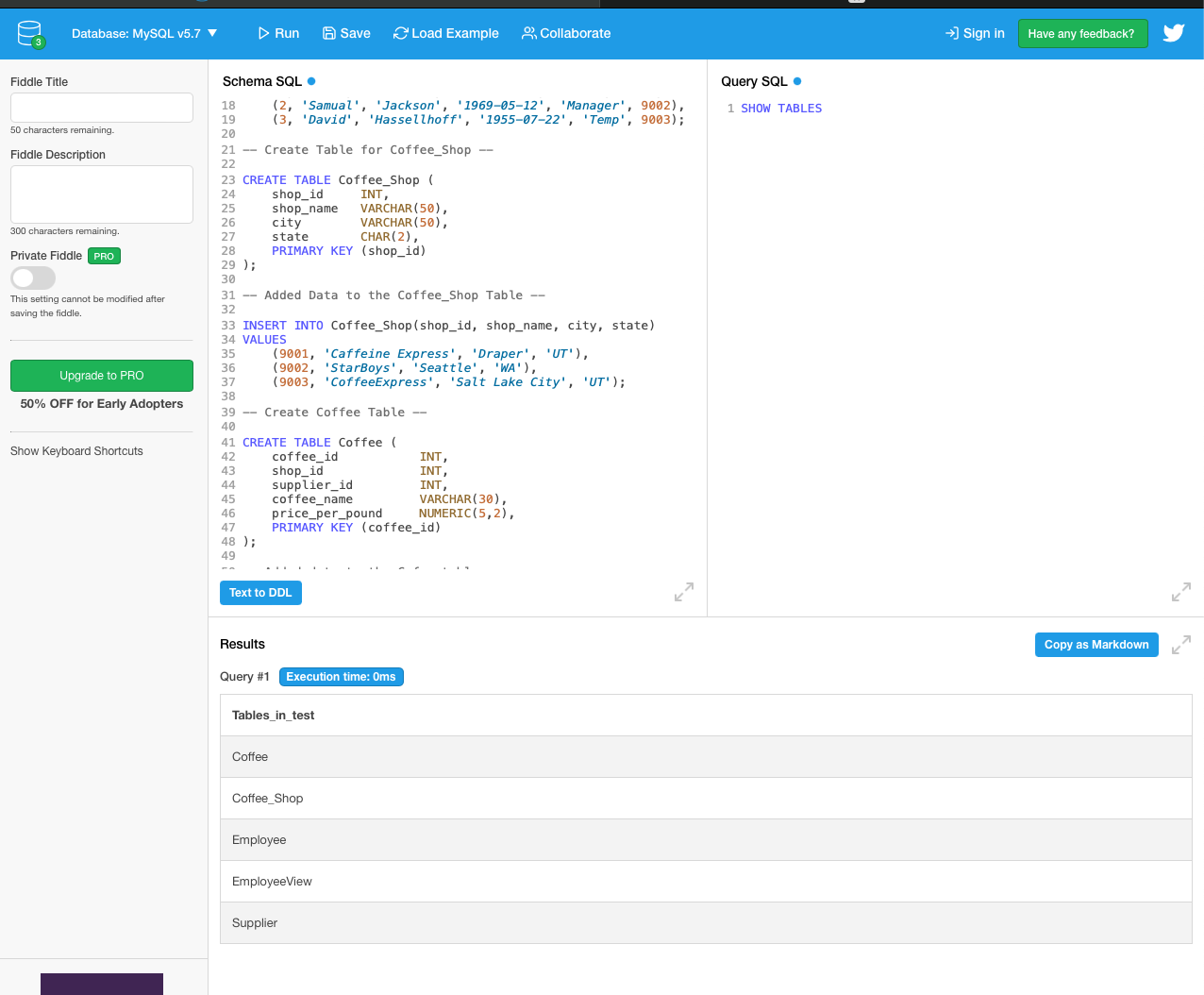
hire\_date,

job\_title,

shop\_id

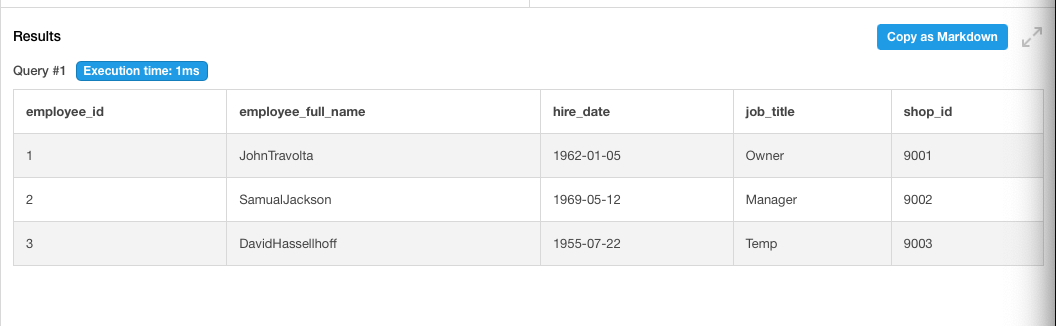
FROM

Employee

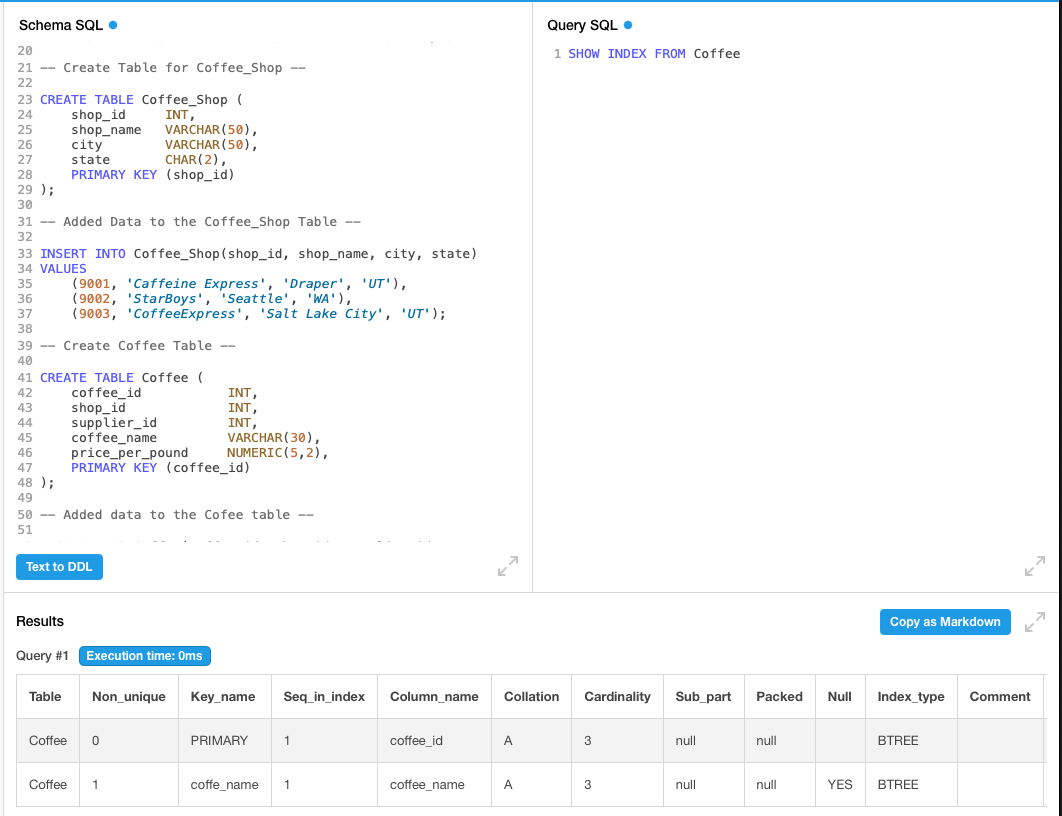


Select Statement Showing Employee View

SELECT \* FROM EmployeeView



**INDEX:** SHOW INDEX FROM Coffee

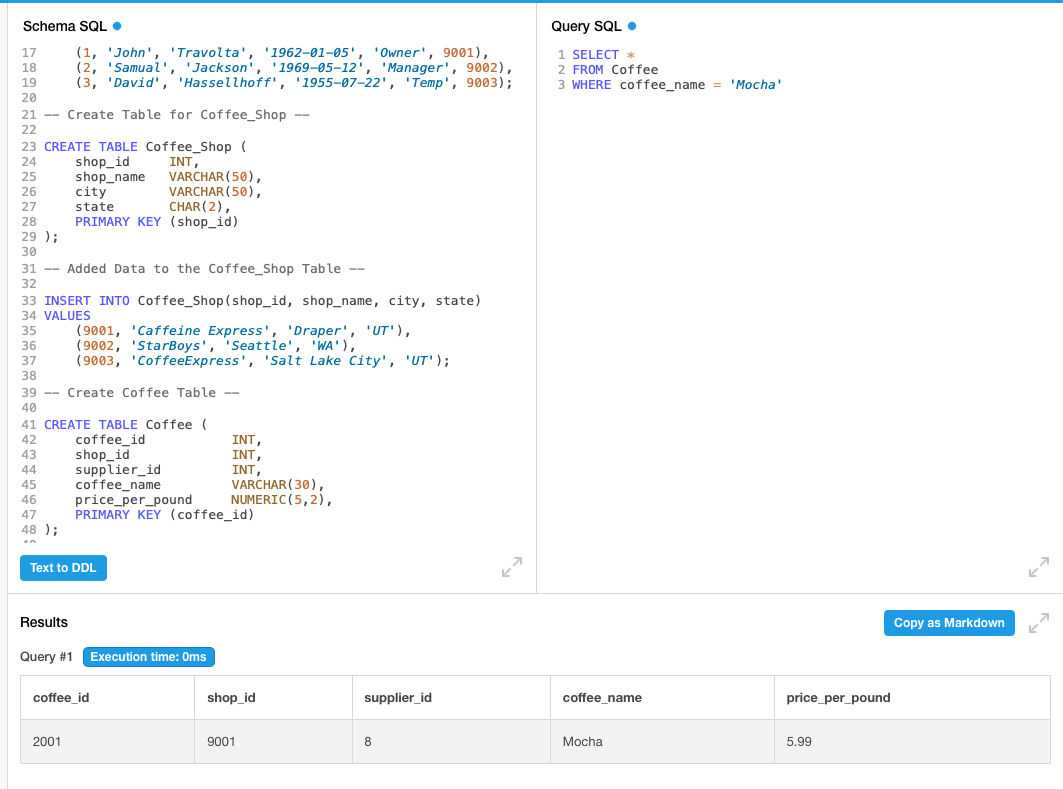


**SFW Statement:**

SELECT \*

FROM Coffee

WHERE coffee\_name = 'Mocha';

****

**Three Table INNER JOIN:**

SELECT Employee.employee\_id, Employee.first\_name, Employee.last\_name, Coffee\_Shop.city, Coffee\_Shop.shop\_name, Coffee.coffee\_name

FROM ((Employee

INNER JOIN Coffee\_Shop ON Employee.shop\_id = Coffee\_Shop.shop\_id)

INNER JOIN Coffee ON Coffee\_Shop.shop\_id = Coffee.shop\_id);

