CIND 119: Introduction to Big Data Analytics Assignment 2 (15% of the final grade)

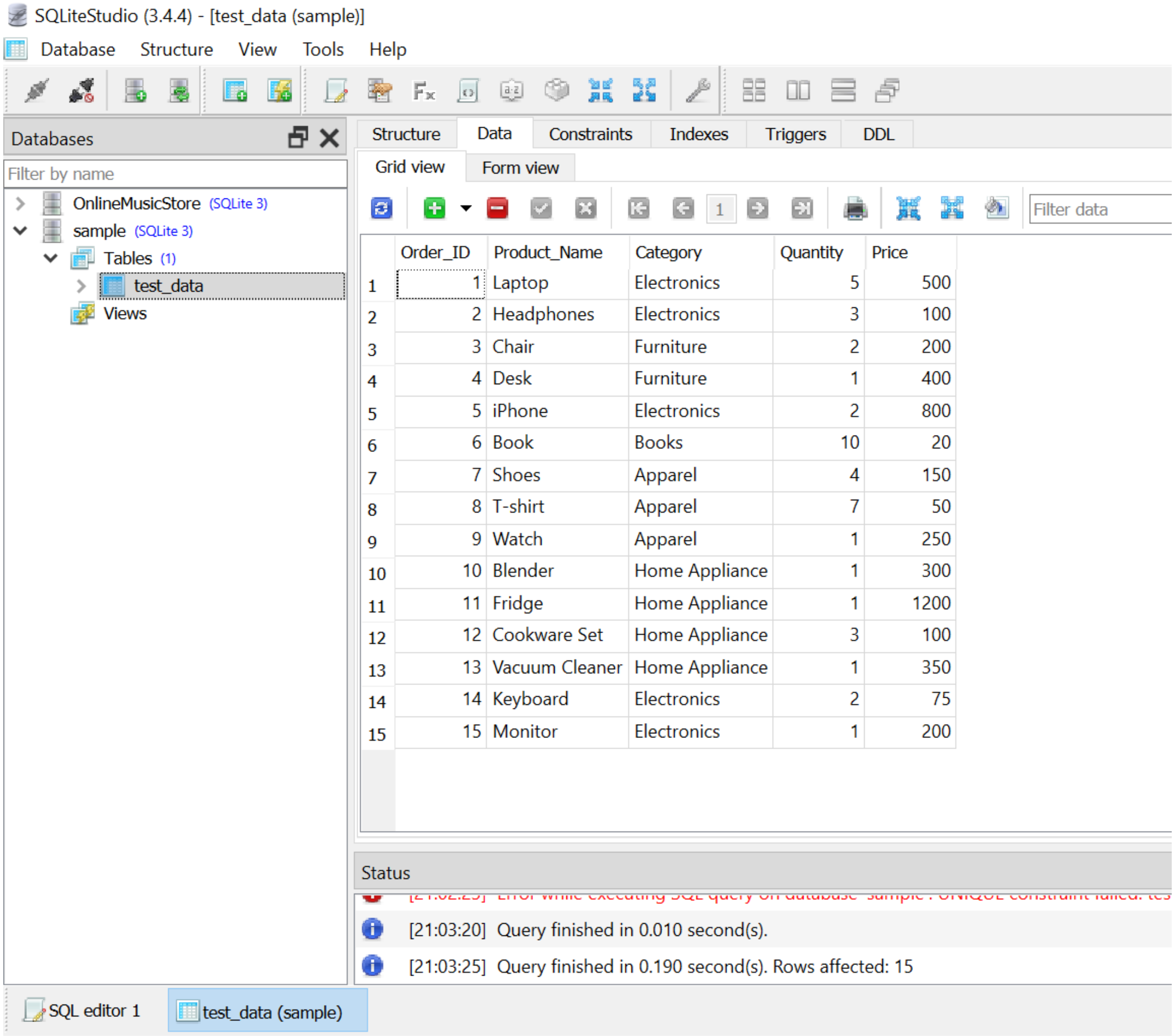
Querying an RDBMS database using SQLite Studio

By: Stephanie Boissonneault

Complete this assignment using SQLite Studio.

Question 1: Create an SQLite database called "sample".

Question 2: Within the "sample" database, create a table called "test_data" and load the following data into the table: (5 points)



	Order_ID	Product_Name	Category	Quantity	Price
1	1	Laptop	Electronics	5	500
2	2	Headphones	Electronics	3	100
3	3	Chair	Furniture	2	200
4	4	Desk	Furniture	1	400
5	5	iPhone	Electronics	2	800
6	6	Book	Books	10	20
7	7	Shoes	Apparel	4	150
8	8	T-shirt	Apparel	7	50
9	9	Watch	Apparel	1	250
10	10	Blender	Home Appliance	1	300
11	11	Fridge	Home Appliance	1	1200
12	12	Cookware Set	Home Appliance	3	100
13	13	Vacuum Cleaner	Home Appliance	1	350
14	14	Keyboard	Electronics	2	75
15	15	Monitor	Electronics	1	200

```
[21:02:23] Error while executing SQL query on database 'sample': UNIQUE constraint failed: tes
[21:03:20] Query finished in 0.010 second(s).
[21:03:25] Query finished in 0.190 second(s). Rows affected: 15
```

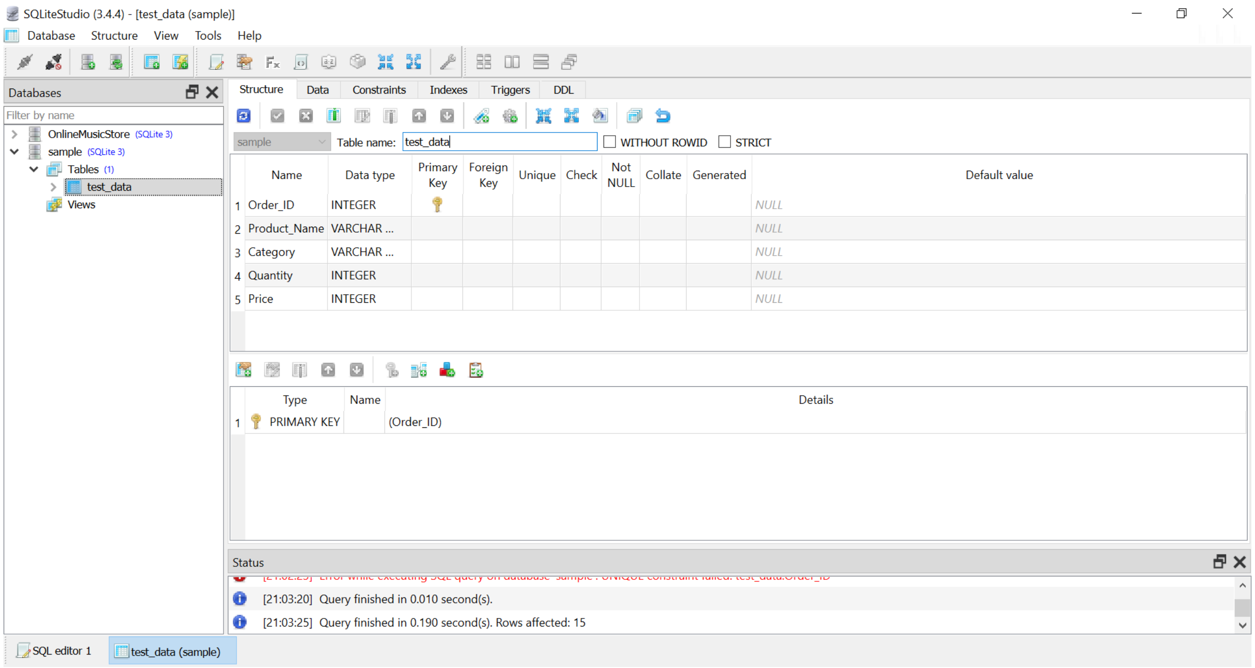
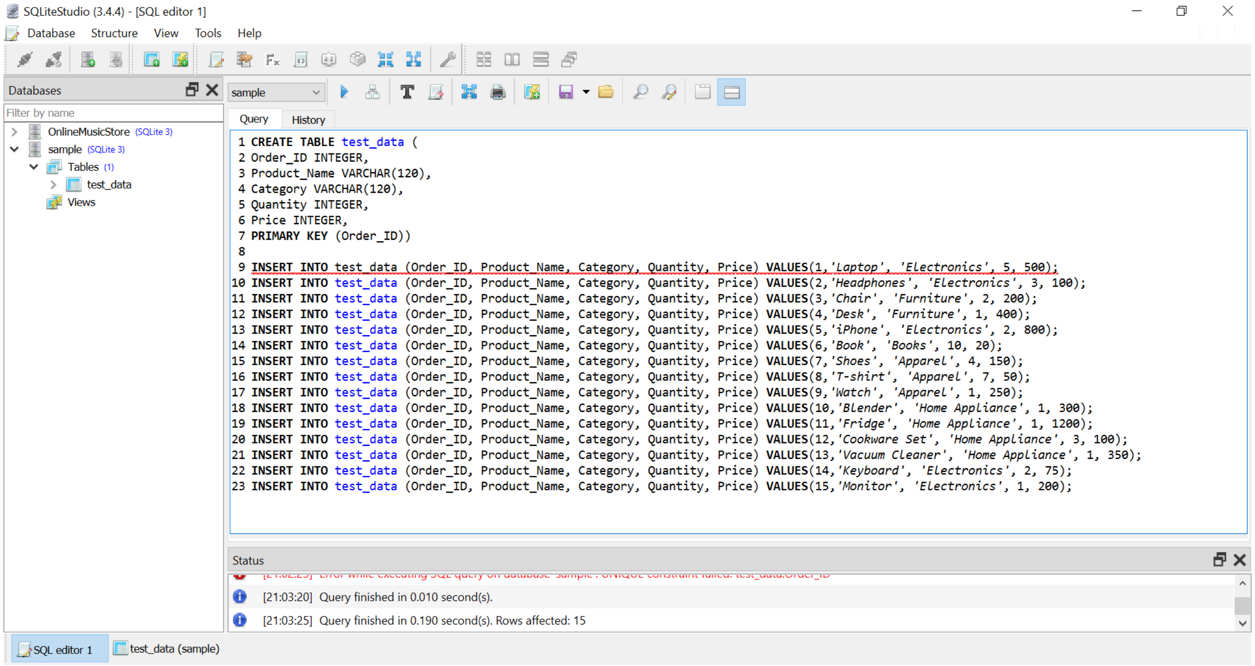


Table name: test_data □ WITHOUT ROWID □ STRICT

	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Generated	Default value
1	Order_ID	INTEGER	🔑							NULL
2	Product_Name	VARCHAR ...								NULL
3	Category	VARCHAR ...								NULL
4	Quantity	INTEGER								NULL
5	Price	INTEGER								NULL

	Type	Name	Details
1	🔑 PRIMARY KEY		(Order_ID)

```
[21:02:25] Error while executing SQL query on database 'sample': UNIQUE constraint failed: test_data.Order_ID
[21:03:20] Query finished in 0.010 second(s).
[21:03:25] Query finished in 0.190 second(s). Rows affected: 15
```



```sql
CREATE TABLE test_data (
Order_ID INTEGER,
Product_Name VARCHAR(120),
Category VARCHAR(120),
Quantity INTEGER,
Price INTEGER,
PRIMARY KEY (Order_ID))

INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price) VALUES(1,'Laptop', 'Electronics', 5, 500);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price) VALUES(2,'Headphones', 'Electronics', 3, 100);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price) VALUES(3,'Chair', 'Furniture', 2, 200);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price) VALUES(4,'Desk', 'Furniture', 1, 400);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price) VALUES(5,'iPhone', 'Electronics', 2, 800);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price) VALUES(6,'Book', 'Books', 10, 20);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price) VALUES(7,'Shoes', 'Apparel', 4, 150);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price) VALUES(8,'T-shirt', 'Apparel', 7, 50);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price) VALUES(9,'Watch', 'Apparel', 1, 250);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price) VALUES(10,'Blender', 'Home Appliance', 1, 300);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price) VALUES(11,'Fridge', 'Home Appliance', 1, 1200);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price) VALUES(12,'Cookware Set', 'Home Appliance', 3, 100);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price) VALUES(13,'Vacuum Cleaner', 'Home Appliance', 1, 350);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price) VALUES(14,'Keyboard', 'Electronics', 2, 75);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price) VALUES(15,'Monitor', 'Electronics', 1, 200);
```

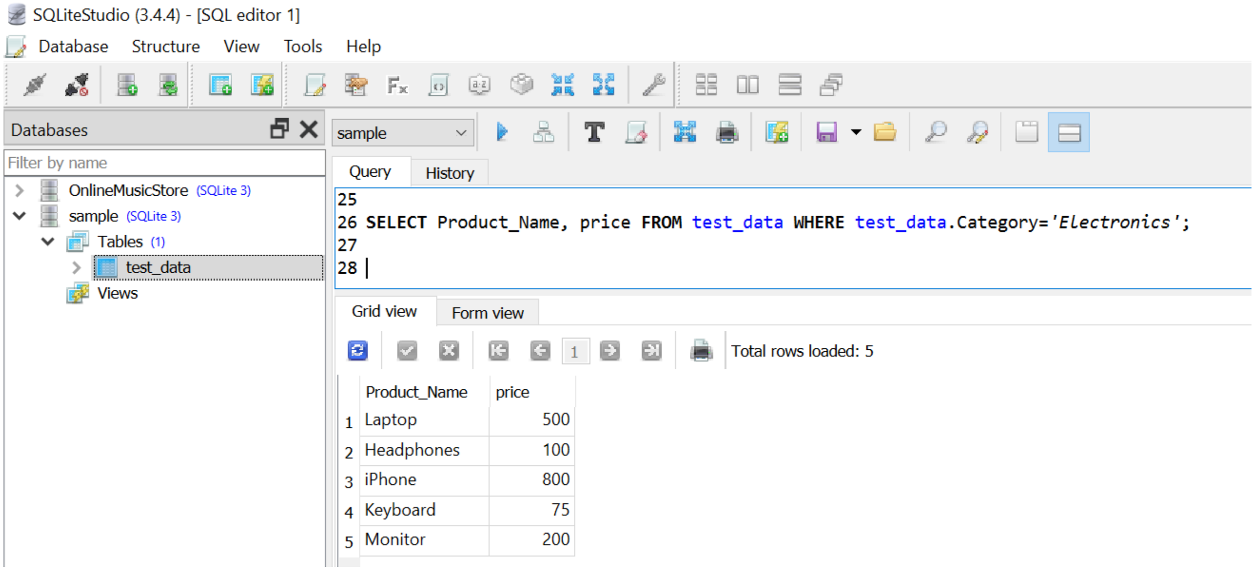
```
[21:02:25] Error while executing SQL query on database 'sample': UNIQUE constraint failed: test_data.Order_ID
[21:03:20] Query finished in 0.010 second(s).
[21:03:25] Query finished in 0.190 second(s). Rows affected: 15
```

```
CREATE TABLE test_data (
Order_ID INTEGER,
Product_Name VARCHAR(120),
Category VARCHAR(120),
Quantity INTEGER,
Price INTEGER,

PRIMARY KEY (Order_ID))
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price)
VALUES(1,'Laptop', 'Electronics', 5, 500);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price)
VALUES(2,'Headphones', 'Electronics', 3, 100);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price)
VALUES(3,'Chair', 'Furniture', 2, 200);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price)
VALUES(4,'Desk', 'Furniture', 1, 400);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price)
VALUES(5,'iPhone', 'Electronics', 2, 800);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price)
VALUES(6,'Book', 'Books', 10, 20);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price)
VALUES(7,'Shoes', 'Apparel', 4, 150);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price)
VALUES(8,'T-shirt', 'Apparel', 7, 50);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price)
VALUES(9,'Watch', 'Apparel', 1, 250);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price)
VALUES(10,'Blender', 'Home Appliance', 1, 300);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price)
VALUES(11,'Fridge', 'Home Appliance', 1, 1200);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price)
VALUES(12,'Cookware Set', 'Home Appliance', 3, 100);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price)
VALUES(13,'Vacuum Cleaner', 'Home Appliance', 1, 350);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price)
VALUES(14,'Keyboard', 'Electronics', 2, 75);
INSERT INTO test_data (Order_ID, Product_Name, Category, Quantity, Price)
VALUES(15,'Monitor', 'Electronics', 1, 200);
```

Question 3: Write SQL queries to select/compute data from the "test_data" table. (2 points each)

a. Select the Product_Name and Price of products where the Category is 'Electronics'.



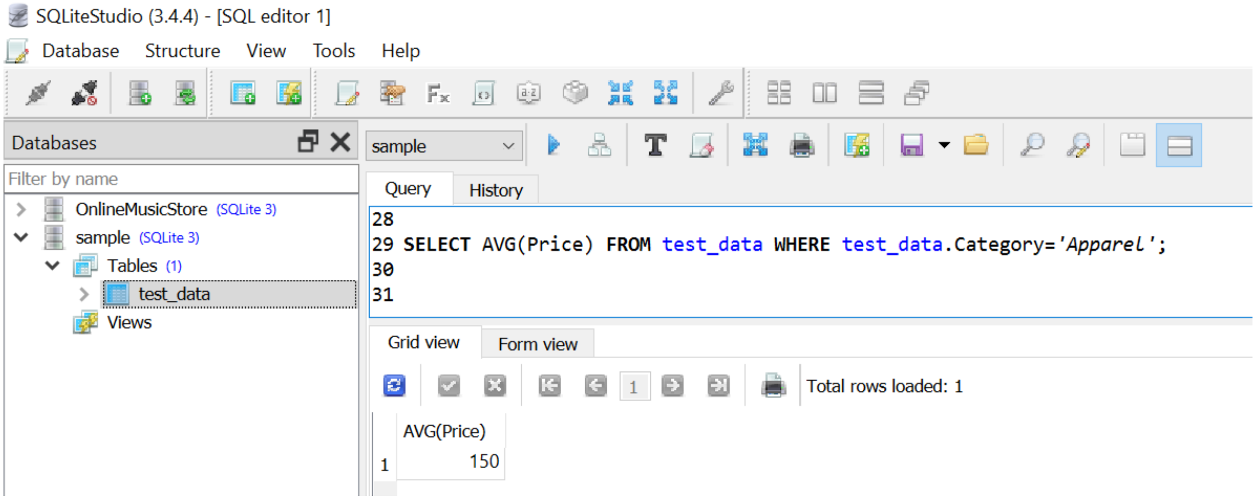
```
SELECT Product_Name, price FROM test_data WHERE test_data.Category='Electronics';
```

Total rows loaded: 5

	Product_Name	price
1	Laptop	500
2	Headphones	100
3	iPhone	800
4	Keyboard	75
5	Monitor	200

SELECT Product_Name, price FROM test_data WHERE test_data.Category='Electronics';

b. Compute the average price of products in the 'Apparel' category.



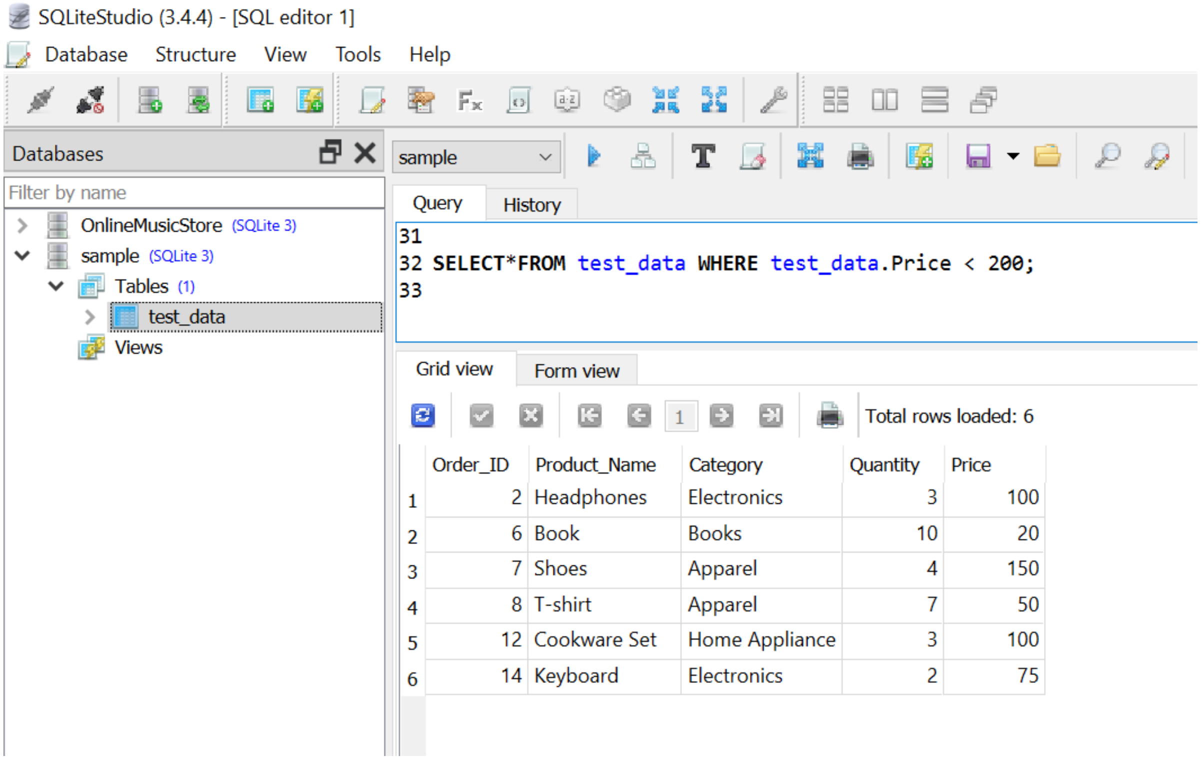
```
SELECT AVG(Price) FROM test_data WHERE test_data.Category='Apparel';
```

Total rows loaded: 1

	AVG(Price)
1	150

SELECT AVG(Price) FROM test_data WHERE test_data.Category='Apparel';

c. Select all fields of products where the price is less than 200.



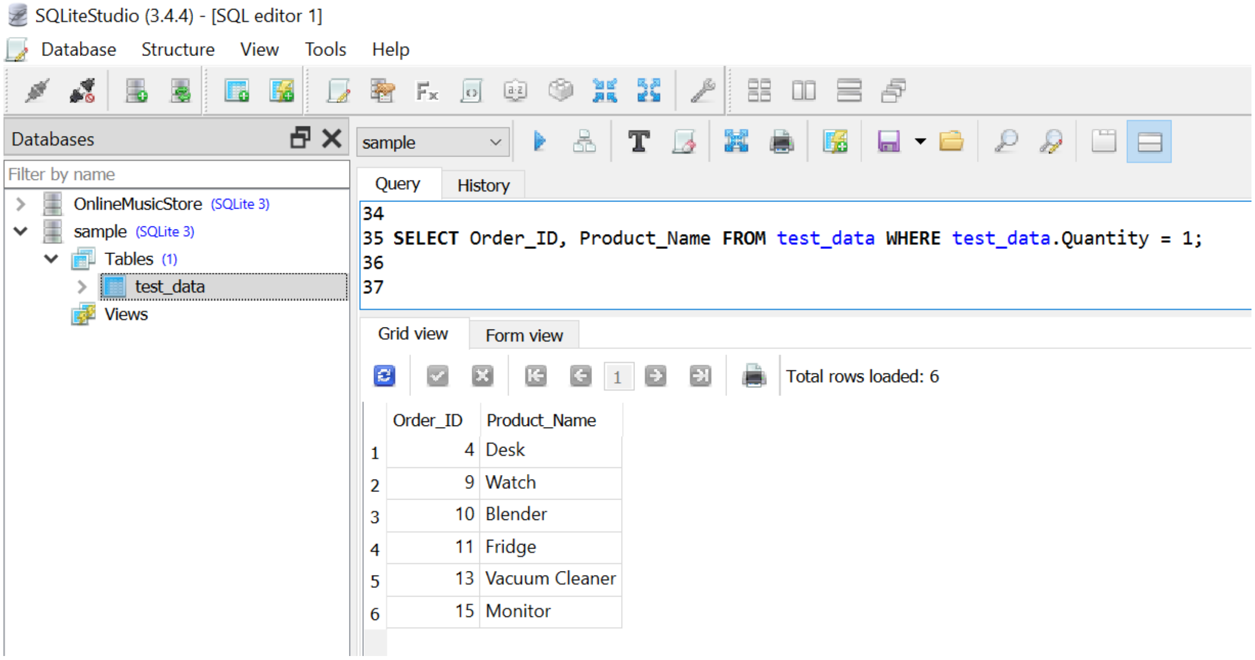
```
SELECT*FROM test_data WHERE test_data.Price < 200;
```

Total rows loaded: 6

	Order_ID	Product_Name	Category	Quantity	Price
1	2	Headphones	Electronics	3	100
2	6	Book	Books	10	20
3	7	Shoes	Apparel	4	150
4	8	T-shirt	Apparel	7	50
5	12	Cookware Set	Home Appliance	3	100
6	14	Keyboard	Electronics	2	75

SELECT*FROM test_data WHERE test_data.Price < 200;

d. Select the Order_ID and Product_Name of products where the Quantity is equal to 1.



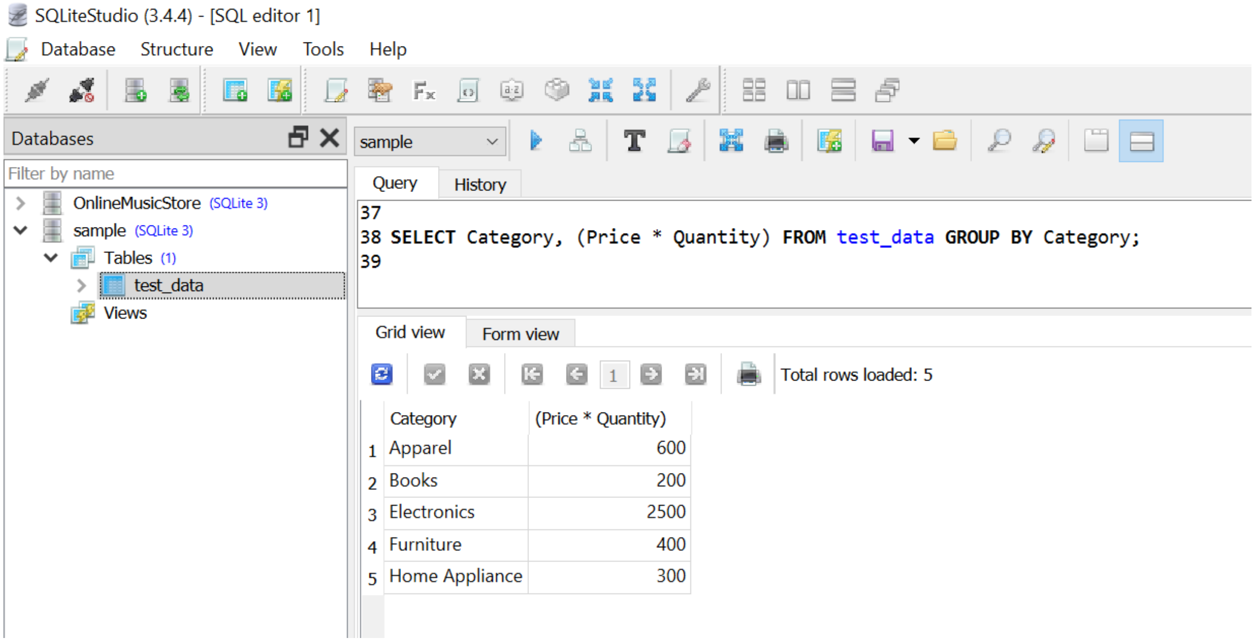
```
SELECT Order_ID, Product_Name FROM test_data WHERE test_data.Quantity = 1;
```

Total rows loaded: 6

	Order_ID	Product_Name
1	4	Desk
2	9	Watch
3	10	Blender
4	11	Fridge
5	13	Vacuum Cleaner
6	15	Monitor

SELECT Order_ID, Product_Name FROM test_data WHERE test_data.Quantity = 1;

e. Compute the total revenue (Price * Quantity) for each Category.



```
SELECT Category, (Price * Quantity) FROM test_data GROUP BY Category;
```

Total rows loaded: 5

	Category	(Price * Quantity)
1	Apparel	600
2	Books	200
3	Electronics	2500
4	Furniture	400
5	Home Appliance	300

SELECT Category, (Price * Quantity) FROM test_data GROUP BY Category;