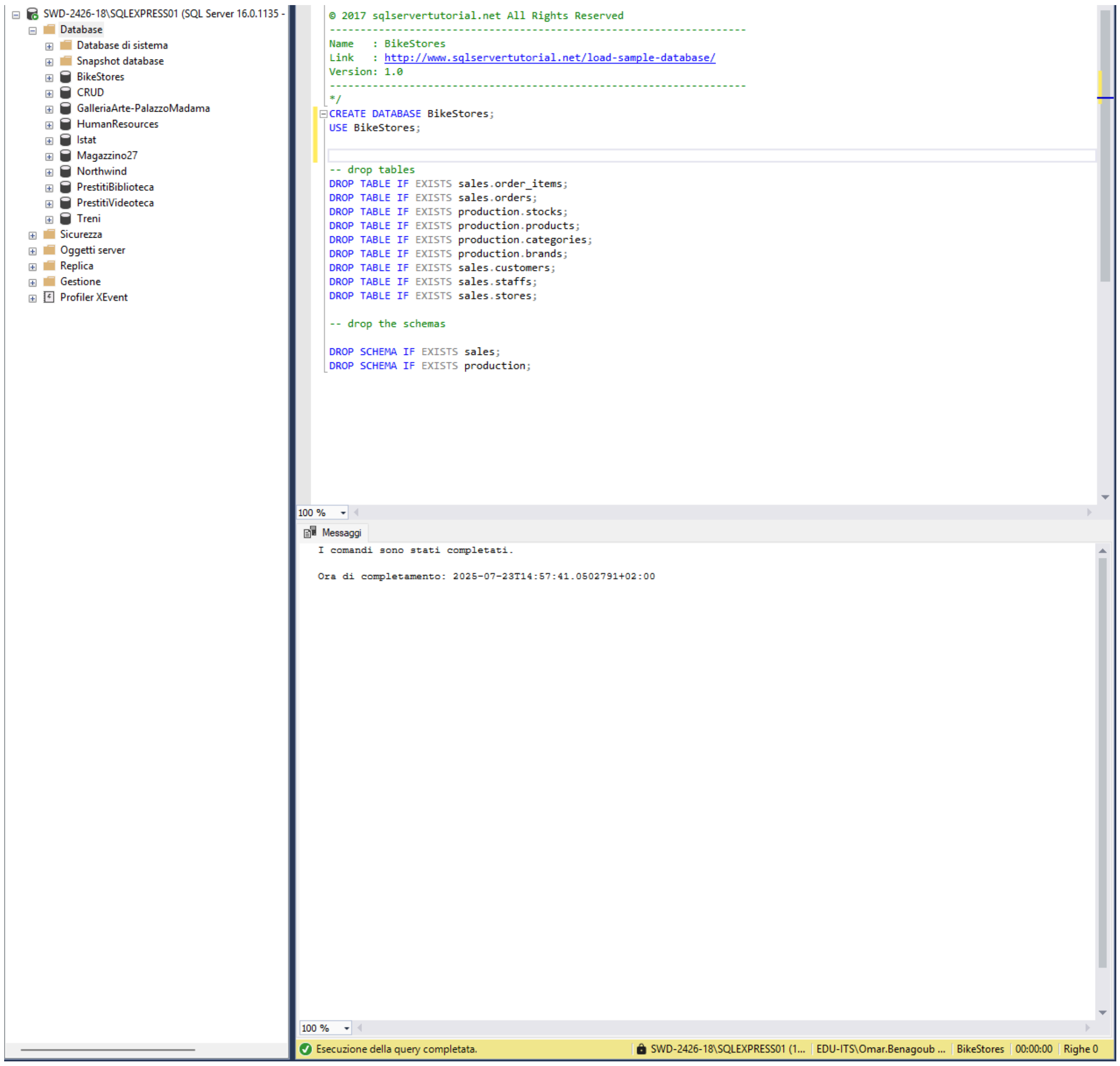
Benagoub
Omar
23-07-2025
Corso Software Developer
Base di Dati - SQL

Operazioni da eseguire:

1. creare il database BikeStores

CREATE DATABASE BikeStores;

2. eseguire il file ... drop all objects (inserire il database da utilizzare)

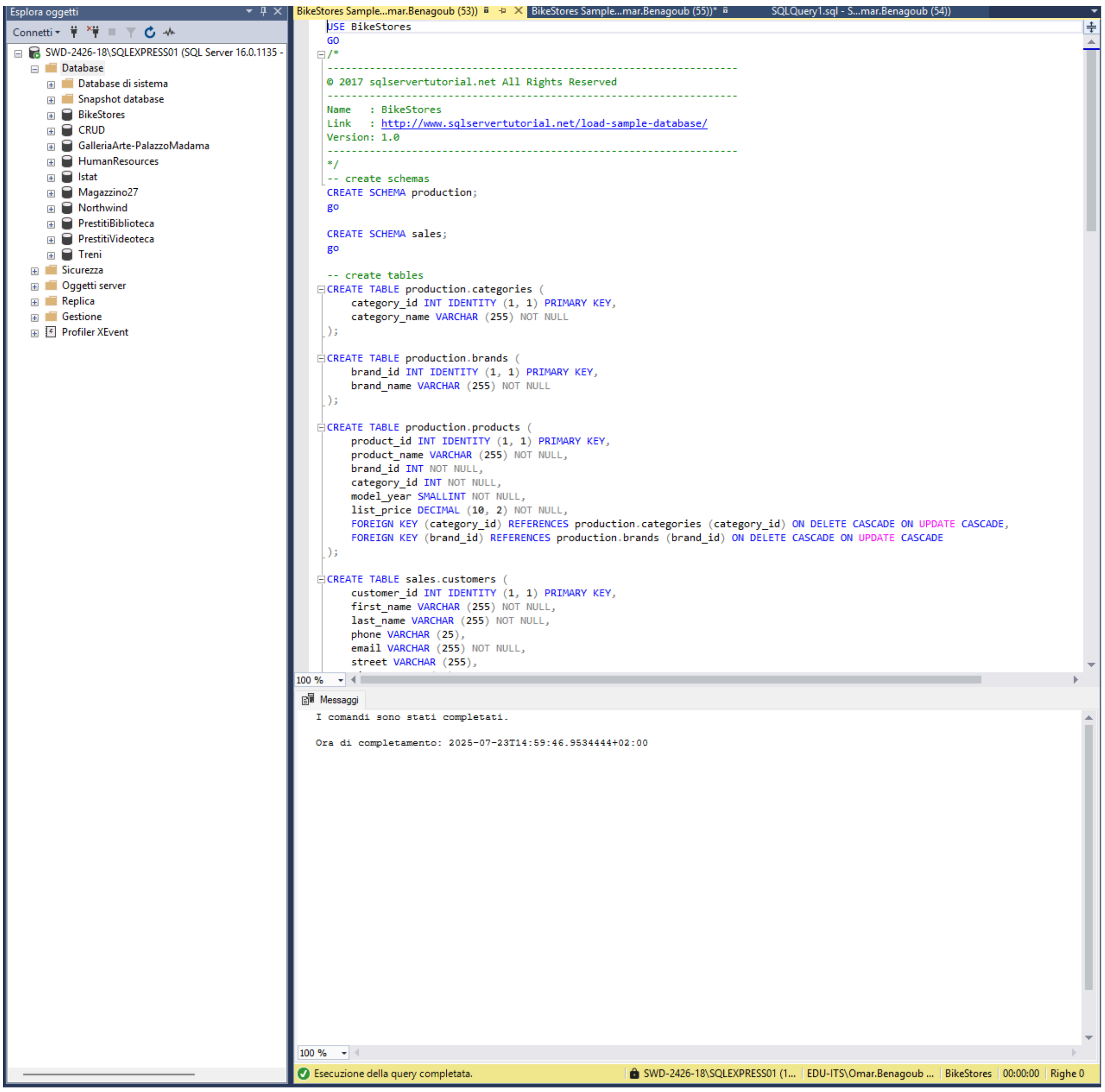


```
© 2017 sqlservertutorial.net All Rights Reserved
--------------------------------------------------------------------
Name   : BikeStores
Link   : http://www.sqlservertutorial.net/load-sample-database/
Version: 1.0
--------------------------------------------------------------------
*/
CREATE DATABASE BikeStores;
USE BikeStores;

-- drop tables
DROP TABLE IF EXISTS sales.order_items;
DROP TABLE IF EXISTS sales.orders;
DROP TABLE IF EXISTS production.stocks;
DROP TABLE IF EXISTS production.products;
DROP TABLE IF EXISTS production.categories;
DROP TABLE IF EXISTS production.brands;
DROP TABLE IF EXISTS sales.customers;
DROP TABLE IF EXISTS sales.staffs;
DROP TABLE IF EXISTS sales.stores;

-- drop the schemas

DROP SCHEMA IF EXISTS sales;
DROP SCHEMA IF EXISTS production;
```

```
I comandi sono stati completati.

Ora di completamento: 2025-07-23T14:57:41.0502791+02:00
```

```
Esecuzione della query completata.   SWD-2426-18\SQLEXPRESS01 (1...   EDU-ITS\Omar.Benagoub ...   BikeStores   00:00:00   Righe 0
```

3. eseguire il file … create object

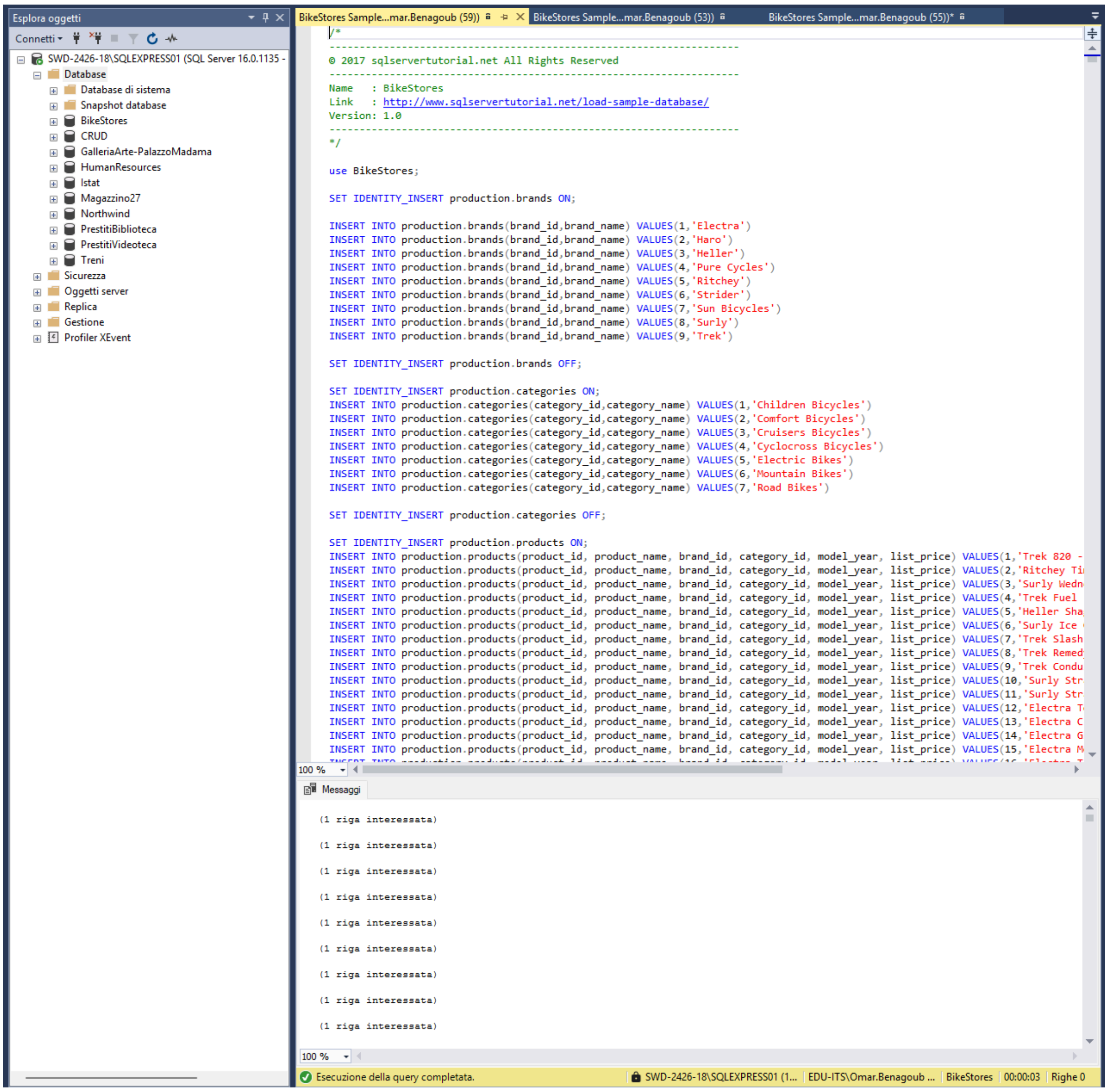


```sql
USE BikeStores
GO
/*
--------------------------------------------------------------------
© 2017 sqlservertutorial.net All Rights Reserved
--------------------------------------------------------------------
Name   : BikeStores
Link   : http://www.sqlservertutorial.net/load-sample-database/
Version: 1.0
--------------------------------------------------------------------
*/
-- create schemas
CREATE SCHEMA production;
go

CREATE SCHEMA sales;
go

-- create tables
CREATE TABLE production.categories (
    category_id INT IDENTITY (1, 1) PRIMARY KEY,
    category_name VARCHAR (255) NOT NULL
);

CREATE TABLE production.brands (
    brand_id INT IDENTITY (1, 1) PRIMARY KEY,
    brand_name VARCHAR (255) NOT NULL
);

CREATE TABLE production.products (
    product_id INT IDENTITY (1, 1) PRIMARY KEY,
    product_name VARCHAR (255) NOT NULL,
    brand_id INT NOT NULL,
    category_id INT NOT NULL,
    model_year SMALLINT NOT NULL,
    list_price DECIMAL (10, 2) NOT NULL,
    FOREIGN KEY (category_id) REFERENCES production.categories (category_id) ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (brand_id) REFERENCES production.brands (brand_id) ON DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE sales.customers (
    customer_id INT IDENTITY (1, 1) PRIMARY KEY,
    first_name VARCHAR (255) NOT NULL,
    last_name VARCHAR (255) NOT NULL,
    phone VARCHAR (25),
    email VARCHAR (255) NOT NULL,
    street VARCHAR (255),
```

```
I comandi sono stati completati.

Ora di completamento: 2025-07-23T14:59:46.9534444+02:00
```

```
Esecuzione della query completata.    SWD-2426-18\SQLEXPRESS01 (1...    EDU-ITS\Omar.Benagoub ...    BikeStores    00:00:00    Righe 0
```

4. eseguire il file … load data



```sql
/*
--------------------------------------------------------------------
© 2017 sqlservertutorial.net All Rights Reserved
--------------------------------------------------------------------
Name   : BikeStores
Link   : http://www.sqlservertutorial.net/load-sample-database/
Version: 1.0
--------------------------------------------------------------------
*/

use BikeStores;

SET IDENTITY_INSERT production.brands ON;

INSERT INTO production.brands(brand_id,brand_name) VALUES(1,'Electra')
INSERT INTO production.brands(brand_id,brand_name) VALUES(2,'Haro')
INSERT INTO production.brands(brand_id,brand_name) VALUES(3,'Heller')
INSERT INTO production.brands(brand_id,brand_name) VALUES(4,'Pure Cycles')
INSERT INTO production.brands(brand_id,brand_name) VALUES(5,'Ritchey')
INSERT INTO production.brands(brand_id,brand_name) VALUES(6,'Strider')
INSERT INTO production.brands(brand_id,brand_name) VALUES(7,'Sun Bicycles')
INSERT INTO production.brands(brand_id,brand_name) VALUES(8,'Surly')
INSERT INTO production.brands(brand_id,brand_name) VALUES(9,'Trek')

SET IDENTITY_INSERT production.brands OFF;

SET IDENTITY_INSERT production.categories ON;
INSERT INTO production.categories(category_id,category_name) VALUES(1,'Children Bicycles')
INSERT INTO production.categories(category_id,category_name) VALUES(2,'Comfort Bicycles')
INSERT INTO production.categories(category_id,category_name) VALUES(3,'Cruisers Bicycles')
INSERT INTO production.categories(category_id,category_name) VALUES(4,'Cyclocross Bicycles')
INSERT INTO production.categories(category_id,category_name) VALUES(5,'Electric Bikes')
INSERT INTO production.categories(category_id,category_name) VALUES(6,'Mountain Bikes')
INSERT INTO production.categories(category_id,category_name) VALUES(7,'Road Bikes')

SET IDENTITY_INSERT production.categories OFF;

SET IDENTITY_INSERT production.products ON;
INSERT INTO production.products(product_id, product_name, brand_id, category_id, model_year, list_price) VALUES(1,'Trek 820 -
INSERT INTO production.products(product_id, product_name, brand_id, category_id, model_year, list_price) VALUES(2,'Ritchey Ti
INSERT INTO production.products(product_id, product_name, brand_id, category_id, model_year, list_price) VALUES(3,'Surly Wedn
INSERT INTO production.products(product_id, product_name, brand_id, category_id, model_year, list_price) VALUES(4,'Trek Fuel
INSERT INTO production.products(product_id, product_name, brand_id, category_id, model_year, list_price) VALUES(5,'Heller Sha
INSERT INTO production.products(product_id, product_name, brand_id, category_id, model_year, list_price) VALUES(6,'Surly Ice
INSERT INTO production.products(product_id, product_name, brand_id, category_id, model_year, list_price) VALUES(7,'Trek Slash
INSERT INTO production.products(product_id, product_name, brand_id, category_id, model_year, list_price) VALUES(8,'Trek Remed
INSERT INTO production.products(product_id, product_name, brand_id, category_id, model_year, list_price) VALUES(9,'Trek Condu
INSERT INTO production.products(product_id, product_name, brand_id, category_id, model_year, list_price) VALUES(10,'Surly Str
INSERT INTO production.products(product_id, product_name, brand_id, category_id, model_year, list_price) VALUES(11,'Surly Str
INSERT INTO production.products(product_id, product_name, brand_id, category_id, model_year, list_price) VALUES(12,'Electra T
INSERT INTO production.products(product_id, product_name, brand_id, category_id, model_year, list_price) VALUES(13,'Electra C
INSERT INTO production.products(product_id, product_name, brand_id, category_id, model_year, list_price) VALUES(14,'Electra G
INSERT INTO production.products(product_id, product_name, brand_id, category_id, model_year, list_price) VALUES(15,'Electra M
```

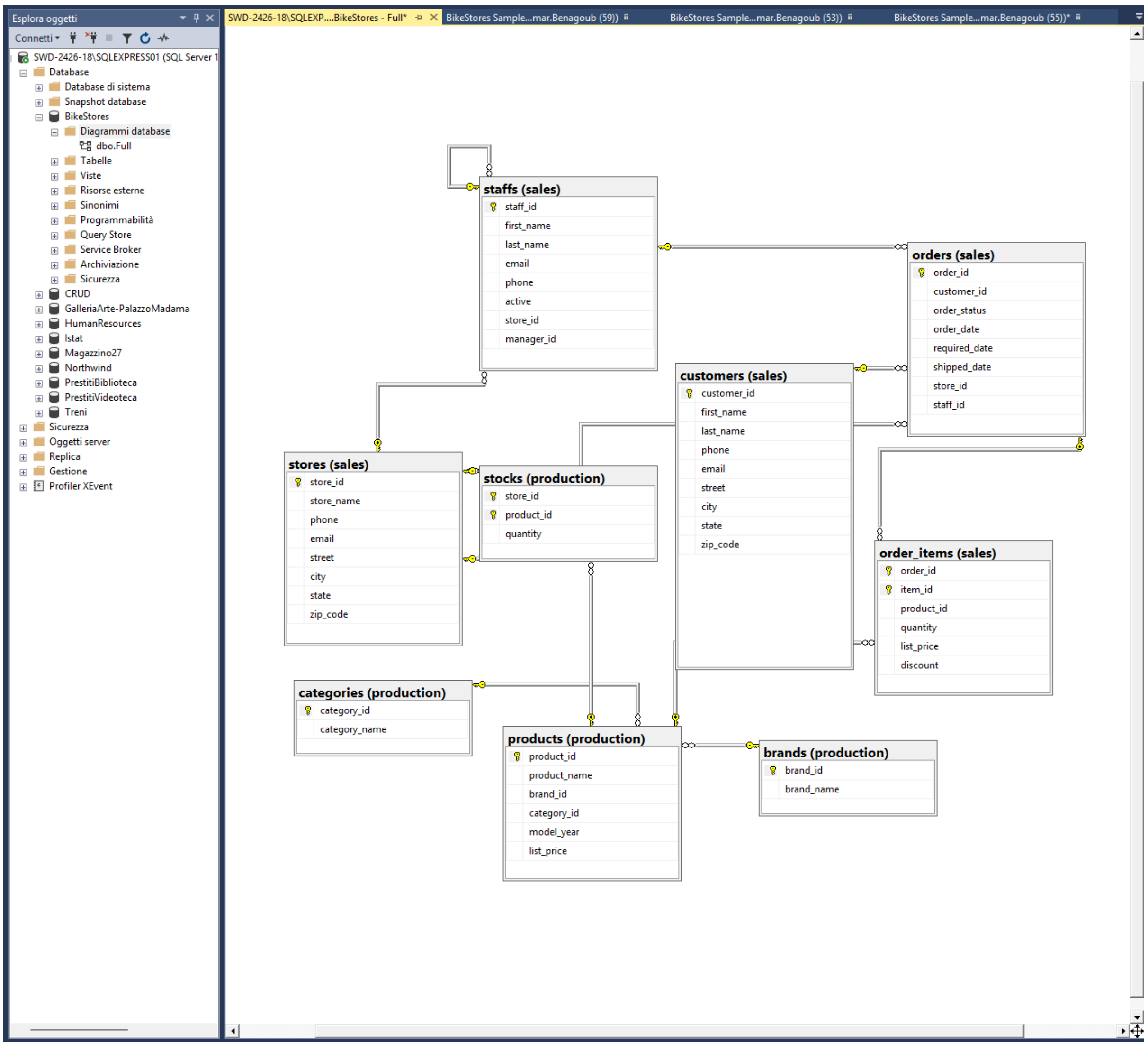
```
(1 riga interessata)

(1 riga interessata)

(1 riga interessata)

(1 riga interessata)

(1 riga interessata)

(1 riga interessata)

(1 riga interessata)

(1 riga interessata)

(1 riga interessata)
```

```
Esecuzione della query completata.   SWD-2426-18\SQLEXPRESS01 (1...   EDU-ITS\Omar.Benagoub ...   BikeStores   00:00:03   Righe 0
```

Creare i seguenti diagrammi:

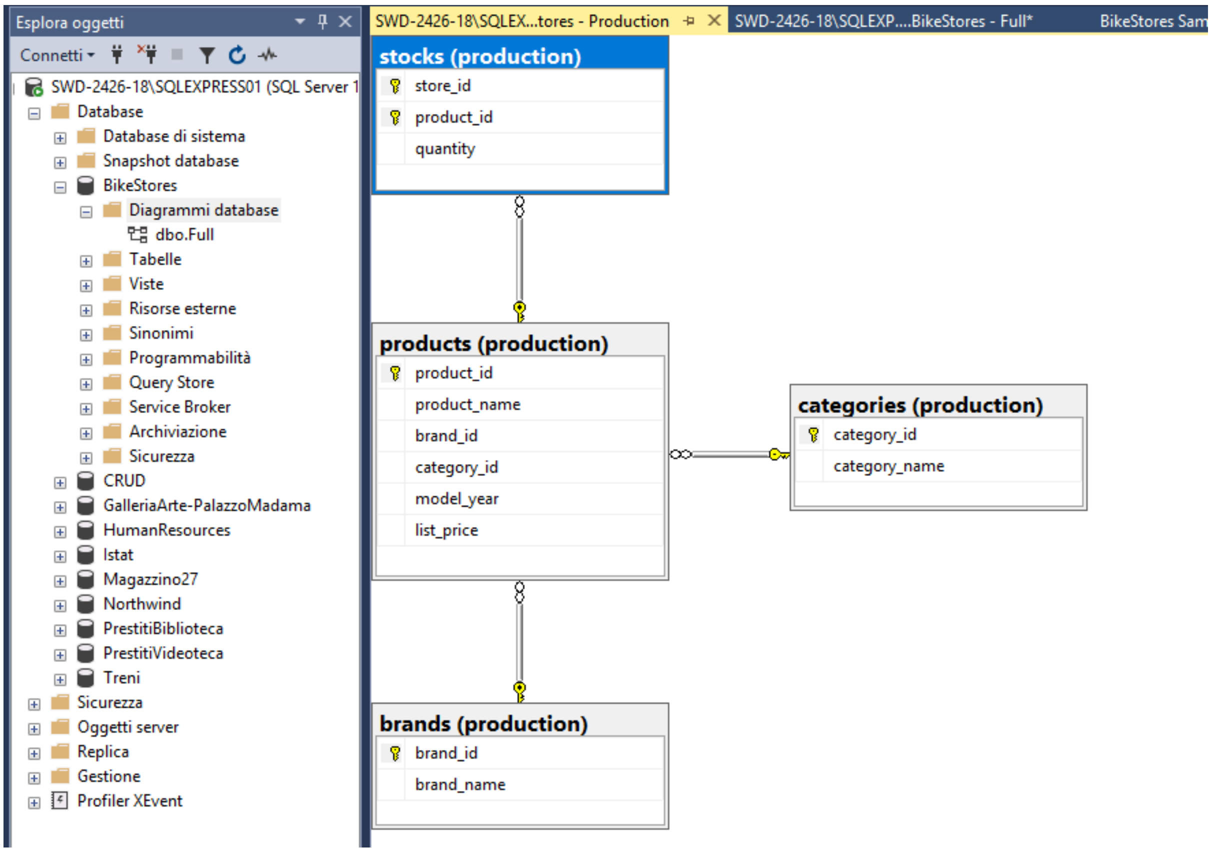
1. Full - fa riferimento a tutte le tabelle del database

Diagramma Full



2. Production - fa riferimento alle tabelle dello schema production

Diagramma Production


Esplora oggetti
Connetti
SWD-2426-18\SQLEX...tores - Production
SWD-2426-18\SQLEXP....BikeStores - Full*
BikeStores Sam

SWD-2426-18\SQLEXPRESS01 (SQL Server 1
- Database
  - Database di sistema
  - Snapshot database
  - BikeStores
    - Diagrammi database
      - dbo.Full
    - Tabelle
    - Viste
    - Risorse esterne
    - Sinonimi
    - Programmabilità
    - Query Store
    - Service Broker
    - Archiviazione
    - Sicurezza
  - CRUD
  - GalleriaArte-PalazzoMadama
  - HumanResources
  - Istat
  - Magazzino27
  - Northwind
  - PrestitiBiblioteca
  - PrestitiVideoteca
  - Treni
- Sicurezza
- Oggetti server
- Replica
- Gestione
- Profiler XEvent

stocks (production)
store_id
product_id
quantity

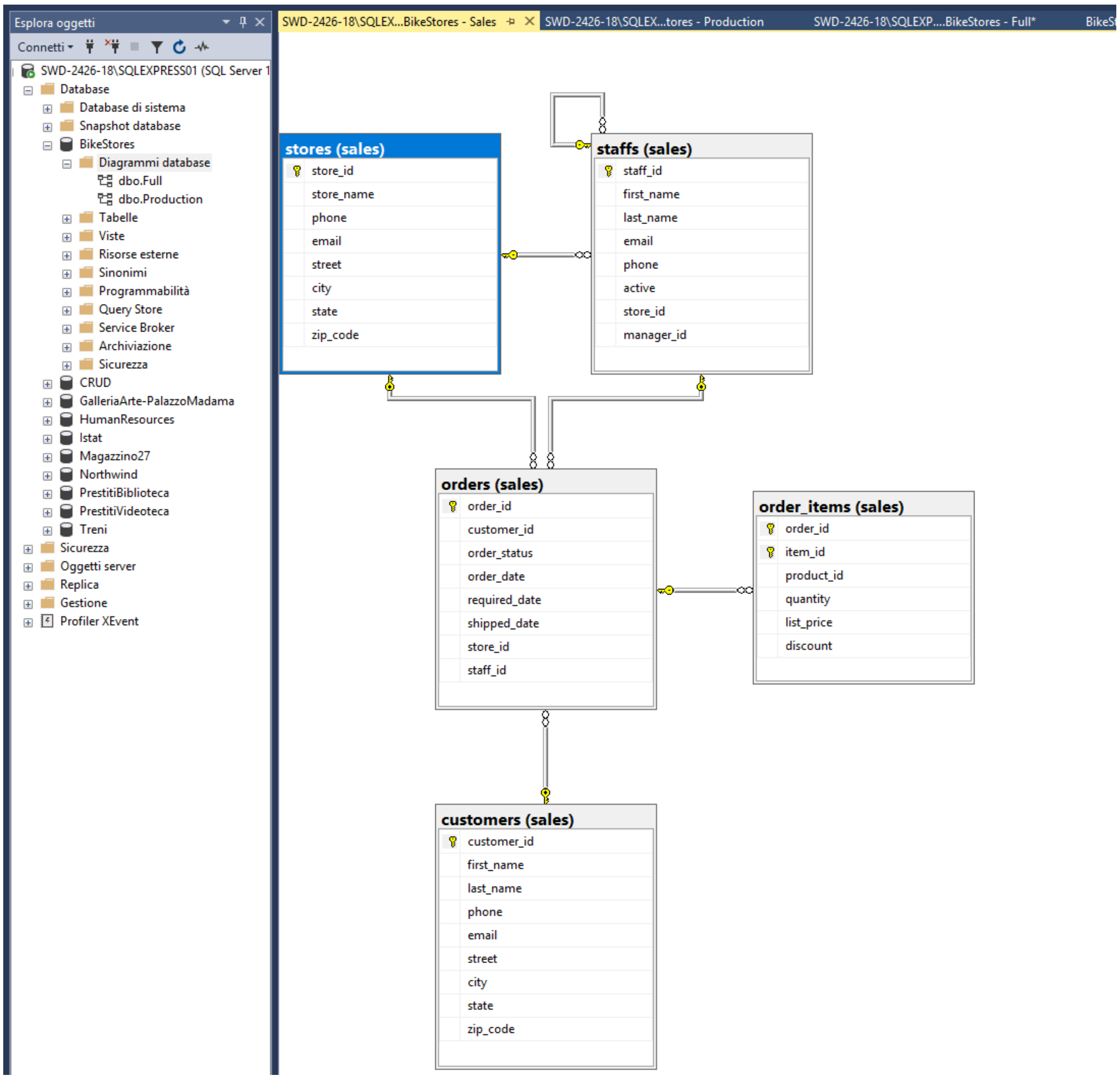
products (production)
product_id
product_name
brand_id
category_id
model_year
list_price

categories (production)
category_id
category_name

brands (production)
brand_id
brand_name

3. Sales - fa riferimento alle tabelle dello schema sales

   Diagramma Sales



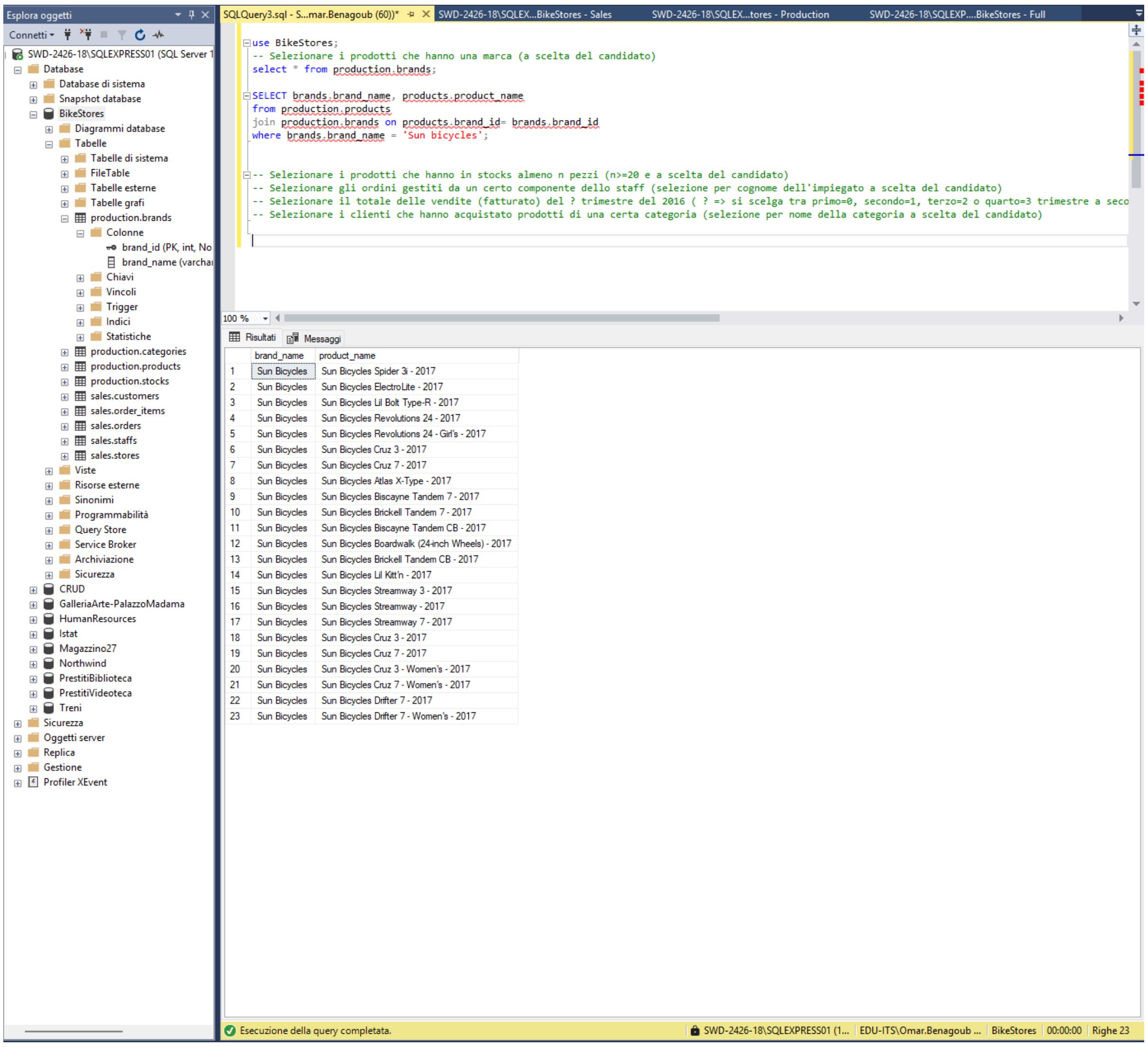
Eseguire le seguenti queries

1. Selezionare i prodotti che hanno una marca (a scelta del candidato)

   SELECT brands.brand_name, products.product_name

   from production.products

   join production.brands on products.brand_id= brands.brand_id where brands.brand_name = 'Sun bicycles';



```
use BikeStores;
-- Selezionare i prodotti che hanno una marca (a scelta del candidato)
select * from production.brands;

SELECT brands.brand_name, products.product_name
from production.products
join production.brands on products.brand_id= brands.brand_id
where brands.brand_name = 'Sun bicycles';


-- Selezionare i prodotti che hanno in stocks almeno n pezzi (n>=20 e a scelta del candidato)
-- Selezionare gli ordini gestiti da un certo componente dello staff (selezione per cognome dell'impiegato a scelta del candidato)
-- Selezionare il totale delle vendite (fatturato) del ? trimestre del 2016 ( ? => si scelga tra primo=0, secondo=1, terzo=2 o quarto=3 trimestre a seco
-- Selezionare i clienti che hanno acquistato prodotti di una certa categoria (selezione per nome della categoria a scelta del candidato)
```

	brand_name	product_name
1	Sun Bicycles	Sun Bicycles Spider 3i - 2017
2	Sun Bicycles	Sun Bicycles ElectroLite - 2017
3	Sun Bicycles	Sun Bicycles Lil Bolt Type-R - 2017
4	Sun Bicycles	Sun Bicycles Revolutions 24 - 2017
5	Sun Bicycles	Sun Bicycles Revolutions 24 - Girl's - 2017
6	Sun Bicycles	Sun Bicycles Cruz 3 - 2017
7	Sun Bicycles	Sun Bicycles Cruz 7 - 2017
8	Sun Bicycles	Sun Bicycles Atlas X-Type - 2017
9	Sun Bicycles	Sun Bicycles Biscayne Tandem 7 - 2017
10	Sun Bicycles	Sun Bicycles Brickell Tandem 7 - 2017
11	Sun Bicycles	Sun Bicycles Biscayne Tandem CB - 2017
12	Sun Bicycles	Sun Bicycles Boardwalk (24-inch Wheels) - 2017
13	Sun Bicycles	Sun Bicycles Brickell Tandem CB - 2017
14	Sun Bicycles	Sun Bicycles Lil Kitt'n - 2017
15	Sun Bicycles	Sun Bicycles Streamway 3 - 2017
16	Sun Bicycles	Sun Bicycles Streamway - 2017
17	Sun Bicycles	Sun Bicycles Streamway 7 - 2017
18	Sun Bicycles	Sun Bicycles Cruz 3 - 2017
19	Sun Bicycles	Sun Bicycles Cruz 7 - 2017
20	Sun Bicycles	Sun Bicycles Cruz 3 - Women's - 2017
21	Sun Bicycles	Sun Bicycles Cruz 7 - Women's - 2017
22	Sun Bicycles	Sun Bicycles Drifter 7 - 2017
23	Sun Bicycles	Sun Bicycles Drifter 7 - Women's - 2017

Esecuzione della query completata.

2. Selezionare i prodotti che hanno in stocks almeno n pezzi (n>=20 e a scelta del candidato)

SELECT production.products.product_id,

production.products.product_name,

production.stocks.quantity,

sales.stores.store_name

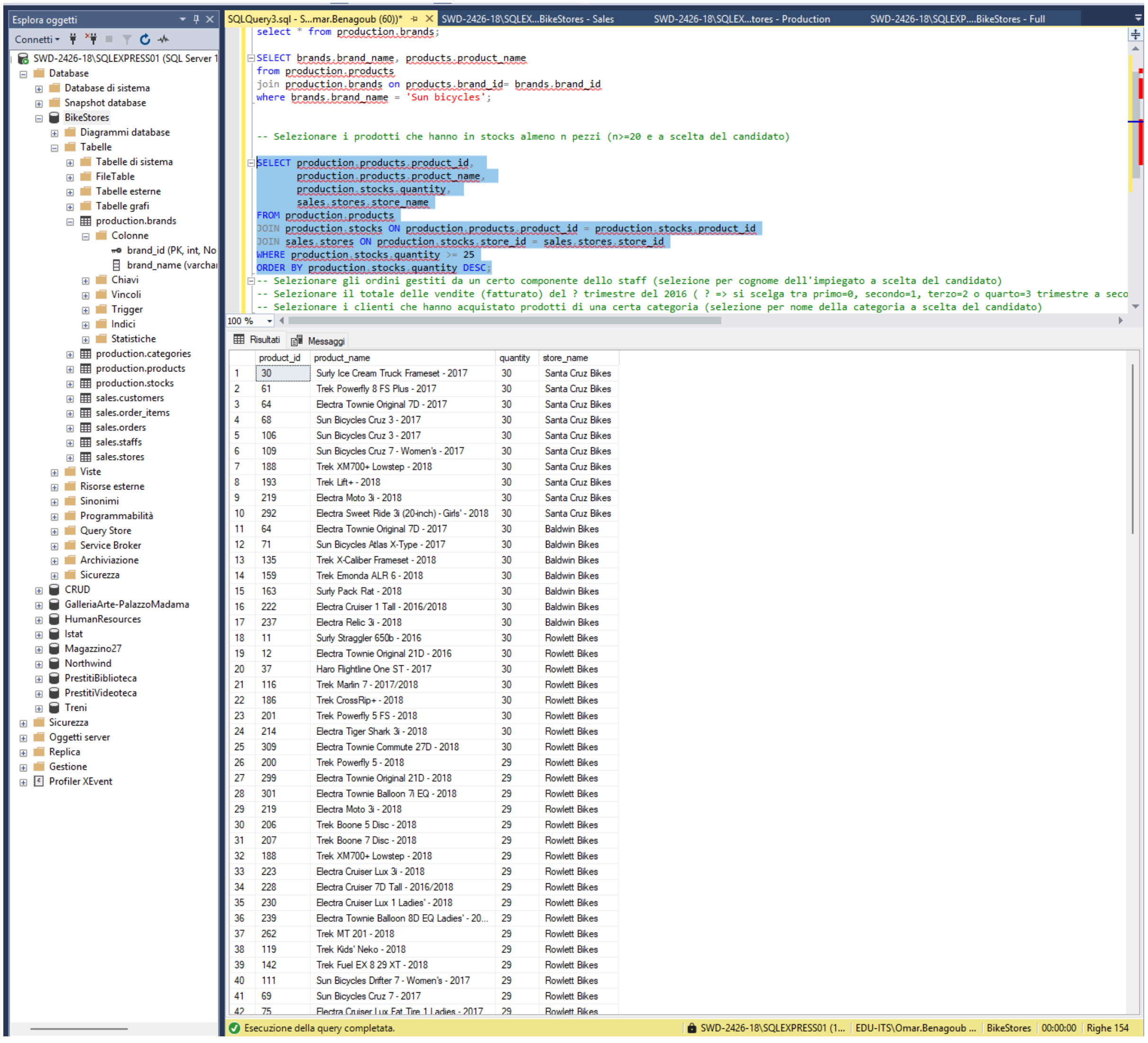
FROM production.products

JOIN production.stocks ON production.products.product_id = production.stocks.product_id

JOIN sales.stores ON production.stocks.store_id = sales.stores.store_id

WHERE production.stocks.quantity >= 25

ORDER BY production.stocks.quantity DESC;



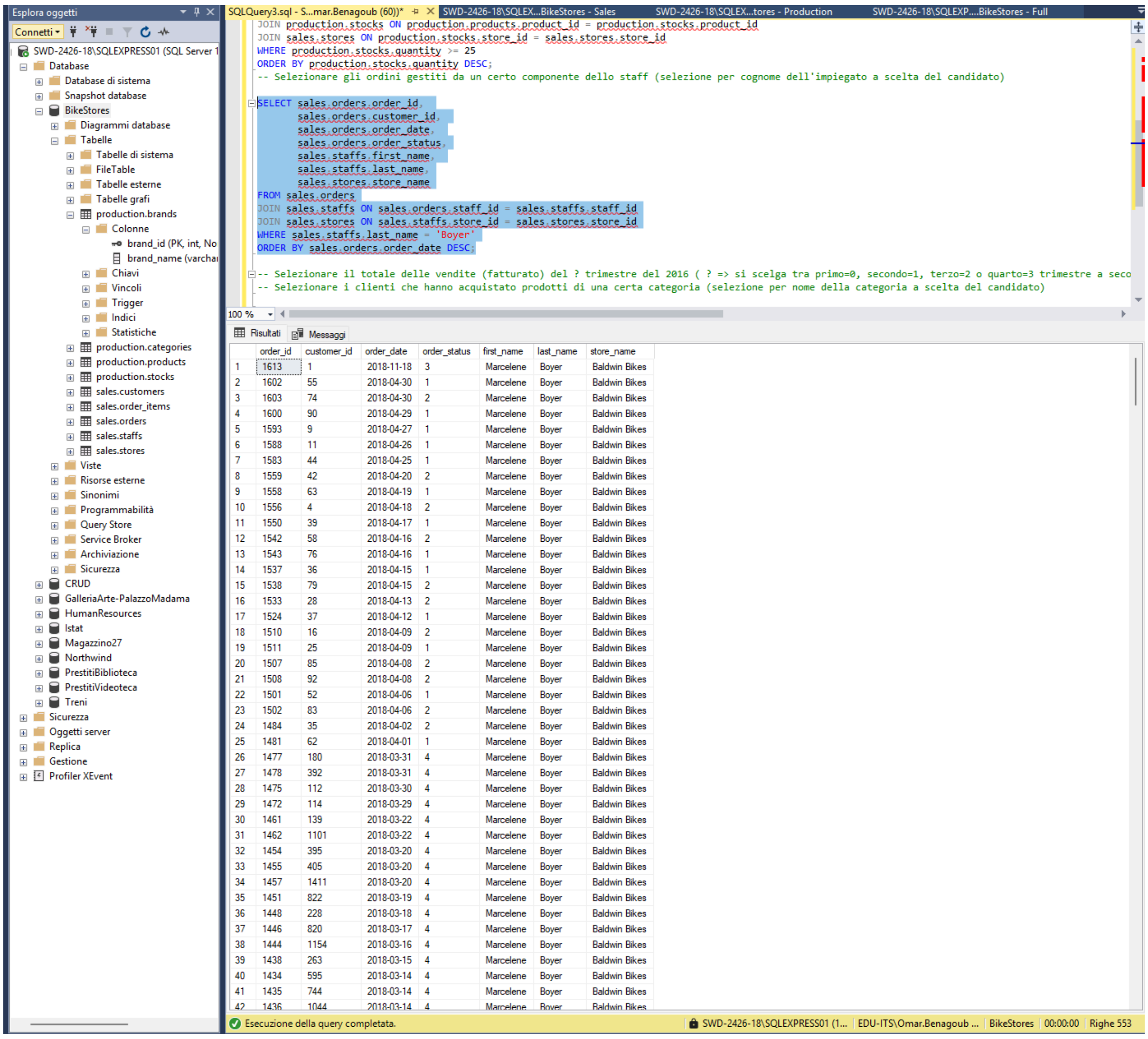
```
select * from production.brands;

SELECT brands.brand_name, products.product_name
from production.products
join production.brands on products.brand_id= brands.brand_id
where brands.brand_name = 'Sun bicycles';


-- Selezionare i prodotti che hanno in stocks almeno n pezzi (n>=20 e a scelta del candidato)

SELECT production.products.product_id,
       production.products.product_name,
       production.stocks.quantity,
       sales.stores.store_name
FROM production.products
JOIN production.stocks ON production.products.product_id = production.stocks.product_id
JOIN sales.stores ON production.stocks.store_id = sales.stores.store_id
WHERE production.stocks.quantity >= 25
ORDER BY production.stocks.quantity DESC;
-- Selezionare gli ordini gestiti da un certo componente dello staff (selezione per cognome dell'impiegato a scelta del candidato)
-- Selezionare il totale delle vendite (fatturato) del ? trimestre del 2016 ( ? => si scelga tra primo=0, secondo=1, terzo=2 o quarto=3 trimestre a seco
-- Selezionare i clienti che hanno acquistato prodotti di una certa categoria (selezione per nome della categoria a scelta del candidato)
```

	product_id	product_name	quantity	store_name
1	30	Surly Ice Cream Truck Frameset - 2017	30	Santa Cruz Bikes
2	61	Trek Powerfly 8 FS Plus - 2017	30	Santa Cruz Bikes
3	64	Electra Townie Original 7D - 2017	30	Santa Cruz Bikes
4	68	Sun Bicycles Cruz 3 - 2017	30	Santa Cruz Bikes
5	106	Sun Bicycles Cruz 3 - 2017	30	Santa Cruz Bikes
6	109	Sun Bicycles Cruz 7 - Women's - 2017	30	Santa Cruz Bikes
7	188	Trek XM700+ Lowstep - 2018	30	Santa Cruz Bikes
8	193	Trek Lift+ - 2018	30	Santa Cruz Bikes
9	219	Electra Moto 3i - 2018	30	Santa Cruz Bikes
10	292	Electra Sweet Ride 3i (20-inch) - Girls' - 2018	30	Santa Cruz Bikes
11	64	Electra Townie Original 7D - 2017	30	Baldwin Bikes
12	71	Sun Bicycles Atlas X-Type - 2017	30	Baldwin Bikes
13	135	Trek X-Caliber Frameset - 2018	30	Baldwin Bikes
14	159	Trek Emonda ALR 6 - 2018	30	Baldwin Bikes
15	163	Surly Pack Rat - 2018	30	Baldwin Bikes
16	222	Electra Cruiser 1 Tall - 2016/2018	30	Baldwin Bikes
17	237	Electra Relic 3i - 2018	30	Baldwin Bikes
18	11	Surly Straggler 650b - 2016	30	Rowlett Bikes
19	12	Electra Townie Original 21D - 2016	30	Rowlett Bikes
20	37	Haro Flightline One ST - 2017	30	Rowlett Bikes
21	116	Trek Marlin 7 - 2017/2018	30	Rowlett Bikes
22	186	Trek CrossRip+ - 2018	30	Rowlett Bikes
23	201	Trek Powerfly 5 FS - 2018	30	Rowlett Bikes
24	214	Electra Tiger Shark 3i - 2018	30	Rowlett Bikes
25	309	Electra Townie Commute 27D - 2018	30	Rowlett Bikes
26	200	Trek Powerfly 5 - 2018	29	Rowlett Bikes
27	299	Electra Townie Original 21D - 2018	29	Rowlett Bikes
28	301	Electra Townie Balloon 7i EQ - 2018	29	Rowlett Bikes
29	219	Electra Moto 3i - 2018	29	Rowlett Bikes
30	206	Trek Boone 5 Disc - 2018	29	Rowlett Bikes
31	207	Trek Boone 7 Disc - 2018	29	Rowlett Bikes
32	188	Trek XM700+ Lowstep - 2018	29	Rowlett Bikes
33	223	Electra Cruiser Lux 3i - 2018	29	Rowlett Bikes
34	228	Electra Cruiser 7D Tall - 2016/2018	29	Rowlett Bikes
35	230	Electra Cruiser Lux 1 Ladies' - 2018	29	Rowlett Bikes
36	239	Electra Townie Balloon 8D EQ Ladies' - 20...	29	Rowlett Bikes
37	262	Trek MT 201 - 2018	29	Rowlett Bikes
38	119	Trek Kids' Neko - 2018	29	Rowlett Bikes
39	142	Trek Fuel EX 8 29 XT - 2018	29	Rowlett Bikes
40	111	Sun Bicycles Drifter 7 - Women's - 2017	29	Rowlett Bikes
41	69	Sun Bicycles Cruz 7 - 2017	29	Rowlett Bikes
42	75	Electra Cruiser Lux Fat Tire 1 Ladies - 2017	29	Rowlett Bikes

Esecuzione della query completata. | SWD-2426-18\SQLEXPRESS01 (1... | EDU-ITS\Omar.Benagoub ... | BikeStores | 00:00:00 | Righe 154

3. Selezionare gli ordini gestiti da un certo componente dello staff (selezione per cognome dell'impiegato a scelta del candidato)

```
SELECT sales.orders.order_id,

      sales.orders.customer_id,

      sales.orders.order_date,

      sales.orders.order_status,

      sales.staffs.first_name,

      sales.staffs.last_name,

      sales.stores.store_name

FROM sales.orders

JOIN sales.staffs ON sales.orders.staff_id = sales.staffs.staff_id

JOIN sales.stores ON sales.staffs.store_id = sales.stores.store_id

WHERE sales.staffs.last_name = 'Boyer'

ORDER BY sales.orders.order_date DESC;
```



```sql
JOIN production.stocks ON production.products.product_id = production.stocks.product_id
JOIN sales.stores ON production.stocks.store_id = sales.stores.store_id
WHERE production.stocks.quantity >= 25
ORDER BY production.stocks.quantity DESC;
-- Selezionare gli ordini gestiti da un certo componente dello staff (selezione per cognome dell'impiegato a scelta del candidato)

SELECT sales.orders.order_id,
       sales.orders.customer_id,
       sales.orders.order_date,
       sales.orders.order_status,
       sales.staffs.first_name,
       sales.staffs.last_name,
       sales.stores.store_name
FROM sales.orders
JOIN sales.staffs ON sales.orders.staff_id = sales.staffs.staff_id
JOIN sales.stores ON sales.staffs.store_id = sales.stores.store_id
WHERE sales.staffs.last_name = 'Boyer'
ORDER BY sales.orders.order_date DESC;

-- Selezionare il totale delle vendite (fatturato) del ? trimestre del 2016 ( ? => si scelga tra primo=0, secondo=1, terzo=2 o quarto=3 trimestre a seco
-- Selezionare i clienti che hanno acquistato prodotti di una certa categoria (selezione per nome della categoria a scelta del candidato)
```

	order_id	customer_id	order_date	order_status	first_name	last_name	store_name
1	1613	1	2018-11-18	3	Marcelene	Boyer	Baldwin Bikes
2	1602	55	2018-04-30	1	Marcelene	Boyer	Baldwin Bikes
3	1603	74	2018-04-30	2	Marcelene	Boyer	Baldwin Bikes
4	1600	90	2018-04-29	1	Marcelene	Boyer	Baldwin Bikes
5	1593	9	2018-04-27	1	Marcelene	Boyer	Baldwin Bikes
6	1588	11	2018-04-26	1	Marcelene	Boyer	Baldwin Bikes
7	1583	44	2018-04-25	1	Marcelene	Boyer	Baldwin Bikes
8	1559	42	2018-04-20	2	Marcelene	Boyer	Baldwin Bikes
9	1558	63	2018-04-19	1	Marcelene	Boyer	Baldwin Bikes
10	1556	4	2018-04-18	2	Marcelene	Boyer	Baldwin Bikes
11	1550	39	2018-04-17	1	Marcelene	Boyer	Baldwin Bikes
12	1542	58	2018-04-16	2	Marcelene	Boyer	Baldwin Bikes
13	1543	76	2018-04-16	1	Marcelene	Boyer	Baldwin Bikes
14	1537	36	2018-04-15	1	Marcelene	Boyer	Baldwin Bikes
15	1538	79	2018-04-15	2	Marcelene	Boyer	Baldwin Bikes
16	1533	28	2018-04-13	2	Marcelene	Boyer	Baldwin Bikes
17	1524	37	2018-04-12	1	Marcelene	Boyer	Baldwin Bikes
18	1510	16	2018-04-09	2	Marcelene	Boyer	Baldwin Bikes
19	1511	25	2018-04-09	1	Marcelene	Boyer	Baldwin Bikes
20	1507	85	2018-04-08	2	Marcelene	Boyer	Baldwin Bikes
21	1508	92	2018-04-08	2	Marcelene	Boyer	Baldwin Bikes
22	1501	52	2018-04-06	1	Marcelene	Boyer	Baldwin Bikes
23	1502	83	2018-04-06	2	Marcelene	Boyer	Baldwin Bikes
24	1484	35	2018-04-02	2	Marcelene	Boyer	Baldwin Bikes
25	1481	62	2018-04-01	1	Marcelene	Boyer	Baldwin Bikes
26	1477	180	2018-03-31	4	Marcelene	Boyer	Baldwin Bikes
27	1478	392	2018-03-31	4	Marcelene	Boyer	Baldwin Bikes
28	1475	112	2018-03-30	4	Marcelene	Boyer	Baldwin Bikes
29	1472	114	2018-03-29	4	Marcelene	Boyer	Baldwin Bikes
30	1461	139	2018-03-22	4	Marcelene	Boyer	Baldwin Bikes
31	1462	1101	2018-03-22	4	Marcelene	Boyer	Baldwin Bikes
32	1454	395	2018-03-20	4	Marcelene	Boyer	Baldwin Bikes
33	1455	405	2018-03-20	4	Marcelene	Boyer	Baldwin Bikes
34	1457	1411	2018-03-20	4	Marcelene	Boyer	Baldwin Bikes
35	1451	822	2018-03-19	4	Marcelene	Boyer	Baldwin Bikes
36	1448	228	2018-03-18	4	Marcelene	Boyer	Baldwin Bikes
37	1446	820	2018-03-17	4	Marcelene	Boyer	Baldwin Bikes
38	1444	1154	2018-03-16	4	Marcelene	Boyer	Baldwin Bikes
39	1438	263	2018-03-15	4	Marcelene	Boyer	Baldwin Bikes
40	1434	595	2018-03-14	4	Marcelene	Boyer	Baldwin Bikes
41	1435	744	2018-03-14	4	Marcelene	Boyer	Baldwin Bikes
42	1436	1044	2018-03-14	4	Marcelene	Boyer	Baldwin Bikes

```
Esecuzione della query completata.    SWD-2426-18\SQLEXPRESS01 (1... | EDU-ITS\Omar.Benagoub ... | BikeStores | 00:00:00 | Righe 553
```

4. Selezionare il totale delle vendite (fatturato) del ? trimestre del 2016 (? => si scelga tra primo=0, secondo=1, terzo=2 o quarto=3 trimestre a seconda del risultato della seguente formula: n % 4, dove n è la posizione del registro di classe del candidato)

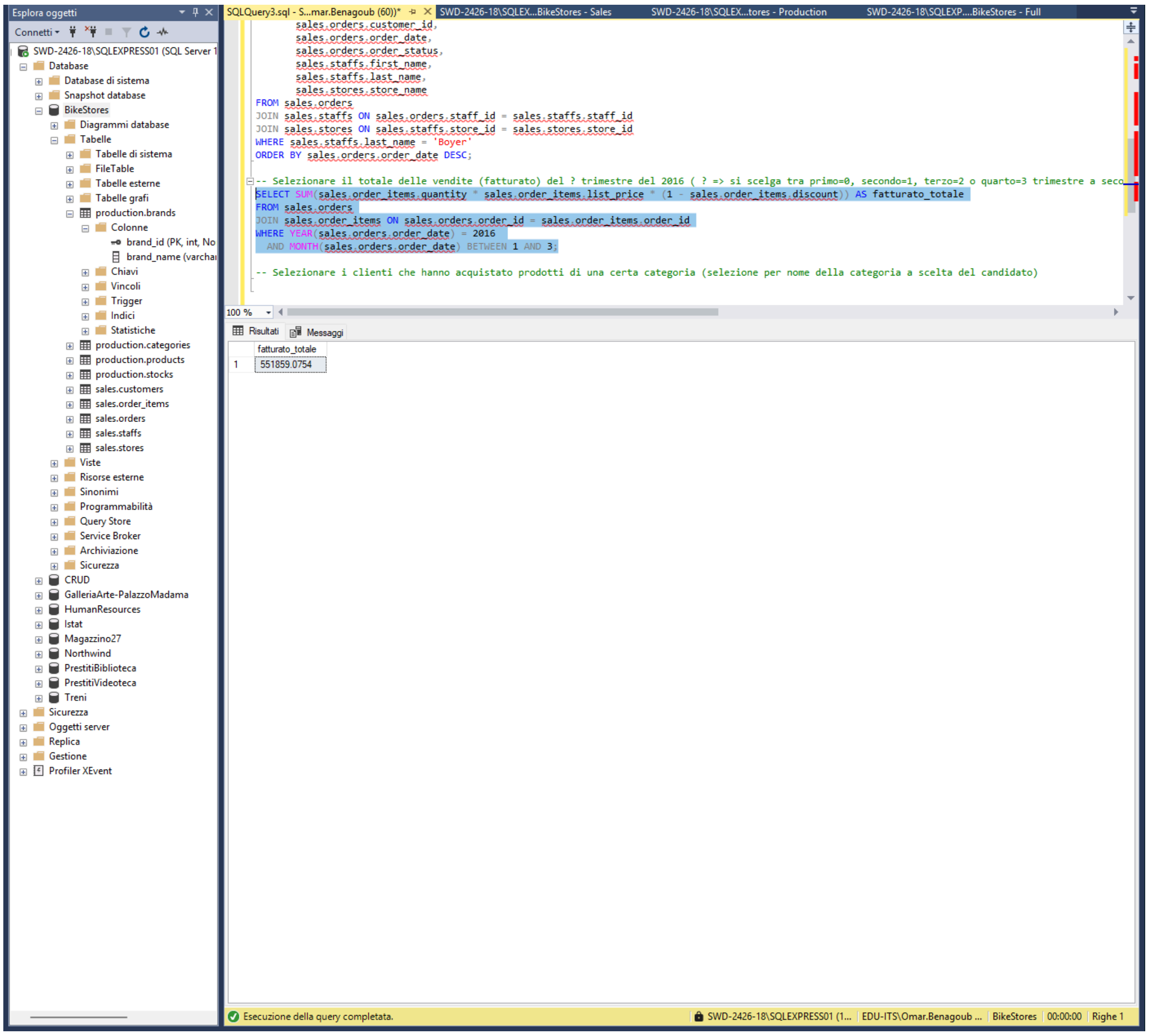
SELECT SUM(sales.order_items.quantity * sales.order_items.list_price * (1 - sales.order_items.discount)) AS fatturato_totale

FROM sales.orders

JOIN sales.order_items ON sales.orders.order_id = sales.order_items.order_id

WHERE YEAR(sales.orders.order_date) = 2016

  AND MONTH(sales.orders.order_date) BETWEEN 1 AND 3;


Esplora oggetti
Connetti
SQLQuery3.sql - S...mar.Benagoub (60))*
SWD-2426-18\SQLEX...BikeStores - Sales
SWD-2426-18\SQLEX...tores - Production
SWD-2426-18\SQLEXP....BikeStores - Full
100 %
Risultati
Messaggi

```
SWD-2426-18\SQLEXPRESS01 (SQL Server 1
  Database
    Database di sistema
    Snapshot database
    BikeStores
      Diagrammi database
      Tabelle
        Tabelle di sistema
        FileTable
        Tabelle esterne
        Tabelle grafi
        production.brands
          Colonne
            brand_id (PK, int, No
            brand_name (varchar
          Chiavi
          Vincoli
          Trigger
          Indici
          Statistiche
        production.categories
        production.products
        production.stocks
        sales.customers
        sales.order_items
        sales.orders
        sales.staffs
        sales.stores
      Viste
      Risorse esterne
      Sinonimi
      Programmabilità
      Query Store
      Service Broker
      Archiviazione
      Sicurezza
    CRUD
    GalleriaArte-PalazzoMadama
    HumanResources
    Istat
    Magazzino27
    Northwind
    PrestitiBiblioteca
    PrestitiVideoteca
    Treni
  Sicurezza
  Oggetti server
  Replica
  Gestione
  Profiler XEvent
```

```sql
       sales.orders.customer_id,
       sales.orders.order_date,
       sales.orders.order_status,
       sales.staffs.first_name,
       sales.staffs.last_name,
       sales.stores.store_name
FROM sales.orders
JOIN sales.staffs ON sales.orders.staff_id = sales.staffs.staff_id
JOIN sales.stores ON sales.staffs.store_id = sales.stores.store_id
WHERE sales.staffs.last_name = 'Boyer'
ORDER BY sales.orders.order_date DESC;

-- Selezionare il totale delle vendite (fatturato) del ? trimestre del 2016 ( ? => si scelga tra primo=0, secondo=1, terzo=2 o quarto=3 trimestre a seco
SELECT SUM(sales.order_items.quantity * sales.order_items.list_price * (1 - sales.order_items.discount)) AS fatturato_totale
FROM sales.orders
JOIN sales.order_items ON sales.orders.order_id = sales.order_items.order_id
WHERE YEAR(sales.orders.order_date) = 2016
  AND MONTH(sales.orders.order_date) BETWEEN 1 AND 3;

-- Selezionare i clienti che hanno acquistato prodotti di una certa categoria (selezione per nome della categoria a scelta del candidato)
```

	fatturato_totale
1	551859.0754

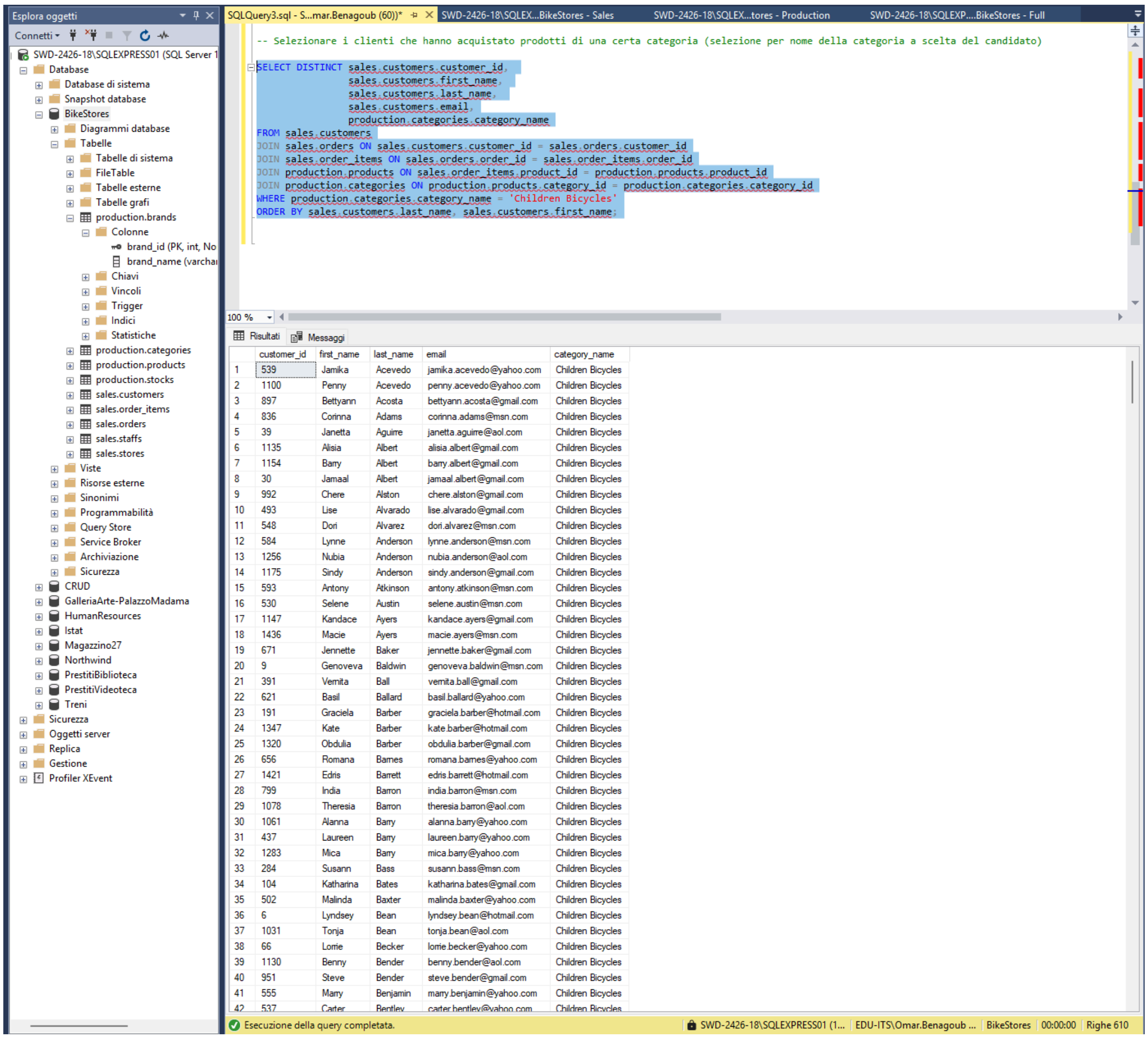
Esecuzione della query completata. | SWD-2426-18\SQLEXPRESS01 (1... | EDU-ITS\Omar.Benagoub ... | BikeStores | 00:00:00 | Righe 1

5. Selezionare i clienti che hanno acquistato prodotti di una certa categoria (selezione per nome della categoria a scelta del candidato)

SELECT DISTINCT sales.customers.customer_id,

        sales.customers.first_name,

```
            sales.customers.last_name,

            sales.customers.email,

            production.categories.category_name

FROM sales.customers

JOIN sales.orders ON sales.customers.customer_id =
sales.orders.customer_id

JOIN sales.order_items ON sales.orders.order_id =
sales.order_items.order_id

JOIN production.products ON sales.order_items.product_id =
production.products.product_id

JOIN production.categories ON production.products.category_id =
production.categories.category_id

WHERE production.categories.category_name = 'Children Bicycles'

ORDER BY sales.customers.last_name,
sales.customers.first_name;
```



```sql
-- Selezionare i clienti che hanno acquistato prodotti di una certa categoria (selezione per nome della categoria a scelta del candidato)

SELECT DISTINCT sales.customers.customer_id,
                sales.customers.first_name,
                sales.customers.last_name,
                sales.customers.email,
                production.categories.category_name
FROM sales.customers
JOIN sales.orders ON sales.customers.customer_id = sales.orders.customer_id
JOIN sales.order_items ON sales.orders.order_id = sales.order_items.order_id
JOIN production.products ON sales.order_items.product_id = production.products.product_id
JOIN production.categories ON production.products.category_id = production.categories.category_id
WHERE production.categories.category_name = 'Children Bicycles'
ORDER BY sales.customers.last_name, sales.customers.first_name;
```

	customer_id	first_name	last_name	email	category_name
1	539	Jamika	Acevedo	jamika.acevedo@yahoo.com	Children Bicycles
2	1100	Penny	Acevedo	penny.acevedo@yahoo.com	Children Bicycles
3	897	Bettyann	Acosta	bettyann.acosta@gmail.com	Children Bicycles
4	836	Corinna	Adams	corinna.adams@msn.com	Children Bicycles
5	39	Janetta	Aguirre	janetta.aguirre@aol.com	Children Bicycles
6	1135	Alisia	Albert	alisia.albert@gmail.com	Children Bicycles
7	1154	Barry	Albert	barry.albert@gmail.com	Children Bicycles
8	30	Jamaal	Albert	jamaal.albert@gmail.com	Children Bicycles
9	992	Chere	Alston	chere.alston@gmail.com	Children Bicycles
10	493	Lise	Alvarado	lise.alvarado@gmail.com	Children Bicycles
11	548	Dori	Alvarez	dori.alvarez@msn.com	Children Bicycles
12	584	Lynne	Anderson	lynne.anderson@msn.com	Children Bicycles
13	1256	Nubia	Anderson	nubia.anderson@aol.com	Children Bicycles
14	1175	Sindy	Anderson	sindy.anderson@gmail.com	Children Bicycles
15	593	Antony	Atkinson	antony.atkinson@msn.com	Children Bicycles
16	530	Selene	Austin	selene.austin@msn.com	Children Bicycles
17	1147	Kandace	Ayers	kandace.ayers@gmail.com	Children Bicycles
18	1436	Macie	Ayers	macie.ayers@msn.com	Children Bicycles
19	671	Jennette	Baker	jennette.baker@gmail.com	Children Bicycles
20	9	Genoveva	Baldwin	genoveva.baldwin@msn.com	Children Bicycles
21	391	Vernita	Ball	vernita.ball@gmail.com	Children Bicycles
22	621	Basil	Ballard	basil.ballard@yahoo.com	Children Bicycles
23	191	Graciela	Barber	graciela.barber@hotmail.com	Children Bicycles
24	1347	Kate	Barber	kate.barber@hotmail.com	Children Bicycles
25	1320	Obdulia	Barber	obdulia.barber@gmail.com	Children Bicycles
26	656	Romana	Barnes	romana.barnes@yahoo.com	Children Bicycles
27	1421	Edris	Barrett	edris.barrett@hotmail.com	Children Bicycles
28	799	India	Barron	india.barron@msn.com	Children Bicycles
29	1078	Theresia	Barron	theresia.barron@aol.com	Children Bicycles
30	1061	Alanna	Barry	alanna.barry@yahoo.com	Children Bicycles
31	437	Laureen	Barry	laureen.barry@yahoo.com	Children Bicycles
32	1283	Mica	Barry	mica.barry@yahoo.com	Children Bicycles
33	284	Susann	Bass	susann.bass@msn.com	Children Bicycles
34	104	Katharina	Bates	katharina.bates@gmail.com	Children Bicycles
35	502	Malinda	Baxter	malinda.baxter@yahoo.com	Children Bicycles
36	6	Lyndsey	Bean	lyndsey.bean@hotmail.com	Children Bicycles
37	1031	Tonja	Bean	tonja.bean@aol.com	Children Bicycles
38	66	Lorrie	Becker	lorrie.becker@yahoo.com	Children Bicycles
39	1130	Benny	Bender	benny.bender@aol.com	Children Bicycles
40	951	Steve	Bender	steve.bender@gmail.com	Children Bicycles
41	555	Marry	Benjamin	marry.benjamin@yahoo.com	Children Bicycles
42	537	Carter	Bentley	carter.bentley@yahoo.com	Children Bicycles

```
Esecuzione della query completata.    SWD-2426-18\SQLEXPRESS01 (1... | EDU-ITS\Omar.Benagoub ... | BikeStores | 00:00:00 | Righe 610
```

Creare le seguenti views

1. Creare la vista vw_ProductionByQuantity con category_name, brand_name, product_name, model_year, list_price, quantity dello schema production. Interrogare la view richiedendo solo i dati con quantity strettamente inferiori a n unità (con n a scelta del candidato)

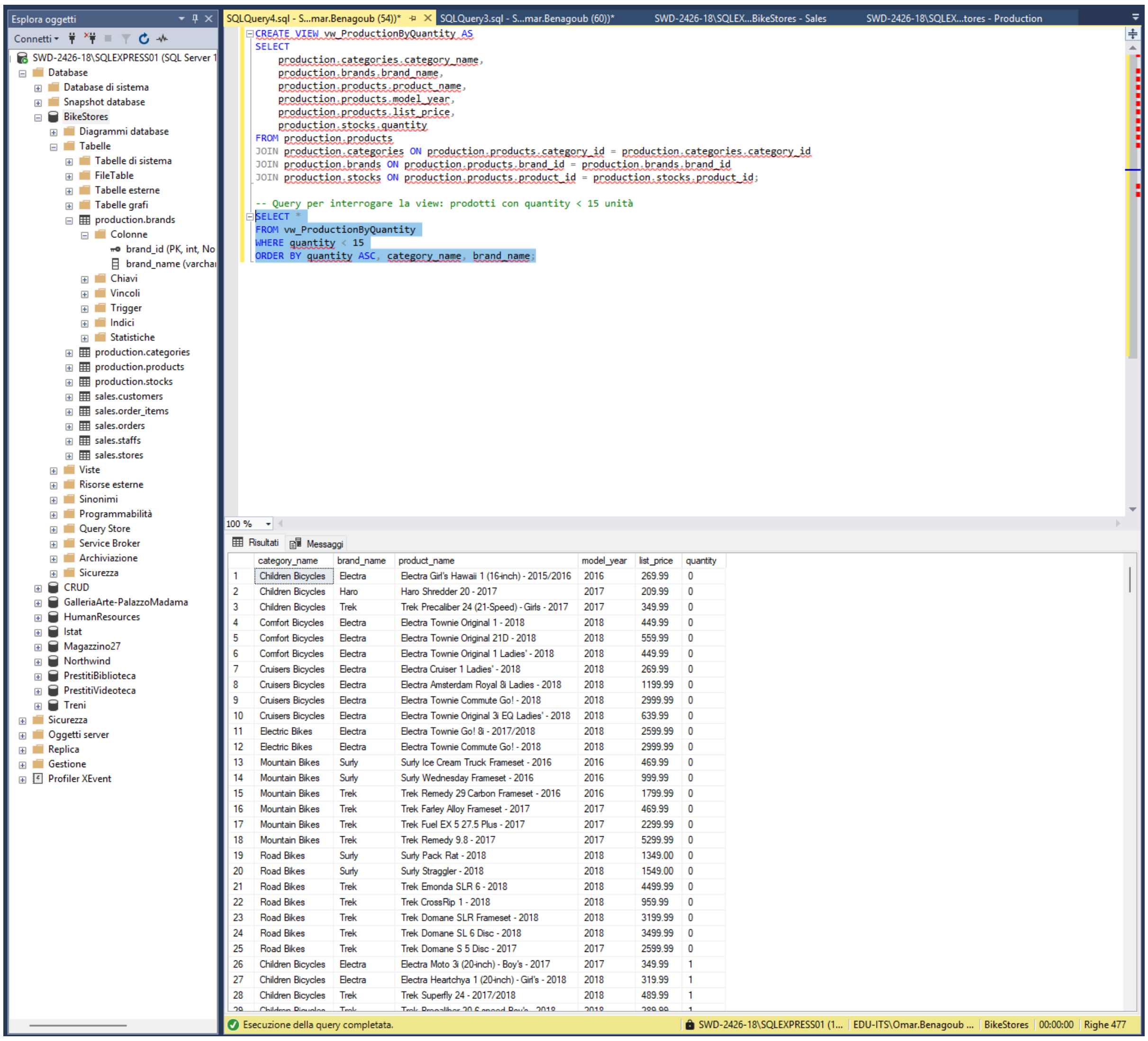
CREATE VIEW vw_ProductionByQuantity AS

SELECT

    production.categories.category_name,

    production.brands.brand_name,

```
    production.products.product_name,

    production.products.model_year,

    production.products.list_price,

    production.stocks.quantity

FROM production.products

JOIN production.categories ON production.products.category_id =
production.categories.category_id

JOIN production.brands ON production.products.brand_id =
production.brands.brand_id

JOIN production.stocks ON production.products.product_id =
production.stocks.product_id;


-- Query per interrogare la view: prodotti con quantity < 15 unità

SELECT *

FROM vw_ProductionByQuantity

WHERE quantity < 15

ORDER BY quantity ASC, category_name, brand_name;
```



```sql
CREATE VIEW vw_ProductionByQuantity AS
SELECT
    production.categories.category_name,
    production.brands.brand_name,
    production.products.product_name,
    production.products.model_year,
    production.products.list_price,
    production.stocks.quantity
FROM production.products
JOIN production.categories ON production.products.category_id = production.categories.category_id
JOIN production.brands ON production.products.brand_id = production.brands.brand_id
JOIN production.stocks ON production.products.product_id = production.stocks.product_id;

-- Query per interrogare la view: prodotti con quantity < 15 unità
SELECT *
FROM vw_ProductionByQuantity
WHERE quantity < 15
ORDER BY quantity ASC, category_name, brand_name;
```

	category_name	brand_name	product_name	model_year	list_price	quantity
1	Children Bicycles	Electra	Electra Girl's Hawaii 1 (16-inch) - 2015/2016	2016	269.99	0
2	Children Bicycles	Haro	Haro Shredder 20 - 2017	2017	209.99	0
3	Children Bicycles	Trek	Trek Precaliber 24 (21-Speed) - Girls - 2017	2017	349.99	0
4	Comfort Bicycles	Electra	Electra Townie Original 1 - 2018	2018	449.99	0
5	Comfort Bicycles	Electra	Electra Townie Original 21D - 2018	2018	559.99	0
6	Comfort Bicycles	Electra	Electra Townie Original 1 Ladies' - 2018	2018	449.99	0
7	Cruisers Bicycles	Electra	Electra Cruiser 1 Ladies' - 2018	2018	269.99	0
8	Cruisers Bicycles	Electra	Electra Amsterdam Royal 8i Ladies - 2018	2018	1199.99	0
9	Cruisers Bicycles	Electra	Electra Townie Commute Go! - 2018	2018	2999.99	0
10	Cruisers Bicycles	Electra	Electra Townie Original 3i EQ Ladies' - 2018	2018	639.99	0
11	Electric Bikes	Electra	Electra Townie Go! 8i - 2017/2018	2018	2599.99	0
12	Electric Bikes	Electra	Electra Townie Commute Go! - 2018	2018	2999.99	0
13	Mountain Bikes	Surly	Surly Ice Cream Truck Frameset - 2016	2016	469.99	0
14	Mountain Bikes	Surly	Surly Wednesday Frameset - 2016	2016	999.99	0
15	Mountain Bikes	Trek	Trek Remedy 29 Carbon Frameset - 2016	2016	1799.99	0
16	Mountain Bikes	Trek	Trek Farley Alloy Frameset - 2017	2017	469.99	0
17	Mountain Bikes	Trek	Trek Fuel EX 5 27.5 Plus - 2017	2017	2299.99	0
18	Mountain Bikes	Trek	Trek Remedy 9.8 - 2017	2017	5299.99	0
19	Road Bikes	Surly	Surly Pack Rat - 2018	2018	1349.00	0
20	Road Bikes	Surly	Surly Straggler - 2018	2018	1549.00	0
21	Road Bikes	Trek	Trek Emonda SLR 6 - 2018	2018	4499.99	0
22	Road Bikes	Trek	Trek CrossRip 1 - 2018	2018	959.99	0
23	Road Bikes	Trek	Trek Domane SLR Frameset - 2018	2018	3199.99	0
24	Road Bikes	Trek	Trek Domane SL 6 Disc - 2018	2018	3499.99	0
25	Road Bikes	Trek	Trek Domane S 5 Disc - 2017	2017	2599.99	0
26	Children Bicycles	Electra	Electra Moto 3i (20-inch) - Boy's - 2017	2017	349.99	1
27	Children Bicycles	Electra	Electra Heartchya 1 (20-inch) - Girl's - 2018	2018	319.99	1
28	Children Bicycles	Trek	Trek Superfly 24 - 2017/2018	2018	489.99	1

```
Esecuzione della query completata.    SWD-2426-18\SQLEXPRESS01 (1... | EDU-ITS\Omar.Benagoub ... | BikeStores | 00:00:00 | Righe 477
```

2. Creare la vista vw_StoresByQuantity con brand_name, product_name, store_name, city, quantity. Interrogare la view per visualizzare i dati solo di una certa città

CREATE VIEW vw_StoresByQuantity

AS SELECT   b.brand_name,

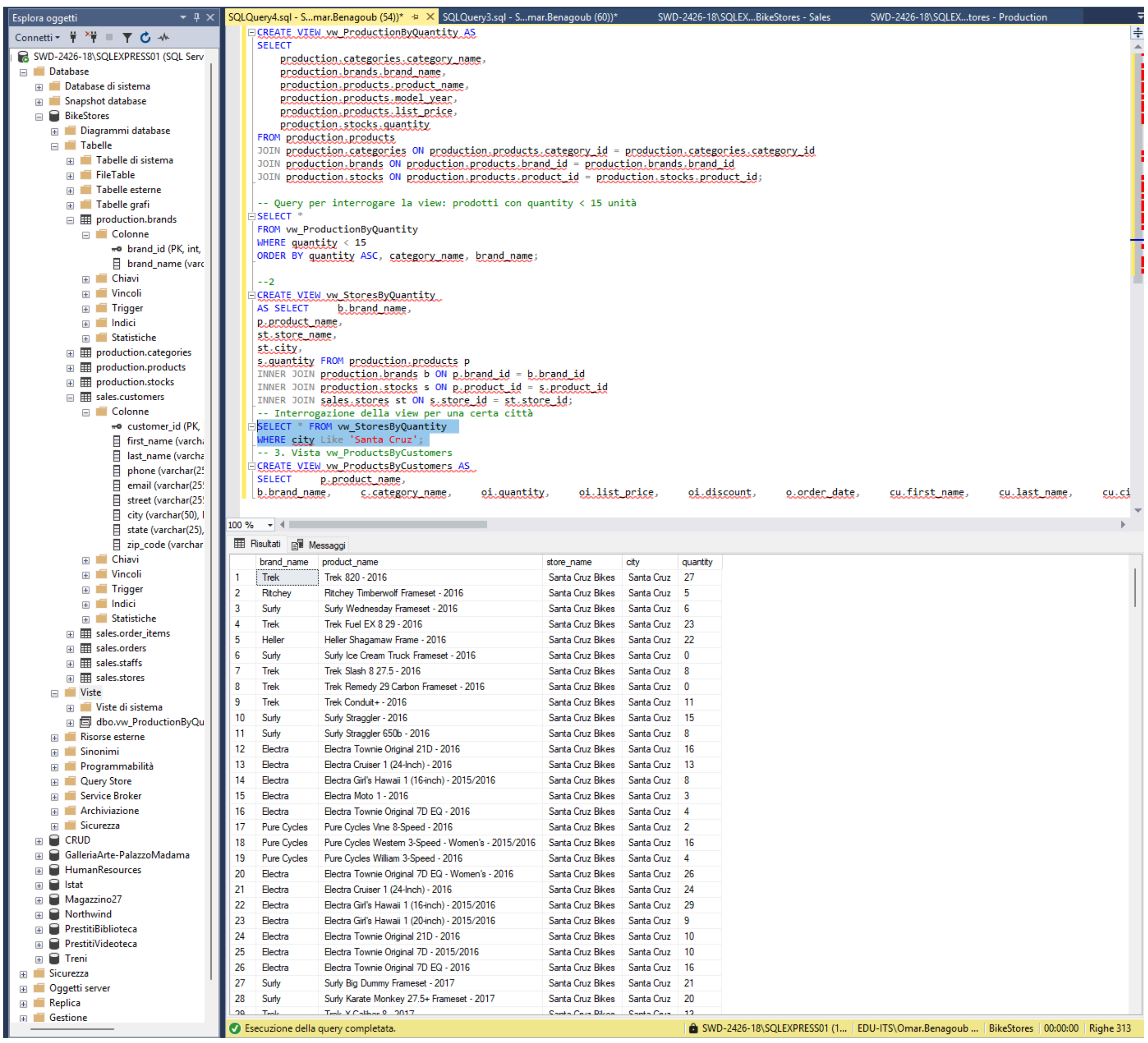
p.product_name,

st.store_name,

st.city,

s.quantity FROM production.products p

```
INNER JOIN production.brands b ON p.brand_id = b.brand_id

INNER JOIN production.stocks s ON p.product_id = s.product_id

INNER JOIN sales.stores st ON s.store_id = st.store_id;

-- Interrogazione della view per una certa città

SELECT * FROM vw_StoresByQuantity

WHERE city Like 'Santa Cruz';
```



```
CREATE VIEW vw_ProductionByQuantity AS
SELECT
    production.categories.category_name,
    production.brands.brand_name,
    production.products.product_name,
    production.products.model_year,
    production.products.list_price,
    production.stocks.quantity
FROM production.products
JOIN production.categories ON production.products.category_id = production.categories.category_id
JOIN production.brands ON production.products.brand_id = production.brands.brand_id
JOIN production.stocks ON production.products.product_id = production.stocks.product_id;

-- Query per interrogare la view: prodotti con quantity < 15 unità
SELECT *
FROM vw_ProductionByQuantity
WHERE quantity < 15
ORDER BY quantity ASC, category_name, brand_name;

--2
CREATE VIEW vw_StoresByQuantity
AS SELECT     b.brand_name,
p.product_name,
st.store_name,
st.city,
s.quantity FROM production.products p
INNER JOIN production.brands b ON p.brand_id = b.brand_id
INNER JOIN production.stocks s ON p.product_id = s.product_id
INNER JOIN sales.stores st ON s.store_id = st.store_id;
-- Interrogazione della view per una certa città
SELECT * FROM vw_StoresByQuantity
WHERE city Like 'Santa Cruz';
-- 3. Vista vw_ProductsByCustomers
CREATE VIEW vw_ProductsByCustomers AS
SELECT     p.product_name,
b.brand_name,     c.category_name,     oi.quantity,     oi.list_price,     oi.discount,     o.order_date,     cu.first_name,     cu.last_name,     cu.ci
```

	brand_name	product_name	store_name	city	quantity
1	Trek	Trek 820 - 2016	Santa Cruz Bikes	Santa Cruz	27
2	Ritchey	Ritchey Timberwolf Frameset - 2016	Santa Cruz Bikes	Santa Cruz	5
3	Surly	Surly Wednesday Frameset - 2016	Santa Cruz Bikes	Santa Cruz	6
4	Trek	Trek Fuel EX 8 29 - 2016	Santa Cruz Bikes	Santa Cruz	23
5	Heller	Heller Shagamaw Frame - 2016	Santa Cruz Bikes	Santa Cruz	22
6	Surly	Surly Ice Cream Truck Frameset - 2016	Santa Cruz Bikes	Santa Cruz	0
7	Trek	Trek Slash 8 27.5 - 2016	Santa Cruz Bikes	Santa Cruz	8
8	Trek	Trek Remedy 29 Carbon Frameset - 2016	Santa Cruz Bikes	Santa Cruz	0
9	Trek	Trek Conduit+ - 2016	Santa Cruz Bikes	Santa Cruz	11
10	Surly	Surly Straggler - 2016	Santa Cruz Bikes	Santa Cruz	15
11	Surly	Surly Straggler 650b - 2016	Santa Cruz Bikes	Santa Cruz	8
12	Electra	Electra Townie Original 21D - 2016	Santa Cruz Bikes	Santa Cruz	16
13	Electra	Electra Cruiser 1 (24-Inch) - 2016	Santa Cruz Bikes	Santa Cruz	13
14	Electra	Electra Girl's Hawaii 1 (16-inch) - 2015/2016	Santa Cruz Bikes	Santa Cruz	8
15	Electra	Electra Moto 1 - 2016	Santa Cruz Bikes	Santa Cruz	3
16	Electra	Electra Townie Original 7D EQ - 2016	Santa Cruz Bikes	Santa Cruz	4
17	Pure Cycles	Pure Cycles Vine 8-Speed - 2016	Santa Cruz Bikes	Santa Cruz	2
18	Pure Cycles	Pure Cycles Western 3-Speed - Women's - 2015/2016	Santa Cruz Bikes	Santa Cruz	16
19	Pure Cycles	Pure Cycles William 3-Speed - 2016	Santa Cruz Bikes	Santa Cruz	4
20	Electra	Electra Townie Original 7D EQ - Women's - 2016	Santa Cruz Bikes	Santa Cruz	26
21	Electra	Electra Cruiser 1 (24-Inch) - 2016	Santa Cruz Bikes	Santa Cruz	24
22	Electra	Electra Girl's Hawaii 1 (16-inch) - 2015/2016	Santa Cruz Bikes	Santa Cruz	29
23	Electra	Electra Girl's Hawaii 1 (20-inch) - 2015/2016	Santa Cruz Bikes	Santa Cruz	9
24	Electra	Electra Townie Original 21D - 2016	Santa Cruz Bikes	Santa Cruz	10
25	Electra	Electra Townie Original 7D - 2015/2016	Santa Cruz Bikes	Santa Cruz	10
26	Electra	Electra Townie Original 7D EQ - 2016	Santa Cruz Bikes	Santa Cruz	16
27	Surly	Surly Big Dummy Frameset - 2017	Santa Cruz Bikes	Santa Cruz	21
28	Surly	Surly Karate Monkey 27.5+ Frameset - 2017	Santa Cruz Bikes	Santa Cruz	20

```
Esecuzione della query completata.    SWD-2426-18\SQLEXPRESS01 (1... | EDU-ITS\Omar.Benagoub ... | BikeStores | 00:00:00 | Righe 313
```

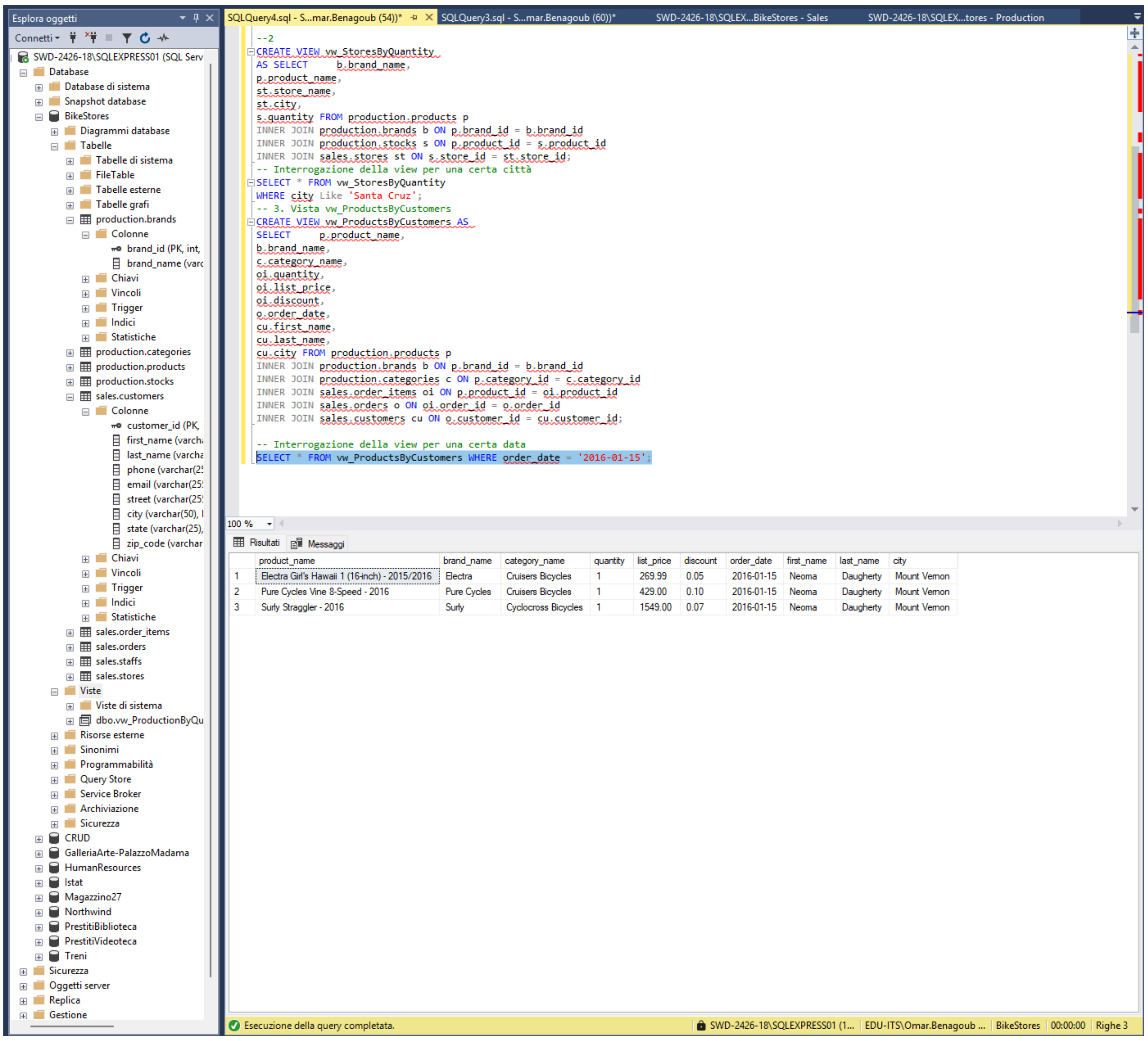
3. Creare la vista vw_ProductsByCustomers con product_name, brand_name, category_name, quantity (quantità di acquisto),

list_price (prezzo finale di acquisto), discount (sconto di acquisto), order_date, first_name, last_name, city. Interrogare la view vw_ProductsByCustomers per visualizzare i prodotti ordinati in una certa data

```
CREATE VIEW vw_ProductsByCustomers AS

SELECT     p.product_name,

b.brand_name,

c.category_name,

oi.quantity,

oi.list_price,

oi.discount,

o.order_date,

cu.first_name,

cu.last_name,

cu.city FROM production.products p

INNER JOIN production.brands b ON p.brand_id = b.brand_id

INNER JOIN production.categories c ON p.category_id =
c.category_id

INNER JOIN sales.order_items oi ON p.product_id =
oi.product_id

INNER JOIN sales.orders o ON oi.order_id = o.order_id

INNER JOIN sales.customers cu ON o.customer_id =
cu.customer_id;
```

-- Interrogazione della view per una certa data

SELECT * FROM vw_ProductsByCustomers WHERE order_date = '2016-01-15';


SQLQuery4.sql - S...mar.Benagoub (54))*
SQLQuery3.sql - S...mar.Benagoub (60))*
SWD-2426-18\SQLEX...BikeStores - Sales
SWD-2426-18\SQLEX...tores - Production
Esplora oggetti
Risultati
Messaggi

```sql
--2
CREATE VIEW vw_StoresByQuantity
AS SELECT     b.brand_name,
p.product_name,
st.store_name,
st.city,
s.quantity FROM production.products p
INNER JOIN production.brands b ON p.brand_id = b.brand_id
INNER JOIN production.stocks s ON p.product_id = s.product_id
INNER JOIN sales.stores st ON s.store_id = st.store_id;
-- Interrogazione della view per una certa città
SELECT * FROM vw_StoresByQuantity
WHERE city Like 'Santa Cruz';
-- 3. Vista vw_ProductsByCustomers
CREATE VIEW vw_ProductsByCustomers AS
SELECT     p.product_name,
b.brand_name,
c.category_name,
oi.quantity,
oi.list_price,
oi.discount,
o.order_date,
cu.first_name,
cu.last_name,
cu.city FROM production.products p
INNER JOIN production.brands b ON p.brand_id = b.brand_id
INNER JOIN production.categories c ON p.category_id = c.category_id
INNER JOIN sales.order_items oi ON p.product_id = oi.product_id
INNER JOIN sales.orders o ON oi.order_id = o.order_id
INNER JOIN sales.customers cu ON o.customer_id = cu.customer_id;

-- Interrogazione della view per una certa data
SELECT * FROM vw_ProductsByCustomers WHERE order_date = '2016-01-15';
```

	product_name	brand_name	category_name	quantity	list_price	discount	order_date	first_name	last_name	city
1	Electra Girl's Hawaii 1 (16-inch) - 2015/2016	Electra	Cruisers Bicycles	1	269.99	0.05	2016-01-15	Neoma	Daugherty	Mount Vernon
2	Pure Cycles Vine 8-Speed - 2016	Pure Cycles	Cruisers Bicycles	1	429.00	0.10	2016-01-15	Neoma	Daugherty	Mount Vernon
3	Surly Straggler - 2016	Surly	Cyclocross Bicycles	1	1549.00	0.07	2016-01-15	Neoma	Daugherty	Mount Vernon

```
Esecuzione della query completata.   SWD-2426-18\SQLEXPRESS01 (1... | EDU-ITS\Omar.Benagoub ... | BikeStores | 00:00:00 | Righe 3
```

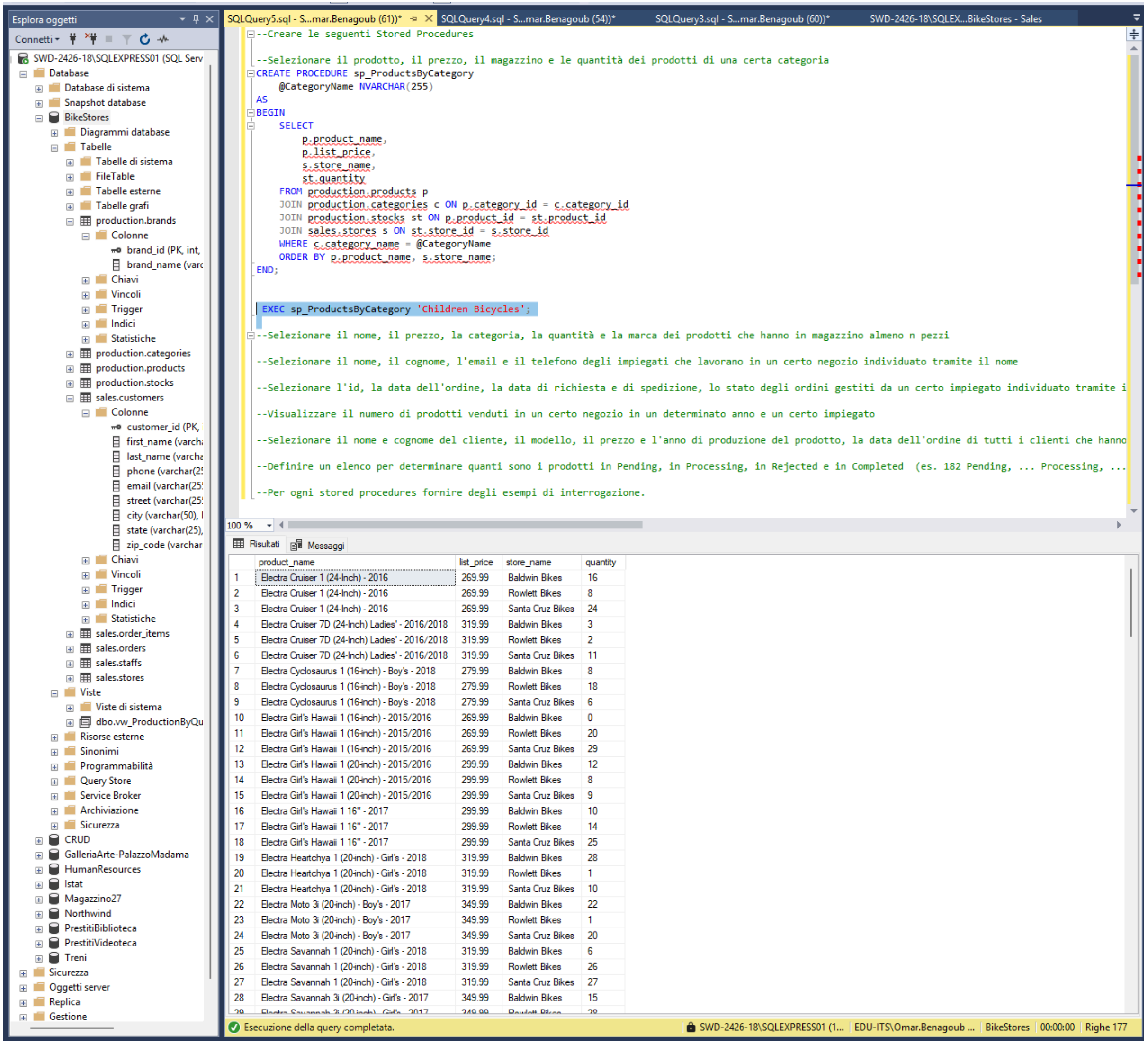
Creare le seguenti Stored Procedures

1. Selezionare il prodotto, il prezzo, il magazzino e le quantità dei prodotti di una certa categoria

CREATE PROCEDURE sp_ProductsByCategory

@CategoryName NVARCHAR(255)

```
AS
BEGIN
    SELECT
        p.product_name,
        p.list_price,
        s.store_name,
        st.quantity
    FROM production.products p
    JOIN production.categories c ON p.category_id = c.category_id
    JOIN production.stocks st ON p.product_id = st.product_id
    JOIN sales.stores s ON st.store_id = s.store_id
    WHERE c.category_name = @CategoryName
    ORDER BY p.product_name, s.store_name;
END;


 EXEC sp_ProductsByCategory 'Children Bicycles';
```



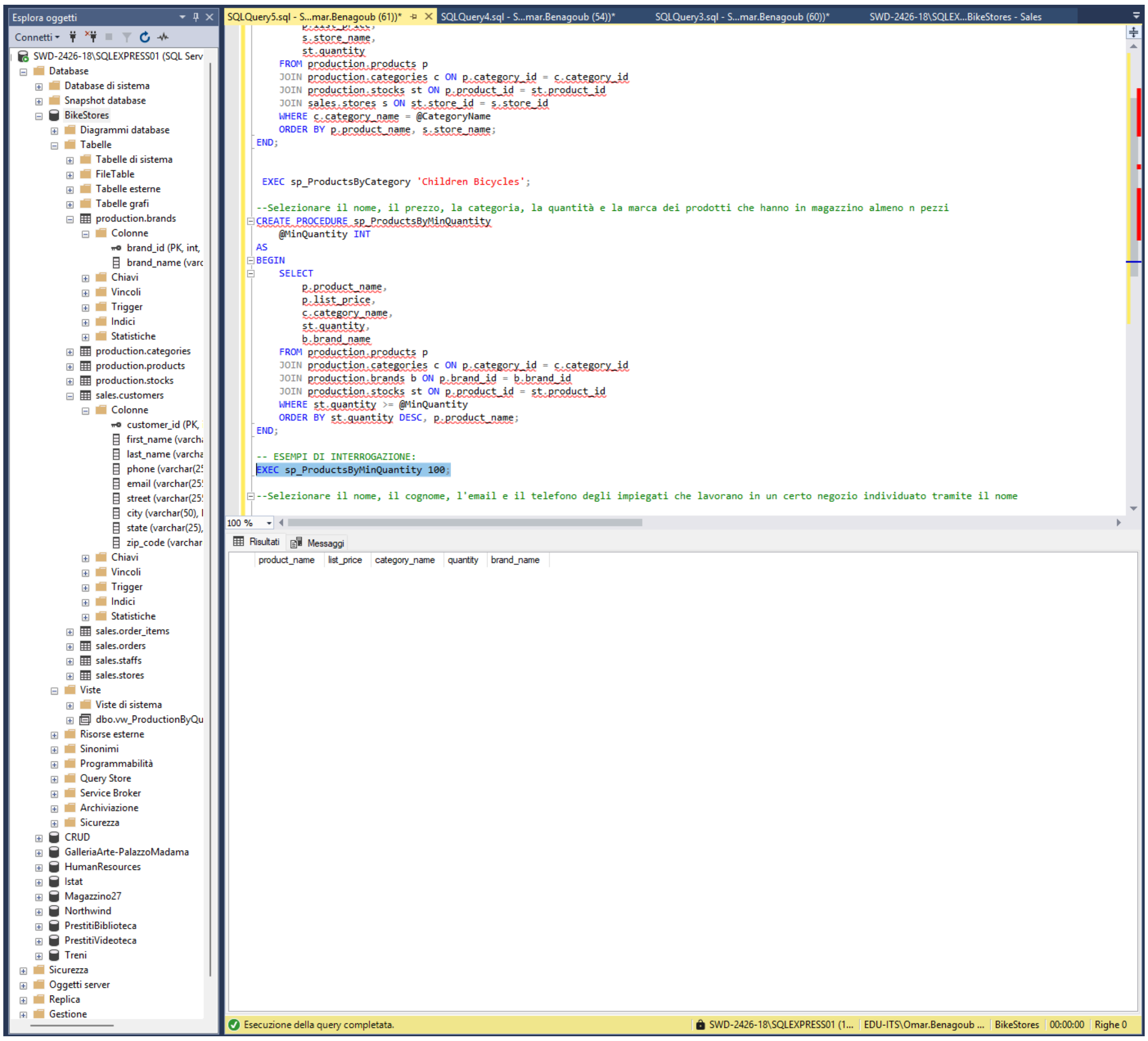
```sql
--Creare le seguenti Stored Procedures

--Selezionare il prodotto, il prezzo, il magazzino e le quantità dei prodotti di una certa categoria
CREATE PROCEDURE sp_ProductsByCategory
    @CategoryName NVARCHAR(255)
AS
BEGIN
    SELECT
        p.product_name,
        p.list_price,
        s.store_name,
        st.quantity
    FROM production.products p
    JOIN production.categories c ON p.category_id = c.category_id
    JOIN production.stocks st ON p.product_id = st.product_id
    JOIN sales.stores s ON st.store_id = s.store_id
    WHERE c.category_name = @CategoryName
    ORDER BY p.product_name, s.store_name;
END;


 EXEC sp_ProductsByCategory 'Children Bicycles';

--Selezionare il nome, il prezzo, la categoria, la quantità e la marca dei prodotti che hanno in magazzino almeno n pezzi

--Selezionare il nome, il cognome, l'email e il telefono degli impiegati che lavorano in un certo negozio individuato tramite il nome

--Selezionare l'id, la data dell'ordine, la data di richiesta e di spedizione, lo stato degli ordini gestiti da un certo impiegato individuato tramite i

--Visualizzare il numero di prodotti venduti in un certo negozio in un determinato anno e un certo impiegato

--Selezionare il nome e cognome del cliente, il modello, il prezzo e l'anno di produzione del prodotto, la data dell'ordine di tutti i clienti che hanno

--Definire un elenco per determinare quanti sono i prodotti in Pending, in Processing, in Rejected e in Completed  (es. 182 Pending, ... Processing, ...

--Per ogni stored procedures fornire degli esempi di interrogazione.
```

	product_name	list_price	store_name	quantity
1	Electra Cruiser 1 (24-Inch) - 2016	269.99	Baldwin Bikes	16
2	Electra Cruiser 1 (24-Inch) - 2016	269.99	Rowlett Bikes	8
3	Electra Cruiser 1 (24-Inch) - 2016	269.99	Santa Cruz Bikes	24
4	Electra Cruiser 7D (24-Inch) Ladies' - 2016/2018	319.99	Baldwin Bikes	3
5	Electra Cruiser 7D (24-Inch) Ladies' - 2016/2018	319.99	Rowlett Bikes	2
6	Electra Cruiser 7D (24-Inch) Ladies' - 2016/2018	319.99	Santa Cruz Bikes	11
7	Electra Cyclosaurus 1 (16-inch) - Boy's - 2018	279.99	Baldwin Bikes	8
8	Electra Cyclosaurus 1 (16-inch) - Boy's - 2018	279.99	Rowlett Bikes	18
9	Electra Cyclosaurus 1 (16-inch) - Boy's - 2018	279.99	Santa Cruz Bikes	6
10	Electra Girl's Hawaii 1 (16-inch) - 2015/2016	269.99	Baldwin Bikes	0
11	Electra Girl's Hawaii 1 (16-inch) - 2015/2016	269.99	Rowlett Bikes	20
12	Electra Girl's Hawaii 1 (16-inch) - 2015/2016	269.99	Santa Cruz Bikes	29
13	Electra Girl's Hawaii 1 (20-inch) - 2015/2016	299.99	Baldwin Bikes	12
14	Electra Girl's Hawaii 1 (20-inch) - 2015/2016	299.99	Rowlett Bikes	8
15	Electra Girl's Hawaii 1 (20-inch) - 2015/2016	299.99	Santa Cruz Bikes	9
16	Electra Girl's Hawaii 1 16" - 2017	299.99	Baldwin Bikes	10
17	Electra Girl's Hawaii 1 16" - 2017	299.99	Rowlett Bikes	14
18	Electra Girl's Hawaii 1 16" - 2017	299.99	Santa Cruz Bikes	25
19	Electra Heartchya 1 (20-inch) - Girl's - 2018	319.99	Baldwin Bikes	28
20	Electra Heartchya 1 (20-inch) - Girl's - 2018	319.99	Rowlett Bikes	1
21	Electra Heartchya 1 (20-inch) - Girl's - 2018	319.99	Santa Cruz Bikes	10
22	Electra Moto 3i (20-inch) - Boy's - 2017	349.99	Baldwin Bikes	22
23	Electra Moto 3i (20-inch) - Boy's - 2017	349.99	Rowlett Bikes	1
24	Electra Moto 3i (20-inch) - Boy's - 2017	349.99	Santa Cruz Bikes	20
25	Electra Savannah 1 (20-inch) - Girl's - 2018	319.99	Baldwin Bikes	6
26	Electra Savannah 1 (20-inch) - Girl's - 2018	319.99	Rowlett Bikes	26
27	Electra Savannah 1 (20-inch) - Girl's - 2018	319.99	Santa Cruz Bikes	27
28	Electra Savannah 3i (20-inch) - Girl's - 2017	349.99	Baldwin Bikes	15
29	Electra Savannah 3i (20-inch) - Girl's - 2017	349.99	Rowlett Bikes	28

```
Esecuzione della query completata.    SWD-2426-18\SQLEXPRESS01 (1...  EDU-ITS\Omar.Benagoub ...  BikeStores  00:00:00  Righe 177
```

2. Selezionare il nome, il prezzo, la categoria, la quantità e la marca dei prodotti che hanno in magazzino almeno n pezzi

```
CREATE PROCEDURE sp_ProductsByMinQuantity
    @MinQuantity INT
AS
BEGIN
    SELECT
        p.product_name,
        p.list_price,
        c.category_name,
        st.quantity,
        b.brand_name
    FROM production.products p
    JOIN production.categories c ON p.category_id = c.category_id
    JOIN production.brands b ON p.brand_id = b.brand_id
    JOIN production.stocks st ON p.product_id = st.product_id
    WHERE st.quantity >= @MinQuantity
    ORDER BY st.quantity DESC, p.product_name;
END;

-- ESEMPI DI INTERROGAZIONE:
EXEC sp_ProductsByMinQuantity 100;
```



```
        s.store_name,
        st.quantity
    FROM production.products p
    JOIN production.categories c ON p.category_id = c.category_id
    JOIN production.stocks st ON p.product_id = st.product_id
    JOIN sales.stores s ON st.store_id = s.store_id
    WHERE c.category_name = @CategoryName
    ORDER BY p.product_name, s.store_name;
END;


 EXEC sp_ProductsByCategory 'Children Bicycles';

--Selezionare il nome, il prezzo, la categoria, la quantità e la marca dei prodotti che hanno in magazzino almeno n pezzi
CREATE PROCEDURE sp_ProductsByMinQuantity
    @MinQuantity INT
AS
BEGIN
    SELECT
        p.product_name,
        p.list_price,
        c.category_name,
        st.quantity,
        b.brand_name
    FROM production.products p
    JOIN production.categories c ON p.category_id = c.category_id
    JOIN production.brands b ON p.brand_id = b.brand_id
    JOIN production.stocks st ON p.product_id = st.product_id
    WHERE st.quantity >= @MinQuantity
    ORDER BY st.quantity DESC, p.product_name;
END;

-- ESEMPI DI INTERROGAZIONE:
EXEC sp_ProductsByMinQuantity 100;

--Selezionare il nome, il cognome, l'email e il telefono degli impiegati che lavorano in un certo negozio individuato tramite il nome
```

```
product_name    list_price    category_name    quantity    brand_name
```

Esecuzione della query completata.

3. Selezionare il nome, il cognome, l'email e il telefono degli impiegati che lavorano in un certo negozio individuato tramite il nome

CREATE PROCEDURE sp_StaffByStore

  @StoreName NVARCHAR(255)

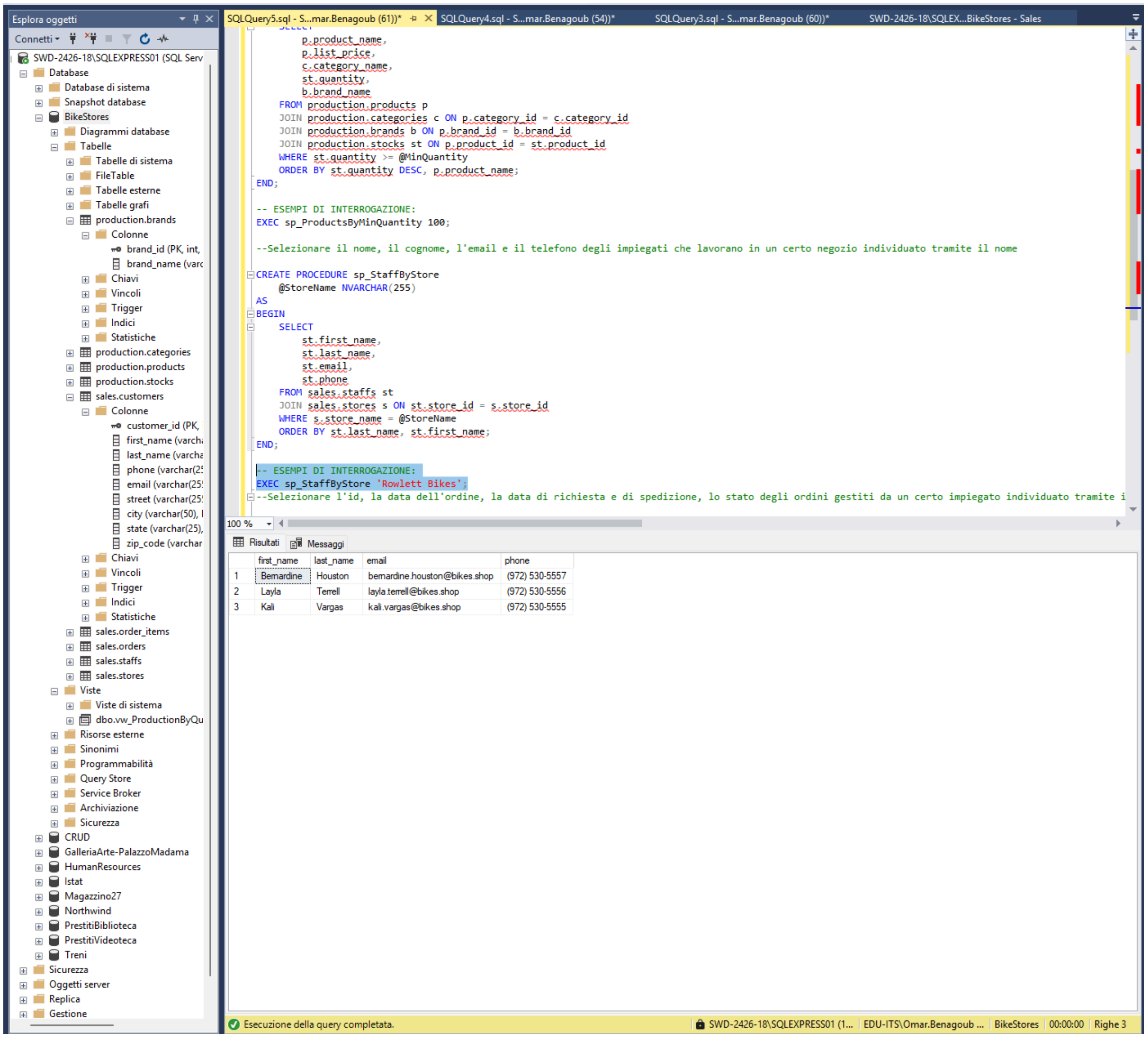
AS

BEGIN

  SELECT

    st.first_name,

```
        st.last_name,
        st.email,
        st.phone
    FROM sales.staffs st
    JOIN sales.stores s ON st.store_id = s.store_id
    WHERE s.store_name = @StoreName
    ORDER BY st.last_name, st.first_name;
END;

-- ESEMPI DI INTERROGAZIONE:
EXEC sp_StaffByStore 'Rowlett Bikes';
```



```
        p.product_name,
        p.list_price,
        c.category_name,
        st.quantity,
        b.brand_name
    FROM production.products p
    JOIN production.categories c ON p.category_id = c.category_id
    JOIN production.brands b ON p.brand_id = b.brand_id
    JOIN production.stocks st ON p.product_id = st.product_id
    WHERE st.quantity >= @MinQuantity
    ORDER BY st.quantity DESC, p.product_name;
END;

-- ESEMPI DI INTERROGAZIONE:
EXEC sp_ProductsByMinQuantity 100;

--Selezionare il nome, il cognome, l'email e il telefono degli impiegati che lavorano in un certo negozio individuato tramite il nome

CREATE PROCEDURE sp_StaffByStore
    @StoreName NVARCHAR(255)
AS
BEGIN
    SELECT
        st.first_name,
        st.last_name,
        st.email,
        st.phone
    FROM sales.staffs st
    JOIN sales.stores s ON st.store_id = s.store_id
    WHERE s.store_name = @StoreName
    ORDER BY st.last_name, st.first_name;
END;

-- ESEMPI DI INTERROGAZIONE:
EXEC sp_StaffByStore 'Rowlett Bikes';
--Selezionare l'id, la data dell'ordine, la data di richiesta e di spedizione, lo stato degli ordini gestiti da un certo impiegato individuato tramite i
```

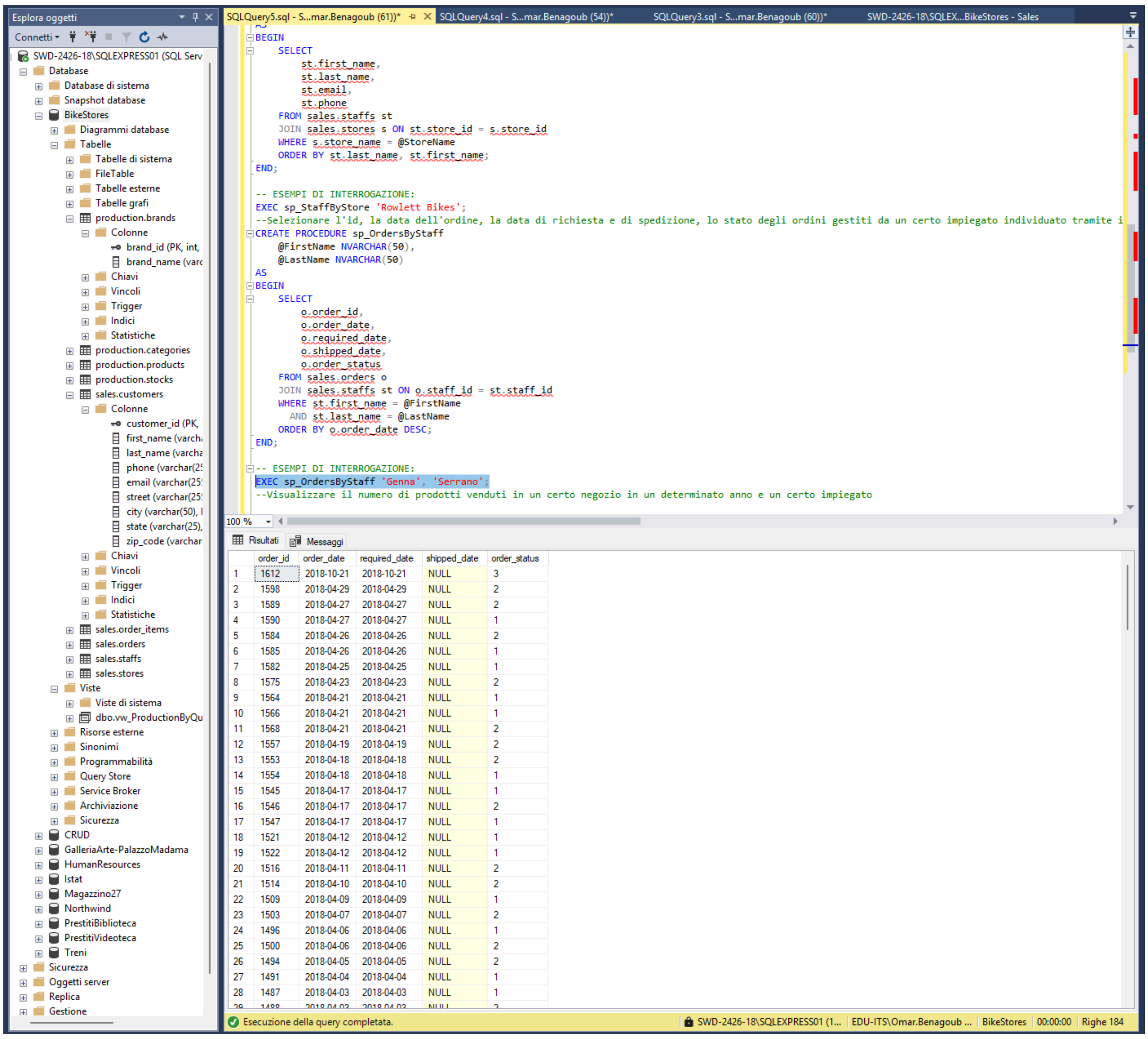
	first_name	last_name	email	phone
1	Bernardine	Houston	bernardine.houston@bikes.shop	(972) 530-5557
2	Layla	Terrell	layla.terrell@bikes.shop	(972) 530-5556
3	Kali	Vargas	kali.vargas@bikes.shop	(972) 530-5555

Esecuzione della query completata.

4. Selezionare l'id, la data dell'ordine, la data di richiesta e di spedizione, lo stato degli ordini gestiti da un certo impiegato individuato tramite il cognome e il nome

```
CREATE PROCEDURE sp_OrdersByStaff

   @FirstName NVARCHAR(50),

   @LastName NVARCHAR(50)

AS

BEGIN

   SELECT
```

```
        o.order_id,
        o.order_date,
        o.required_date,
        o.shipped_date,
        o.order_status
    FROM sales.orders o
    JOIN sales.staffs st ON o.staff_id = st.staff_id
    WHERE st.first_name = @FirstName
      AND st.last_name = @LastName
    ORDER BY o.order_date DESC;
END;

-- ESEMPI DI INTERROGAZIONE:
EXEC sp_OrdersByStaff 'Genna', 'Serrano';
```



```
BEGIN
    SELECT
        st.first_name,
        st.last_name,
        st.email,
        st.phone
    FROM sales.staffs st
    JOIN sales.stores s ON st.store_id = s.store_id
    WHERE s.store_name = @StoreName
    ORDER BY st.last_name, st.first_name;
END;

-- ESEMPI DI INTERROGAZIONE:
EXEC sp_StaffByStore 'Rowlett Bikes';
--Selezionare l'id, la data dell'ordine, la data di richiesta e di spedizione, lo stato degli ordini gestiti da un certo impiegato individuato tramite i
CREATE PROCEDURE sp_OrdersByStaff
    @FirstName NVARCHAR(50),
    @LastName NVARCHAR(50)
AS
BEGIN
    SELECT
        o.order_id,
        o.order_date,
        o.required_date,
        o.shipped_date,
        o.order_status
    FROM sales.orders o
    JOIN sales.staffs st ON o.staff_id = st.staff_id
    WHERE st.first_name = @FirstName
      AND st.last_name = @LastName
    ORDER BY o.order_date DESC;
END;

-- ESEMPI DI INTERROGAZIONE:
EXEC sp_OrdersByStaff 'Genna', 'Serrano';
--Visualizzare il numero di prodotti venduti in un certo negozio in un determinato anno e un certo impiegato
```

	order_id	order_date	required_date	shipped_date	order_status
1	1612	2018-10-21	2018-10-21	NULL	3
2	1598	2018-04-29	2018-04-29	NULL	2
3	1589	2018-04-27	2018-04-27	NULL	2
4	1590	2018-04-27	2018-04-27	NULL	1
5	1584	2018-04-26	2018-04-26	NULL	2
6	1585	2018-04-26	2018-04-26	NULL	1
7	1582	2018-04-25	2018-04-25	NULL	1
8	1575	2018-04-23	2018-04-23	NULL	2
9	1564	2018-04-21	2018-04-21	NULL	1
10	1566	2018-04-21	2018-04-21	NULL	1
11	1568	2018-04-21	2018-04-21	NULL	2
12	1557	2018-04-19	2018-04-19	NULL	2
13	1553	2018-04-18	2018-04-18	NULL	2
14	1554	2018-04-18	2018-04-18	NULL	1
15	1545	2018-04-17	2018-04-17	NULL	1
16	1546	2018-04-17	2018-04-17	NULL	2
17	1547	2018-04-17	2018-04-17	NULL	1
18	1521	2018-04-12	2018-04-12	NULL	1
19	1522	2018-04-12	2018-04-12	NULL	1
20	1516	2018-04-11	2018-04-11	NULL	2
21	1514	2018-04-10	2018-04-10	NULL	2
22	1509	2018-04-09	2018-04-09	NULL	1
23	1503	2018-04-07	2018-04-07	NULL	2
24	1496	2018-04-06	2018-04-06	NULL	1
25	1500	2018-04-06	2018-04-06	NULL	2
26	1494	2018-04-05	2018-04-05	NULL	2
27	1491	2018-04-04	2018-04-04	NULL	1
28	1487	2018-04-03	2018-04-03	NULL	1

Esecuzione della query completata.

5. Visualizzare il numero di prodotti venduti in un certo negozio in un determinato anno e un certo impiegato

```
CREATE PROCEDURE sp_SalesCountByStoreYearStaff

    @StoreName NVARCHAR(255),

    @Year INT,

    @StaffFirstName NVARCHAR(50),

    @StaffLastName NVARCHAR(50)

AS

BEGIN
```

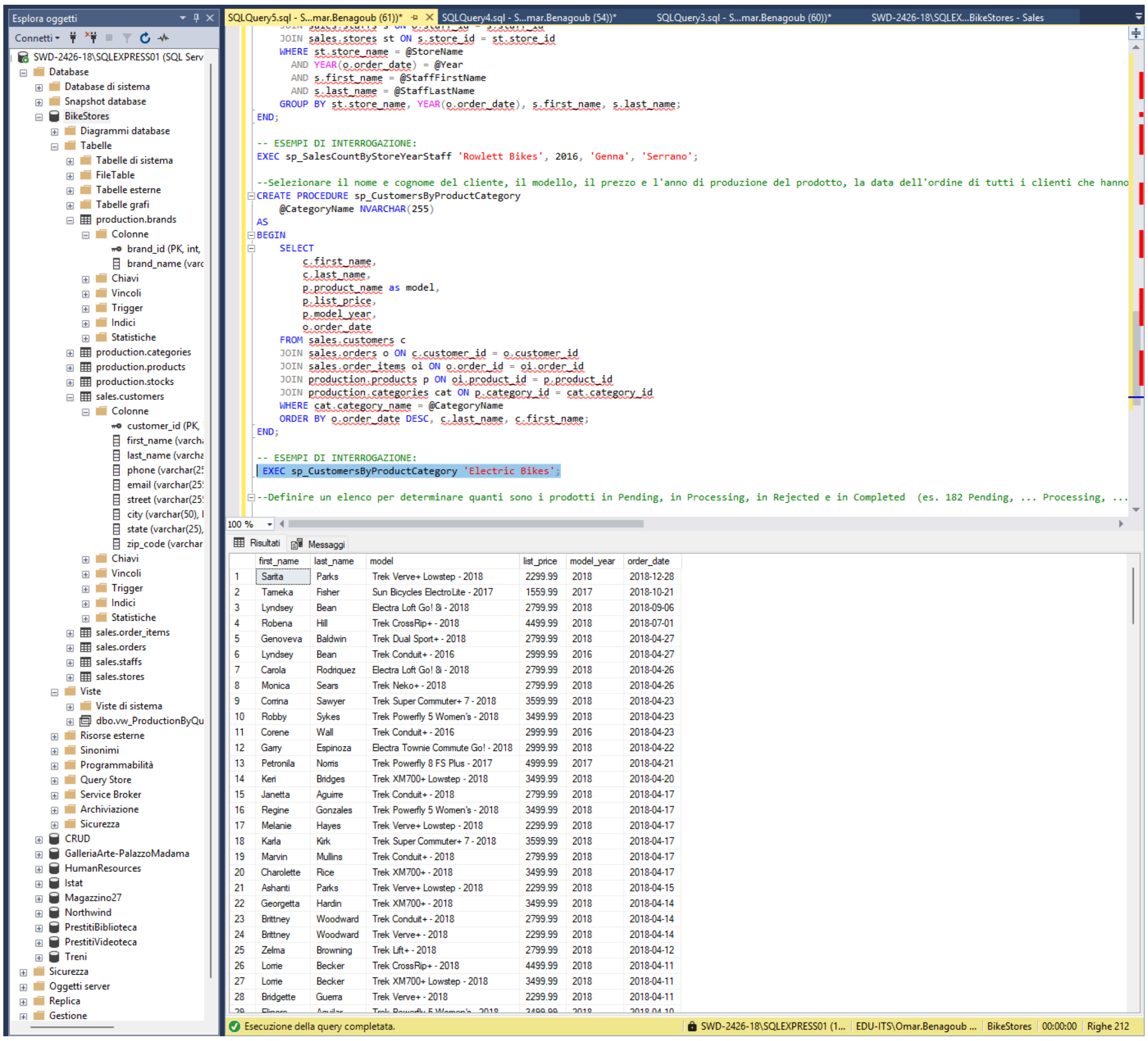
```
    SELECT
        COUNT(oi.product_id) as total_products_sold,
        SUM(oi.quantity) as total_quantity_sold,
        st.store_name,
        YEAR(o.order_date) as sales_year,
        s.first_name + ' ' + s.last_name as staff_name
    FROM sales.orders o
    JOIN sales.order_items oi ON o.order_id = oi.order_id
    JOIN sales.staffs s ON o.staff_id = s.staff_id
    JOIN sales.stores st ON s.store_id = st.store_id
    WHERE st.store_name = @StoreName
      AND YEAR(o.order_date) = @Year
      AND s.first_name = @StaffFirstName
      AND s.last_name = @StaffLastName
    GROUP BY st.store_name, YEAR(o.order_date), s.first_name,
s.last_name;
END;


-- ESEMPI DI INTERROGAZIONE:
EXEC sp_SalesCountByStoreYearStaff 'Rowlett Bikes', 2016,
'Genna', 'Serrano';
```

6. Selezionare il nome e cognome del cliente, il modello, il prezzo e l'anno di produzione del prodotto, la data dell'ordine di tutti i clienti che hanno acquistato prodotti di una certa categoria individuata tramite il nome della categoria

```
CREATE PROCEDURE sp_CustomersByProductCategory

   @CategoryName NVARCHAR(255)

AS

BEGIN

   SELECT

      c.first_name,

      c.last_name,

      p.product_name as model,

      p.list_price,

      p.model_year,

      o.order_date

   FROM sales.customers c

   JOIN sales.orders o ON c.customer_id = o.customer_id

   JOIN sales.order_items oi ON o.order_id = oi.order_id

   JOIN production.products p ON oi.product_id = p.product_id

   JOIN production.categories cat ON p.category_id =
cat.category_id

   WHERE cat.category_name = @CategoryName

   ORDER BY o.order_date DESC, c.last_name, c.first_name;

END;
```

-- ESEMPI DI INTERROGAZIONE:

EXEC sp_CustomersByProductCategory 'Electric Bikes';



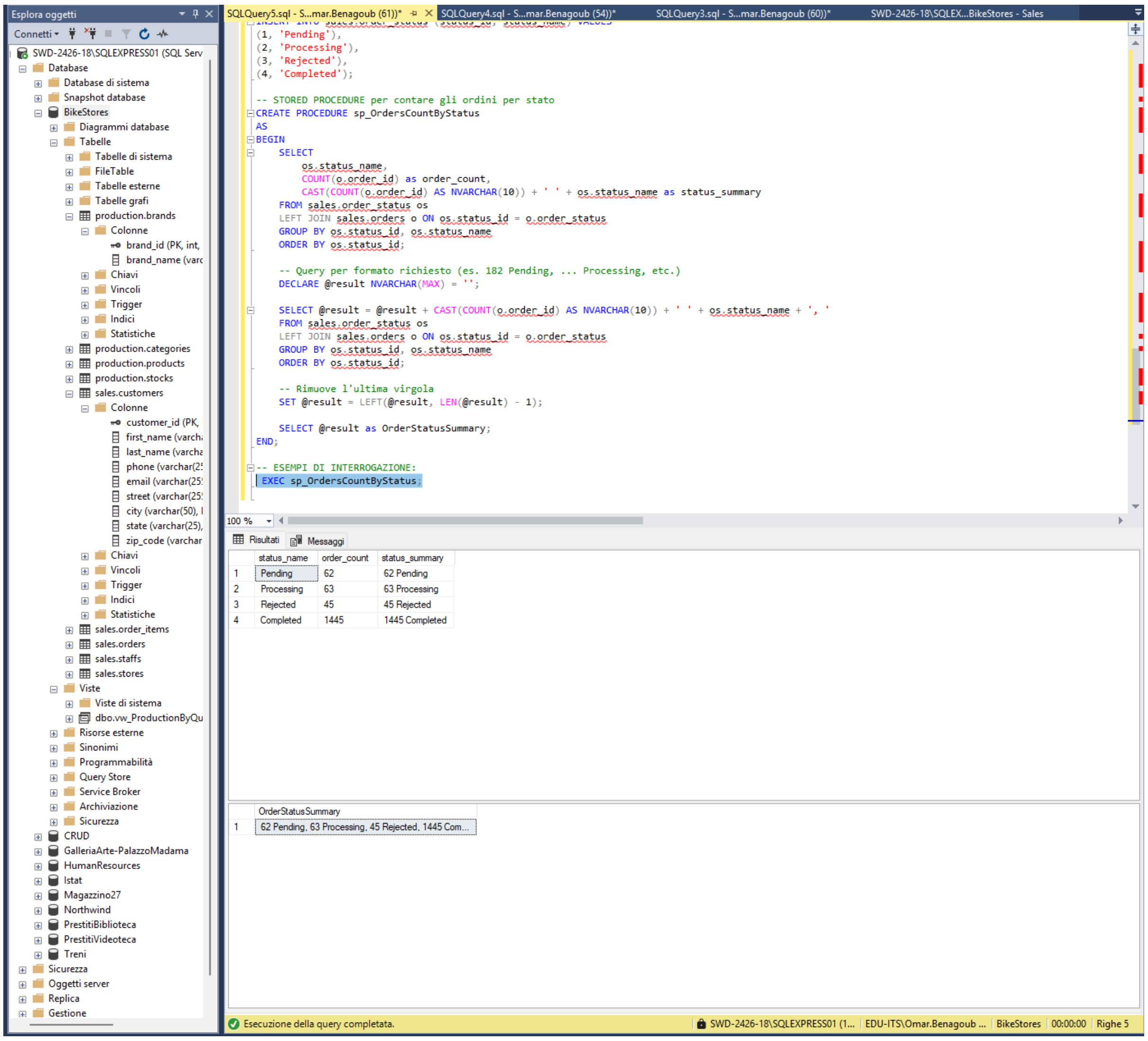
```sql
    JOIN sales.stores st ON s.store_id = st.store_id
    WHERE st.store_name = @StoreName
      AND YEAR(o.order_date) = @Year
      AND s.first_name = @StaffFirstName
      AND s.last_name = @StaffLastName
    GROUP BY st.store_name, YEAR(o.order_date), s.first_name, s.last_name;
END;

-- ESEMPI DI INTERROGAZIONE:
EXEC sp_SalesCountByStoreYearStaff 'Rowlett Bikes', 2016, 'Genna', 'Serrano';

--Selezionare il nome e cognome del cliente, il modello, il prezzo e l'anno di produzione del prodotto, la data dell'ordine di tutti i clienti che hanno
CREATE PROCEDURE sp_CustomersByProductCategory
    @CategoryName NVARCHAR(255)
AS
BEGIN
    SELECT
        c.first_name,
        c.last_name,
        p.product_name as model,
        p.list_price,
        p.model_year,
        o.order_date
    FROM sales.customers c
    JOIN sales.orders o ON c.customer_id = o.customer_id
    JOIN sales.order_items oi ON o.order_id = oi.order_id
    JOIN production.products p ON oi.product_id = p.product_id
    JOIN production.categories cat ON p.category_id = cat.category_id
    WHERE cat.category_name = @CategoryName
    ORDER BY o.order_date DESC, c.last_name, c.first_name;
END;

-- ESEMPI DI INTERROGAZIONE:
EXEC sp_CustomersByProductCategory 'Electric Bikes';

--Definire un elenco per determinare quanti sono i prodotti in Pending, in Processing, in Rejected e in Completed  (es. 182 Pending, ... Processing, ...
```

	first_name	last_name	model	list_price	model_year	order_date
1	Sarita	Parks	Trek Verve+ Lowstep - 2018	2299.99	2018	2018-12-28
2	Tameka	Fisher	Sun Bicycles ElectroLite - 2017	1559.99	2017	2018-10-21
3	Lyndsey	Bean	Electra Loft Go! 8i - 2018	2799.99	2018	2018-09-06
4	Robena	Hill	Trek CrossRip+ - 2018	4499.99	2018	2018-07-01
5	Genoveva	Baldwin	Trek Dual Sport+ - 2018	2799.99	2018	2018-04-27
6	Lyndsey	Bean	Trek Conduit+ - 2016	2999.99	2016	2018-04-27
7	Carola	Rodriquez	Electra Loft Go! 8i - 2018	2799.99	2018	2018-04-26
8	Monica	Sears	Trek Neko+ - 2018	2799.99	2018	2018-04-26
9	Corrina	Sawyer	Trek Super Commuter+ 7 - 2018	3599.99	2018	2018-04-23
10	Robby	Sykes	Trek Powerfly 5 Women's - 2018	3499.99	2018	2018-04-23
11	Corene	Wall	Trek Conduit+ - 2016	2999.99	2016	2018-04-23
12	Garry	Espinoza	Electra Townie Commute Go! - 2018	2999.99	2018	2018-04-22
13	Petronila	Norris	Trek Powerfly 8 FS Plus - 2017	4999.99	2017	2018-04-21
14	Keri	Bridges	Trek XM700+ Lowstep - 2018	3499.99	2018	2018-04-20
15	Janetta	Aguirre	Trek Conduit+ - 2018	2799.99	2018	2018-04-17
16	Regine	Gonzales	Trek Powerfly 5 Women's - 2018	3499.99	2018	2018-04-17
17	Melanie	Hayes	Trek Verve+ Lowstep - 2018	2299.99	2018	2018-04-17
18	Karla	Kirk	Trek Super Commuter+ 7 - 2018	3599.99	2018	2018-04-17
19	Marvin	Mullins	Trek Conduit+ - 2018	2799.99	2018	2018-04-17
20	Charolette	Rice	Trek XM700+ - 2018	3499.99	2018	2018-04-17
21	Ashanti	Parks	Trek Verve+ Lowstep - 2018	2299.99	2018	2018-04-15
22	Georgetta	Hardin	Trek XM700+ - 2018	3499.99	2018	2018-04-14
23	Brittney	Woodward	Trek Conduit+ - 2018	2799.99	2018	2018-04-14
24	Brittney	Woodward	Trek Verve+ - 2018	2299.99	2018	2018-04-14
25	Zelma	Browning	Trek Lift+ - 2018	2799.99	2018	2018-04-12
26	Lorrie	Becker	Trek CrossRip+ - 2018	4499.99	2018	2018-04-11
27	Lorrie	Becker	Trek XM700+ Lowstep - 2018	3499.99	2018	2018-04-11
28	Bridgette	Guerra	Trek Verve+ - 2018	2299.99	2018	2018-04-11

```
Esecuzione della query completata.    SWD-2426-18\SQLEXPRESS01 (1... | EDU-ITS\Omar.Benagoub ... | BikeStores | 00:00:00 | Righe 212
```

7. Definire un elenco per determinare quanti sono i prodotti in Pending, in Processing, in Rejected e in Completed (es. 182 Pending, ... Processing, ... Rejected, ... Completed). Si consideri a tal proposito l'aggiunta di una tabella Order_Status con i riferimenti agli stati Pending=1, Processing=2, Rejected=3, Completed=4

```
CREATE TABLE sales.order_status (

    status_id INT PRIMARY KEY,

    status_name NVARCHAR(50) NOT NULL

);


-- Inserimento degli stati

INSERT INTO sales.order_status (status_id, status_name)
VALUES

(1, 'Pending'),

(2, 'Processing'),

(3, 'Rejected'),

(4, 'Completed');


-- STORED PROCEDURE per contare gli ordini per stato

CREATE PROCEDURE sp_OrdersCountByStatus

AS

BEGIN

    SELECT

        os.status_name,

        COUNT(o.order_id) as order_count,

        CAST(COUNT(o.order_id) AS NVARCHAR(10)) + ' ' +
os.status_name as status_summary

    FROM sales.order_status os
    LEFT JOIN sales.orders o ON os.status_id = o.order_status
    GROUP BY os.status_id, os.status_name
    ORDER BY os.status_id;

    -- Query per formato richiesto (es. 182 Pending, ... Processing,
etc.)
    DECLARE @result NVARCHAR(MAX) = '';

    SELECT @result = @result + CAST(COUNT(o.order_id) AS
NVARCHAR(10)) + ' ' + os.status_name + ', '
    FROM sales.order_status os
    LEFT JOIN sales.orders o ON os.status_id = o.order_status
    GROUP BY os.status_id, os.status_name
    ORDER BY os.status_id;

    -- Rimuove l'ultima virgola
    SET @result = LEFT(@result, LEN(@result) - 1);

    SELECT @result as OrderStatusSummary;
END;

-- ESEMPI DI INTERROGAZIONE:
 EXEC sp_OrdersCountByStatus;
```



```sql
(1, 'Pending'),
(2, 'Processing'),
(3, 'Rejected'),
(4, 'Completed');

-- STORED PROCEDURE per contare gli ordini per stato
CREATE PROCEDURE sp_OrdersCountByStatus
AS
BEGIN
    SELECT
        os.status_name,
        COUNT(o.order_id) as order_count,
        CAST(COUNT(o.order_id) AS NVARCHAR(10)) + ' ' + os.status_name as status_summary
    FROM sales.order_status os
    LEFT JOIN sales.orders o ON os.status_id = o.order_status
    GROUP BY os.status_id, os.status_name
    ORDER BY os.status_id;

    -- Query per formato richiesto (es. 182 Pending, ... Processing, etc.)
    DECLARE @result NVARCHAR(MAX) = '';

    SELECT @result = @result + CAST(COUNT(o.order_id) AS NVARCHAR(10)) + ' ' + os.status_name + ', '
    FROM sales.order_status os
    LEFT JOIN sales.orders o ON os.status_id = o.order_status
    GROUP BY os.status_id, os.status_name
    ORDER BY os.status_id;

    -- Rimuove l'ultima virgola
    SET @result = LEFT(@result, LEN(@result) - 1);

    SELECT @result as OrderStatusSummary;
END;

-- ESEMPI DI INTERROGAZIONE:
EXEC sp_OrdersCountByStatus;
```

	status_name	order_count	status_summary
1	Pending	62	62 Pending
2	Processing	63	63 Processing
3	Rejected	45	45 Rejected
4	Completed	1445	1445 Completed

	OrderStatusSummary
1	62 Pending, 63 Processing, 45 Rejected, 1445 Com...

Esecuzione della query completata.

Per ogni stored procedures fornire degli esempi di interrogazione.