FINAL PROJECT

Bike Store analysis using ‘SQL’

INTRODUCTION

hi , i’m nada sami
i’m AI student
and I will
introduce my
project to you.

MY project is about
Bike Store
it explain the relation
beteween every thing in
the dataset i have used
in my steps in the
project

here i will explain my steps in this project

At first , I open kaggle to optain the dataset
then i upload it to SSMS to start work with it ,
I make some preprocessing steps in the data
So it become ready to make our queries on it.

how I upload the dataset to SSMS

1* Create database BIKESTORE_DB
2* click on my database>> tasks>>import flat files
then upload every table in the dataset from my pc

Here is ERD to the project tables:

*customer

* staffs

*stores

*orders

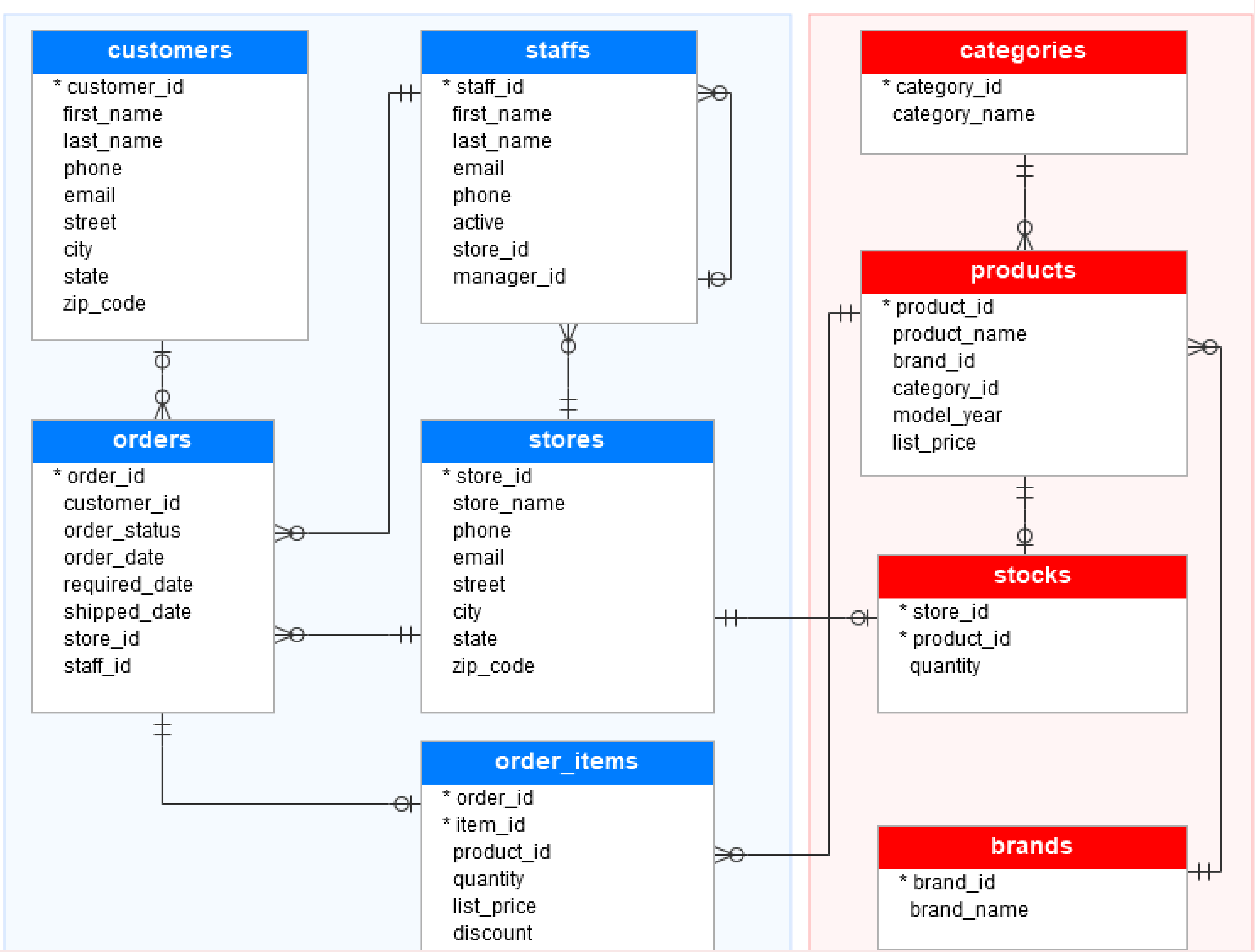
*order_items

*categories

*products

*stocks

*brands



customers
* customer_id
first_name
last_name
phone
email
street
city
state
zip_code

staffs
* staff_id
first_name
last_name
email
phone
active
store_id
manager_id

categories
* category_id
category_name

orders
* order_id
customer_id
order_status
order_date
required_date
shipped_date
store_id
staff_id

stores
* store_id
store_name
phone
email
street
city
state
zip_code

products
* product_id
product_name
brand_id
category_id
model_year
list_price

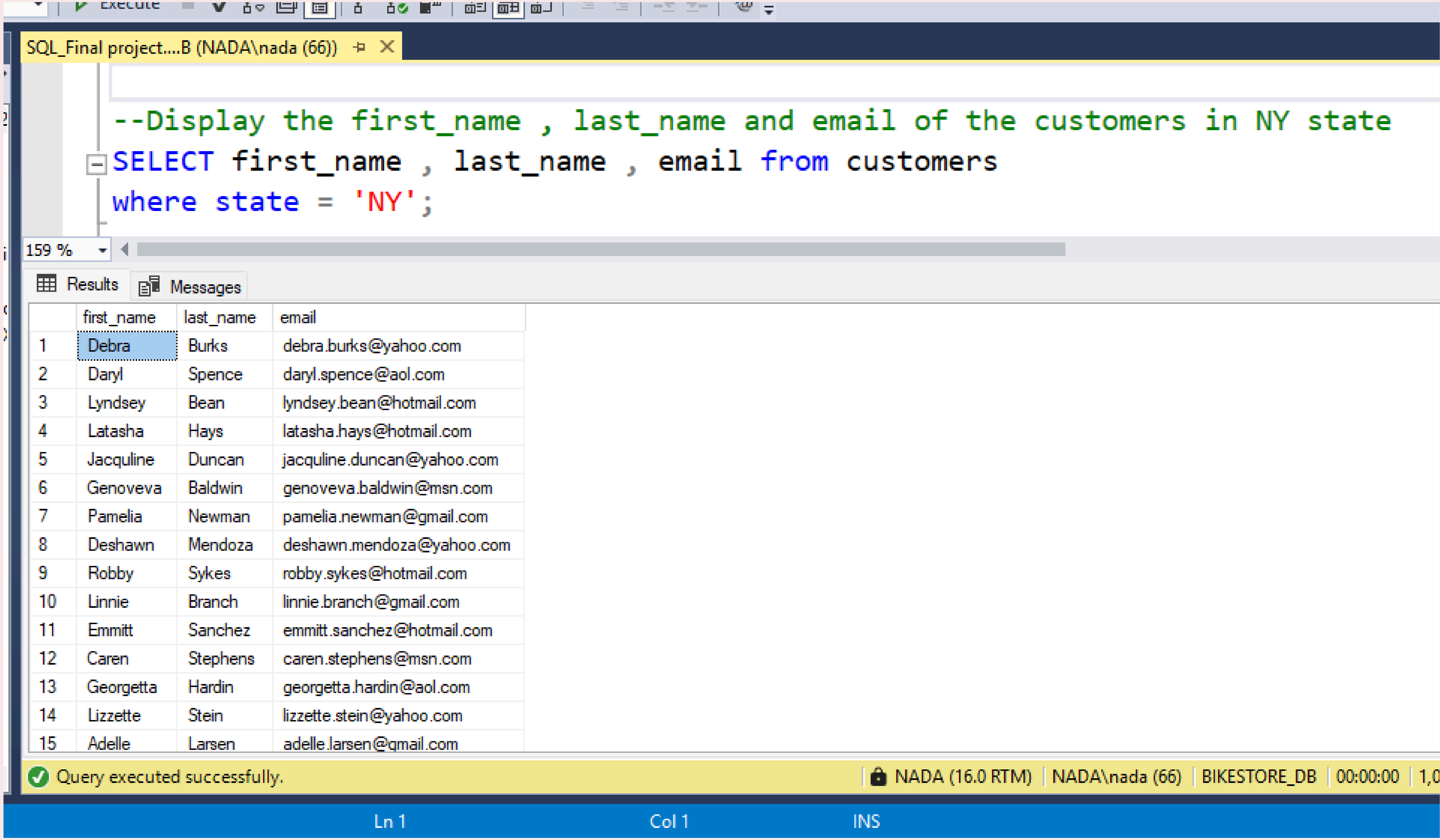
stocks
* store_id
* product_id
quantity

order_items
* order_id
* item_id
product_id
quantity
list_price
discount

brands
* brand_id
brand_name

Display Data

“Here I display customers names and notice that more than 1000 customer from (NY) state”



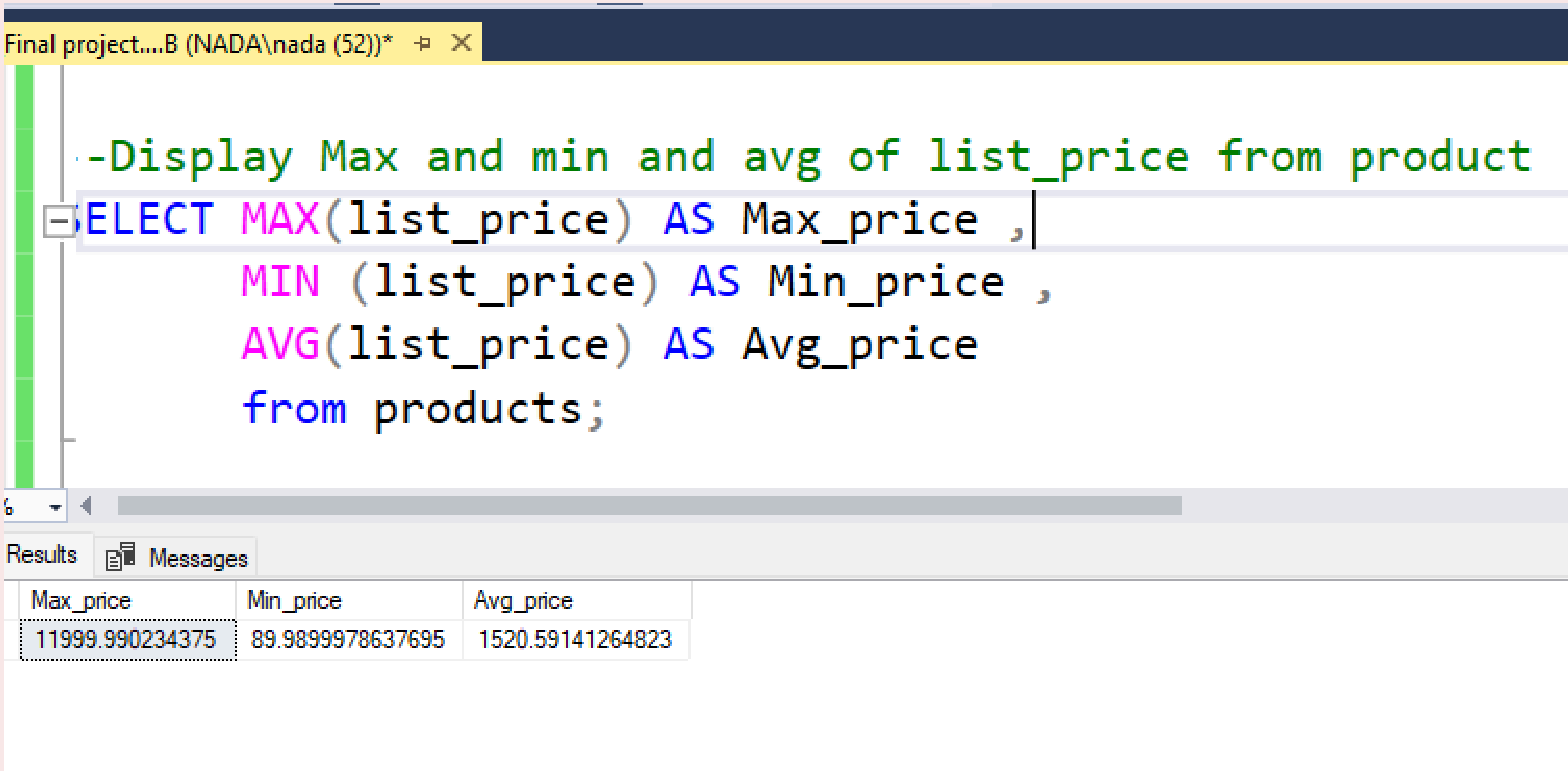
```sql
--Display the first_name , last_name and email of the customers in NY state
SELECT first_name , last_name , email from customers
where state = 'NY';
```

	first_name	last_name	email
1	Debra	Burks	debra.burks@yahoo.com
2	Daryl	Spence	daryl.spence@aol.com
3	Lyndsey	Bean	lyndsey.bean@hotmail.com
4	Latasha	Hays	latasha.hays@hotmail.com
5	Jacquline	Duncan	jacquline.duncan@yahoo.com
6	Genoveva	Baldwin	genoveva.baldwin@msn.com
7	Pamelia	Newman	pamelia.newman@gmail.com
8	Deshawn	Mendoza	deshawn.mendoza@yahoo.com
9	Robby	Sykes	robby.sykes@hotmail.com
10	Linnie	Branch	linnie.branch@gmail.com
11	Emmitt	Sanchez	emmitt.sanchez@hotmail.com
12	Caren	Stephens	caren.stephens@msn.com
13	Georgetta	Hardin	georgetta.hardin@aol.com
14	Lizzette	Stein	lizzette.stein@yahoo.com
15	Adelle	Larsen	adelle.larsen@gmail.com

Query executed successfully. NADA (16.0 RTM) | NADA\nada (66) | BIKESTORE_DB | 00:00:00 | 1,0

Ln 1 Col 1 INS

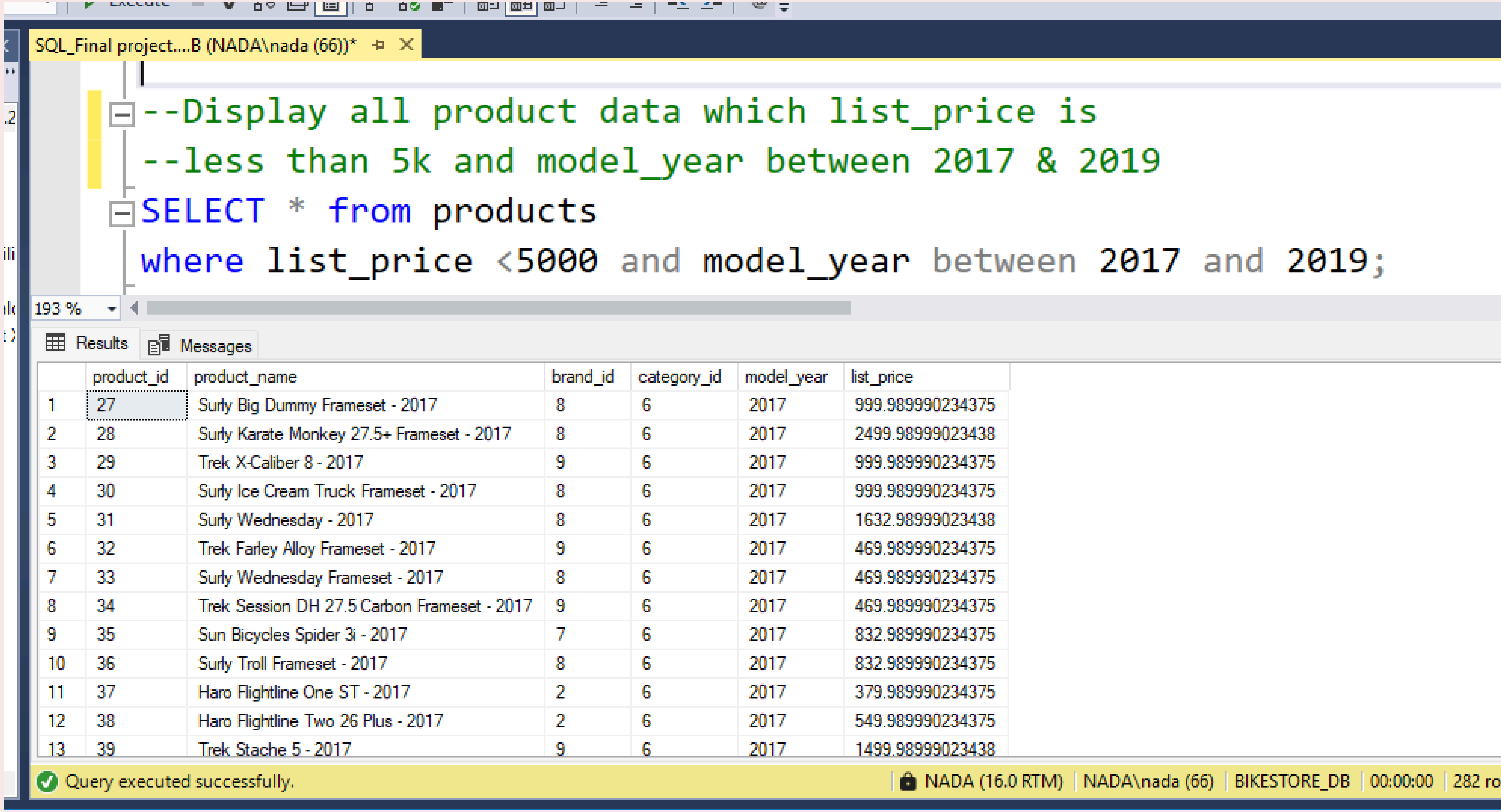
Here I display MAX & MIN & AVG price ‘for each product



```
-Display Max and min and avg of list_price from product
ELECT MAX(list_price) AS Max_price ,
      MIN (list_price) AS Min_price ,
      AVG(list_price) AS Avg_price
      from products;
```

Max_price	Min_price	Avg_price
11999.990234375	89.9899978637695	1520.59141264823

“Here we notice that in time between 2017 & 2019 there are 282 product with price < 5000 “

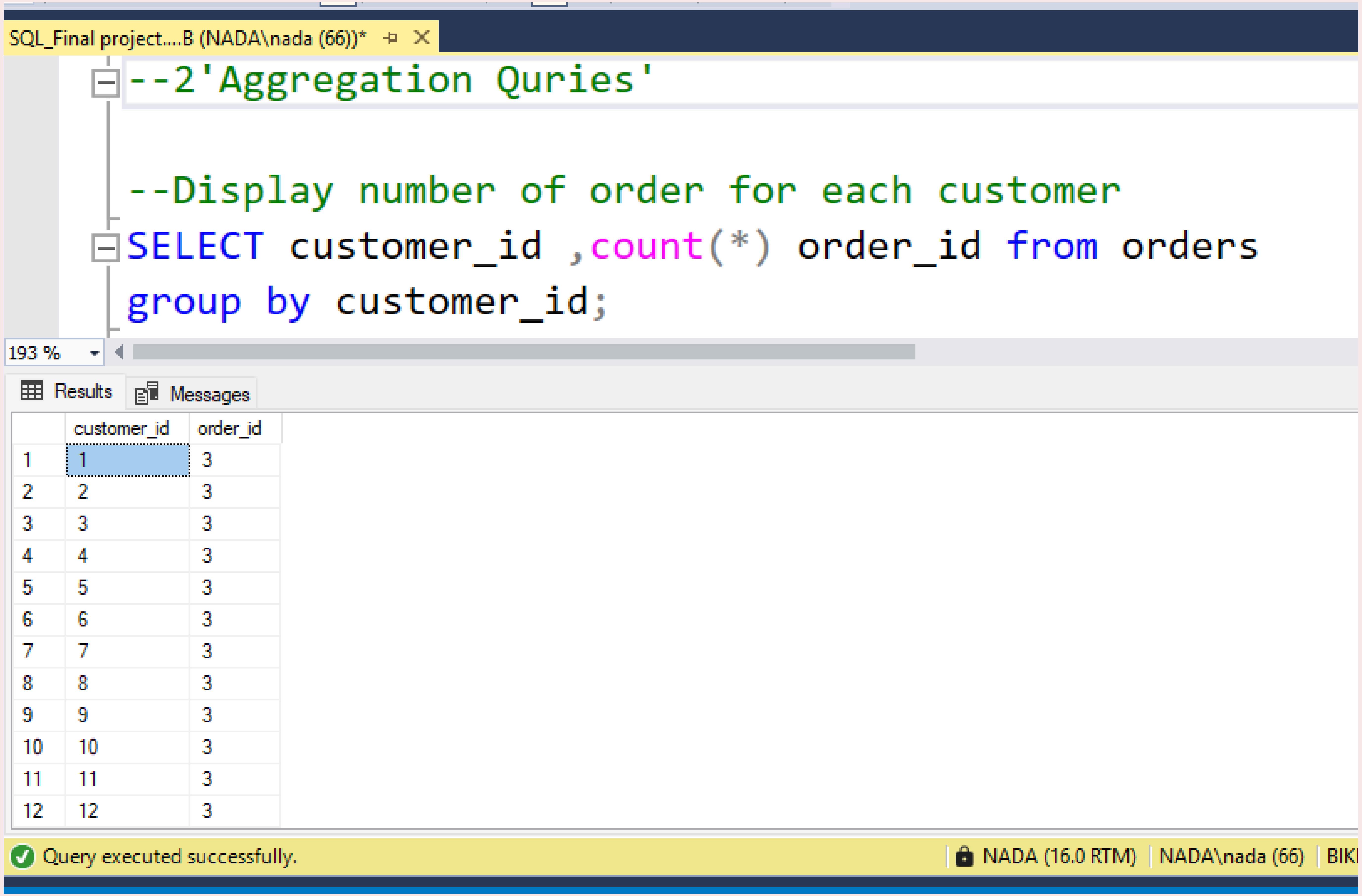


```
--Display all product data which list_price is
--less than 5k and model_year between 2017 & 2019
SELECT * from products
where list_price <5000 and model_year between 2017 and 2019;
```

	product_id	product_name	brand_id	category_id	model_year	list_price
1	27	Surly Big Dummy Frameset - 2017	8	6	2017	999.989990234375
2	28	Surly Karate Monkey 27.5+ Frameset - 2017	8	6	2017	2499.98999023438
3	29	Trek X-Caliber 8 - 2017	9	6	2017	999.989990234375
4	30	Surly Ice Cream Truck Frameset - 2017	8	6	2017	999.989990234375
5	31	Surly Wednesday - 2017	8	6	2017	1632.98999023438
6	32	Trek Farley Alloy Frameset - 2017	9	6	2017	469.989990234375
7	33	Surly Wednesday Frameset - 2017	8	6	2017	469.989990234375
8	34	Trek Session DH 27.5 Carbon Frameset - 2017	9	6	2017	469.989990234375
9	35	Sun Bicycles Spider 3i - 2017	7	6	2017	832.989990234375
10	36	Surly Troll Frameset - 2017	8	6	2017	832.989990234375
11	37	Haro Flightline One ST - 2017	2	6	2017	379.989990234375
12	38	Haro Flightline Two 26 Plus - 2017	2	6	2017	549.989990234375
13	39	Trek Stache 5 - 2017	9	6	2017	1499.98999023438

Query executed successfully.

NADA (16.0 RTM) | NADA\nada (66) | BIKESTORE_DB | 00:00:00 | 282 ro “this query display the number of orders which each customer makes”

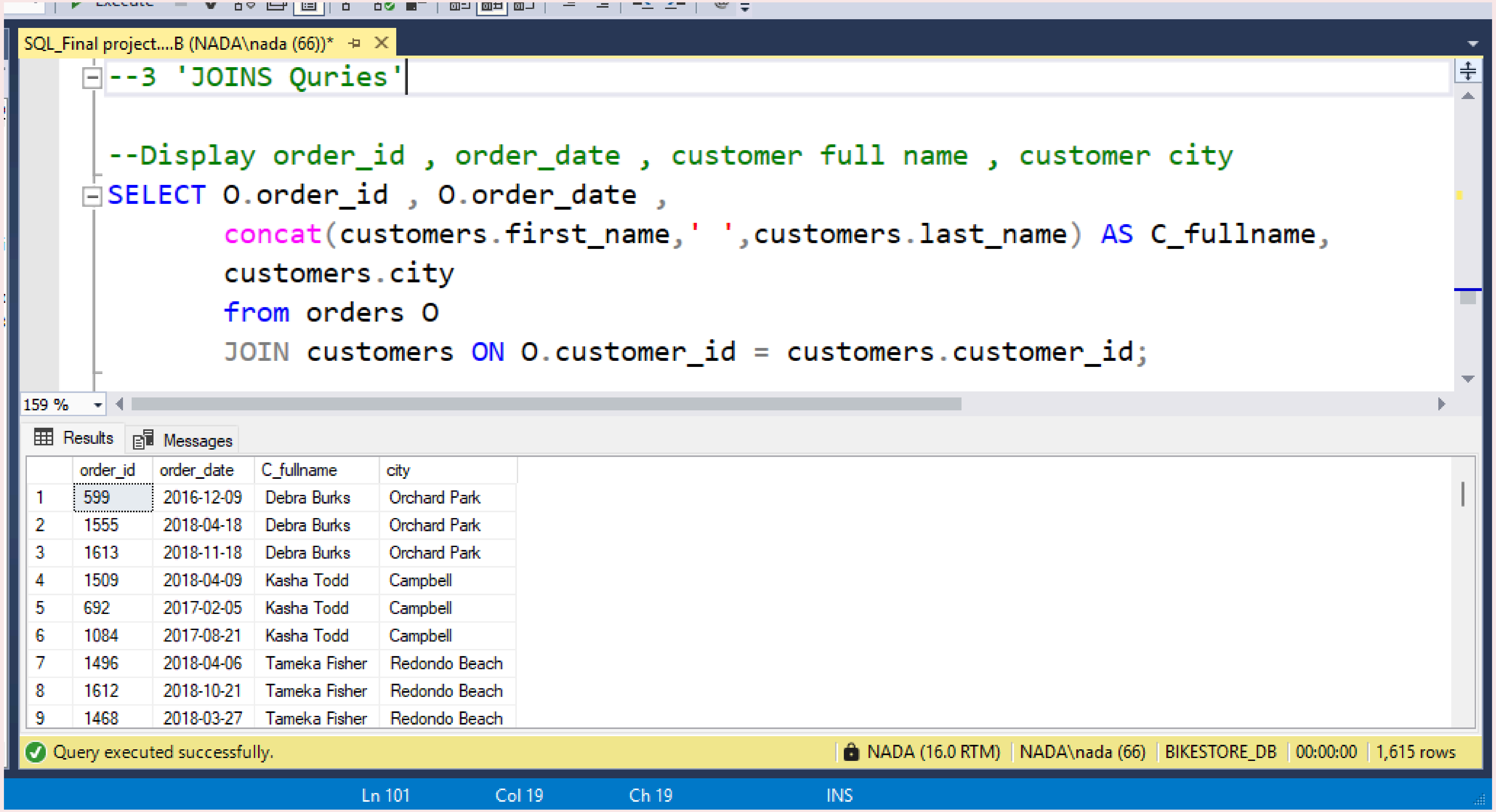


```
--2'Aggregation Quries'

--Display number of order for each customer
SELECT customer_id ,count(*) order_id from orders
group by customer_id;
```

	customer_id	order_id
1	1	3
2	2	3
3	3	3
4	4	3
5	5	3
6	6	3
7	7	3
8	8	3
9	9	3
10	10	3
11	11	3
12	12	3

Query executed successfully.

“These results display all customers names and cities with all orders ID and date they make”



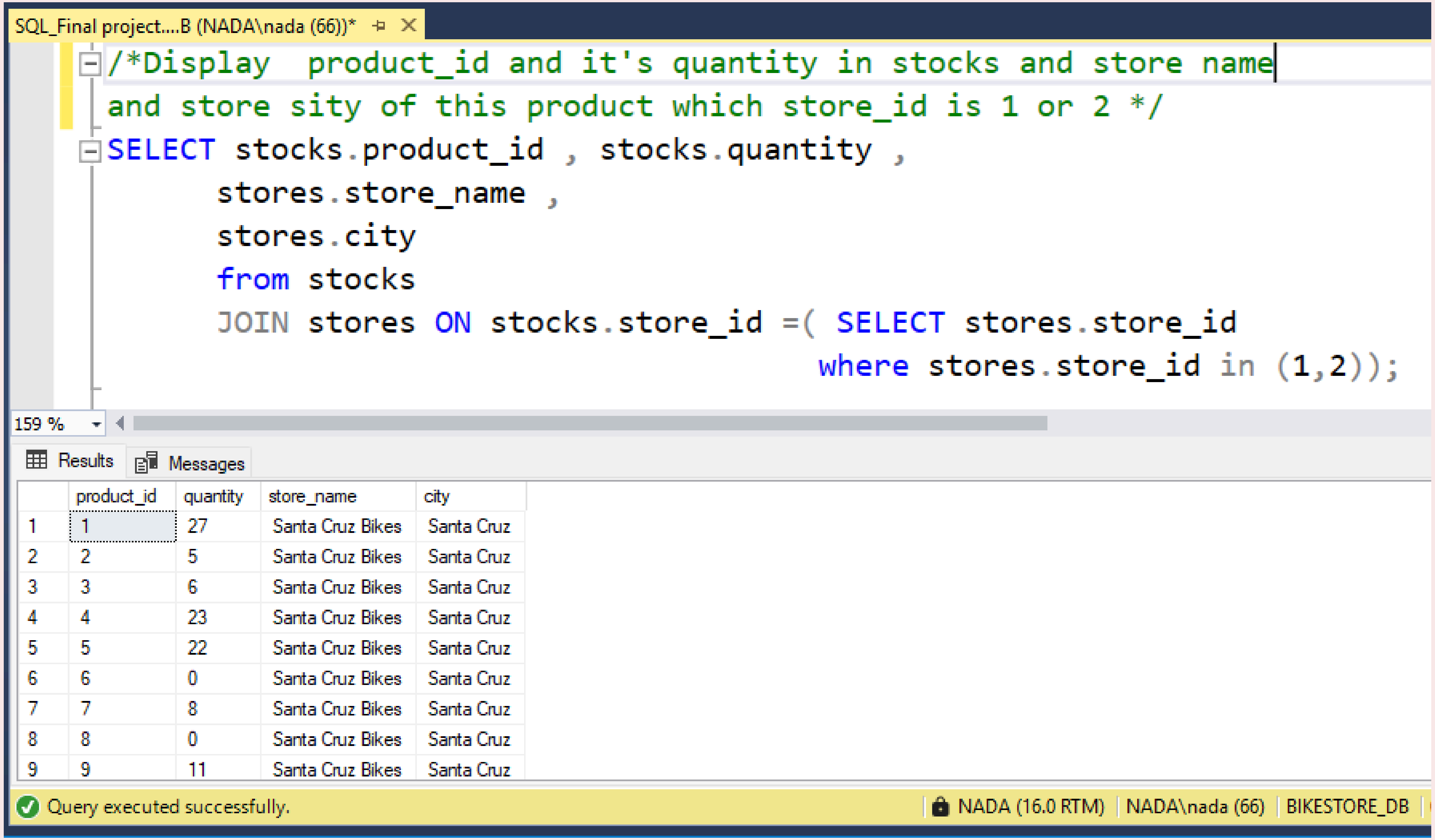
```
--3 'JOINS Quries'

--Display order_id , order_date , customer full name , customer city
SELECT O.order_id , O.order_date ,
       concat(customers.first_name,' ',customers.last_name) AS C_fullname,
       customers.city
       from orders O
       JOIN customers ON O.customer_id = customers.customer_id;
```

	order_id	order_date	C_fullname	city
1	599	2016-12-09	Debra Burks	Orchard Park
2	1555	2018-04-18	Debra Burks	Orchard Park
3	1613	2018-11-18	Debra Burks	Orchard Park
4	1509	2018-04-09	Kasha Todd	Campbell
5	692	2017-02-05	Kasha Todd	Campbell
6	1084	2017-08-21	Kasha Todd	Campbell
7	1496	2018-04-06	Tameka Fisher	Redondo Beach
8	1612	2018-10-21	Tameka Fisher	Redondo Beach
9	1468	2018-03-27	Tameka Fisher	Redondo Beach

Query executed successfully. | NADA (16.0 RTM) | NADA\nada (66) | BIKESTORE_DB | 00:00:00 | 1,615 rows

Ln 101 Col 19 Ch 19 INS

Here this query display product_id & it’s quantity in store num 1 & 2 notice that there is 626 product

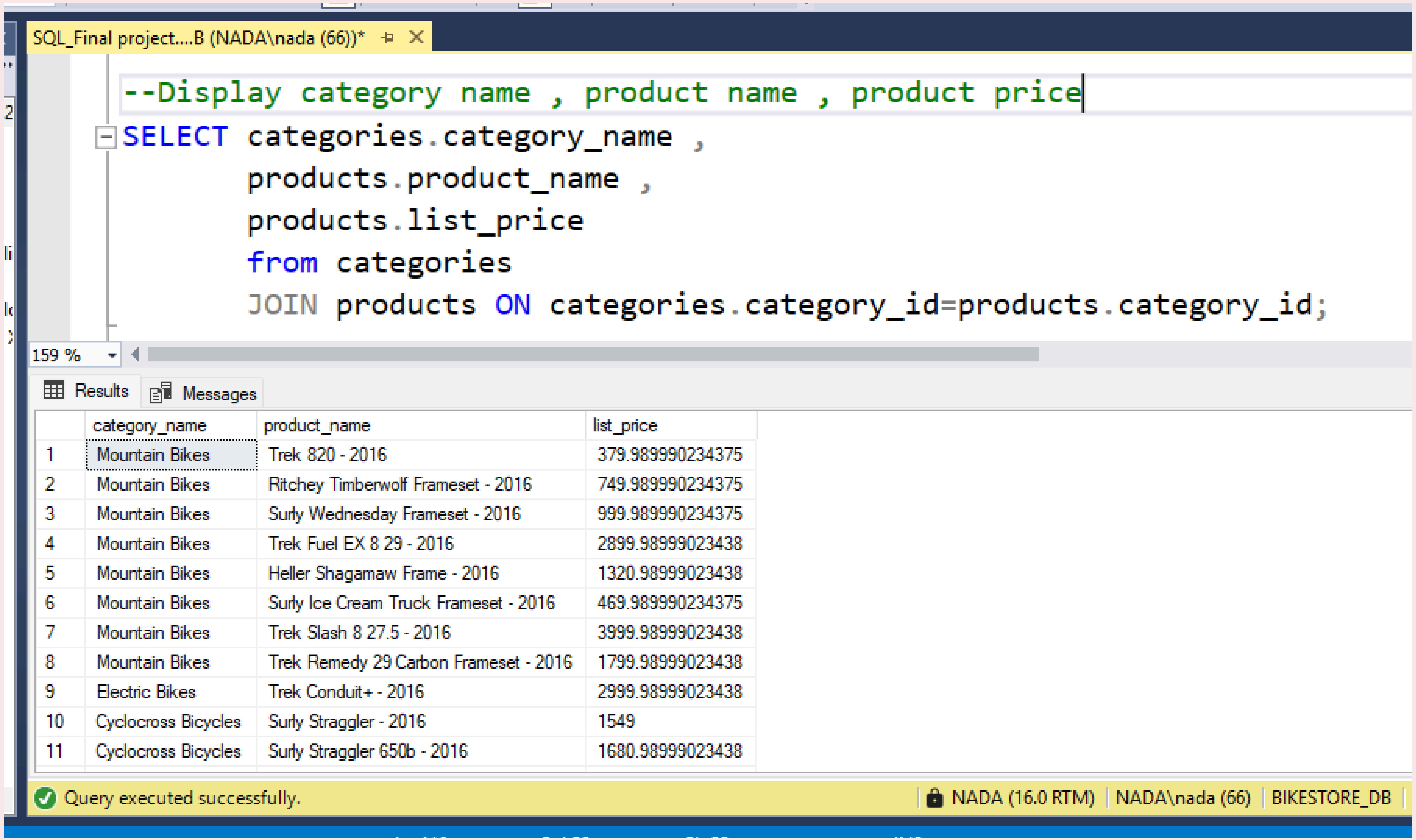


```
/*Display  product_id and it's quantity in stocks and store name
and store sity of this product which store_id is 1 or 2 */
SELECT stocks.product_id , stocks.quantity ,
      stores.store_name ,
      stores.city
      from stocks
      JOIN stores ON stocks.store_id =( SELECT stores.store_id
                                        where stores.store_id in (1,2));
```

	product_id	quantity	store_name	city
1	1	27	Santa Cruz Bikes	Santa Cruz
2	2	5	Santa Cruz Bikes	Santa Cruz
3	3	6	Santa Cruz Bikes	Santa Cruz
4	4	23	Santa Cruz Bikes	Santa Cruz
5	5	22	Santa Cruz Bikes	Santa Cruz
6	6	0	Santa Cruz Bikes	Santa Cruz
7	7	8	Santa Cruz Bikes	Santa Cruz
8	8	0	Santa Cruz Bikes	Santa Cruz
9	9	11	Santa Cruz Bikes	Santa Cruz

Query executed successfully. | NADA (16.0 RTM) | NADA\nada (66) | BIKESTORE_DB

“Here the results display each category name and the products of it (product _n and it’ price)”



```
--Display category name , product name , product price
SELECT categories.category_name ,
       products.product_name ,
       products.list_price
       from categories
       JOIN products ON categories.category_id=products.category_id;
```

	category_name	product_name	list_price
1	Mountain Bikes	Trek 820 - 2016	379.989990234375
2	Mountain Bikes	Ritchey Timberwolf Frameset - 2016	749.989990234375
3	Mountain Bikes	Surly Wednesday Frameset - 2016	999.989990234375
4	Mountain Bikes	Trek Fuel EX 8 29 - 2016	2899.98999023438
5	Mountain Bikes	Heller Shagamaw Frame - 2016	1320.98999023438
6	Mountain Bikes	Surly Ice Cream Truck Frameset - 2016	469.989990234375
7	Mountain Bikes	Trek Slash 8 27.5 - 2016	3999.98999023438
8	Mountain Bikes	Trek Remedy 29 Carbon Frameset - 2016	1799.98999023438
9	Electric Bikes	Trek Conduit+ - 2016	2999.98999023438
10	Cyclocross Bicycles	Surly Straggler - 2016	1549
11	Cyclocross Bicycles	Surly Straggler 650b - 2016	1680.98999023438

Query executed successfully.

ANY QUESTION ?

THANK YOU