Activity #2

Name: Davila, John E.
Section: BSIT – 2D

Instructor: Mr. Alfred Ocampo
Subject: FDBS

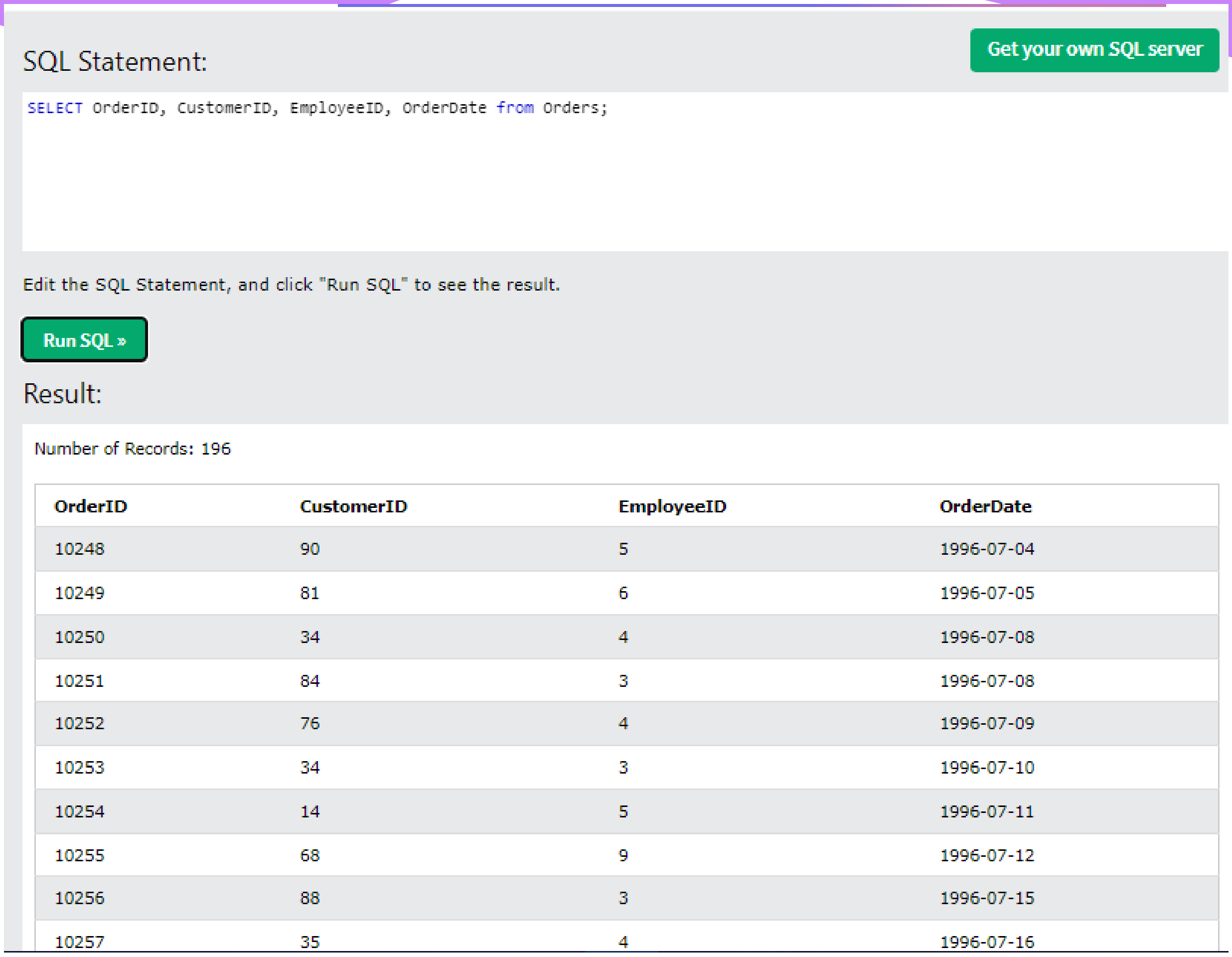
Table of Contents

Fundamental of Database Systems

Min and Max

The MIN() function returns the smallest value of the selected column.

Before using MIN()



SQL Statement:

```
SELECT OrderID, CustomerID, EmployeeID, OrderDate from Orders;
```

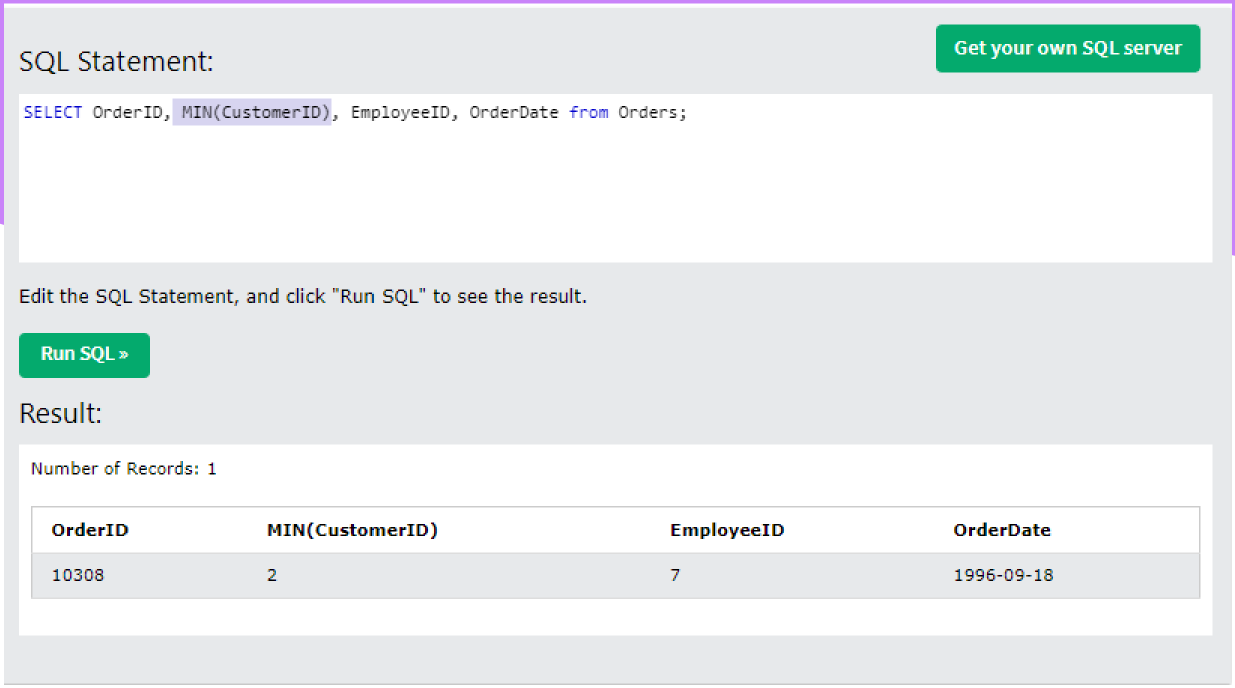
Edit the SQL Statement, and click "Run SQL" to see the result.

Result:

Number of Records: 196

OrderID	CustomerID	EmployeeID	OrderDate
10248	90	5	1996-07-04
10249	81	6	1996-07-05
10250	34	4	1996-07-08
10251	84	3	1996-07-08
10252	76	4	1996-07-09
10253	34	3	1996-07-10
10254	14	5	1996-07-11
10255	68	9	1996-07-12
10256	88	3	1996-07-15
10257	35	4	1996-07-16

After using MIN()



SQL Statement:

Get your own SQL server

```
SELECT OrderID, MIN(CustomerID), EmployeeID, OrderDate from Orders;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

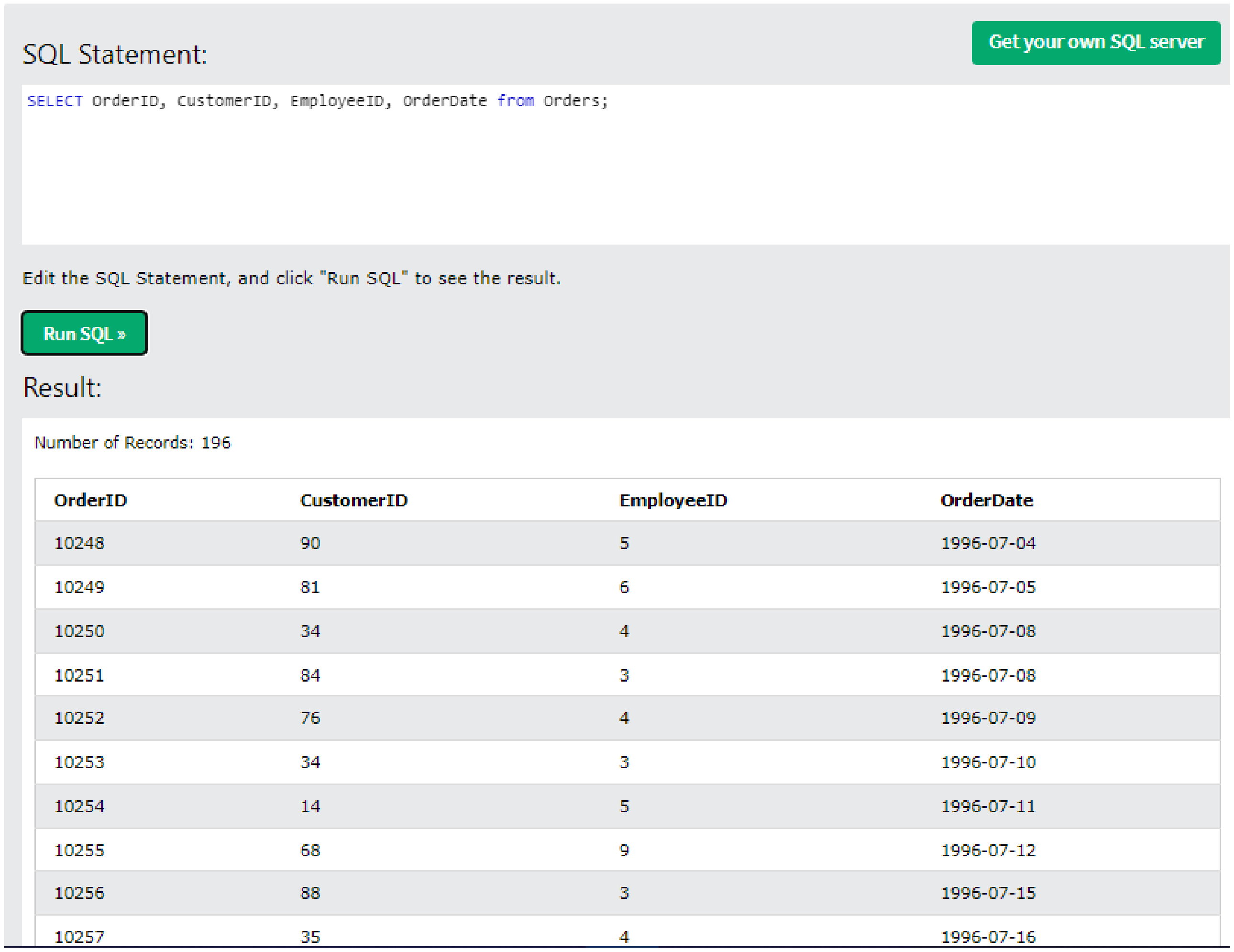
Result:

Number of Records: 1

OrderID	MIN(CustomerID)	EmployeeID	OrderDate
10308	2	7	1996-09-18

The MAX() function returns the largest value of the selected column.

Before using MAX()



SQL Statement:

Get your own SQL server

```
SELECT OrderID, CustomerID, EmployeeID, OrderDate from Orders;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

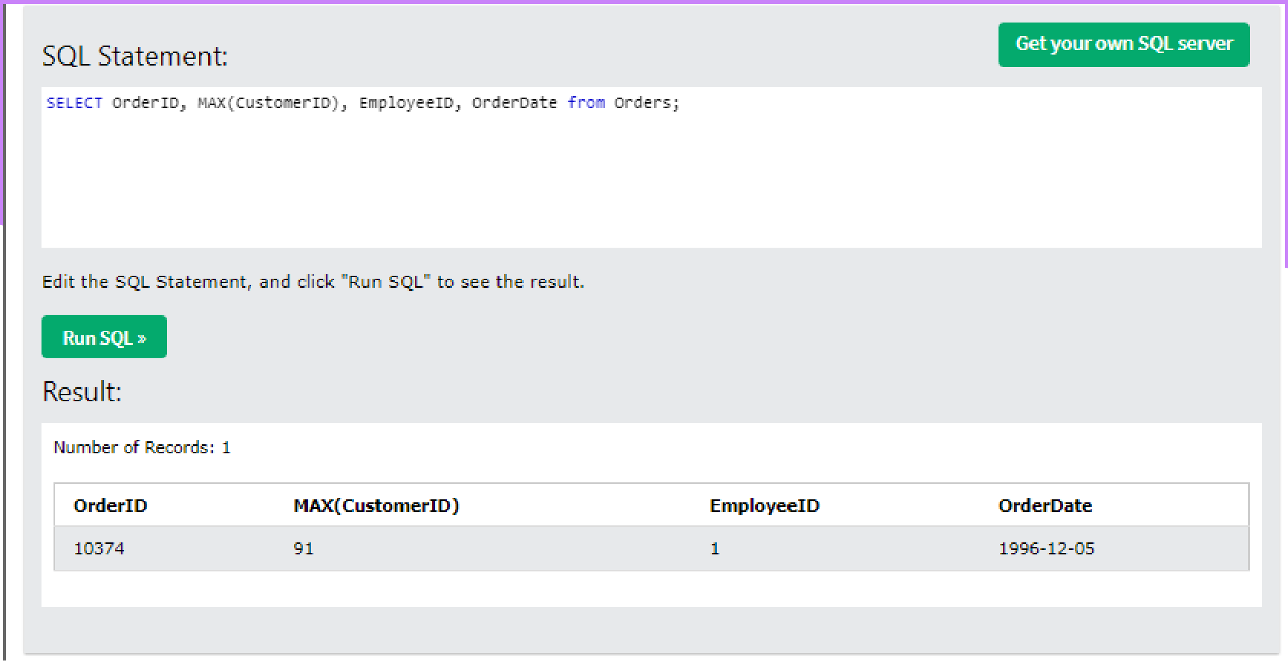
Run SQL »

Result:

Number of Records: 196

OrderID	CustomerID	EmployeeID	OrderDate
10248	90	5	1996-07-04
10249	81	6	1996-07-05
10250	34	4	1996-07-08
10251	84	3	1996-07-08
10252	76	4	1996-07-09
10253	34	3	1996-07-10
10254	14	5	1996-07-11
10255	68	9	1996-07-12
10256	88	3	1996-07-15
10257	35	4	1996-07-16

After using MAX()



SQL Statement:

```
SELECT OrderID, MAX(CustomerID), EmployeeID, OrderDate from Orders;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Result:

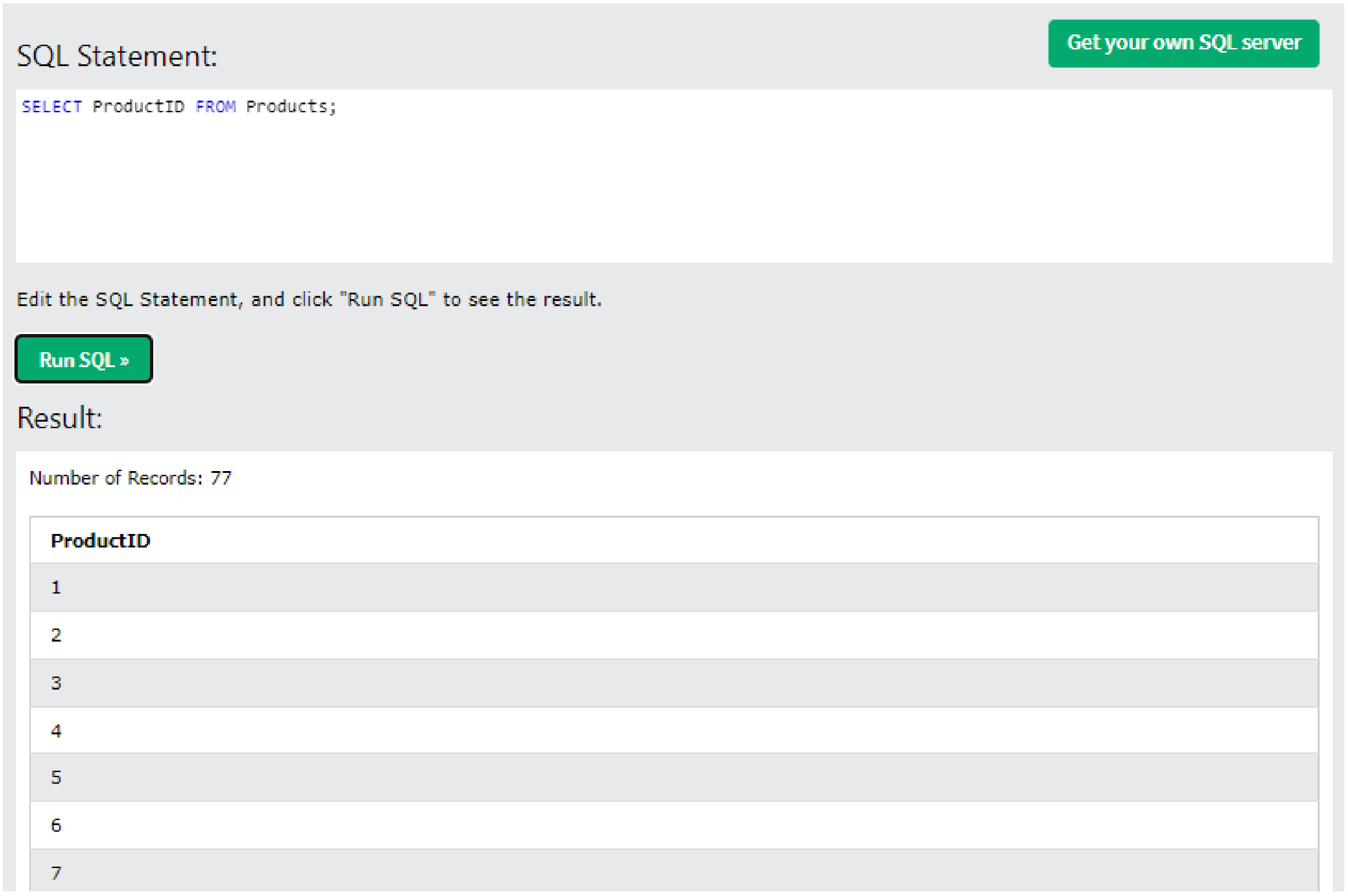
Number of Records: 1

OrderID	MAX(CustomerID)	EmployeeID	OrderDate
10374	91	1	1996-12-05

Count Ave Sum

The COUNT() function returns the number of rows that matches a specified criterion.

Before using COUNT()



SQL Statement:

```
SELECT ProductID FROM Products;
```

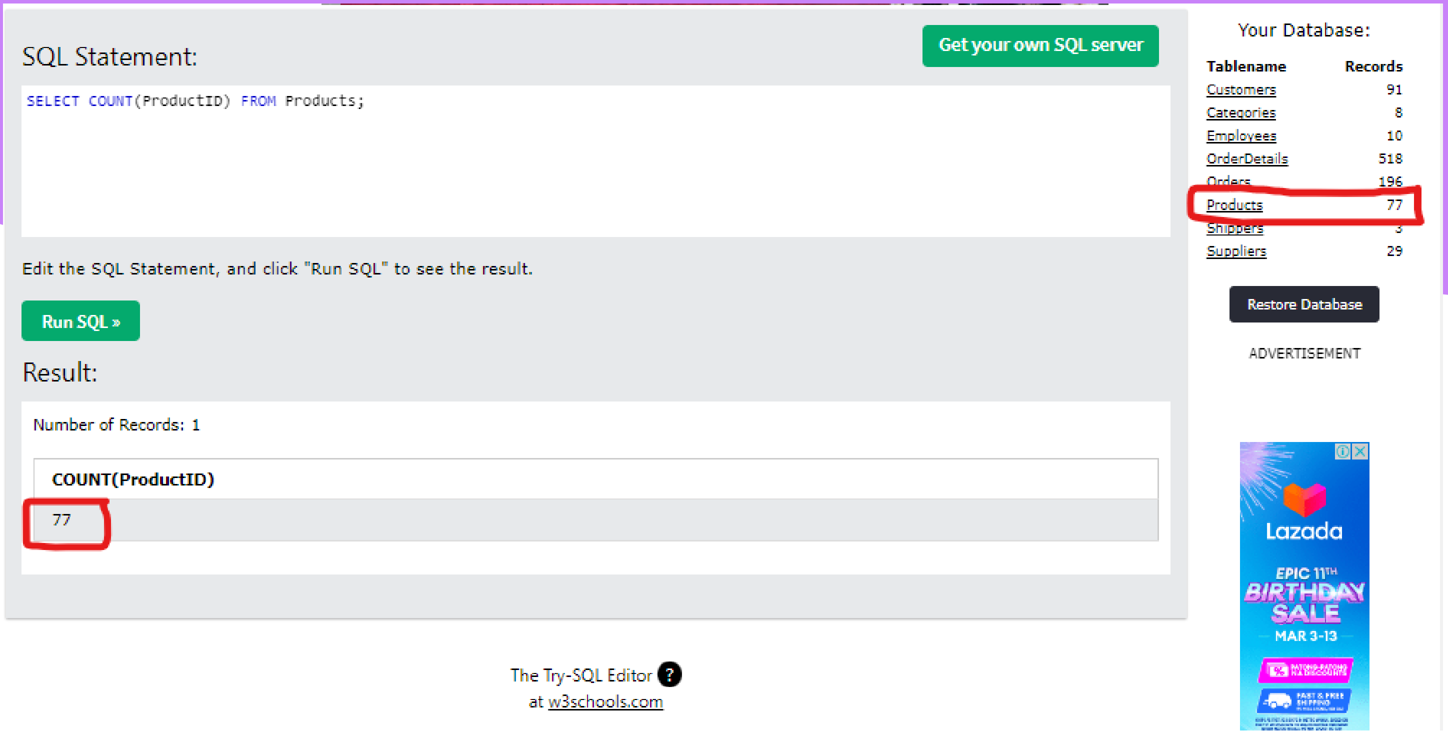
Edit the SQL Statement, and click "Run SQL" to see the result.

Result:

Number of Records: 77

ProductID
1
2
3
4
5
6
7

After using COUNT()



SQL Statement:

```
SELECT COUNT(ProductID) FROM Products;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Result:

Number of Records: 1

COUNT(ProductID)
77

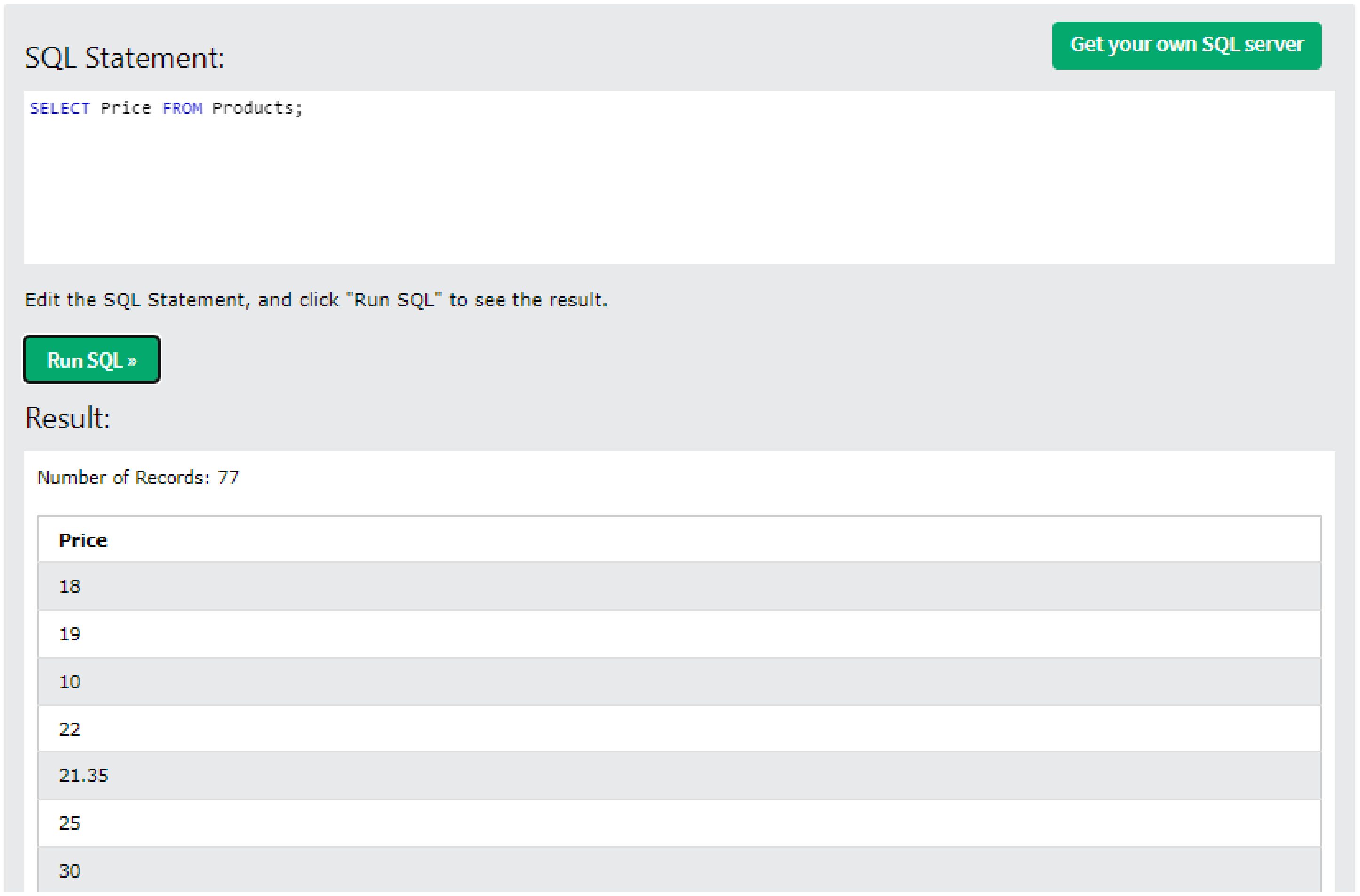
Your Database:

Tablename	Records
Customers	91
Categories	8
Employees	10
OrderDetails	518
Orders	196
Products	77
Shippers	3
Suppliers	29

The Try-SQL Editor ?
at w3schools.com

The AVG() function returns the average value of a numeric column.

Before using AVG()



SQL Statement:

```
SELECT Price FROM Products;
```

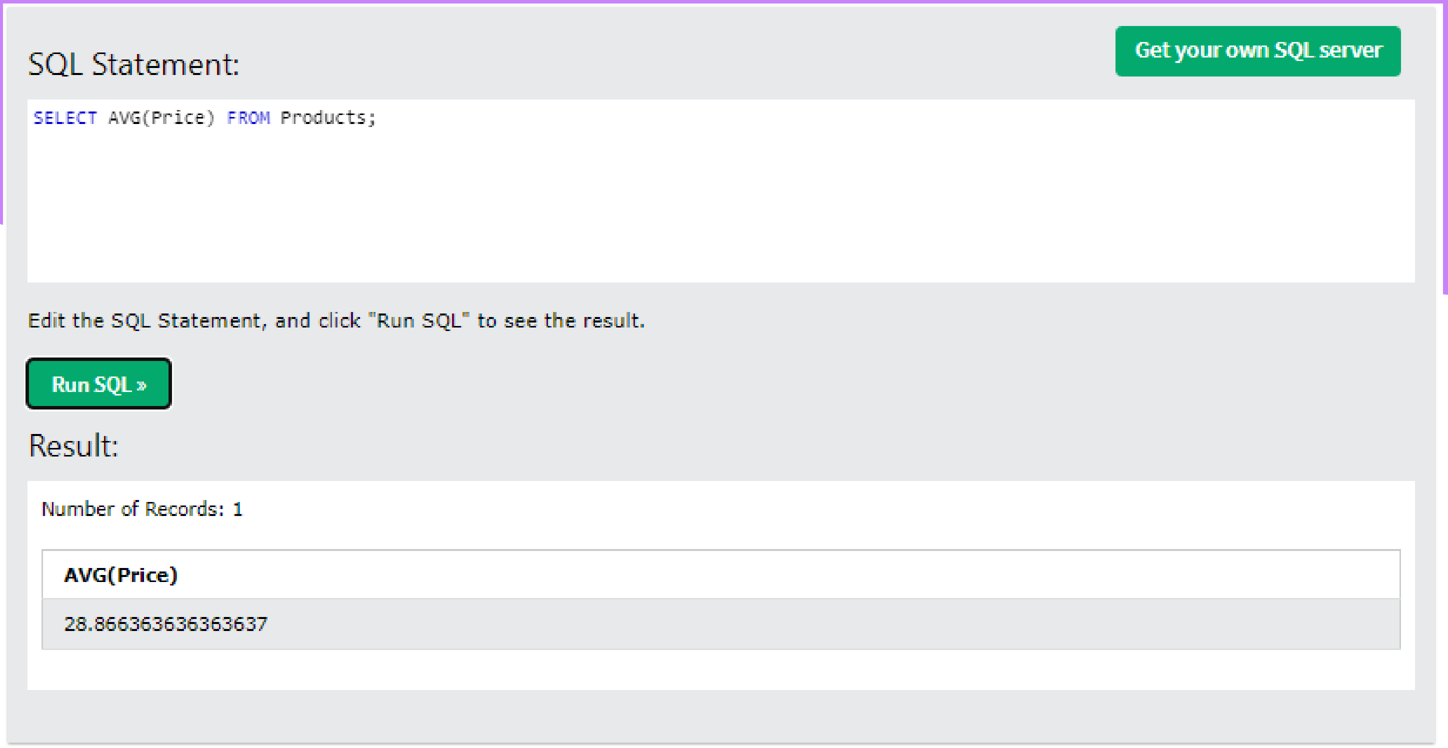
Edit the SQL Statement, and click "Run SQL" to see the result.

Result:

Number of Records: 77

Price
18
19
10
22
21.35
25
30

After using AVG()



```
SELECT AVG(Price) FROM Products;
```

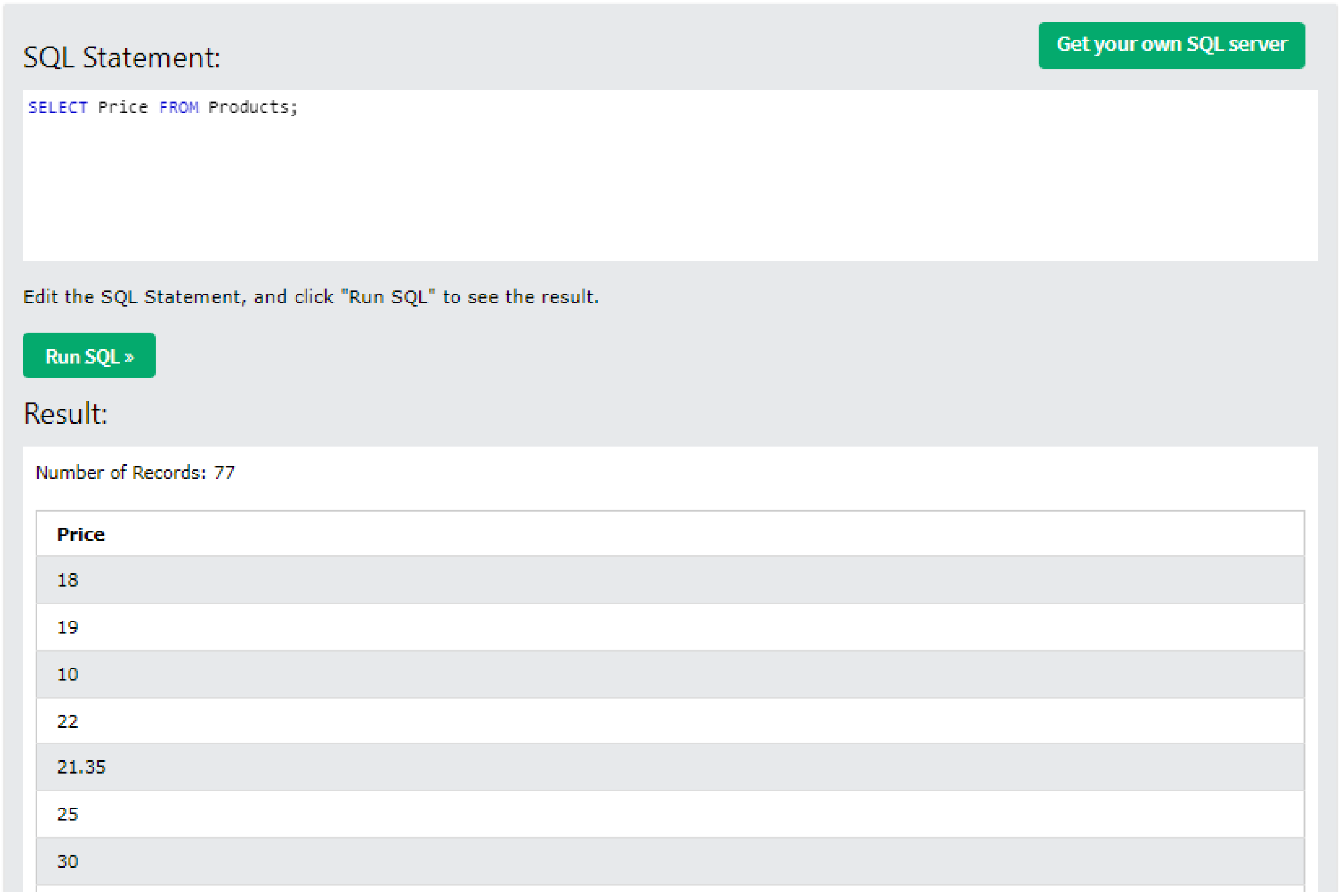
Edit the SQL Statement, and click "Run SQL" to see the result.

Number of Records: 1

AVG(Price)
28.866363636363637

The SUM() function returns the total sum of a numeric column.

Before using SUM()



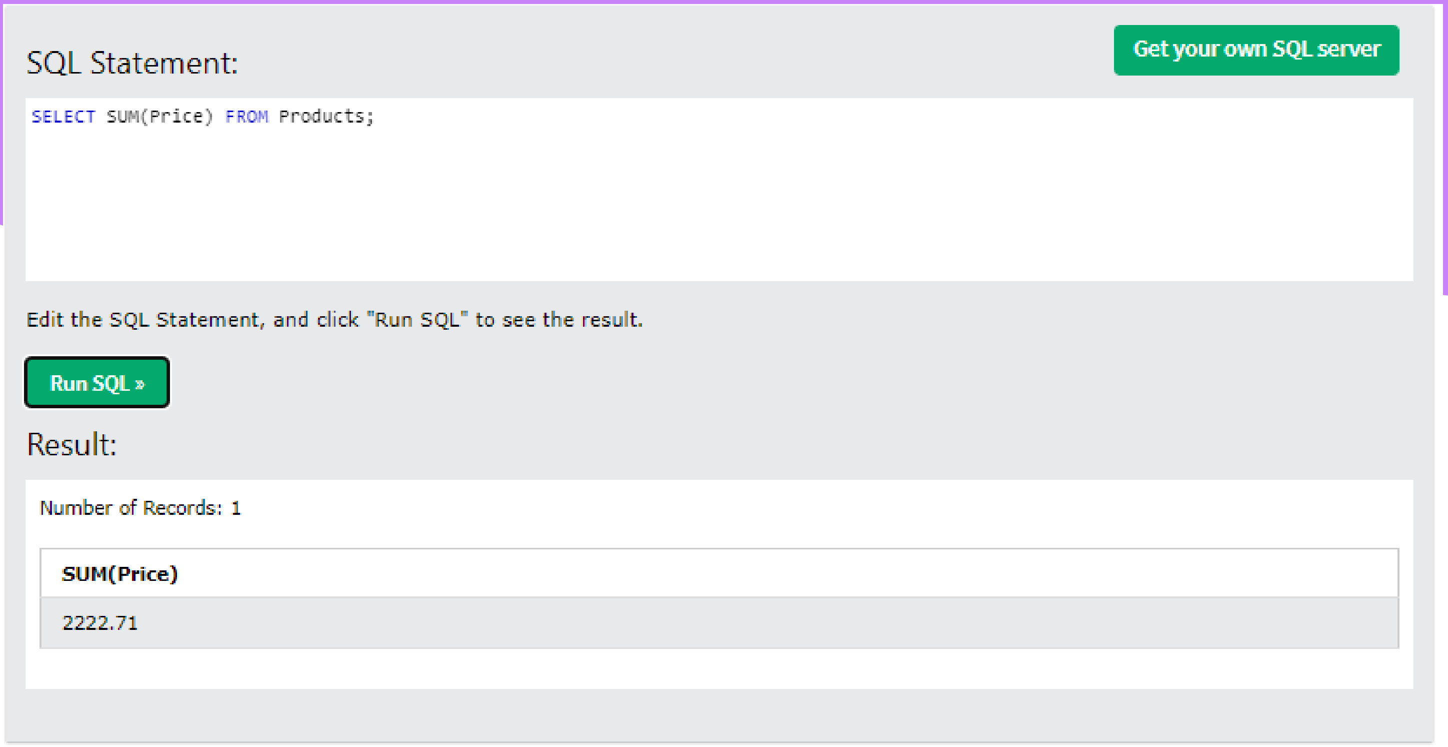
```
SELECT Price FROM Products;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Number of Records: 77

Price
18
19
10
22
21.35
25
30

After using SUM()



SQL Statement:

```
SELECT SUM(Price) FROM Products;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Result:

Number of Records: 1

SUM(Price)
2222.71

Like

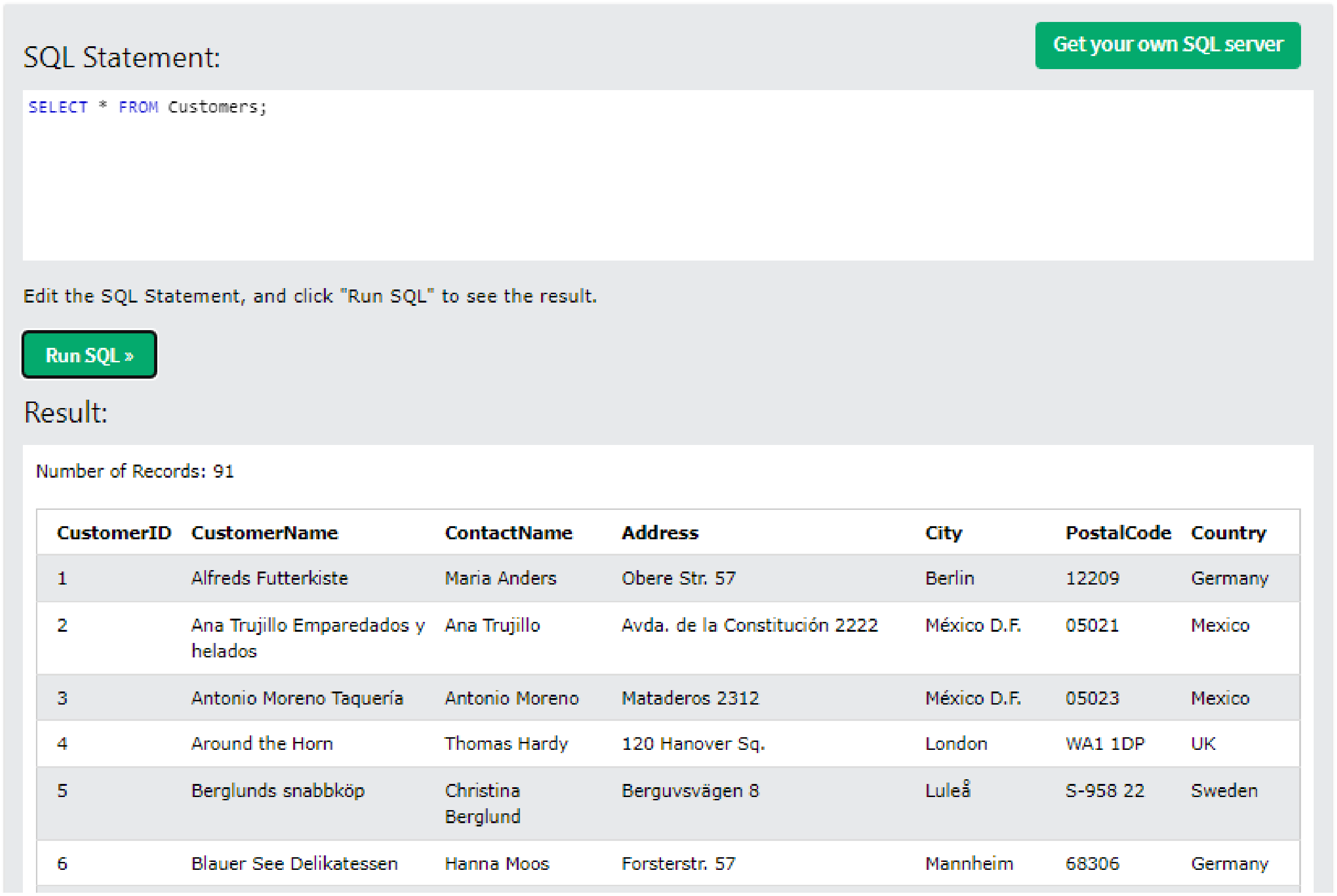
The LIKE operator is used in a WHERE clause to search for a specified pattern in a column.

There are two wildcards often used in conjunction with the LIKE operator:

- The percent sign (%) represents zero, one, or multiple characters
- The underscore sign (_) represents one, single character
- Both of these wildcards can be combined to fetch a specific data

I will fetch columns if the country starts with letter “G”.

Before using LIKE()



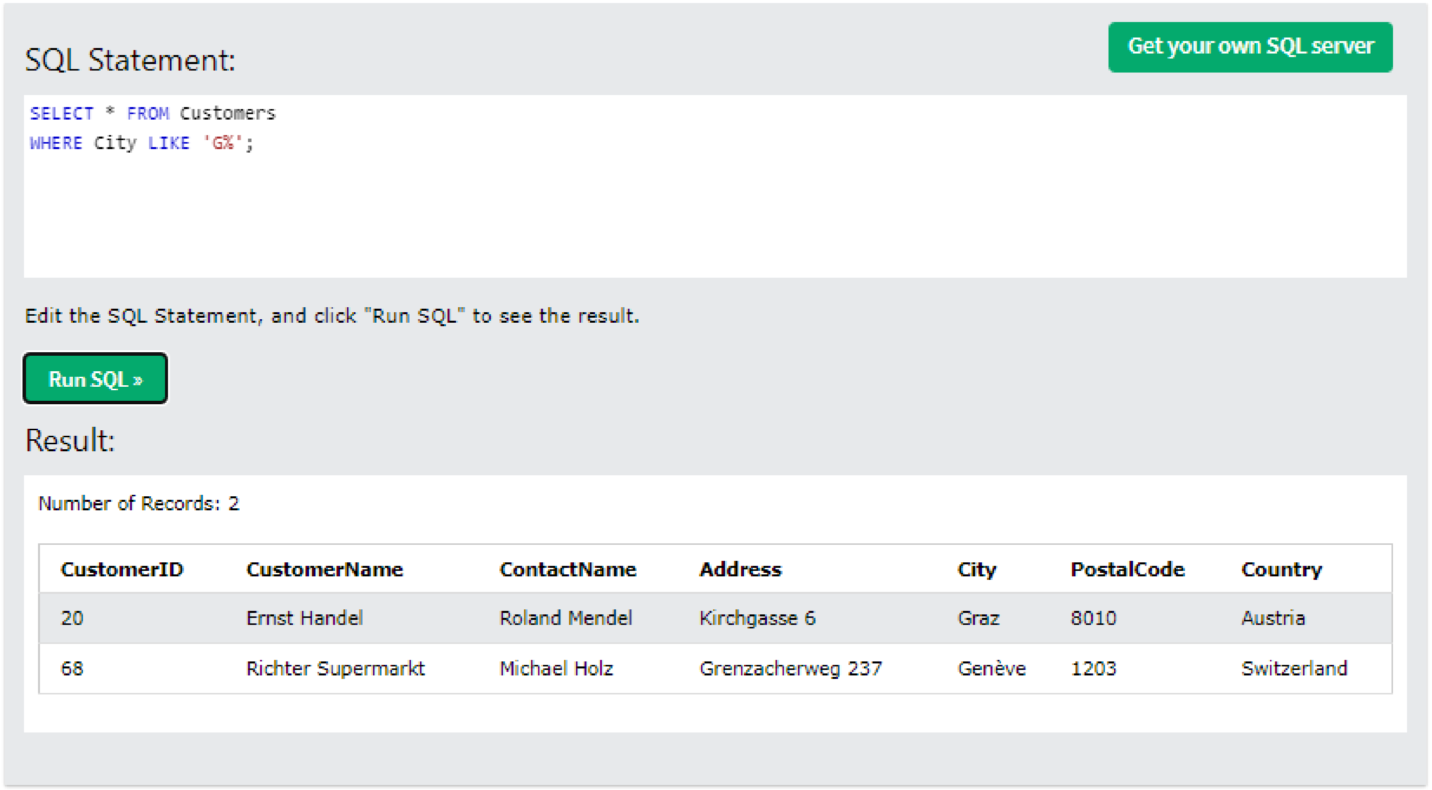
```
SELECT * FROM Customers;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Number of Records: 91

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden
6	Blauer See Delikatessen	Hanna Moos	Forsterstr. 57	Mannheim	68306	Germany

After using LIKE()



```
SELECT * FROM Customers
WHERE City LIKE 'G%';
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Number of Records: 2

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
20	Ernst Handel	Roland Mendel	Kirchgasse 6	Graz	8010	Austria
68	Richter Supermarkt	Michael Holz	Grenzacherweg 237	Genève	1203	Switzerland

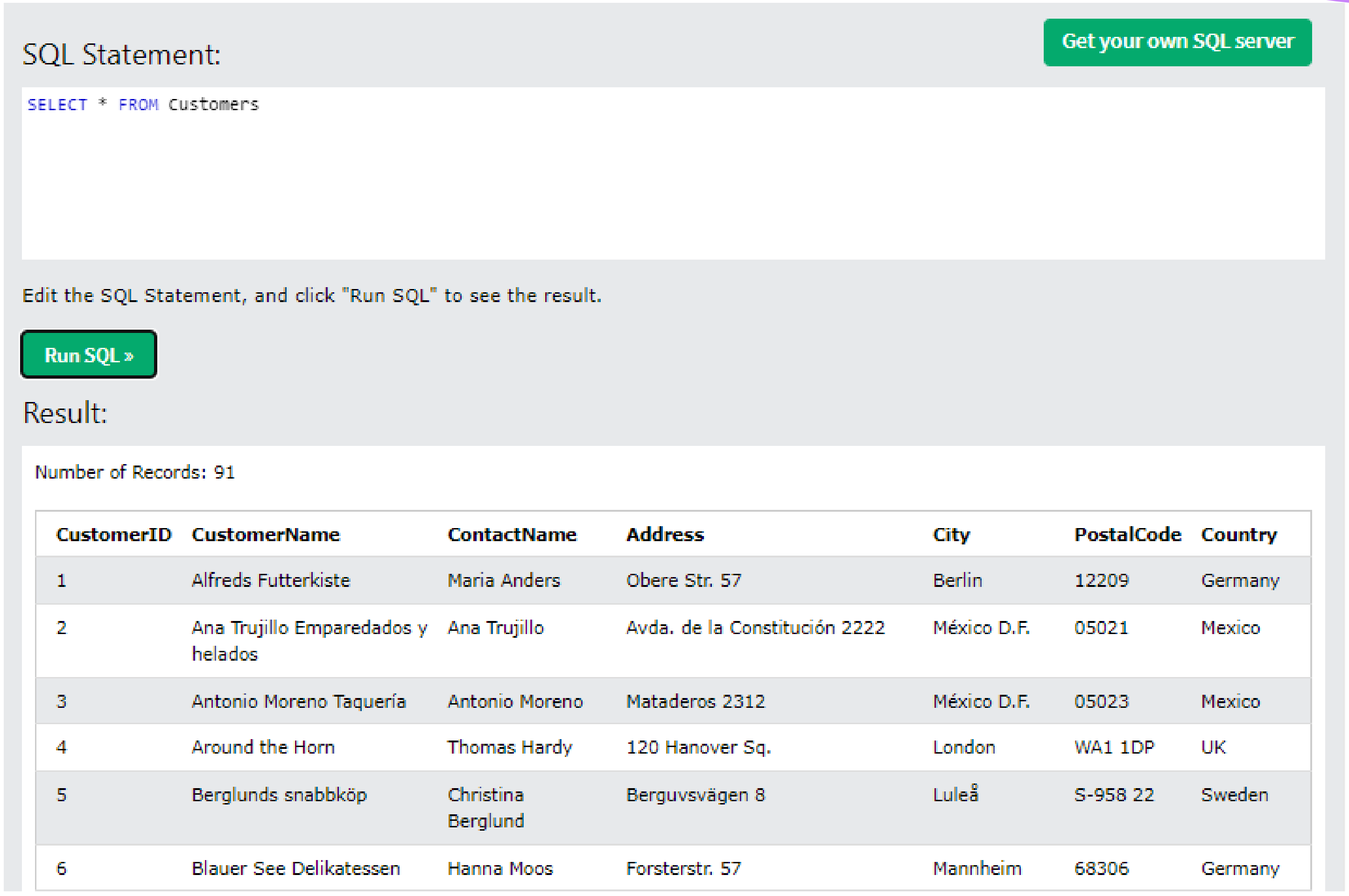
In

The IN operator allows you to specify multiple values in a WHERE clause.

The IN operator is a shorthand for multiple OR conditions.

I will fetch only fetch columns if their city is one of the these following: London, Strasbourg, Munchen.

Before using IN()



SQL Statement:

```
SELECT * FROM Customers
```

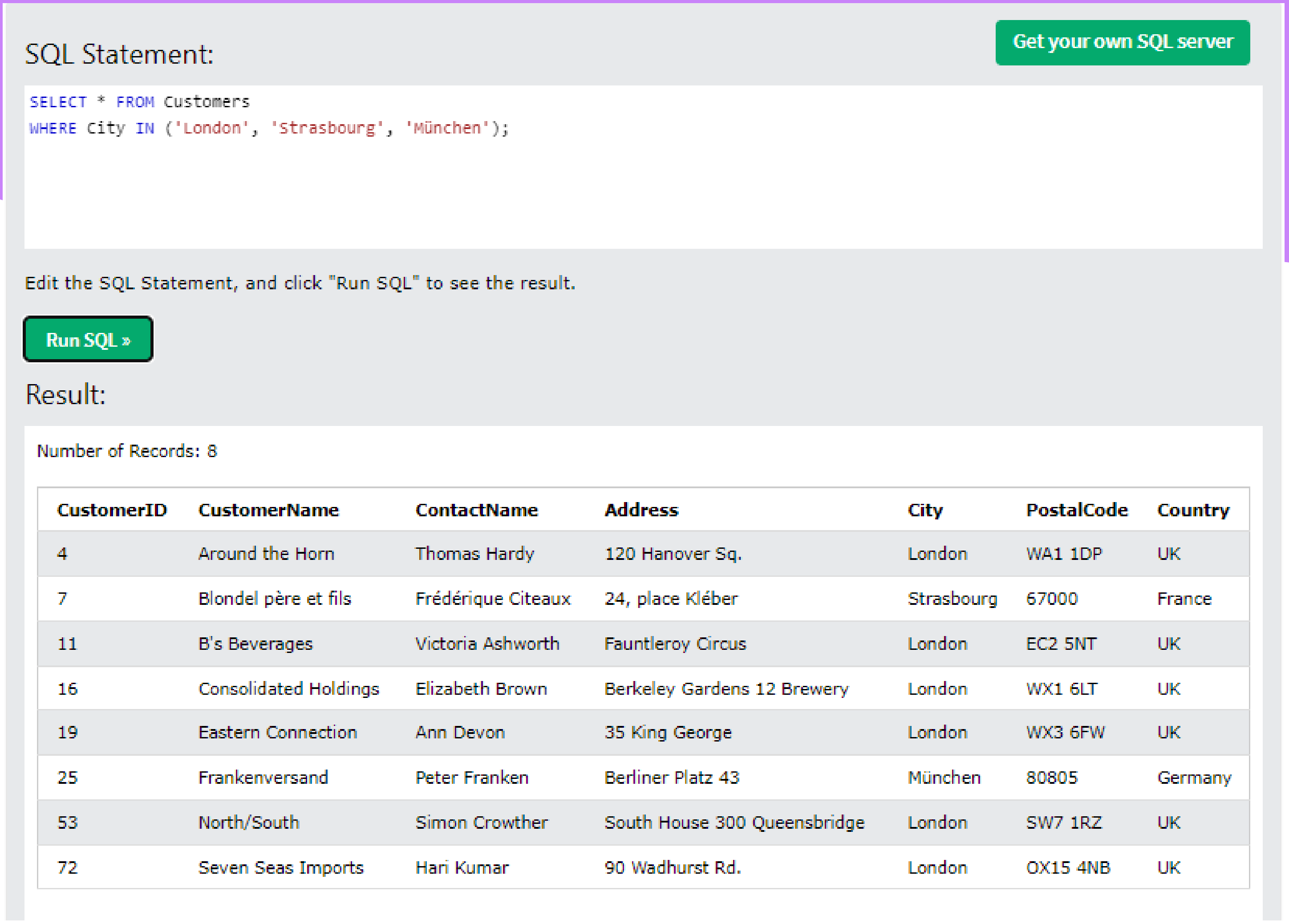
Edit the SQL Statement, and click "Run SQL" to see the result.

Result:

Number of Records: 91

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden
6	Blauer See Delikatessen	Hanna Moos	Forsterstr. 57	Mannheim	68306	Germany

After using IN()



SQL Statement:

Get your own SQL server

```
SELECT * FROM Customers
WHERE City IN ('London', 'Strasbourg', 'München');
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 8

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK
7	Blondel père et fils	Frédérique Citeaux	24, place Kléber	Strasbourg	67000	France
11	B's Beverages	Victoria Ashworth	Fauntleroy Circus	London	EC2 5NT	UK
16	Consolidated Holdings	Elizabeth Brown	Berkeley Gardens 12 Brewery	London	WX1 6LT	UK
19	Eastern Connection	Ann Devon	35 King George	London	WX3 6FW	UK
25	Frankenversand	Peter Franken	Berliner Platz 43	München	80805	Germany
53	North/South	Simon Crowther	South House 300 Queensbridge	London	SW7 1RZ	UK
72	Seven Seas Imports	Hari Kumar	90 Wadhurst Rd.	London	OX15 4NB	UK

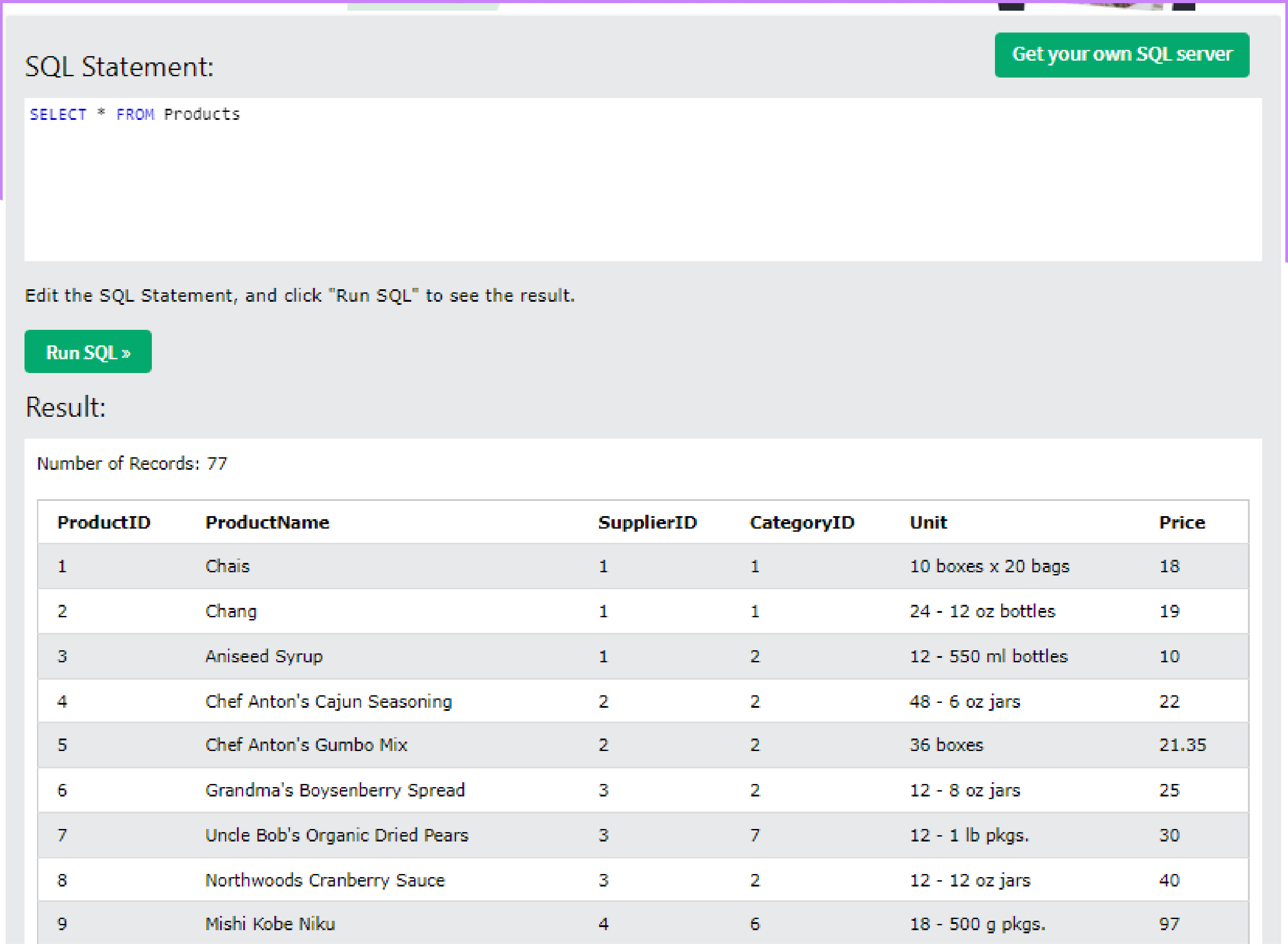
Between

The BETWEEN operator selects values within a given range. The values can be numbers, text, or dates.

The BETWEEN operator is inclusive: begin and end values are included.

I will fetch only fetch columns if the supplier id is in between 3 to 4.

Before using BETWEEN()



SQL Statement:

```
SELECT * FROM Products
```

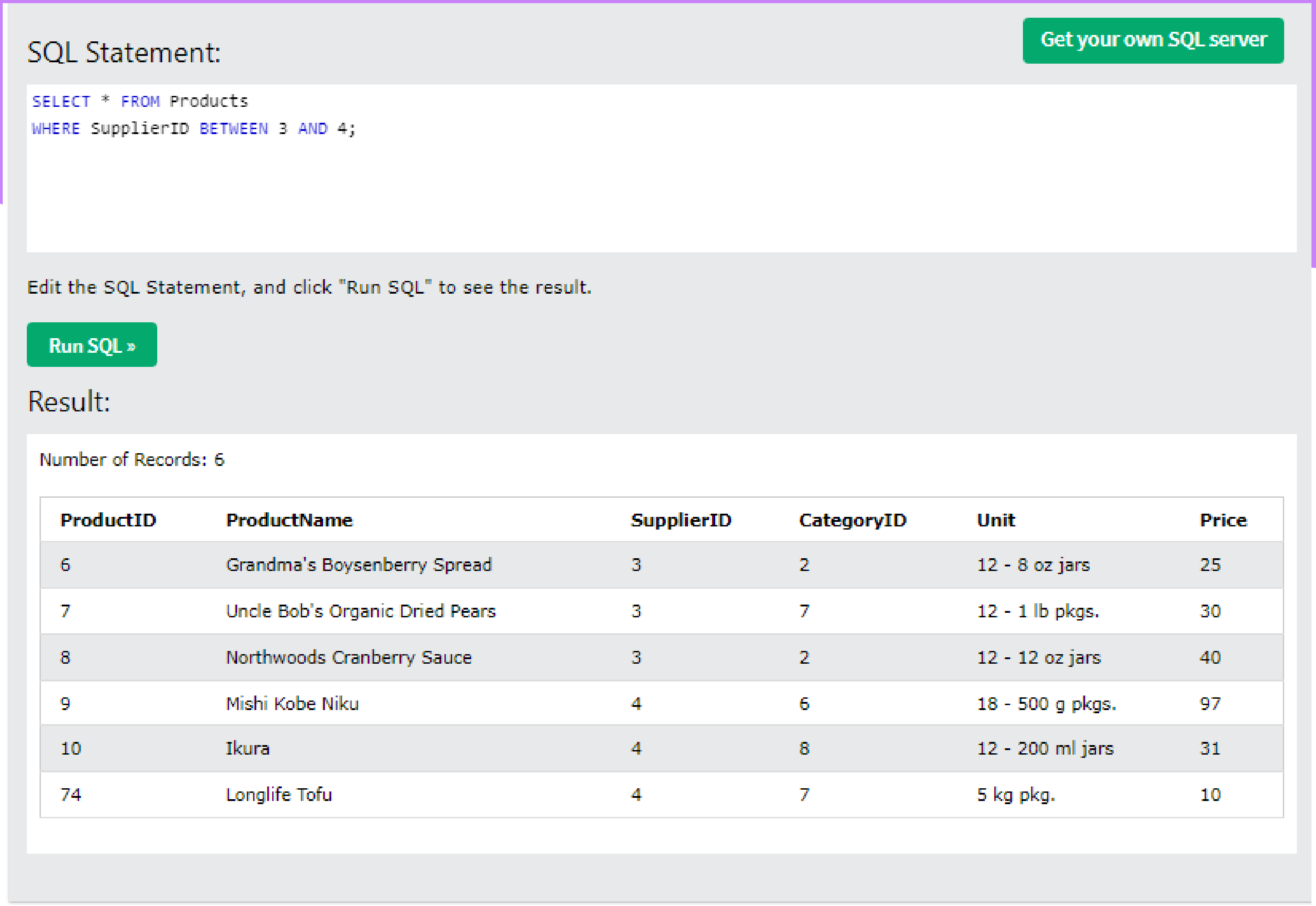
Edit the SQL Statement, and click "Run SQL" to see the result.

Result:

Number of Records: 77

ProductID	ProductName	SupplierID	CategoryID	Unit	Price
1	Chais	1	1	10 boxes x 20 bags	18
2	Chang	1	1	24 - 12 oz bottles	19
3	Aniseed Syrup	1	2	12 - 550 ml bottles	10
4	Chef Anton's Cajun Seasoning	2	2	48 - 6 oz jars	22
5	Chef Anton's Gumbo Mix	2	2	36 boxes	21.35
6	Grandma's Boysenberry Spread	3	2	12 - 8 oz jars	25
7	Uncle Bob's Organic Dried Pears	3	7	12 - 1 lb pkgs.	30
8	Northwoods Cranberry Sauce	3	2	12 - 12 oz jars	40
9	Mishi Kobe Niku	4	6	18 - 500 g pkgs.	97

After using BETWEEN()



SQL Statement:

```
SELECT * FROM Products
WHERE SupplierID BETWEEN 3 AND 4;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Result:

Number of Records: 6

ProductID	ProductName	SupplierID	CategoryID	Unit	Price
6	Grandma's Boysenberry Spread	3	2	12 - 8 oz jars	25
7	Uncle Bob's Organic Dried Pears	3	7	12 - 1 lb pkgs.	30
8	Northwoods Cranberry Sauce	3	2	12 - 12 oz jars	40
9	Mishi Kobe Niku	4	6	18 - 500 g pkgs.	97
10	Ikura	4	8	12 - 200 ml jars	31
74	Longlife Tofu	4	7	5 kg pkg.	10