

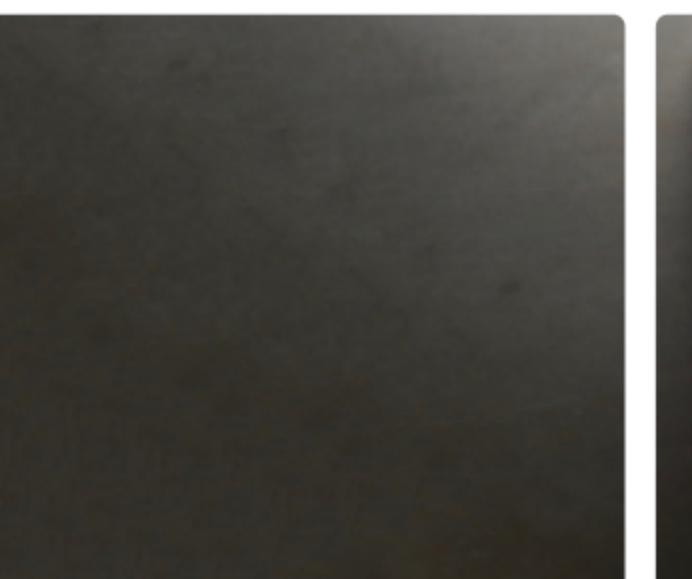
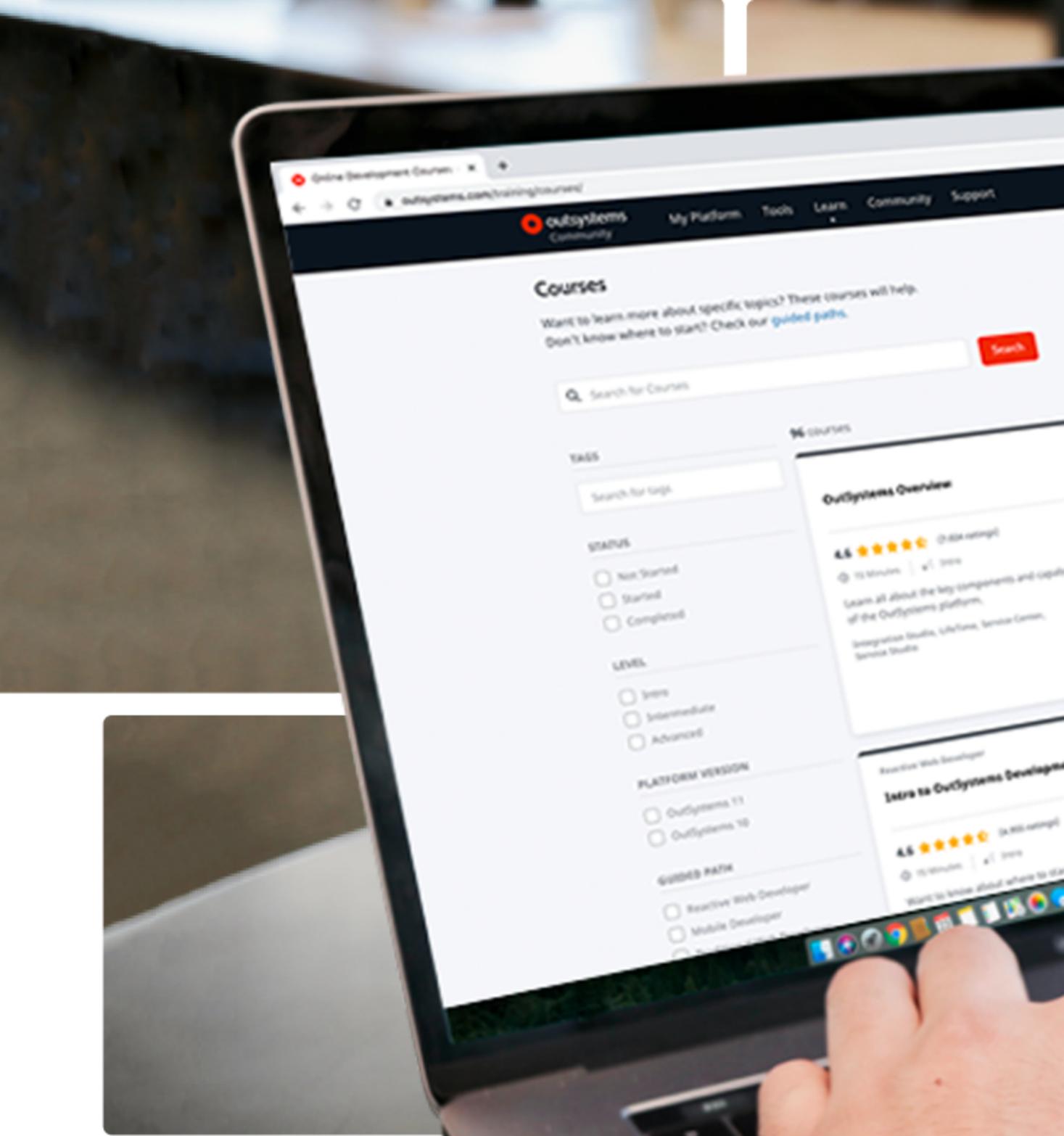


Aggregates Outputs and Properties

Outputs

Properties

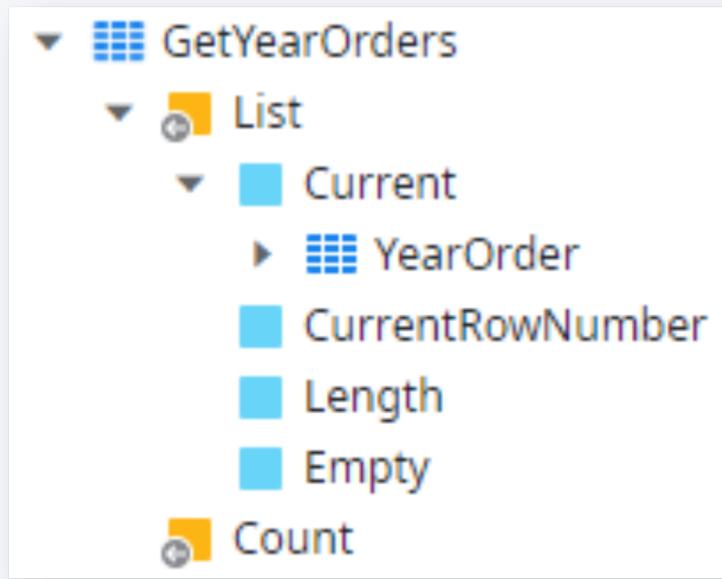
Aggregates as SQL



Outputs

List of Records returned

- Iterator: Current, CurrentRowNumber

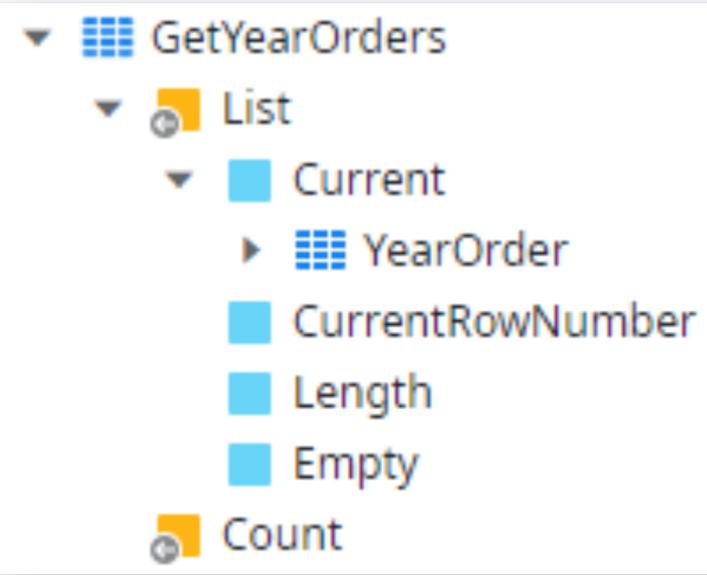


- *Length*
- *Empty*

Outputs

List of Records returned

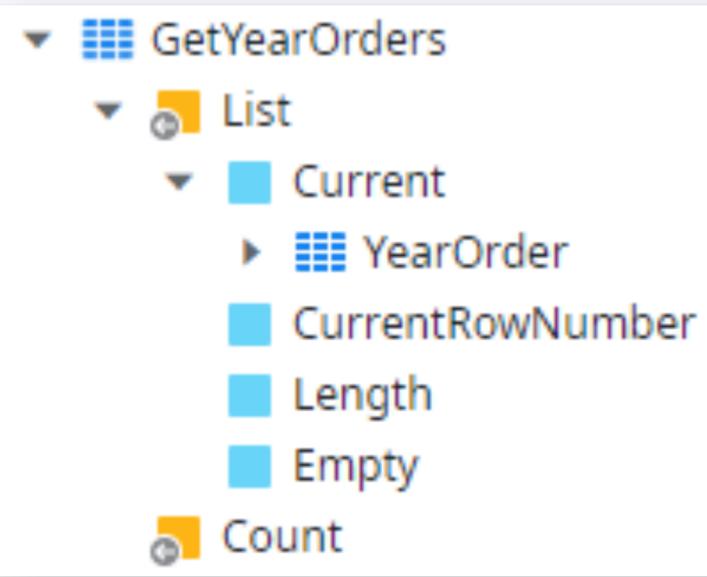
- Iterator: *Current*, *CurrentRowNumber*
 - The *Current* cursor moves through the List when iterating
 - Type matches the definition in the query
 - *Current* points at first row by default
 - Filled in with default values if no rows returned by query
- *Length*
- *Empty*



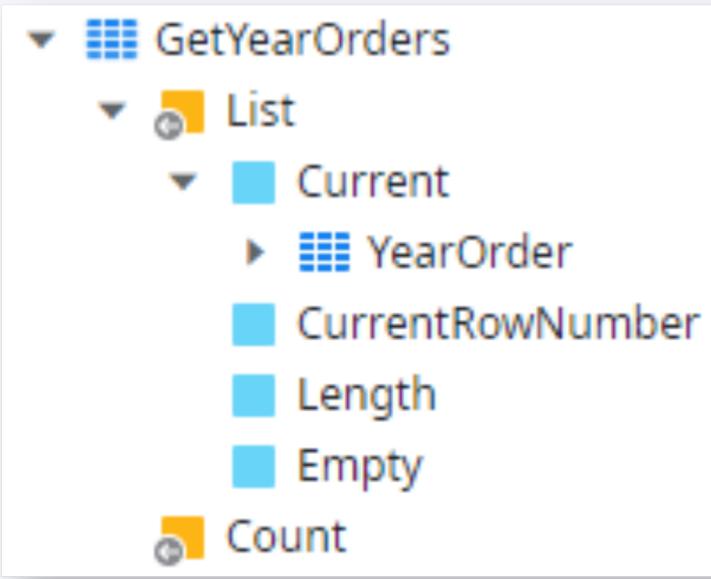
Outputs

List of Records returned

- Iterator: *Current*, *CurrentRowNumber*
 - The *Current* cursor moves through the List when iterating
 - Type matches the definition in the query
 - *Current* points at first row by default
 - Filled in with default values if no rows returned by query
- *Length* - number of elements returned
- *Empty*



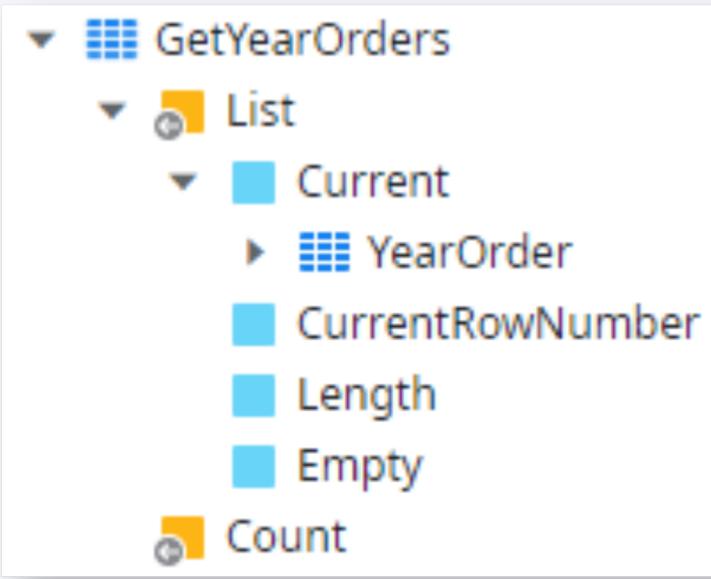
Outputs



List of Records returned

- Iterator: *Current*, *CurrentRowNumber*
 - The Current cursor moves through the List when iterating
 - Type matches the definition in the query
 - Current points at first row by default
 - Filled in with default values if no rows returned by query
- Length* - number of elements returned
- Empty* - True if no records were returned

Outputs



List of Records returned

- Iterator: *Current*, *CurrentRowNumber*
 - The Current cursor moves through the List when iterating
 - Type matches the definition in the query
 - Current points at first row by default
 - Filled in with default values if no rows returned by query
- Length - number of elements returned
- Empty - True if no records were returned

Count has the total number of records that match the criterias defined in the Aggregate

Properties

 **GetYearOrders**
Aggregate

Name	GetYearOrders
Description	
Server Request Time...	(Module Default Timeout)
Start Index	StartIndex2
Max. Records	10
Fetch	At start
Events	
On After Fetch	
Executed SQL	SELECT TOP (10) 
Sources	
 Order	
Filters	
 Year(Order.CreatedOn) = Year	
Sorting	
 Order.CreatedOn (DESC)	
 Order.Priority (DESC)	
Test Values	
 Year	

Properties

It is possible to limit the Aggregate's output to a maximum number of records

- **Max. Records** property
- Does not impact the Count output

The screenshot shows the configuration interface for the 'GetYearOrders' Aggregate. The 'Max. Records' field is highlighted with a red border and contains the value '10'. Other visible settings include 'Start Index' set to 'StartIndex2', 'Fetch' set to 'At start', and 'Executed SQL' set to 'SELECT TOP (10)'. The 'Sources' section lists 'Order', and the 'Filters' section includes a condition 'Year(Order.CreatedOn) = Year'. The 'Sorting' section shows two fields: 'Order.CreatedOn (DESC)' and 'Order.Priority (DESC)'. A 'Test Values' section at the bottom contains a value 'Year'.

Properties

It is possible to limit the Aggregate's output to a maximum number of records

- **Max. Records** property
- Does not impact the Count output

Aggregate Editor is SQL-dialect agnostic

- **Executed SQL** property shows the SQL statement generated from the Aggregate
- SQL is generated according to the DBMS used

The screenshot shows the 'GetYearOrders' aggregate configuration in the Aggregate Editor. The 'Executed SQL' field is highlighted with a red border and contains the text 'SELECT TOP (10)'. Other visible settings include 'Name' (GetYearOrders), 'Start Index' (StartIndex2), 'Max. Records' (10), and 'Fetch' (At start). The 'Sources' section lists 'Order'. The 'Filters' section contains the filter 'Year(Order.CreatedOn) = Year'. The 'Sorting' section shows two fields: 'Order.CreatedOn (DESC)' and 'Order.Priority (DESC)'. The 'Test Values' section has a 'Year' entry.

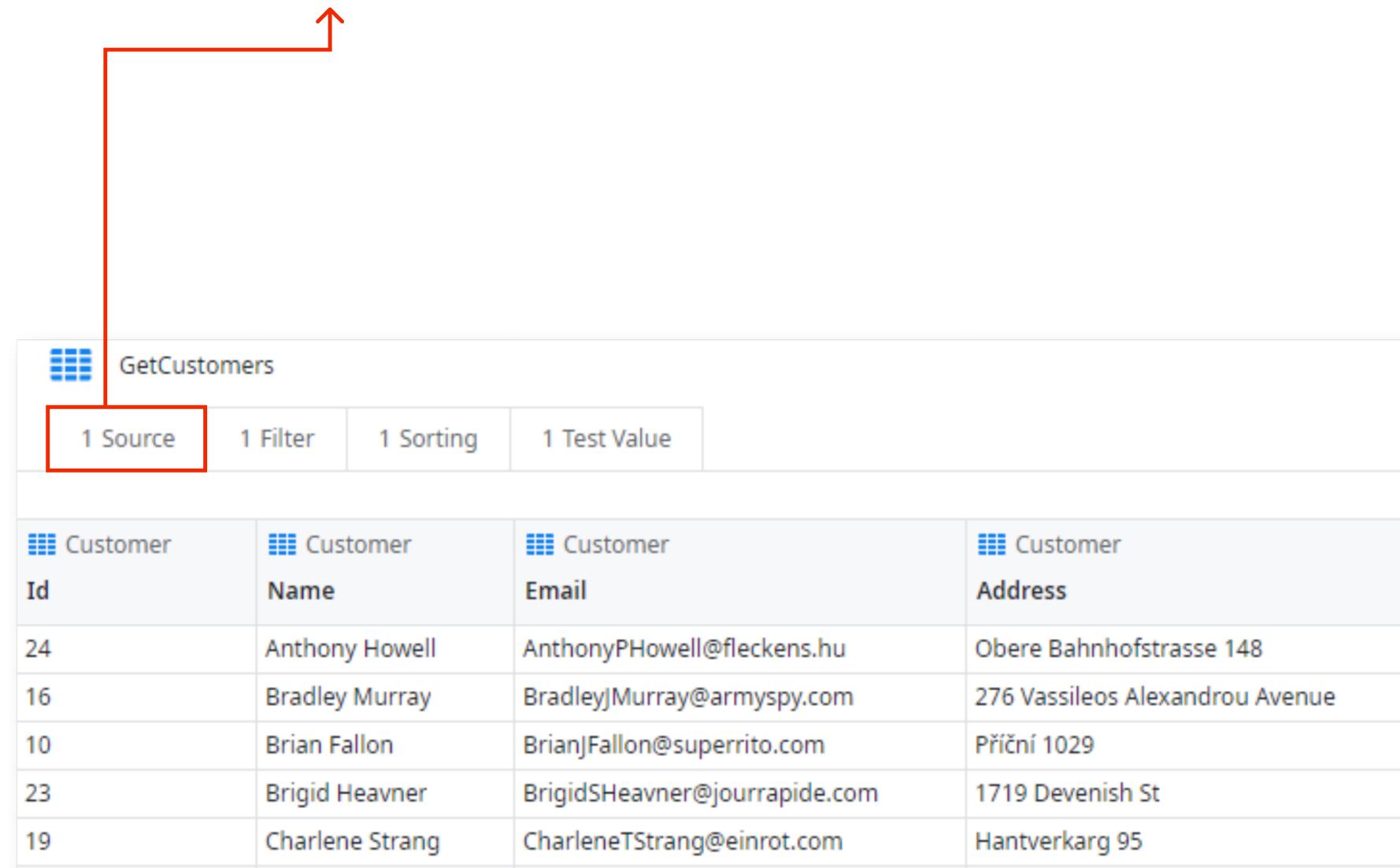
Aggregates as SQL

SELECT . . . FROM {Entities} WHERE {Filters} ORDER BY {Sorting}

GetCustomers			
1 Source	1 Filter	1 Sorting	1 Test Value
Customer Id	Customer Name	Customer Email	Customer Address
24	Anthony Howell	AnthonyPHowell@fleckens.hu	Obere Bahnhofstrasse 148
16	Bradley Murray	BradleyJMurray@armyspy.com	276 Vassileos Alexandrou Avenue
10	Brian Fallon	BrianJFallon@superrito.com	Příční 1029
23	Brigid Heavner	BrigidSHeavner@jourrapide.com	1719 Devenish St
19	Charlene Strang	CharleneTStrang@einrot.com	Hantverkarg 95

Aggregates as SQL

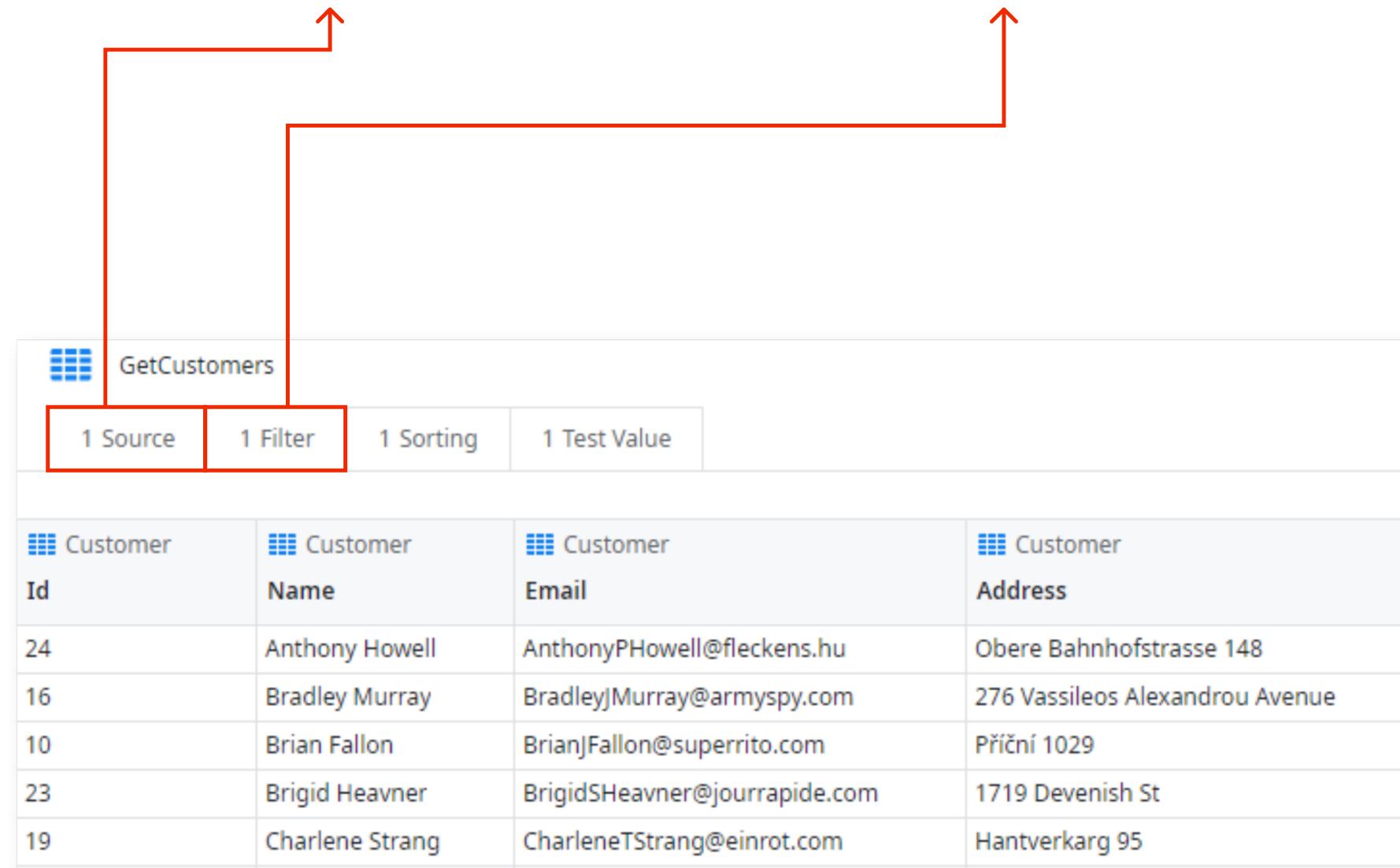
SELECT . . . FROM {Entities} WHERE {Filters} ORDER BY {Sorting}



GetCustomers			
1 Source	1 Filter	1 Sorting	1 Test Value
Customer Id	Customer Name	Customer Email	Customer Address
24	Anthony Howell	AnthonyPHowell@fleckens.hu	Obere Bahnhofstrasse 148
16	Bradley Murray	BradleyJMurray@armyspy.com	276 Vassileos Alexandrou Avenue
10	Brian Fallon	BrianJFallon@superrito.com	Příční 1029
23	Brigid Heavner	BrigidSHeavner@jourrapide.com	1719 Devenish St
19	Charlene Strang	CharleneTStrang@einrot.com	Hantverkarg 95

Aggregates as SQL

SELECT . . . FROM {Entities} WHERE {Filters} ORDER BY {Sorting}

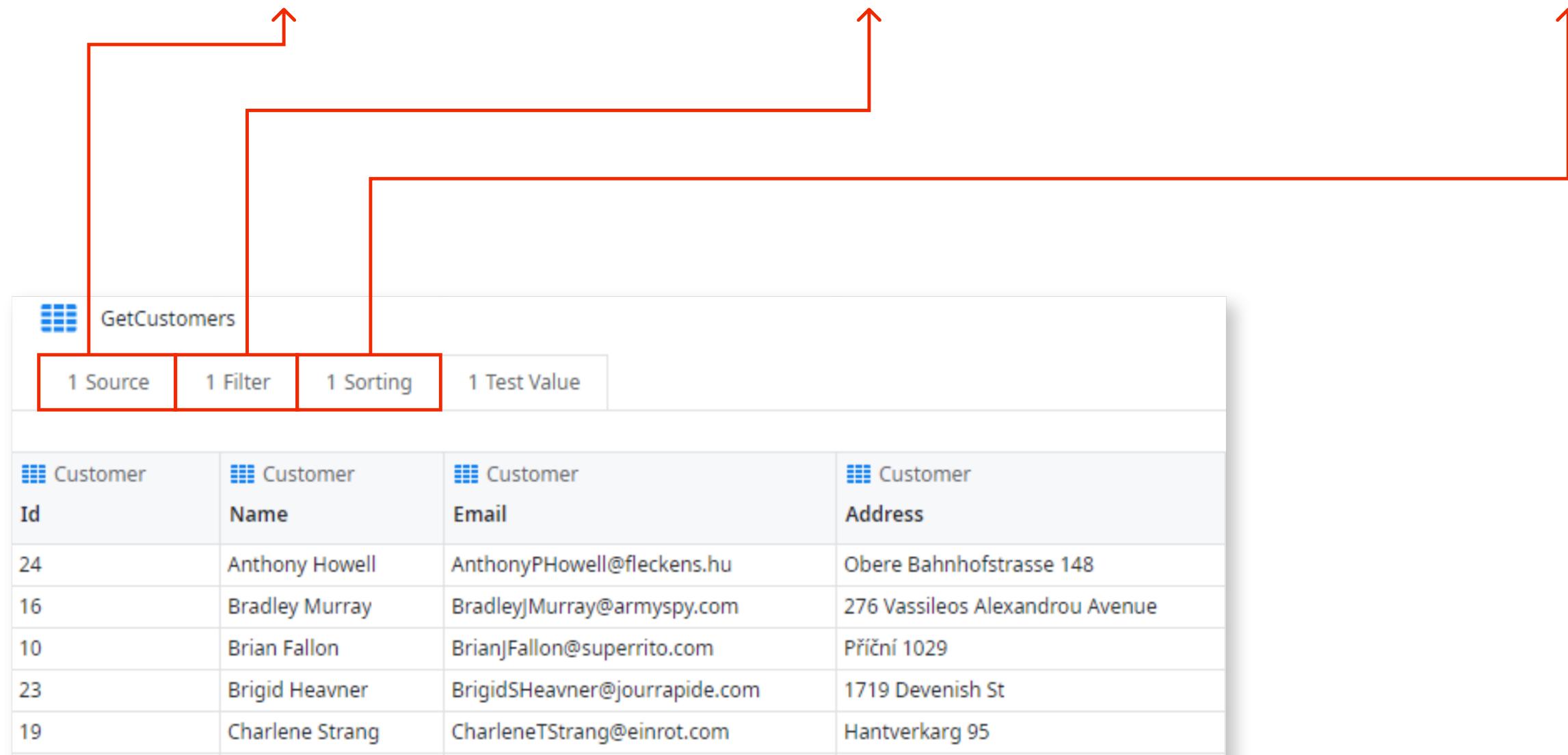


The diagram illustrates the mapping of an aggregate query to a specific data source. A red box encloses the 'GetCustomers' entity, and two red arrows point from the '1 Source' and '1 Filter' fields in its configuration to the corresponding columns in the resulting table below.

Customer Id	Customer Name	Customer Email	Customer Address
24	Anthony Howell	AnthonyPHowell@fleckens.hu	Obere Bahnhofstrasse 148
16	Bradley Murray	BradleyJMurray@armyspy.com	276 Vassileos Alexandrou Avenue
10	Brian Fallon	BrianJFallon@superrito.com	Příční 1029
23	Brigid Heavner	BrigidSHeavner@jourrapide.com	1719 Devenish St
19	Charlene Strang	CharleneTStrang@einrot.com	Hantverkarg 95

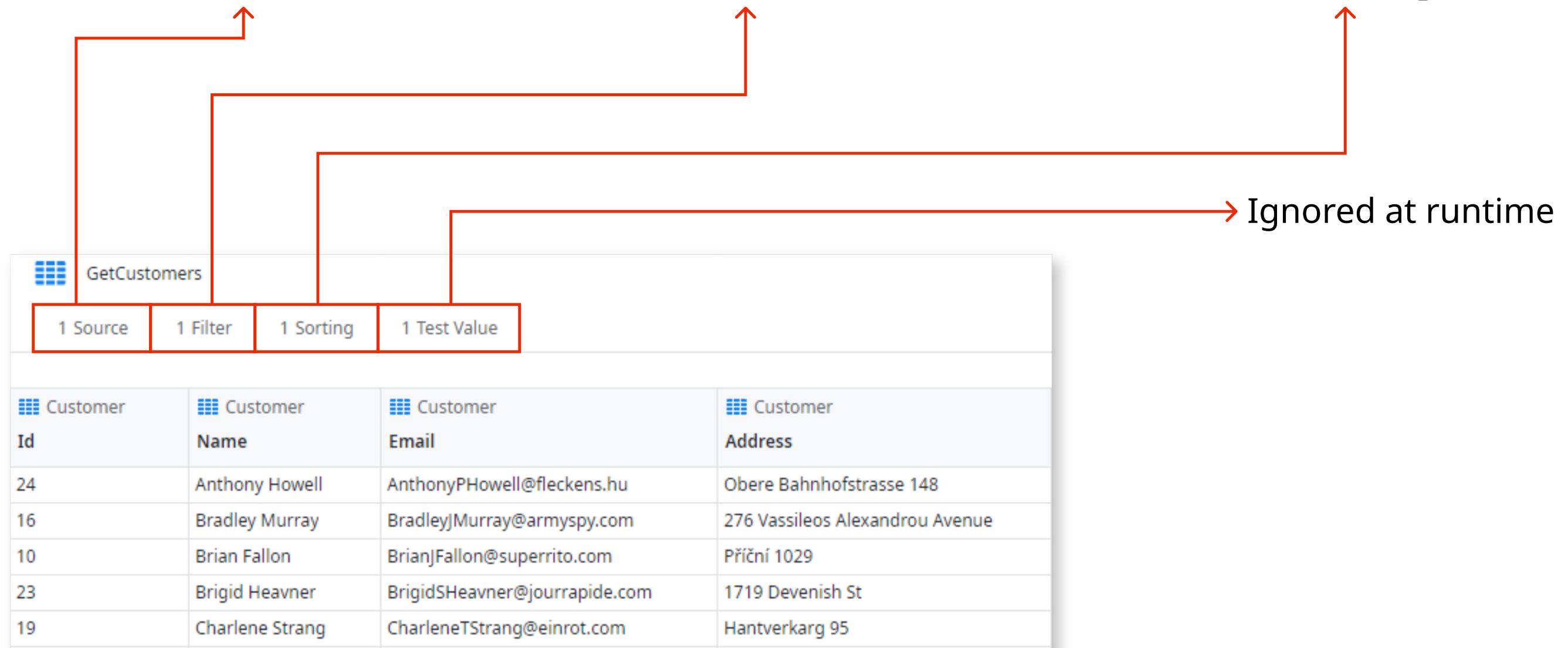
Aggregates as SQL

SELECT . . . FROM {Entities} WHERE {Filters} ORDER BY {Sorting}



Aggregates as SQL

SELECT . . . FROM {Entities} WHERE {Filters} ORDER BY {Sorting}



Aggregates as SQL

SELECT . . . FROM {Entities} WHERE {Filters} ORDER BY {Sorting}

