



Aggregates 101

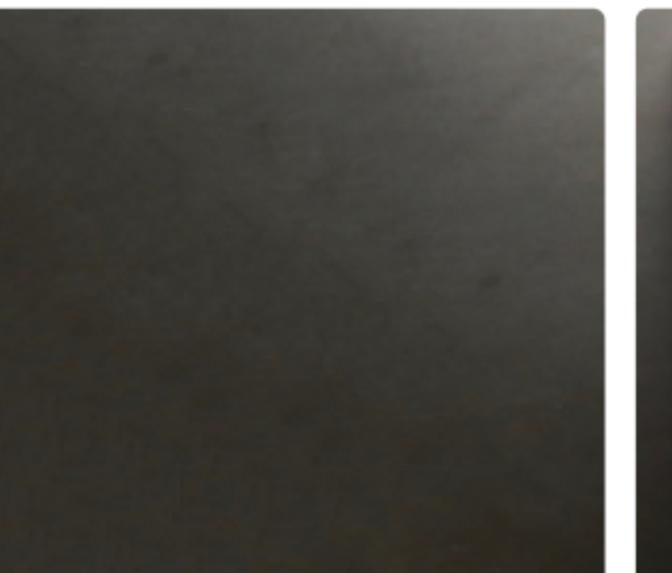
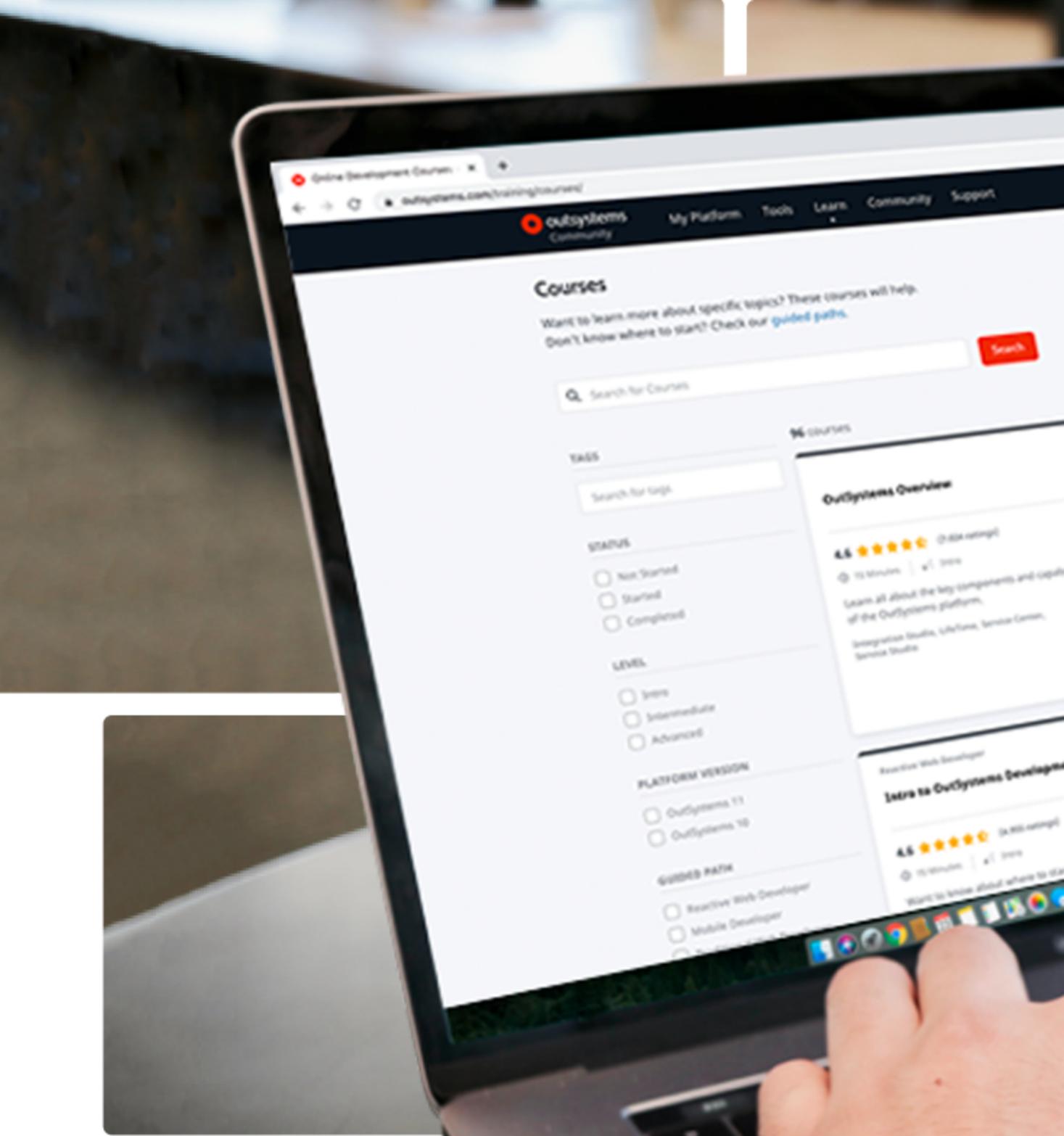
What is an Aggregate?

Sources

Filters

Sorting

Test Values



What is an Aggregate?

An Aggregate is a **visual** element that allows to define a **query to fetch data** from entities with sources, filters, and sorting.

Aggregates

- Most applications need to fetch data from the database

GetCustomers

1 Source | 1 Filter | 1 Sorting | 1 Test Value

Customer Id	Customer Name	Customer Email	Customer Address
3	Blogspan	GeraldSBell@fleckens.hu	7813 Sunnyside Park
7	Dabfeed	BradleyJMurray@armyspy.com	8 Fairview Road
10	Devcast	SaraJFreed@ow.ly	7 Towne Lane
9	Edgeify	TimothyKFlores@superrito.com	0 Melrose Street
6	Flashspan	RoyDLund@cuvox.de	74191 Crescent Oaks Trail
5	Flipstorm	NinaFLaird@einrot.com	96 Cambridge Street
4	Innojam	AnthonyPHowell@dayrep.com	68061 Portage Junction
8	Photofeed	DonaldKCummings@teleworm.us	5879 Westend Circle
2	Skinder	HeatherDDejesus@gustr.com	89935 Monterey Court
1	Youopia	FaithHWhite@rhyta.com	7600 Cottonwood Drive

Aggregates

- Most applications need to fetch data from the database
- Aggregates allow us to define database queries in a visual way

GetCustomers

1 Source | 1 Filter | 1 Sorting | 1 Test Value

Customer Id	Customer Name	Customer Email	Customer Address
3	Blogspan	GeraldSBell@fleckens.hu	7813 Sunnyside Park
7	Dabfeed	BradleyJMurray@armyspy.com	8 Fairview Road
10	Devcast	SaraJFreed@ow.ly	7 Towne Lane
9	Edgeify	TimothyKFlores@superrito.com	0 Melrose Street
6	Flashspan	RoyDLund@cuvox.de	74191 Crescent Oaks Trail
5	Flipstorm	NinaFLaird@einrot.com	96 Cambridge Street
4	Innojam	AnthonyPHowell@dayrep.com	68061 Portage Junction
8	Photofeed	DonaldKCummings@telemorph.us	5879 Westend Circle
2	Skinder	HeatherDDejesus@gustr.com	89935 Monterey Court
1	Youopia	FaithHWhite@rhyta.com	7600 Cottonwood Drive

Aggregates

- Most applications need to fetch data from the database
- Aggregates allow us to define database queries in a visual way
 - Add Sources
 - Create Filters
 - Define Sorting
 - Test

GetCustomers

1 Source | 1 Filter | 1 Sorting | 1 Test Value

Customer	Customer	Customer	Customer
ID	Name	Email	Address
3	Blogspan	GeraldSBell@fleckens.hu	7813 Sunnyside Park
7	Dabfeed	BradleyJMurray@armyspy.com	8 Fairview Road
10	Devcast	SaraJFreed@ow.ly	7 Towne Lane
9	Edgeify	TimothyKFlores@superrito.com	0 Melrose Street
6	Flashspan	RoyDLund@cuvox.de	74191 Crescent Oaks Trail
5	Flipstorm	NinaFLaird@einrot.com	96 Cambridge Street
4	Innojam	AnthonyPHowell@dayrep.com	68061 Portage Junction
8	Photofeed	DonaldKCummings@teleworm.us	5879 Westend Circle
2	Skinder	HeatherDDejesus@gustr.com	89935 Monterey Court
1	Youopia	FaithHWhite@rhyta.com	7600 Cottonwood Drive

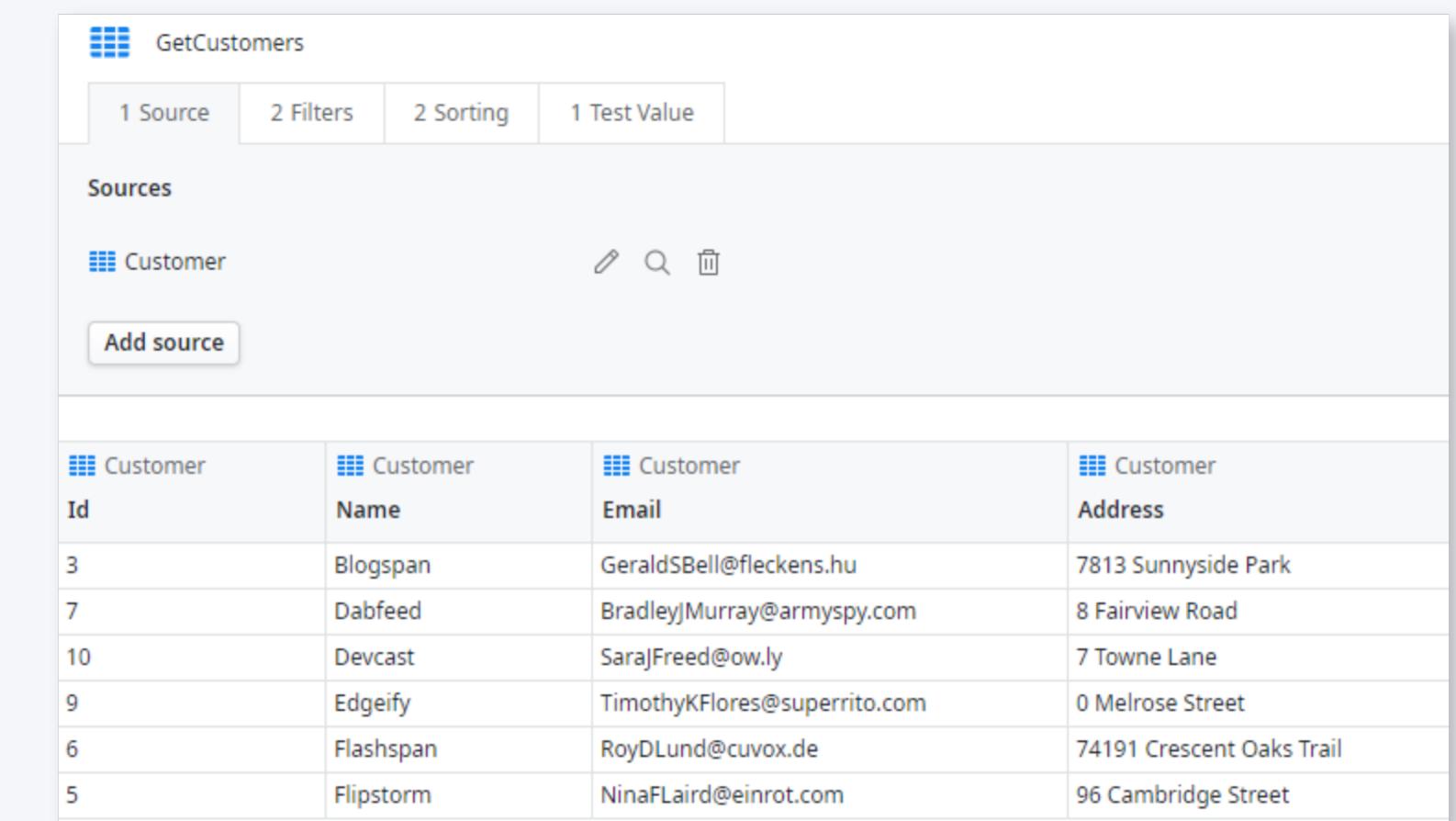
Aggregates

GetCustomers			
1 Source	1 Filter	1 Sorting	1 Test Value
Customer Id	Customer Name	Customer Email	Customer Address
3	Blogspan	GeraldSBell@fleckens.hu	7813 Sunnyside Park
7	Dabfeed	BradleyJMurray@armyspy.com	8 Fairview Road
10	Devcast	SaraJFreed@ow.ly	7 Towne Lane
9	Edgeify	TimothyKFlores@superrito.com	0 Melrose Street
6	Flashspan	RoyDLund@cuvox.de	74191 Crescent Oaks Trail
5	Flipstorm	NinaFLaird@einrot.com	96 Cambridge Street
4	Innojam	AnthonyPHowell@dayrep.com	68061 Portage Junction
8	Photofeed	DonaldKCummings@teleworm.us	5879 Westend Circle
2	Skinder	HeatherDDejesus@gustr.com	89935 Monterey Court
1	Youopia	FaithHWhite@rhyta.com	7600 Cottonwood Drive

- Most applications need to fetch data from the database
- Aggregates allow us to define database queries in a visual way
 - Add Sources
 - Create Filters
 - Define Sorting
 - Test
- Aggregates are easy to maintain
 - Excel-like display of real data
 - SQL knowledge NOT required

Sources

- Entities to retrieve data from
- Aggregates support one or more source Entities



The screenshot shows the OutSystems Power Designer interface for creating an aggregate named "GetCustomers". At the top, there are four tabs: "1 Source" (highlighted), "2 Filters", "2 Sorting", and "1 Test Value". Below the tabs, the title "Sources" is displayed, followed by a "Customer" entity icon and a "Customer" label. There are also edit, search, and delete icons. A "Add source" button is visible. The main area displays a table with the following data:

Customer	Customer	Customer	Customer
Id	Name	Email	Address
3	Blogspan	GeraldSBell@fleckens.hu	7813 Sunnyside Park
7	Dabfeed	BradleyJMurray@armyspy.com	8 Fairview Road
10	Devcast	SaraJFreed@ow.ly	7 Towne Lane
9	Edgeify	TimothyKFlores@superrito.com	0 Melrose Street
6	Flashspan	RoyDLund@cuvox.de	74191 Crescent Oaks Trail
5	Flipstorm	NinaFLaird@einrot.com	96 Cambridge Street

Filters

- Adds one or more conditions to the query to filter the output records

GetCustomers

1 Source 2 Filters 2 Sorting 1 Test Value

:: 1 ☐ Length(Customer.Name)>MinLength
:: 2 ☐ Customer.Email <> ""

Add filter

Customer	Customer	Customer	Customer
Id	Name	Email	Address
3	Blogspan	GeraldSBell@fleckens.hu	7813 Sunnyside Park
7	Dabfeed	BradleyJMurray@armyspy.com	8 Fairview Road
10	Devcast	SaraJFreed@ow.ly	7 Towne Lane
9	Edgeify	TimothyKFlores@superrito.com	0 Melrose Street
6	Flashspan	RoyDLund@cuvox.de	74191 Crescent Oaks Trail
5	Flipstorm	NinaFLaird@einrot.com	96 Cambridge Street

Filters

- Adds one or more conditions to the query to filter the output records
 - Support for multiple filters

GetCustomers

1 Source 2 Filters 2 Sorting 1 Test Value

:: 1 ☐ Length(Customer.Name)>MinLength
:: 2 ☐ Customer.Email <> ""

Add filter

Customer	Customer	Customer	Customer
Id	Name	Email	Address
3	Blogspan	GeraldSBell@fleckens.hu	7813 Sunnyside Park
7	Dabfeed	BradleyJMurray@armyspy.com	8 Fairview Road
10	Devcast	SaraJFreed@ow.ly	7 Towne Lane
9	Edgeify	TimothyKFlores@superrito.com	0 Melrose Street
6	Flashspan	RoyDLund@cuvox.de	74191 Crescent Oaks Trail
5	Flipstorm	NinaFLaird@einrot.com	96 Cambridge Street

Filters

- Adds one or more conditions to the query to filter the output records
 - Support for multiple filters
 - Support logical operators
 - =, <>, and, or, ...

GetCustomers

1 Source | 2 Filters | 2 Sorting | 1 Test Value

⋮ 1 ⌂ Length(Customer.Name)>MinLength
⋮ 2 ⌂ Customer.Email <> ""

Add filter

Customer	Customer	Customer	Customer
Id	Name	Email	Address
3	Blogspan	GeraldSBell@fleckens.hu	7813 Sunnyside Park
7	Dabfeed	BradleyJMurray@armyspy.com	8 Fairview Road
10	Devcast	SaraJFreed@ow.ly	7 Towne Lane
9	Edgeify	TimothyKFlores@superrito.com	0 Melrose Street
6	Flashspan	RoyDLund@cuvox.de	74191 Crescent Oaks Trail
5	Flipstorm	NinaFLaird@einrot.com	96 Cambridge Street

Filters

- Adds one or more conditions to the query to filter the output records
 - Support for multiple filters
 - Support logical operators
 - =, <>, and, or, ...
 - Support for some built-in functions
 - CurrDateTime()
 - If(Condition, True, False)

GetCustomers

1 Source | 2 Filters | 2 Sorting | 1 Test Value

⋮ 1 ⌂ Length(Customer.Name)>MinLength
⋮ 2 ⌂ Customer.Email <> ""

Add filter

Customer	Customer	Customer	Customer
Id	Name	Email	Address
3	Blogspan	GeraldSBell@fleckens.hu	7813 Sunnyside Park
7	Dabfeed	BradleyJMurray@armyspy.com	8 Fairview Road
10	Devcast	SaraJFreed@ow.ly	7 Towne Lane
9	Edgeify	TimothyKFlores@superrito.com	0 Melrose Street
6	Flashspan	RoyDLund@cuvox.de	74191 Crescent Oaks Trail
5	Flipstorm	NinaFLaird@einrot.com	96 Cambridge Street

Sorting

- Defines the Entities attribute to sort by and which direction
 - Ascending
 - Descending
- When defining multiple sorts the order is relevant to the result

GetCustomers

1 Source | 2 Filters | 2 Sorting | 1 Test Value

1 Customer.Name
2 Customer.Address

Add sort | Add dynamic sort

Customer	Customer	Customer	Customer
Id	Name	Email	Address
3	Blogspan	GeraldSBell@fleckens.hu	7813 Sunnyside Park
7	Dabfeed	BradleyJMurray@armyspy.com	8 Fairview Road
10	Devcast	SaraJFreed@ow.ly	7 Towne Lane
9	Edgeify	TimothyKFlores@superrito.com	0 Melrose Street
6	Flashspan	RoyDLund@cuvox.de	74191 Crescent Oaks Trail
5	Flipstorm	NinaFLaird@einrot.com	96 Cambridge Street

Test Values

- Allows specifying values for external variables used in the filter or sort conditions
 - Has as many inputs as external variables

GetCustomersByName

1 Source | 2 Filters | 2 Sorting | 1 Test Value

MinLength

5

Customer	Customer	Customer	Customer
Id	Name	Email	Address
3	Blogspan	GeraldSBell@fleckens.hu	7813 Sunnyside Park
7	Dabfeed	BradleyJMurray@armyspy.com	8 Fairview Road
10	Devcast	SaraJFreed@ow.ly	7 Towne Lane
9	Edgeify	TimothyKFlores@superrito.com	0 Melrose Street
6	Flashspan	RoyDLund@cuvox.de	74191 Crescent Oaks Trail
5	Flipstorm	NinaFLaird@einrot.com	96 Cambridge Street

Test Values

- Allows specifying values for external variables used in the filter or sort conditions
 - Has as many inputs as external variables
- Used to test the Aggregate and preview the output records
 - Does not have any influence at runtime

GetCustomersByName

1 Source | 2 Filters | 2 Sorting | 1 Test Value

MinLength

5

Customer Id	Customer Name	Customer Email	Customer Address
3	Blogspan	GeraldSBell@fleckens.hu	7813 Sunnyside Park
7	Dabfeed	BradleyJMurray@armyspy.com	8 Fairview Road
10	Devcast	SaraJFreed@ow.ly	7 Towne Lane
9	Edgeify	TimothyKFlores@superrito.com	0 Melrose Street
6	Flashspan	RoyDLund@cuvox.de	74191 Crescent Oaks Trail
5	Flipstorm	NinaFLaird@einrot.com	96 Cambridge Street