kx | it's about time

# Dashboards for Kx

## "How to" Guide

kx | it's about time

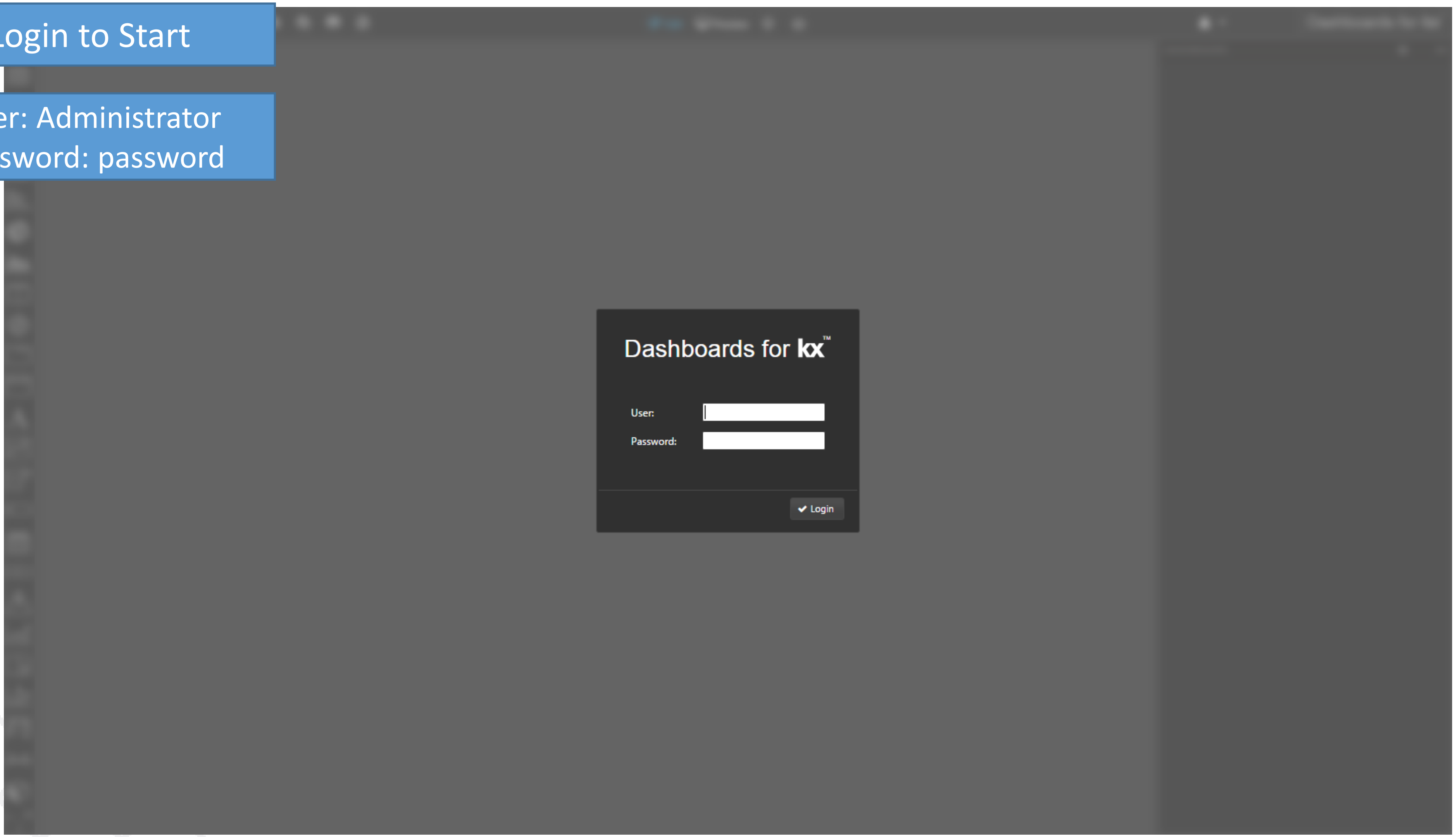
# Setup

## Dashboards v4.0.1 – "How to" Guide

# Getting Started

Login to Start

User: Administrator
Password: password

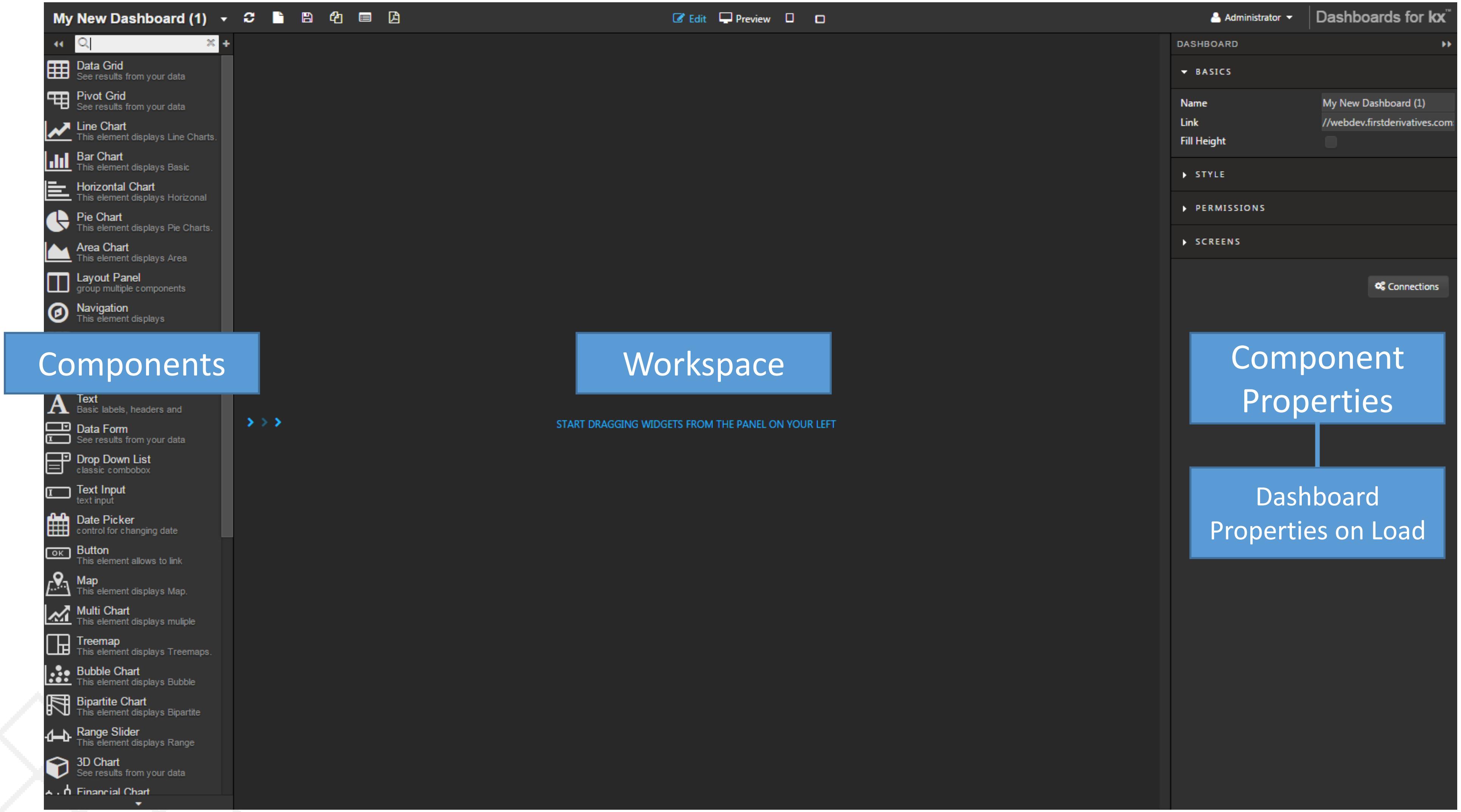


Dashboards for kx™

User:

Password:

# Layout


My New Dashboard (1)
Edit
Preview
Administrator
Dashboards for kx
Data Grid
See results from your data
Pivot Grid
See results from your data
Line Chart
This element displays Line Charts.
Bar Chart
This element displays Basic
Horizontal Chart
This element displays Horizonal
Pie Chart
This element displays Pie Charts.
Area Chart
This element displays Area
Layout Panel
group multiple components
Navigation
This element displays
Text
Basic labels, headers and
Data Form
See results from your data
Drop Down List
classic combobox
Text Input
text input
Date Picker
control for changing date
Button
This element allows to link
Map
This element displays Map.
Multi Chart
This element displays muliple
Treemap
This element displays Treemaps.
Bubble Chart
This element displays Bubble
Bipartite Chart
This element displays Bipartite
Range Slider
This element displays Range
3D Chart
See results from your data
Financial Chart
START DRAGGING WIDGETS FROM THE PANEL ON YOUR LEFT
DASHBOARD
BASICS
Name
My New Dashboard (1)
Link
//webdev.firstderivatives.com
Fill Height
STYLE
PERMISSIONS
SCREENS
Connections

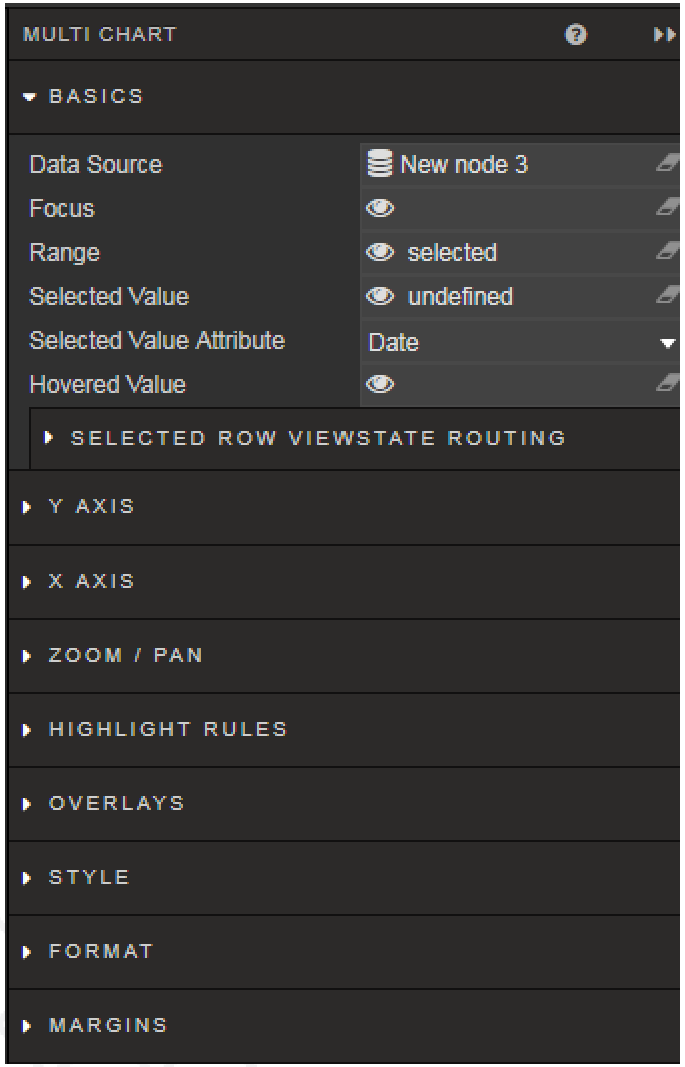
Components

Workspace

Component Properties

Dashboard Properties on Load

# What’s in a Component Property?



Data Source: New node 3
Focus:
Range: selected
Selected Value: undefined
Selected Value Attribute: Date
Hovered Value:

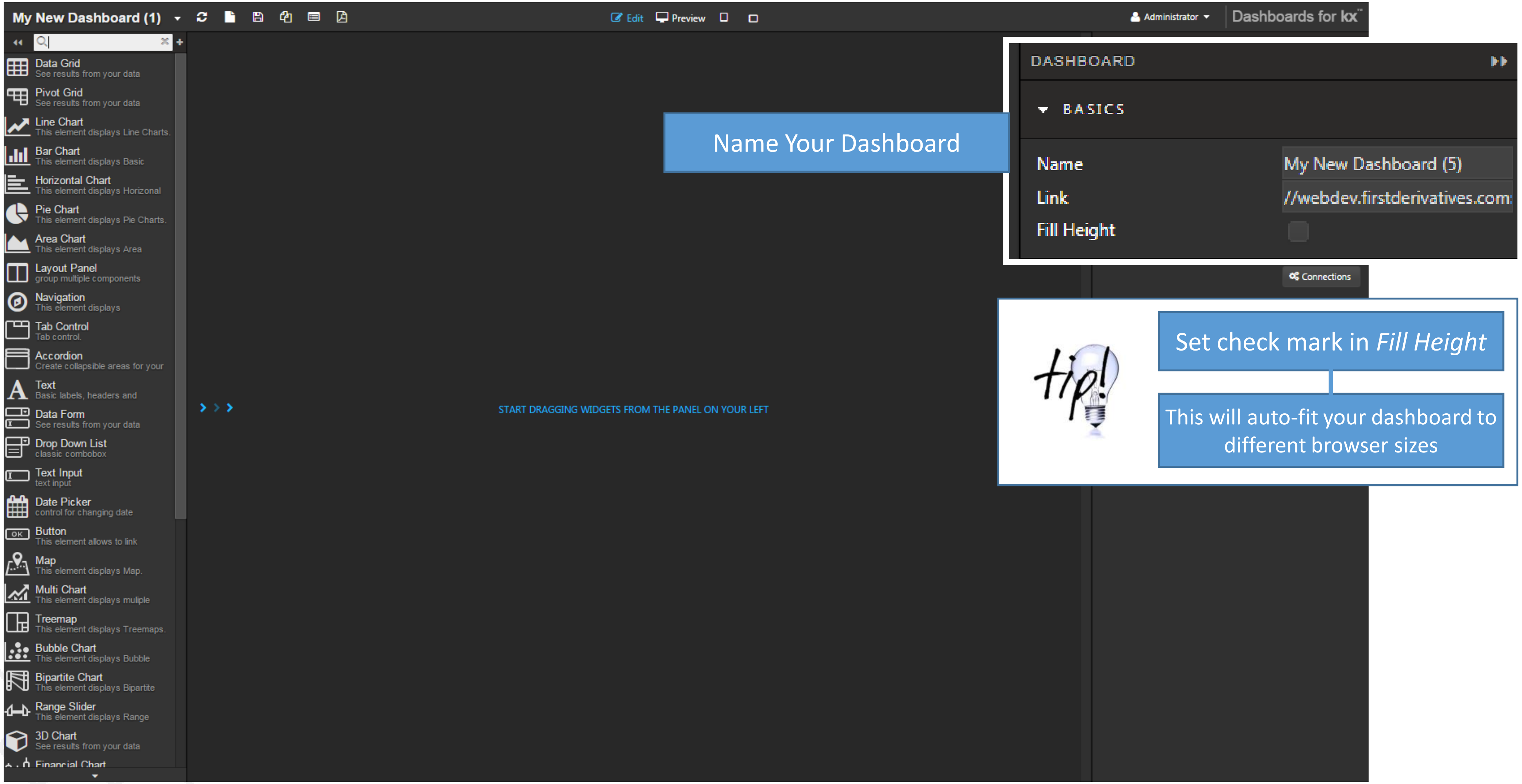
The Property Panel is used to configure component settings

In this document, screenshots of relevant sections of the component will be used to illustrate where configurations are required

Tip!

Further information on configuring components can be found at http://code.kx.com

# Define the Basics


My New Dashboard (1)
Edit
Preview
Administrator
Dashboards for kx
Data Grid
See results from your data
Pivot Grid
See results from your data
Line Chart
This element displays Line Charts.
Bar Chart
This element displays Basic
Horizontal Chart
This element displays Horizonal
Pie Chart
This element displays Pie Charts.
Area Chart
This element displays Area
Layout Panel
group multiple components
Navigation
This element displays
Tab Control
Tab control.
Accordion
Create collapsible areas for your
Text
Basic labels, headers and
Data Form
See results from your data
Drop Down List
classic combobox
Text Input
text input
Date Picker
control for changing date
Button
This element allows to link
Map
This element displays Map.
Multi Chart
This element displays muliple
Treemap
This element displays Treemaps.
Bubble Chart
This element displays Bubble
Bipartite Chart
This element displays Bipartite
Range Slider
This element displays Range
3D Chart
See results from your data
Financial Chart
START DRAGGING WIDGETS FROM THE PANEL ON YOUR LEFT
DASHBOARD
BASICS
Connections
tip!

Name My New Dashboard (5)

Link //webdev.firstderivatives.com

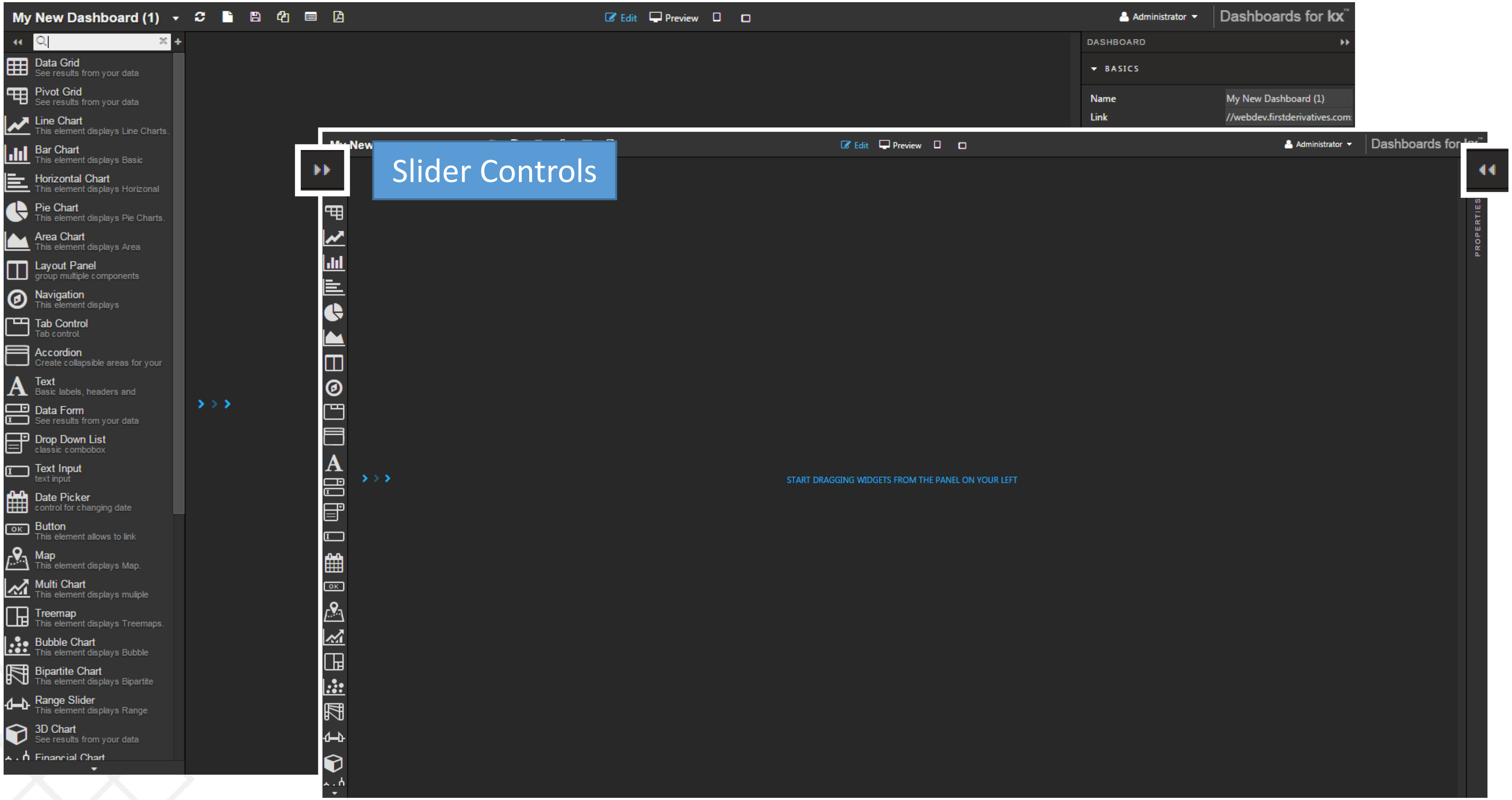
Fill Height □

Name Your Dashboard

Set check mark in *Fill Height*

This will auto-fit your dashboard to different browser sizes

# Slide Panel Control

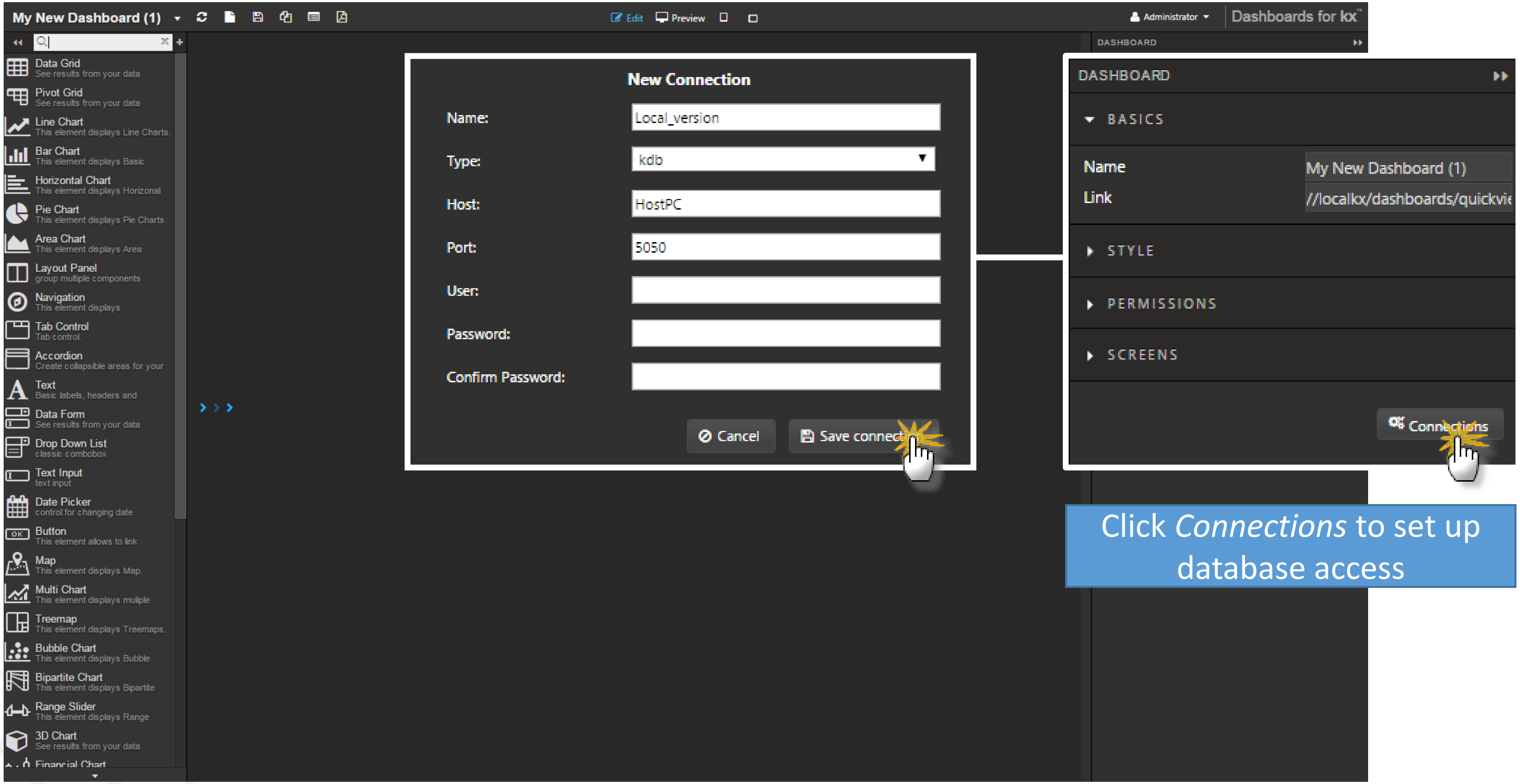

My New Dashboard (1)
Edit
Preview
Administrator
Dashboards for kx
DASHBOARD
BASICS
Data Grid
See results from your data
Pivot Grid
See results from your data
Line Chart
This element displays Line Charts.
Bar Chart
This element displays Basic
Horizontal Chart
This element displays Horizonal
Pie Chart
This element displays Pie Charts.
Area Chart
This element displays Area
Layout Panel
group multiple components
Navigation
This element displays
Tab Control
Tab control.
Accordion
Create collapsible areas for your
Text
Basic labels, headers and
Data Form
See results from your data
Drop Down List
classic combobox
Text Input
text input
Date Picker
control for changing date
Button
This element allows to link
Map
This element displays Map.
Multi Chart
This element displays muliple
Treemap
This element displays Treemaps.
Bubble Chart
This element displays Bubble
Bipartite Chart
This element displays Bipartite
Range Slider
This element displays Range
3D Chart
See results from your data
Financial Chart
Slider Controls
PROPERTIES

Name My New Dashboard (1)

Link //webdev.firstderivatives.com

START DRAGGING WIDGETS FROM THE PANEL ON YOUR LEFT

# Create a Connection – Access to HTMLEvalPack required - contact Kx



My New Dashboard (1)
Edit
Preview
Administrator
Dashboards for kx
Data Grid
See results from your data
Pivot Grid
See results from your data
Line Chart
This element displays Line Charts.
Bar Chart
This element displays Basic
Horizontal Chart
This element displays Horizonal
Pie Chart
This element displays Pie Charts.
Area Chart
This element displays Area
Layout Panel
group multiple components
Navigation
This element displays
Tab Control
Tab control.
Accordion
Create collapsible areas for your
Text
Basic labels, headers and
Data Form
See results from your data
Drop Down List
classic combobox
Text Input
text input
Date Picker
control for changing date
Button
This element allows to link
Map
This element displays Map.
Multi Chart
This element displays muliple
Treemap
This element displays Treemaps.
Bubble Chart
This element displays Bubble
Bipartite Chart
This element displays Bipartite
Range Slider
This element displays Range
3D Chart
See results from your data
Financial Chart
DASHBOARD

**New Connection**

Name: Local_version

Type: kdb

Host: HostPC

Port: 5050

User:

Password:

Confirm Password:

DASHBOARD

BASICS

Name: My New Dashboard (1)

Link: //localkx/dashboards/quickvie

STYLE
PERMISSIONS
SCREENS
Connections

Click *Connections* to set up database access

# Dashboard Connection Settings

## Connection 1

*Name*: html5eval_x
*Type*: kdb
*Host*: webdev
*Port*: 20070*
*User*: Administrator
*Password*: password

## Connection 2

*Name*: html5eval_y
*Type*: kdb
*Host*: webdev
*Port*: 20071*
*User*: Administrator
*Password*: password

## Connection Group

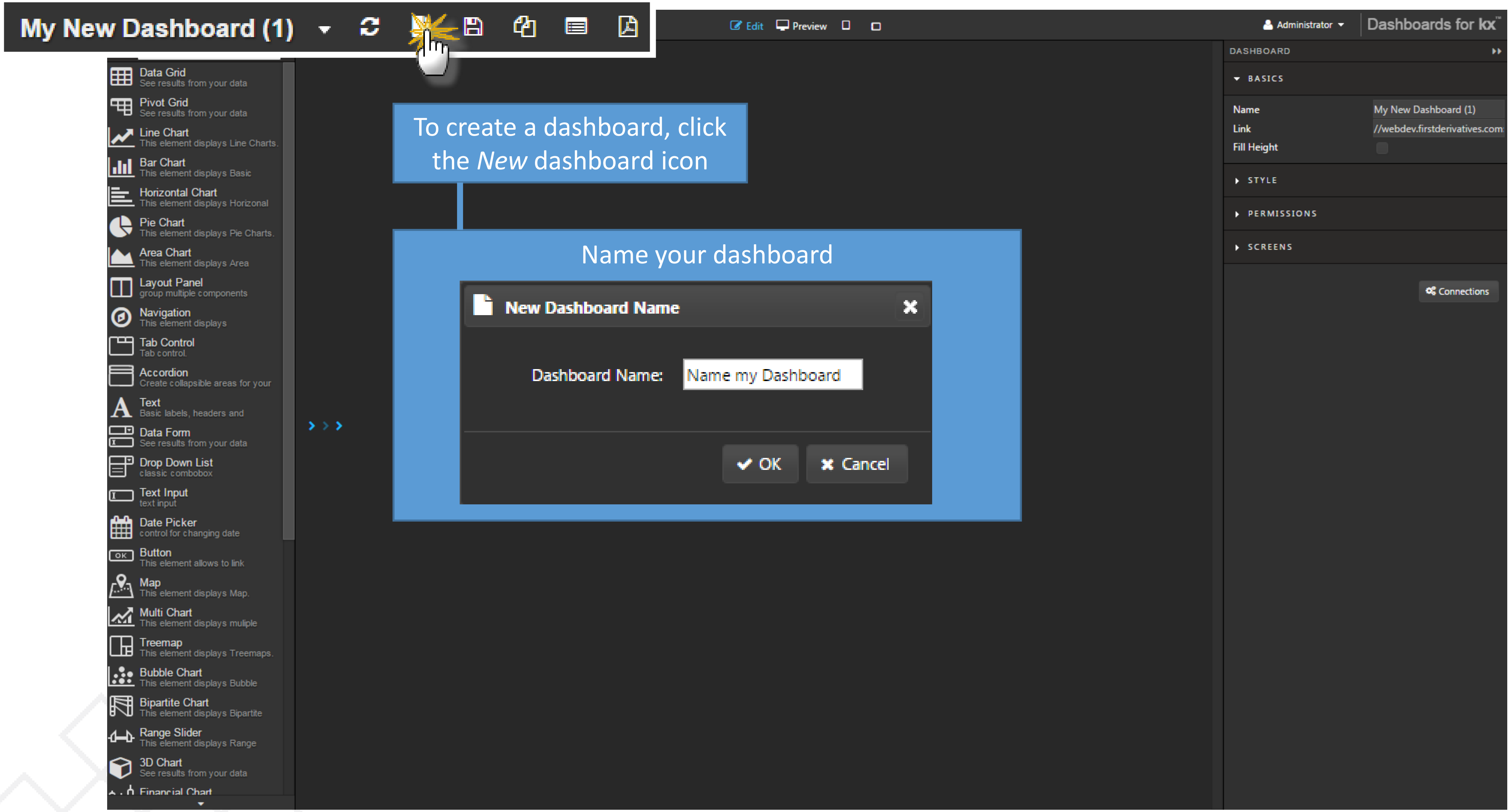
*Group Name*: html5eval_grp
*Type*: Mastered

*html5eval_x*
*html5eval_y*

*Sample Port Numbers – ensure there is no clash with existing port connections

This step is optional. The Dashboard Eval Pack used for this tutorial uses the Connection Group:
htmlevalcongroup

# FYI: When you want to create a New Dashboard


My New Dashboard (1)
Edit
Preview
Administrator
Dashboards for kx
Data Grid
See results from your data
Pivot Grid
See results from your data
Line Chart
This element displays Line Charts.
Bar Chart
This element displays Basic
Horizontal Chart
This element displays Horizonal
Pie Chart
This element displays Pie Charts.
Area Chart
This element displays Area
Layout Panel
group multiple components
Navigation
This element displays
Tab Control
Tab control.
Accordion
Create collapsible areas for your
Text
Basic labels, headers and
Data Form
See results from your data
Drop Down List
classic combobox
Text Input
text input
Date Picker
control for changing date
Button
This element allows to link
Map
This element displays Map.
Multi Chart
This element displays muliple
Treemap
This element displays Treemaps.
Bubble Chart
This element displays Bubble
Bipartite Chart
This element displays Bipartite
Range Slider
This element displays Range
3D Chart
See results from your data
Financial Chart
DASHBOARD
BASICS
STYLE
PERMISSIONS
SCREENS
Connections

Name: My New Dashboard (1)

Link: //webdev.firstderivatives.com

Fill Height: □

To create a dashboard, click the *New* dashboard icon

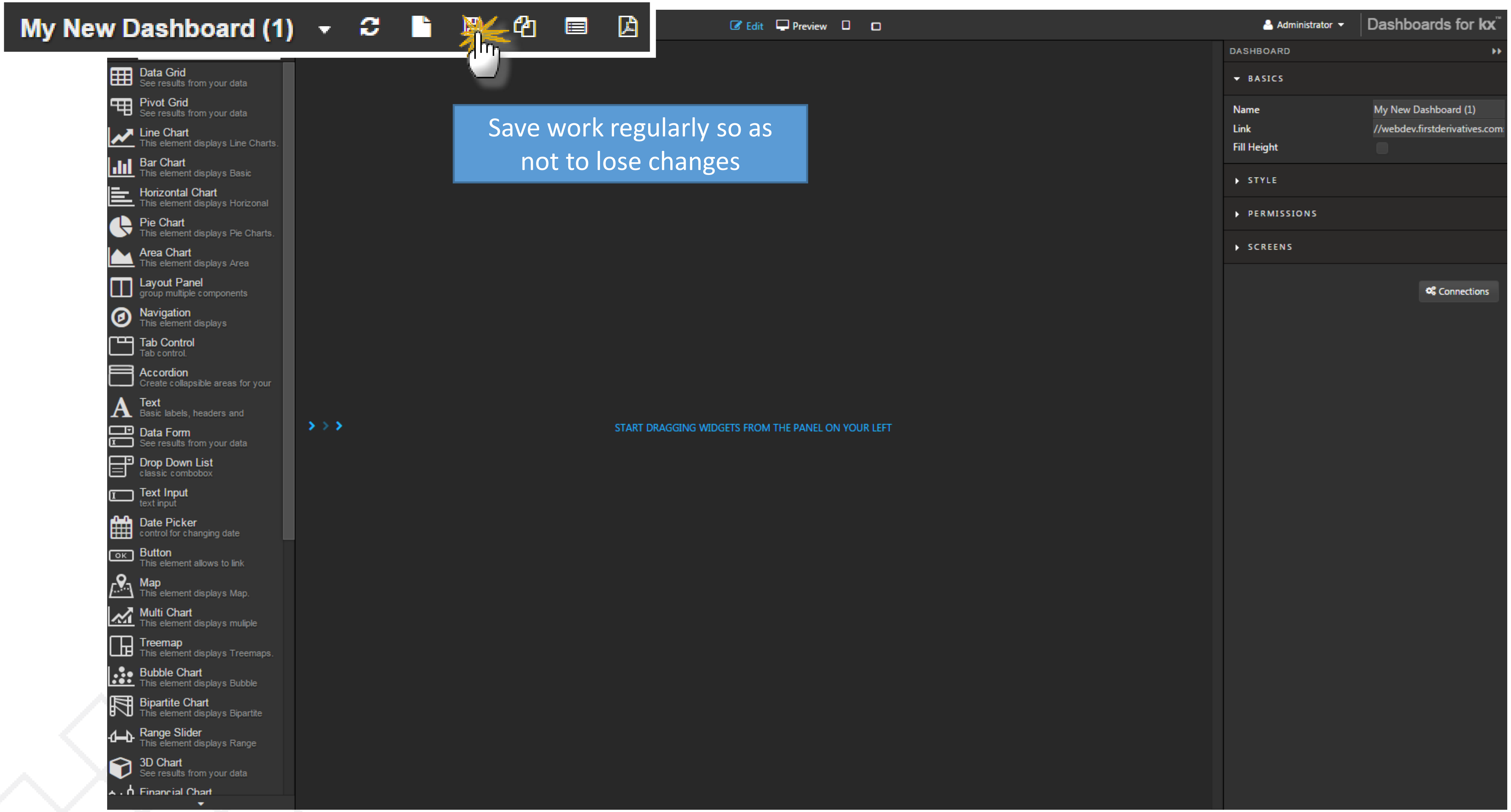
Name your dashboard

**New Dashboard Name**

Dashboard Name: Name my Dashboard

✔ OK ✖ Cancel

# Save a Dashboard

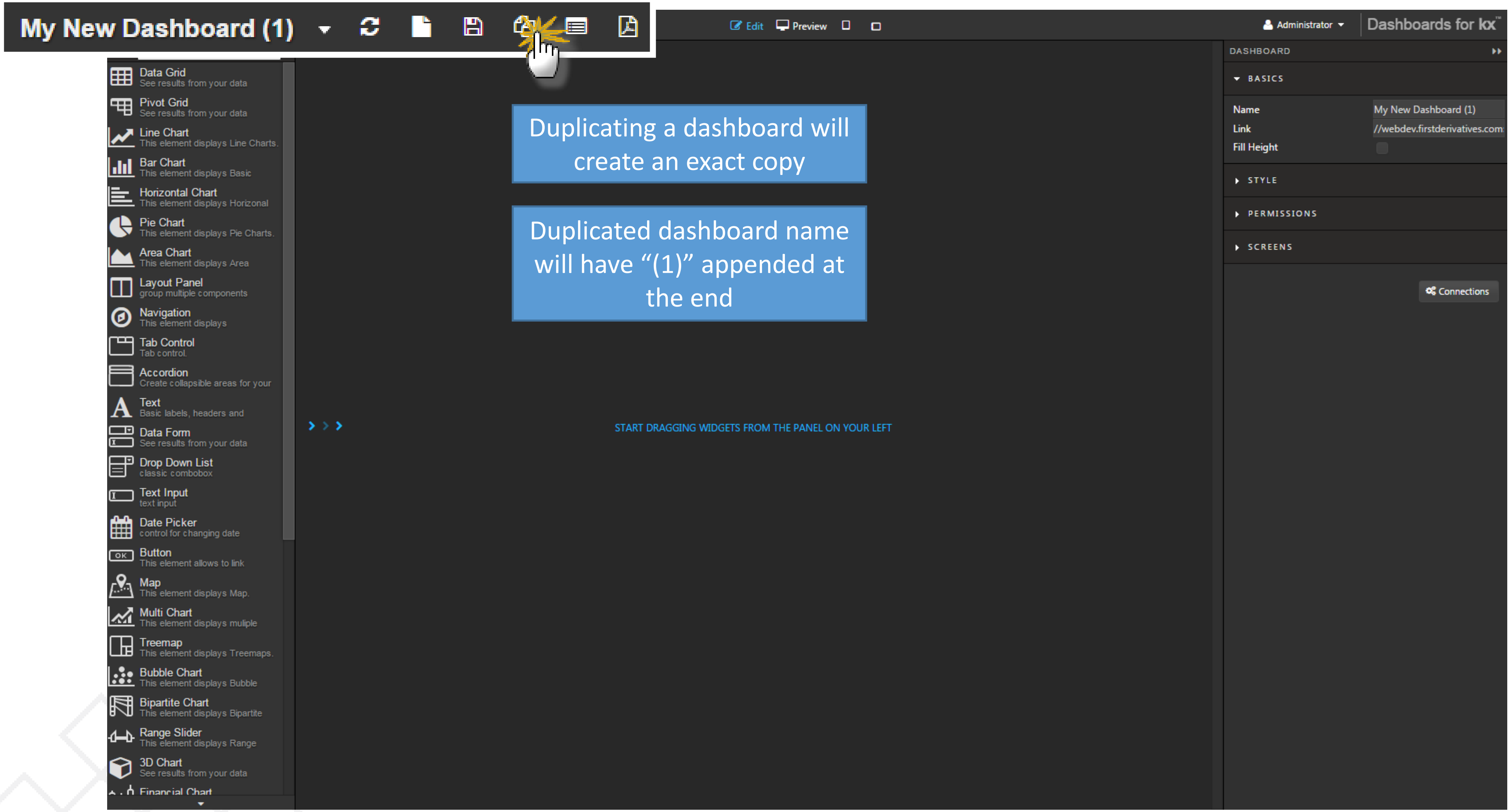


My New Dashboard (1)
Edit
Preview
Administrator
Dashboards for kx
Data Grid
See results from your data
Pivot Grid
See results from your data
Line Chart
This element displays Line Charts.
Bar Chart
This element displays Basic
Horizontal Chart
This element displays Horizonal
Pie Chart
This element displays Pie Charts.
Area Chart
This element displays Area
Layout Panel
group multiple components
Navigation
This element displays
Tab Control
Tab control.
Accordion
Create collapsible areas for your
Text
Basic labels, headers and
Data Form
See results from your data
Drop Down List
classic combobox
Text Input
text input
Date Picker
control for changing date
Button
This element allows to link
Map
This element displays Map.
Multi Chart
This element displays muliple
Treemap
This element displays Treemaps.
Bubble Chart
This element displays Bubble
Bipartite Chart
This element displays Bipartite
Range Slider
This element displays Range
3D Chart
See results from your data
Financial Chart
START DRAGGING WIDGETS FROM THE PANEL ON YOUR LEFT
DASHBOARD
BASICS
STYLE
PERMISSIONS
SCREENS
Connections

Name: My New Dashboard (1)
Link: //webdev.firstderivatives.com
Fill Height

Save work regularly so as not to lose changes

# Duplicate a Dashboard

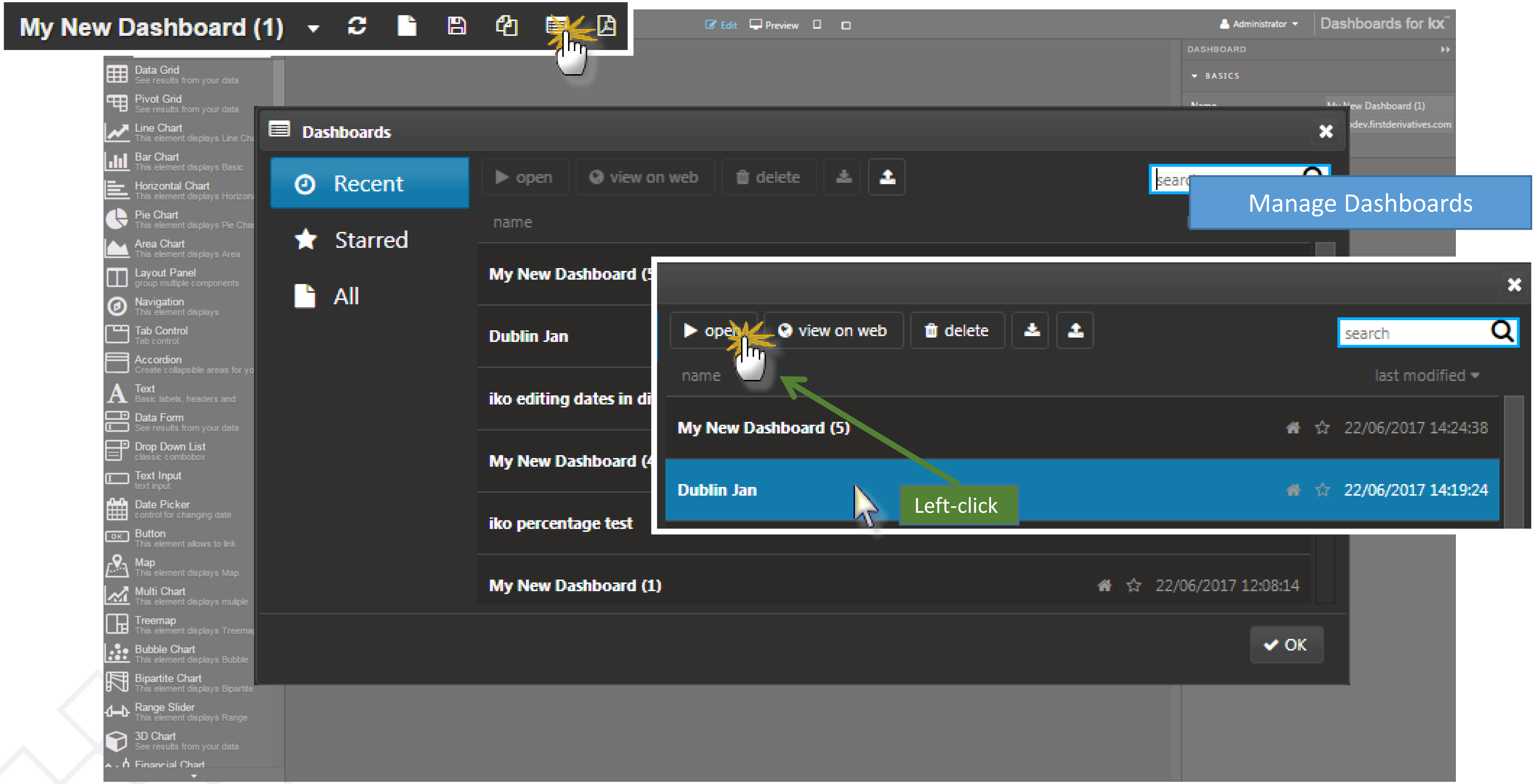


Duplicating a dashboard will create an exact copy

Duplicated dashboard name will have “(1)” appended at the end

My New Dashboard (1)
Edit
Preview
Administrator
Dashboards for kx
DASHBOARD
BASICS
Name
My New Dashboard (1)
Link
//webdev.firstderivatives.com
Fill Height
STYLE
PERMISSIONS
SCREENS
Connections
START DRAGGING WIDGETS FROM THE PANEL ON YOUR LEFT
Data Grid
See results from your data
Pivot Grid
See results from your data
Line Chart
This element displays Line Charts.
Bar Chart
This element displays Basic
Horizontal Chart
This element displays Horizonal
Pie Chart
This element displays Pie Charts.
Area Chart
This element displays Area
Layout Panel
group multiple components
Navigation
This element displays
Tab Control
Tab control.
Accordion
Create collapsible areas for your
Text
Basic labels, headers and
Data Form
See results from your data
Drop Down List
classic combobox
Text Input
text input
Date Picker
control for changing date
Button
This element allows to link
Map
This element displays Map.
Multi Chart
This element displays muliple
Treemap
This element displays Treemaps.
Bubble Chart
This element displays Bubble
Bipartite Chart
This element displays Bipartite
Range Slider
This element displays Range
3D Chart
See results from your data
Financial Chart

# Open / Delete Dashboards

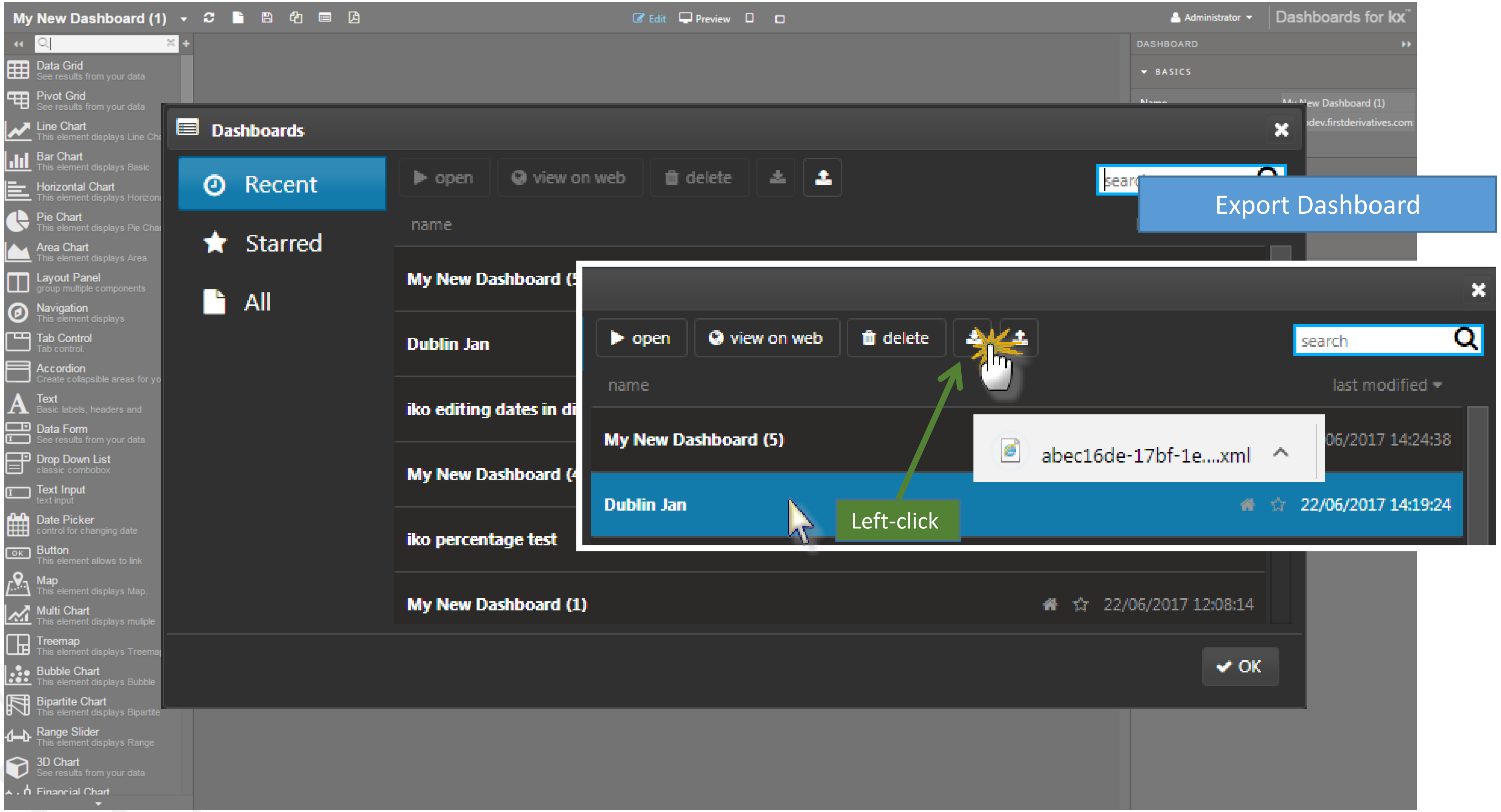


Manage Dashboards

Left-click

| name | last modified |
|---|---|
| My New Dashboard (5) | 22/06/2017 14:24:38 |
| Dublin Jan | 22/06/2017 14:19:24 |
| My New Dashboard (1) | 22/06/2017 12:08:14 |

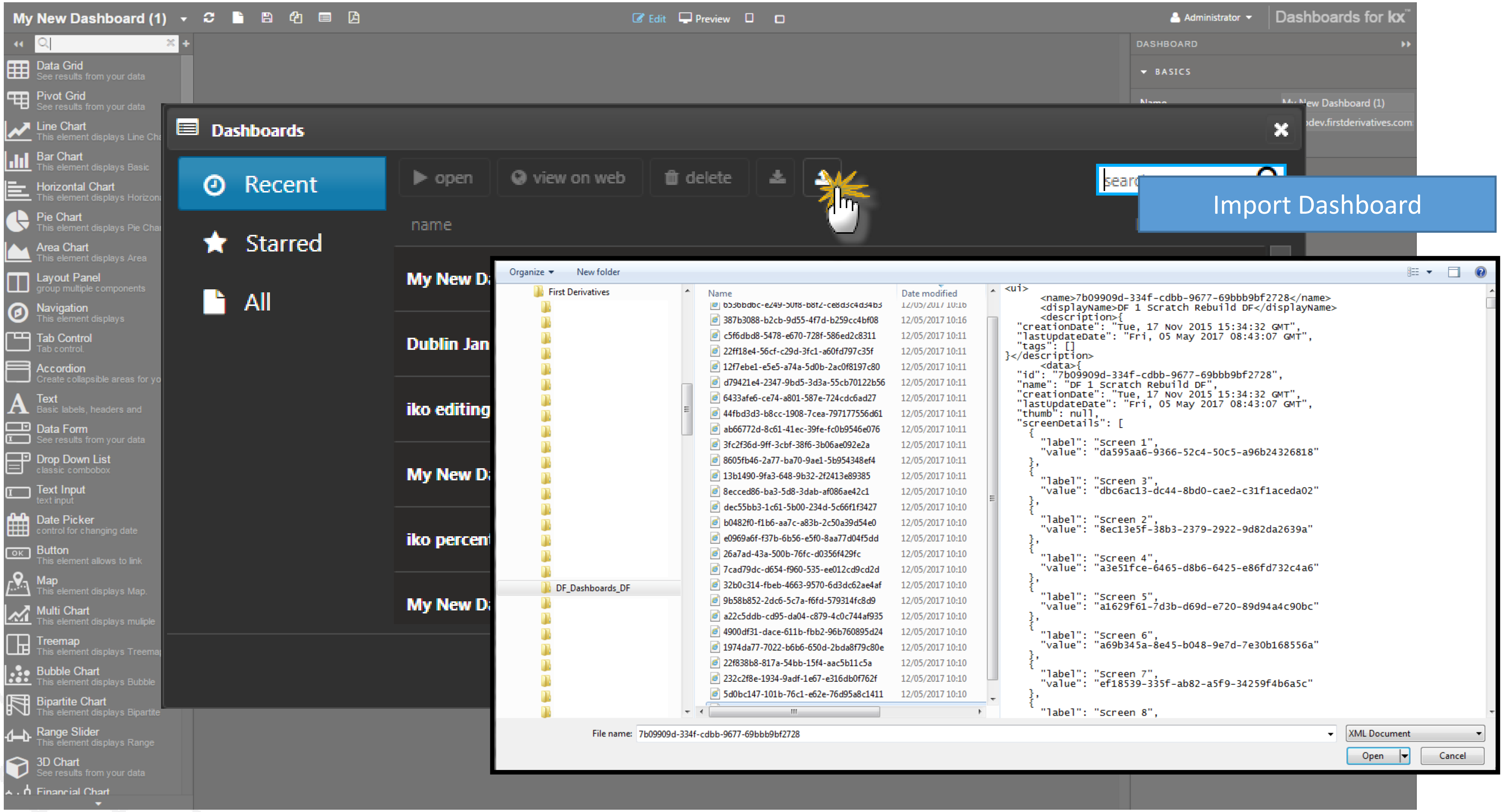
# Export Dashboards



Export Dashboard

Left-click

# Import Dashboards



Import Dashboard

| Name | Date modified |
|---|---|
| b536bdbc-e249-50f8-b8f2-ce8d3c4d34b3 | 12/05/2017 10:16 |
| 387b3088-b2cb-9d55-4f7d-b259cc4bf08 | 12/05/2017 10:16 |
| c5f6dbd8-5478-e670-728f-586ed2c8311 | 12/05/2017 10:11 |
| 22ff18e4-56cf-c29d-3fc1-a60fd797c35f | 12/05/2017 10:11 |
| 12f7ebe1-e5e5-a74a-5d0b-2ac0f8197c80 | 12/05/2017 10:11 |
| d79421e4-2347-9bd5-3d3a-55cb70122b56 | 12/05/2017 10:11 |
| 6433afe6-ce74-a801-587e-724cdc6ad27 | 12/05/2017 10:11 |
| 44fbd3d3-b8cc-1908-7cea-797177556d61 | 12/05/2017 10:11 |
| ab66772d-8c61-41ec-39fe-fc0b9546e076 | 12/05/2017 10:11 |
| 3fc2f36d-9ff-3cbf-38f6-3b06ae092e2a | 12/05/2017 10:11 |
| 8605fb46-2a77-ba70-9ae1-5b954348ef4 | 12/05/2017 10:11 |
| 13b1490-9fa3-648-9b32-2f2413e89385 | 12/05/2017 10:11 |
| 8ecced86-ba3-5d8-3dab-af086ae42c1 | 12/05/2017 10:10 |
| dec55bb3-1c61-5b00-234d-5c66f1f3427 | 12/05/2017 10:10 |
| b0482f0-f1b6-aa7c-a83b-2c50a39d54e0 | 12/05/2017 10:10 |
| e0969a6f-f37b-6b56-e5f0-8aa77d04f5dd | 12/05/2017 10:10 |
| 26a7ad-43a-500b-76fc-d0356f429fc | 12/05/2017 10:10 |
| 7cad79dc-d654-f960-535-ee012cd9cd2d | 12/05/2017 10:10 |
| 32b0c314-fbeb-4663-9570-6d3dc62ae4af | 12/05/2017 10:10 |
| 9b58b852-2dc6-5c7a-f6fd-579314fc8d9 | 12/05/2017 10:10 |
| a22c5ddb-cd95-da04-c879-4c0c744af935 | 12/05/2017 10:10 |
| 4900df31-dace-611b-fbb2-96b760895d24 | 12/05/2017 10:10 |
| 1974da77-7022-b6b6-650d-2bda8f79c80e | 12/05/2017 10:10 |
| 22f838b8-817a-54bb-15f4-aac5b11c5a | 12/05/2017 10:10 |
| 232c2f8e-1934-9adf-1e67-e316db0f762f | 12/05/2017 10:10 |
| 5d0bc147-101b-76c1-e62e-76d95a8c1411 | 12/05/2017 10:10 |

```
<ui>
      <name>7b09909d-334f-cdbb-9677-69bbb9bf2728</name>
      <displayName>DF 1 Scratch Rebuild DF</displayName>
      <description>{
  "creationDate": "Tue, 17 Nov 2015 15:34:32 GMT",
  "lastUpdateDate": "Fri, 05 May 2017 08:43:07 GMT",
  "tags": []
}</description>
      <data>{
  "id": "7b09909d-334f-cdbb-9677-69bbb9bf2728",
  "name": "DF 1 Scratch Rebuild DF",
  "creationDate": "Tue, 17 Nov 2015 15:34:32 GMT",
  "lastUpdateDate": "Fri, 05 May 2017 08:43:07 GMT",
  "thumb": null,
  "screenDetails": [
    {
      "label": "Screen 1",
      "value": "da595aa6-9366-52c4-50c5-a96b24326818"
    },
    {
      "label": "Screen 3",
      "value": "dbc6ac13-dc44-8bd0-cae2-c31f1aceda02"
    },
    {
      "label": "Screen 2",
      "value": "8ec13e5f-38b3-2379-2922-9d82da2639a"
    },
    {
      "label": "Screen 4",
      "value": "a3e51fce-6465-d8b6-6425-e86fd732c4a6"
    },
    {
      "label": "Screen 5",
      "value": "a1629f61-7d3b-d69d-e720-89d94a4c90bc"
    },
    {
      "label": "Screen 6",
      "value": "a69b345a-8e45-b048-9e7d-7e30b168556a"
    },
    {
      "label": "Screen 7",
      "value": "ef18539-335f-ab82-a5f9-34259f4b6a5c"
    },
    {
      "label": "Screen 8",
```

File name: 7b09909d-334f-cdbb-9677-69bbb9bf2728

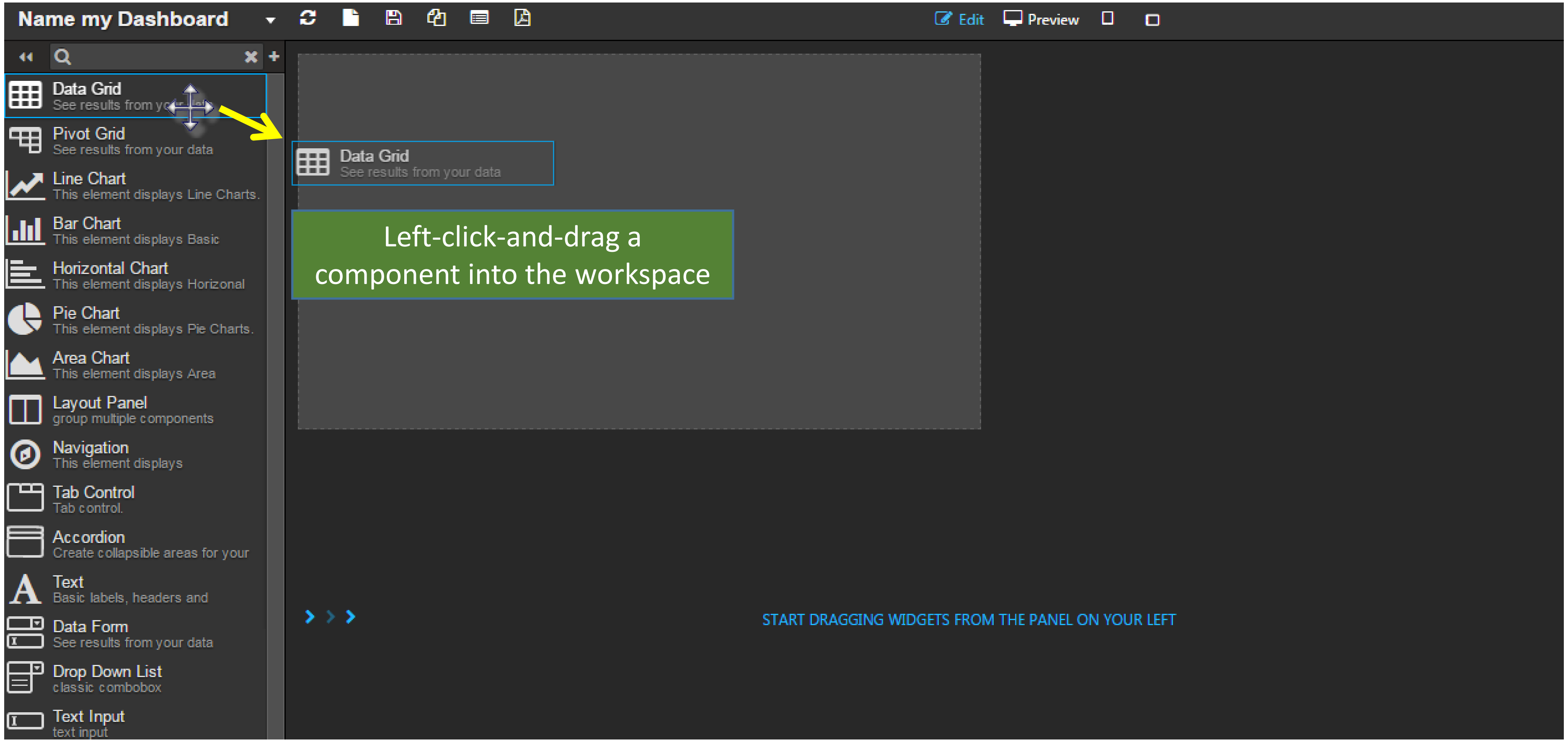
XML Document

it's about time

# Add a Data Grid Component

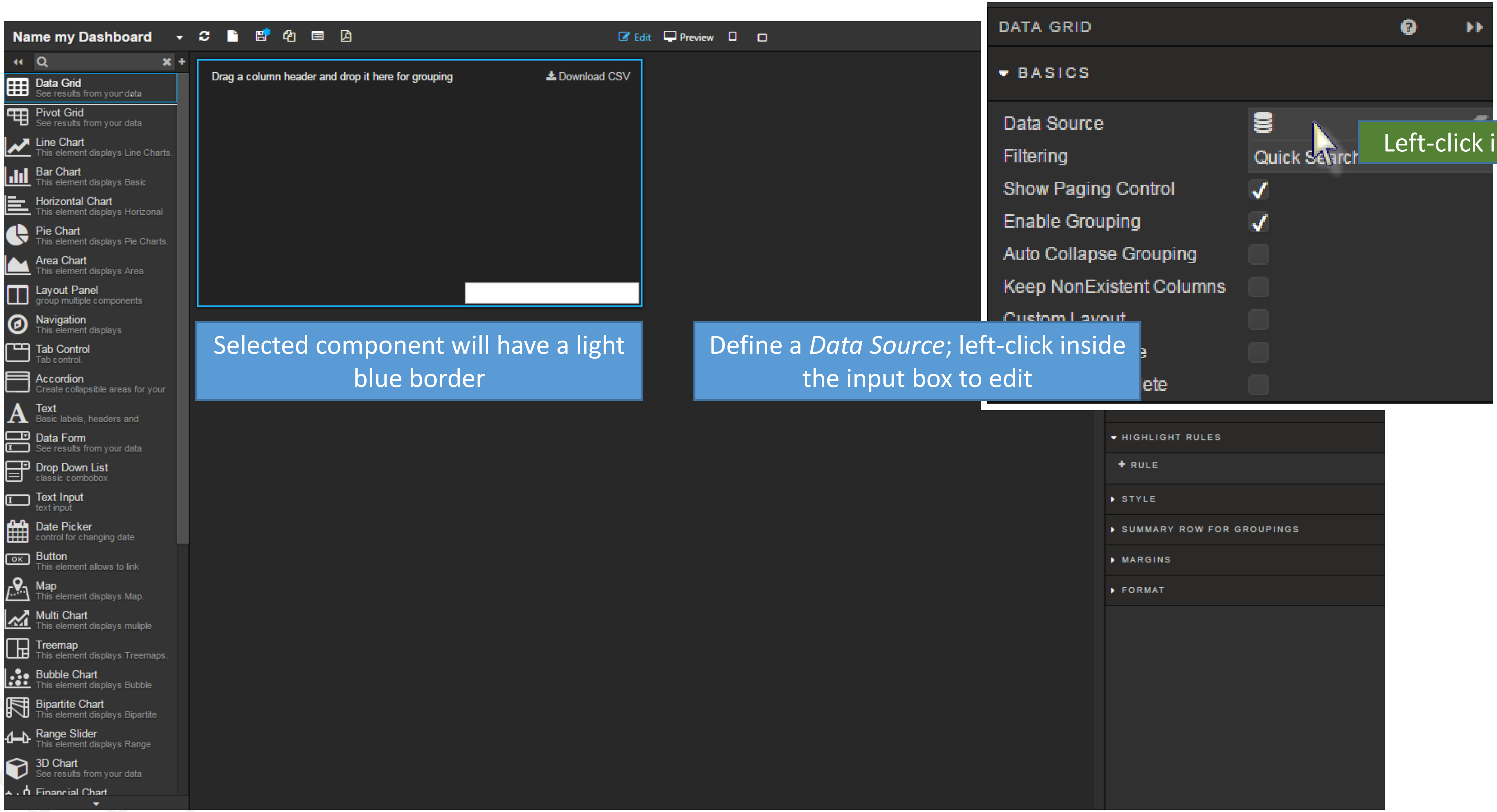
## Dashboards for Kx – "How to" Guide

# Drag a Component into the Workspace; e.g. Data Grid


Name my Dashboard
Edit
Preview
Data Grid
See results from yo
Pivot Grid
See results from your data
Line Chart
This element displays Line Charts.
Bar Chart
This element displays Basic
Horizontal Chart
This element displays Horizonal
Pie Chart
This element displays Pie Charts.
Area Chart
This element displays Area
Layout Panel
group multiple components
Navigation
This element displays
Tab Control
Tab control.
Accordion
Create collapsible areas for your
Text
Basic labels, headers and
Data Form
See results from your data
Drop Down List
classic combobox
Text Input
text input
Data Grid
See results from your data
START DRAGGING WIDGETS FROM THE PANEL ON YOUR LEFT

Left-click-and-drag a component into the workspace

# Configure Properties of a Data Grid


Name my Dashboard
Edit
Preview
Data Grid
See results from your data
Pivot Grid
See results from your data
Line Chart
This element displays Line Charts.
Bar Chart
This element displays Basic
Horizontal Chart
This element displays Horizonal
Pie Chart
This element displays Pie Charts.
Area Chart
This element displays Area
Layout Panel
group multiple components
Navigation
This element displays
Tab Control
Tab control.
Accordion
Create collapsible areas for your
Text
Basic labels, headers and
Data Form
See results from your data
Drop Down List
classic combobox
Text Input
text input
Date Picker
control for changing date
Button
This element allows to link
Map
This element displays Map.
Multi Chart
This element displays muliple
Treemap
This element displays Treemaps.
Bubble Chart
This element displays Bubble
Bipartite Chart
This element displays Bipartite
Range Slider
This element displays Range
3D Chart
See results from your data
Financial Chart
Drag a column header and drop it here for grouping
Download CSV
DATA GRID
BASICS
HIGHLIGHT RULES
+ RULE
STYLE
SUMMARY ROW FOR GROUPINGS
MARGINS
FORMAT

Data Source

Filtering: Quick Search

Show Paging Control ✓

Enable Grouping ✓

Auto Collapse Grouping □

Keep NonExistent Columns □

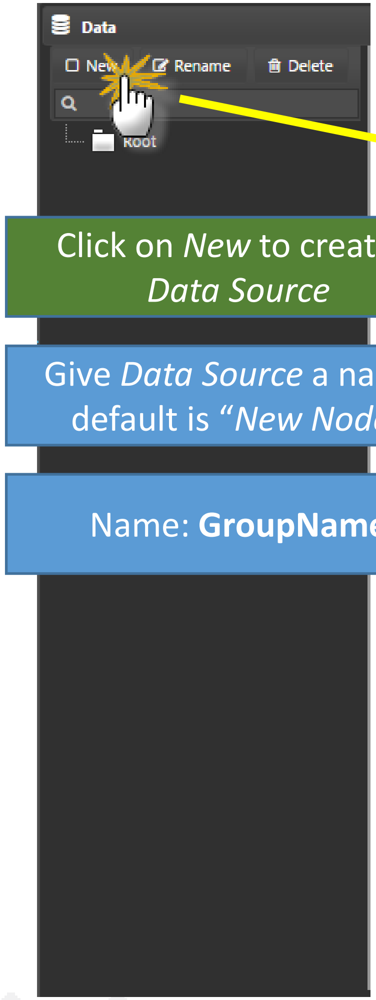
Custom Layout □

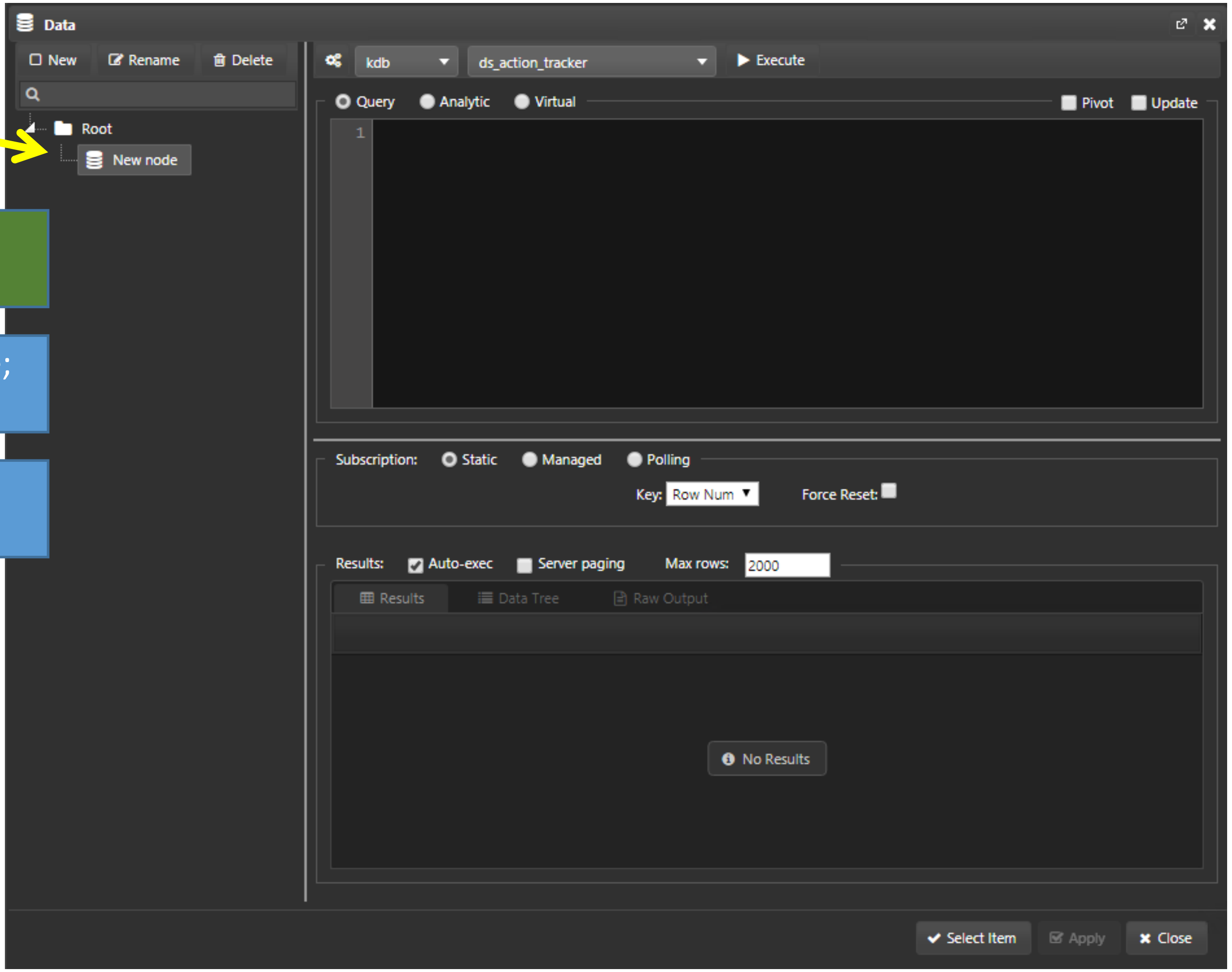
Left-click inside box

Selected component will have a light blue border

Define a *Data Source*; left-click inside the input box to edit

# Create New Data Source: **GroupName**





kdb

ds_action_tracker

○ Query ○ Analytic ○ Virtual □ Pivot □ Update

Subscription: ○ Static ○ Managed ○ Polling

Key: Row Num

Force Reset: □

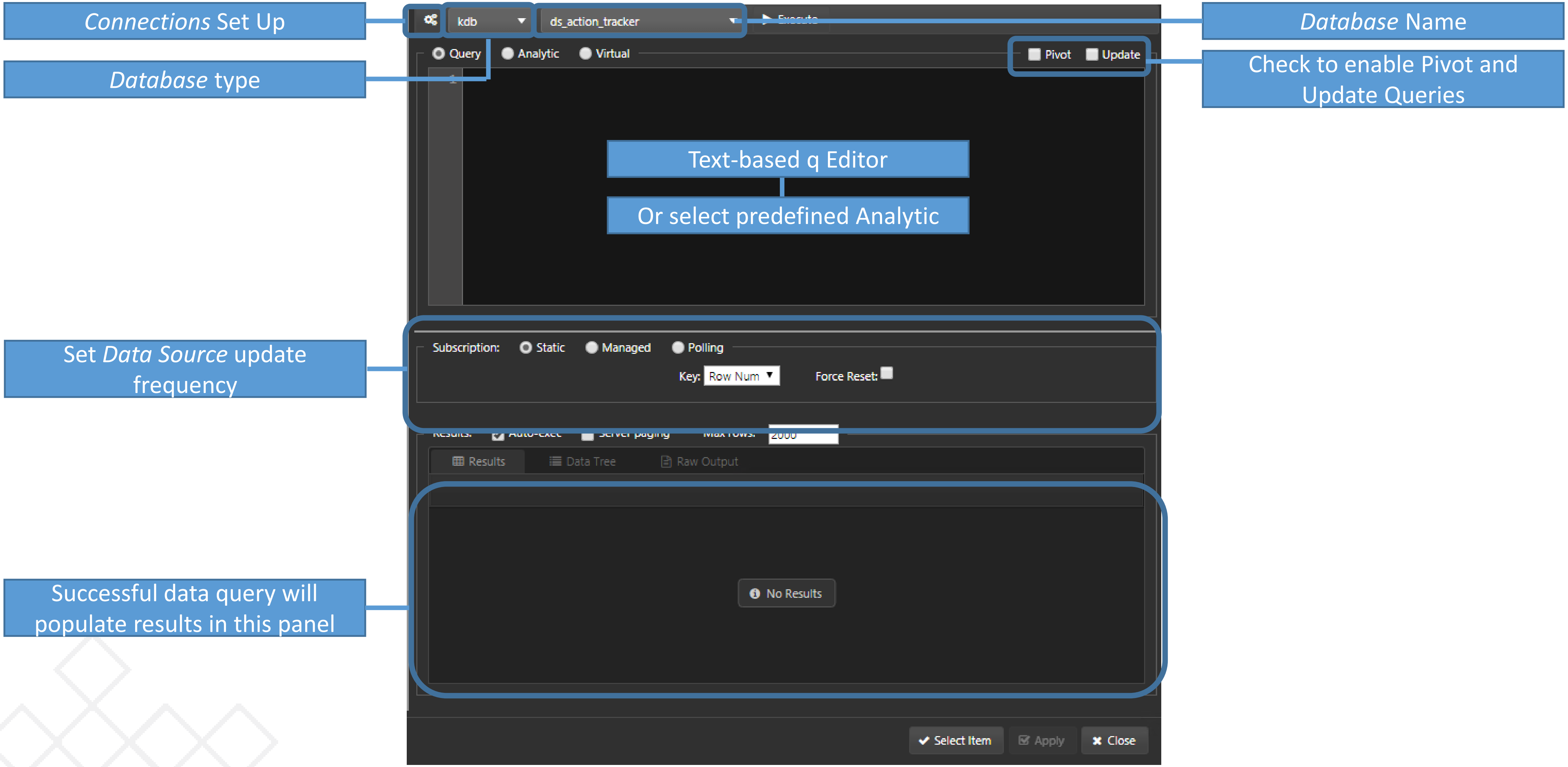
Results: ☑ Auto-exec □ Server paging Max rows: 2000

Click on *New* to create a *Data Source*

Give *Data Source* a name; default is *“New Node”*

Name: **GroupName**

# What’s in a Data Editor?



*Connections* Set Up

*Database* type

*Database* Name

Check to enable Pivot and Update Queries

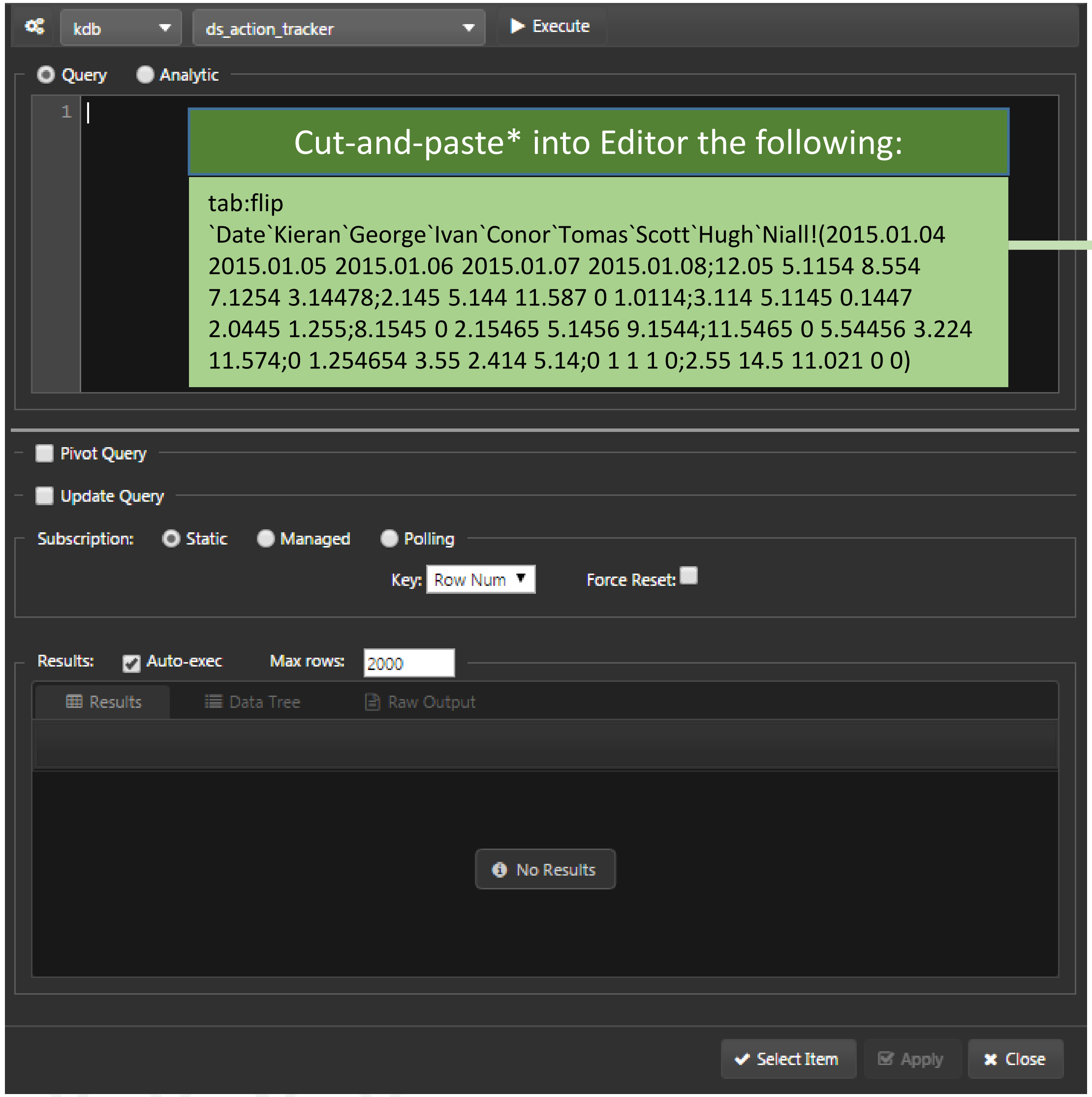
Text-based q Editor

Or select predefined Analytic

Set *Data Source* update frequency

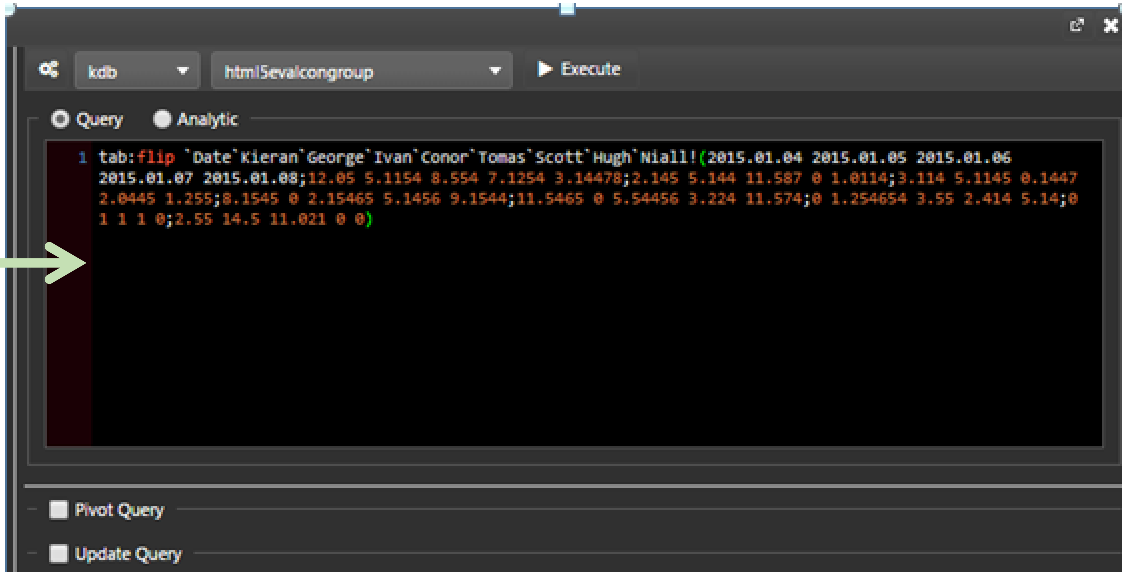
Successful data query will populate results in this panel

# Sample Data Example: **GroupName**



Cut-and-paste* into Editor the following:

```
tab:flip
`Date`Kieran`George`Ivan`Conor`Tomas`Scott`Hugh`Niall!(2015.01.04
2015.01.05 2015.01.06 2015.01.07 2015.01.08;12.05 5.1154 8.554
7.1254 3.14478;2.145 5.144 11.587 0 1.0114;3.114 5.1145 0.1447
2.0445 1.255;8.1545 0 2.15465 5.1456 9.1544;11.5465 0 5.54456 3.224
11.574;0 1.254654 3.55 2.414 5.14;0 1 1 1 0;2.55 14.5 11.021 0 0)
```



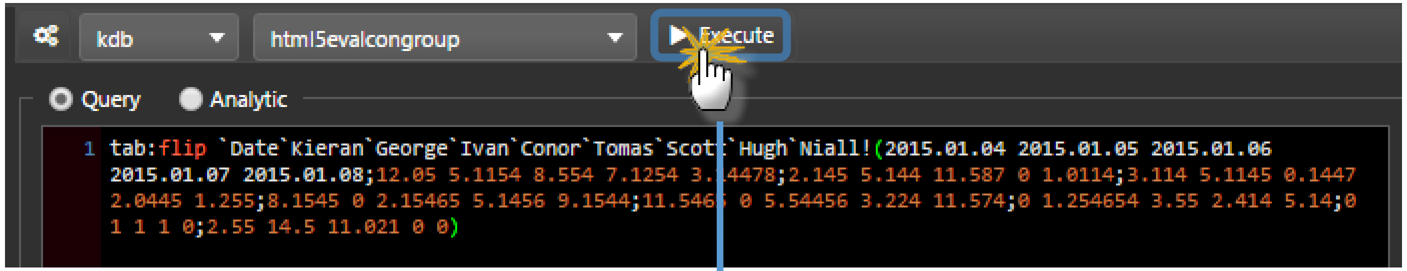
```
tab:flip `Date`Kieran`George`Ivan`Conor`Tomas`Scott`Hugh`Niall!(2015.01.04 2015.01.05 2015.01.06
2015.01.07 2015.01.08;12.05 5.1154 8.554 7.1254 3.14478;2.145 5.144 11.587 0 1.0114;3.114 5.1145 0.1447
2.0445 1.255;8.1545 0 2.15465 5.1456 9.1544;11.5465 0 5.54456 3.224 11.574;0 1.254654 3.55 2.414 5.14;0
1 1 1 0;2.55 14.5 11.021 0 0)
```

Because this is a self-contained query, the database connection doesn’t matter.

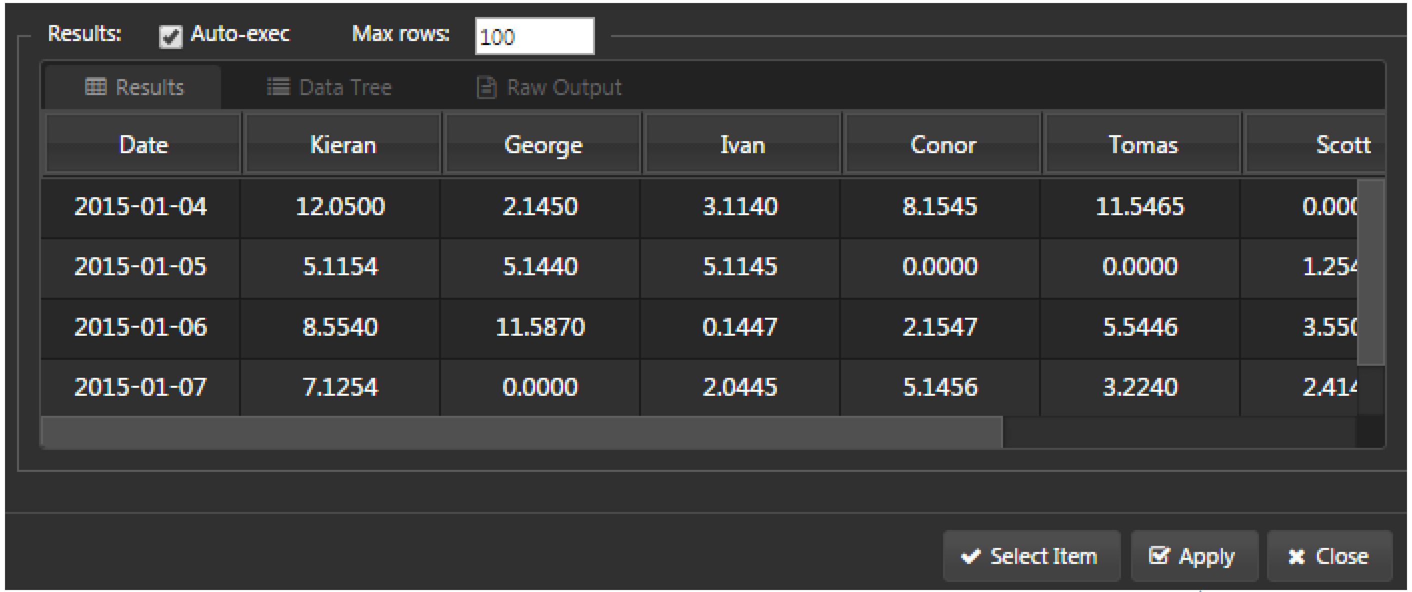
⚠ If you are pulling data from a database, ensure the database connection is correct.

⚠ * When doing a cut-and-paste of queries into the editor watch for line-breaks on new rows which may break the query; delete to remove. Hand-typing a query will prevent this.

# Execute a Data Source



```
tab:flip `Date`Kieran`George`Ivan`Conor`Tomas`Scott`Hugh`Niall!(2015.01.04 2015.01.05 2015.01.06
2015.01.07 2015.01.08;12.05 5.1154 8.554 7.1254 3.14478;2.145 5.144 11.587 0 1.0114;3.114 5.1145 0.1447
2.0445 1.255;8.1545 0 2.15465 5.1456 9.1544;11.5465 0 5.54456 3.224 11.574;0 1.254654 3.55 2.414 5.14;0
1 1 1 0;2.55 14.5 11.021 0 0)
```

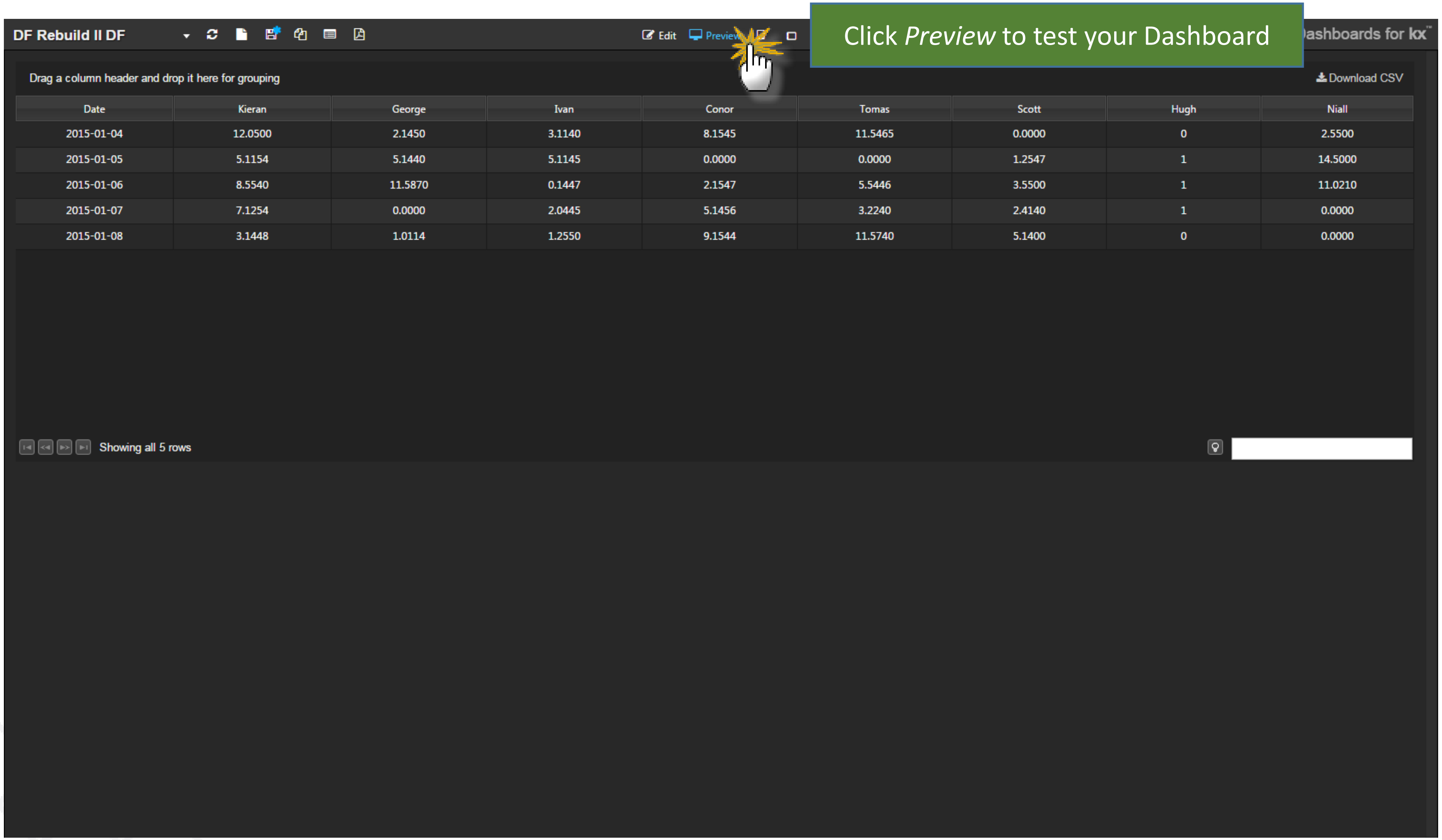


| Date | Kieran | George | Ivan | Conor | Tomas | Scott |
|---|---|---|---|---|---|---|
| 2015-01-04 | 12.0500 | 2.1450 | 3.1140 | 8.1545 | 11.5465 | 0.000 |
| 2015-01-05 | 5.1154 | 5.1440 | 5.1145 | 0.0000 | 0.0000 | 1.254 |
| 2015-01-06 | 8.5540 | 11.5870 | 0.1447 | 2.1547 | 5.5446 | 3.550 |
| 2015-01-07 | 7.1254 | 0.0000 | 2.0445 | 5.1456 | 3.2240 | 2.414 |

Important to populate data using the correct order:

1. Execute
2. Apply
3. Select Item

# Executed Data Populates The Data Grid



Click *Preview* to test your Dashboard

Drag a column header and drop it here for grouping

| Date | Kieran | George | Ivan | Conor | Tomas | Scott | Hugh | Niall |
|---|---|---|---|---|---|---|---|---|
| 2015-01-04 | 12.0500 | 2.1450 | 3.1140 | 8.1545 | 11.5465 | 0.0000 | 0 | 2.5500 |
| 2015-01-05 | 5.1154 | 5.1440 | 5.1145 | 0.0000 | 0.0000 | 1.2547 | 1 | 14.5000 |
| 2015-01-06 | 8.5540 | 11.5870 | 0.1447 | 2.1547 | 5.5446 | 3.5500 | 1 | 11.0210 |
| 2015-01-07 | 7.1254 | 0.0000 | 2.0445 | 5.1456 | 3.2240 | 2.4140 | 1 | 0.0000 |
| 2015-01-08 | 3.1448 | 1.0114 | 1.2550 | 9.1544 | 11.5740 | 5.1400 | 0 | 0.0000 |

Showing all 5 rows

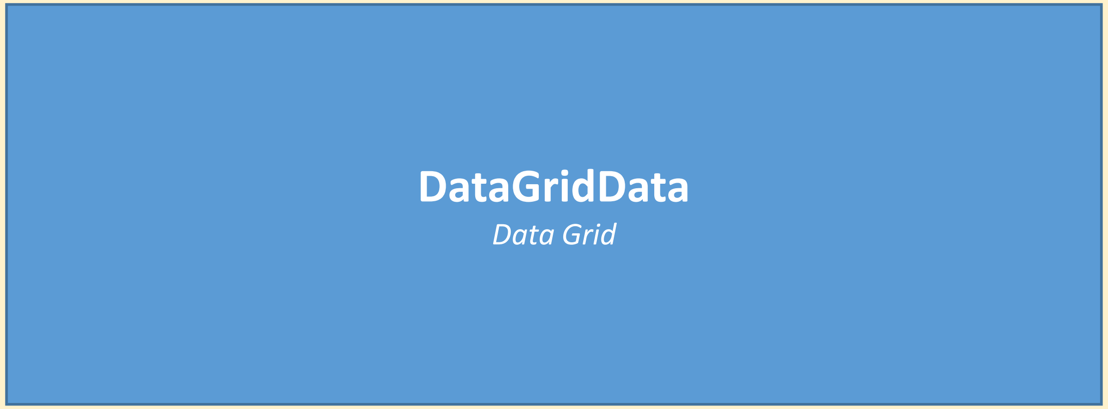
# Try This: Sourcing Data from a kdb+ database

- Name: **DataGridData**
- Connect to html5eval_grp (or html5evalcongroup)
- Max Rows: 1,000

```
select Type, Month, Group, Return, PerReturn, OpenDate, CloseDate
from TradeData
```

If doing a cut-and-paste of queries into the editor watch for line-breaks which can break the query; delete these line-breaks will restore functionality. Hand-typing a query will prevent this.

- Execute -> Apply -> Select



**DataGridData**

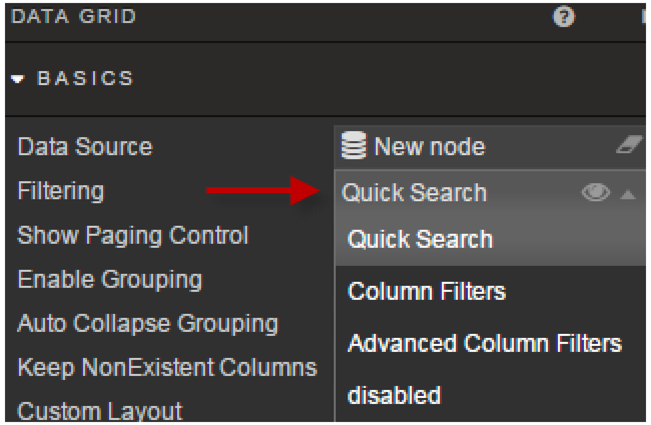
*Data Grid*

it's about time

# Customize Data Grid

## Dashboards for Kx – "How to" Guide

# Configure Search Options for **GroupName**: Default is Quick Search



DATA GRID

BASICS

Data Source: New node

Filtering: Quick Search

Show Paging Control

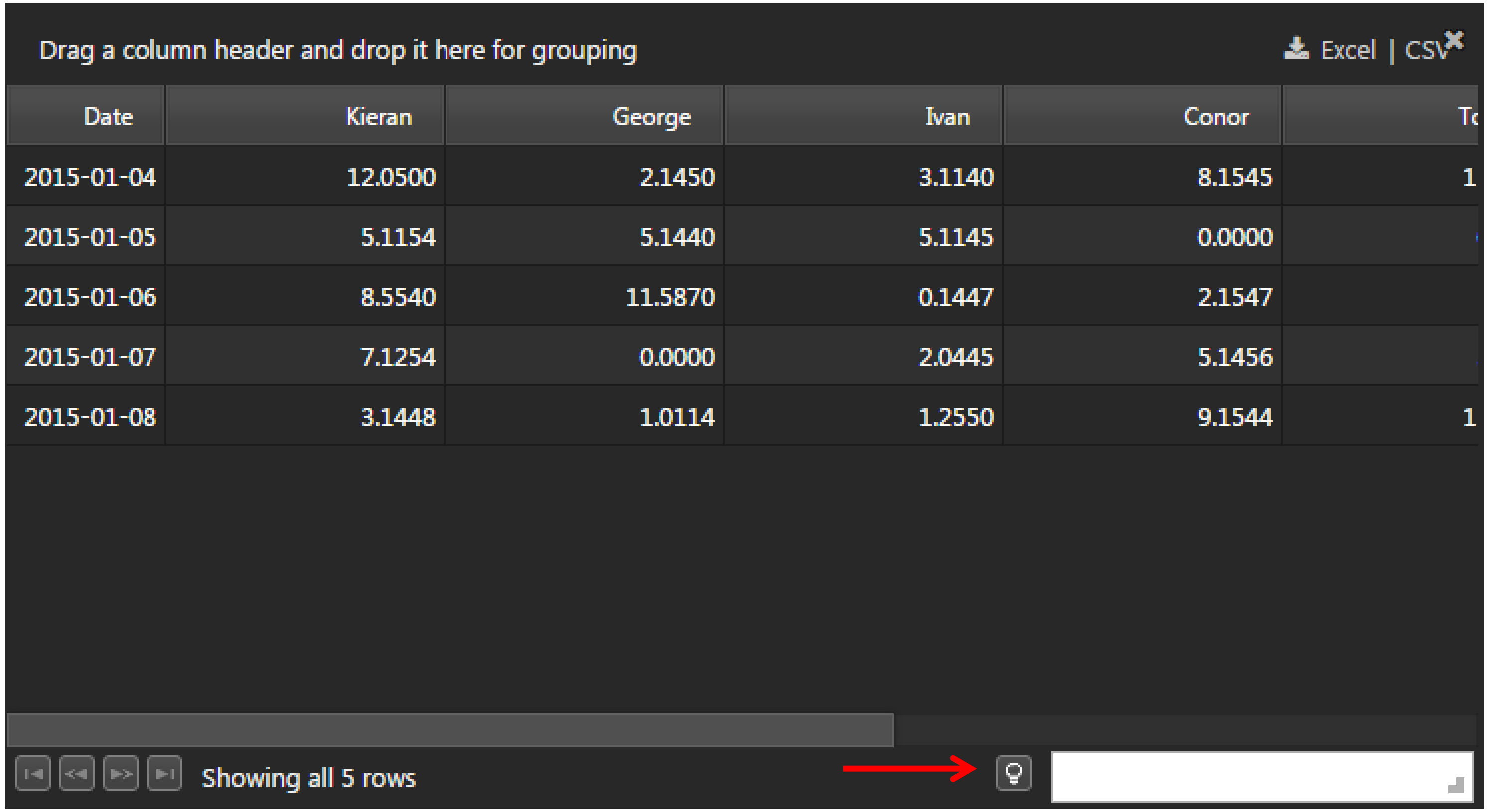
Enable Grouping

Auto Collapse Grouping

Keep NonExistent Columns

Custom Layout

Toggle through Filter options



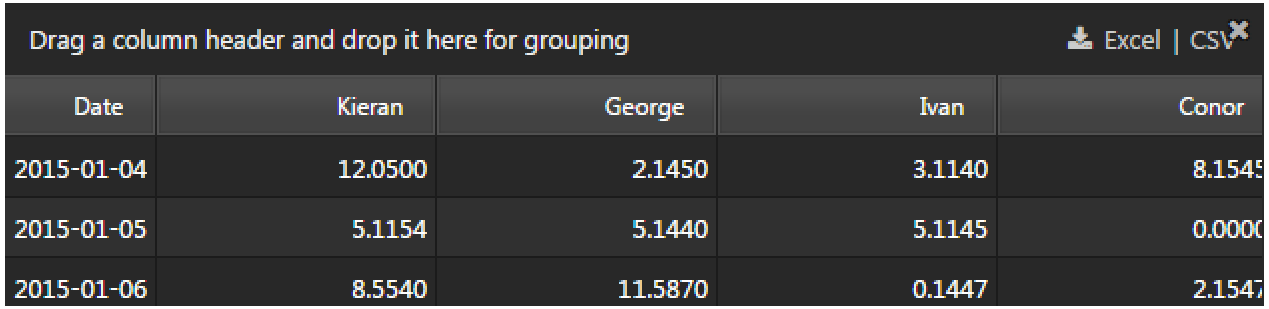
Drag a column header and drop it here for grouping

| Date | Kieran | George | Ivan | Conor | Tc |
|---|---|---|---|---|---|
| 2015-01-04 | 12.0500 | 2.1450 | 3.1140 | 8.1545 | 1 |
| 2015-01-05 | 5.1154 | 5.1440 | 5.1145 | 0.0000 | |
| 2015-01-06 | 8.5540 | 11.5870 | 0.1447 | 2.1547 | |
| 2015-01-07 | 7.1254 | 0.0000 | 2.0445 | 5.1456 | |
| 2015-01-08 | 3.1448 | 1.0114 | 1.2550 | 9.1544 | 1 |

Showing all 5 rows

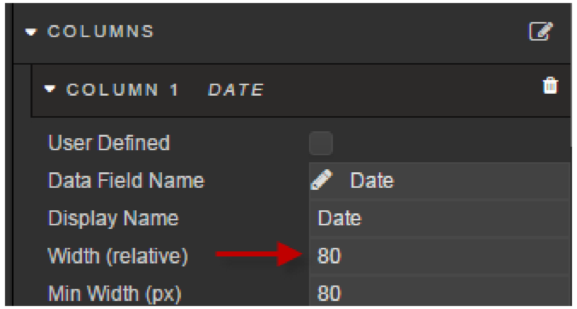
# Optimize Column Widths to Best Display Data

## Change Width (relative) to best display data



Drag a column header and drop it here for grouping

| Date | Kieran | George | Ivan | Conor |
|---|---|---|---|---|
| 2015-01-04 | 12.0500 | 2.1450 | 3.1140 | 8.1545 |
| 2015-01-05 | 5.1154 | 5.1440 | 5.1145 | 0.0000 |
| 2015-01-06 | 8.5540 | 11.5870 | 0.1447 | 2.1547 |



User Defined □

Data Field Name: Date

Display Name: Date

Width (relative): 80

Min Width (px): 80

## How does *Width (relative)* work?

*Width (relative)* applies a scale across all data columns. If *Width (relative)* is set to “1”; i.e. same value, then each column will have the same width.

However, if one column is set to “2”, then the relative width of the column will change, contingent on the total number of columns displayed.

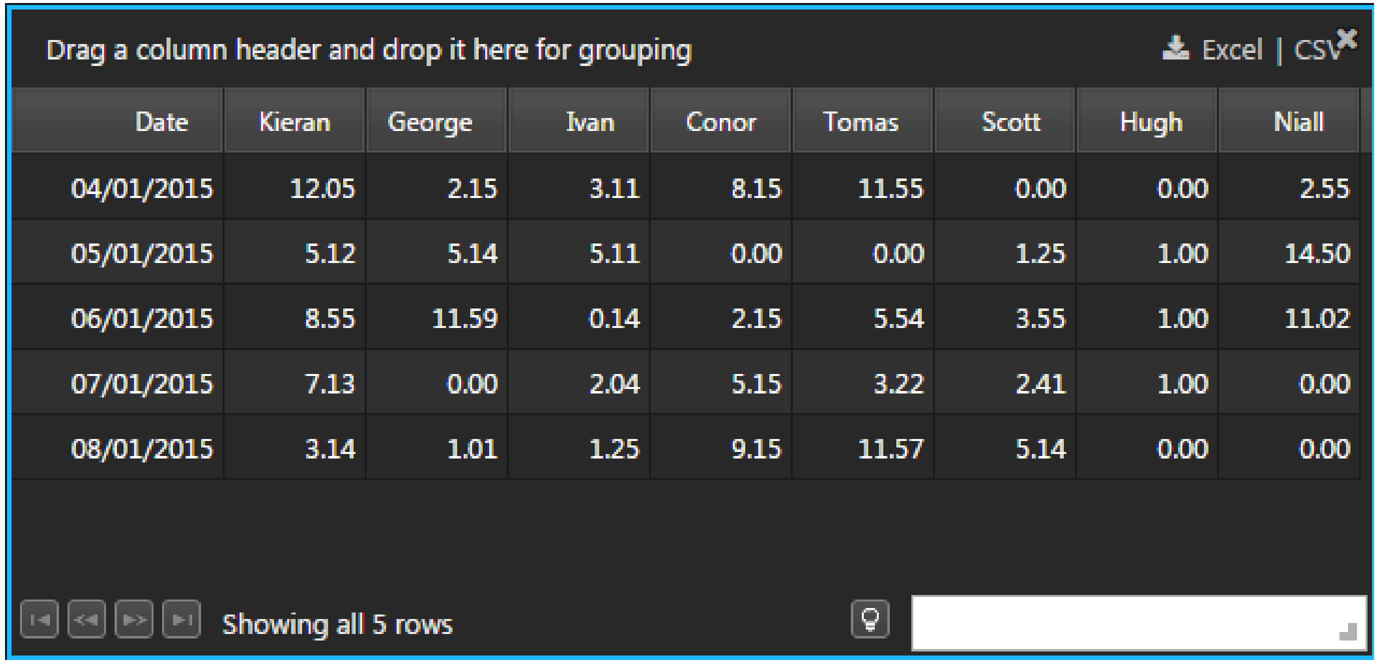
For example: *Width (relative)* for a five column data grid is “2”, “1”, “1”, “1”, “1”.
Total *Width (relative)* = 2+1+1+1+1 = 6
Relative width for each column is therefore: *Width (relative)* / Sum of *Width (relative)*
Relative width = (2/6), (1/6), (1/6), (1/6), (1/6)
Relative width = 33%, 16%, 16%, 16%, 16%

⚠ Note: *Min Width (px)* will also influence *Width (relative).* Set *Min Width (px)* to zero for all columns if only *Width (relative)* is to be used

# Define Column Data Types

Column 1 Date = *Format: Date* and *Date Format: DD/MM/YYYY* and *Width (relative)*: 15

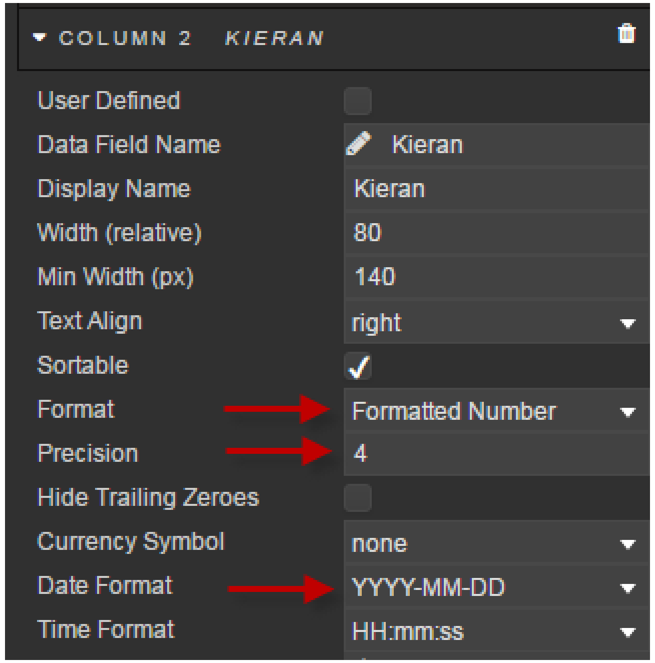
Columns 2-9 = *Format: Formatted Number* and *Precision: 2* and *Width (relative):* 10



Drag a column header and drop it here for grouping

| Date | Kieran | George | Ivan | Conor | Tomas | Scott | Hugh | Niall |
|---|---|---|---|---|---|---|---|---|
| 04/01/2015 | 12.05 | 2.15 | 3.11 | 8.15 | 11.55 | 0.00 | 0.00 | 2.55 |
| 05/01/2015 | 5.12 | 5.14 | 5.11 | 0.00 | 0.00 | 1.25 | 1.00 | 14.50 |
| 06/01/2015 | 8.55 | 11.59 | 0.14 | 2.15 | 5.54 | 3.55 | 1.00 | 11.02 |
| 07/01/2015 | 7.13 | 0.00 | 2.04 | 5.15 | 3.22 | 2.41 | 1.00 | 0.00 |
| 08/01/2015 | 3.14 | 1.01 | 1.25 | 9.15 | 11.57 | 5.14 | 0.00 | 0.00 |

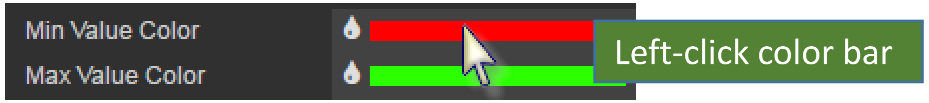
Showing all 5 rows



COLUMN 2 KIERAN

| | |
|---|---|
| User Defined | □ |
| Data Field Name | Kieran |
| Display Name | Kieran |
| Width (relative) | 80 |
| Min Width (px) | 140 |
| Text Align | right |
| Sortable | ✓ |
| Format | Formatted Number |
| Precision | 4 |
| Hide Trailing Zeroes | □ |
| Currency Symbol | none |
| Date Format | YYYY-MM-DD |
| Time Format | HH:mm:ss |

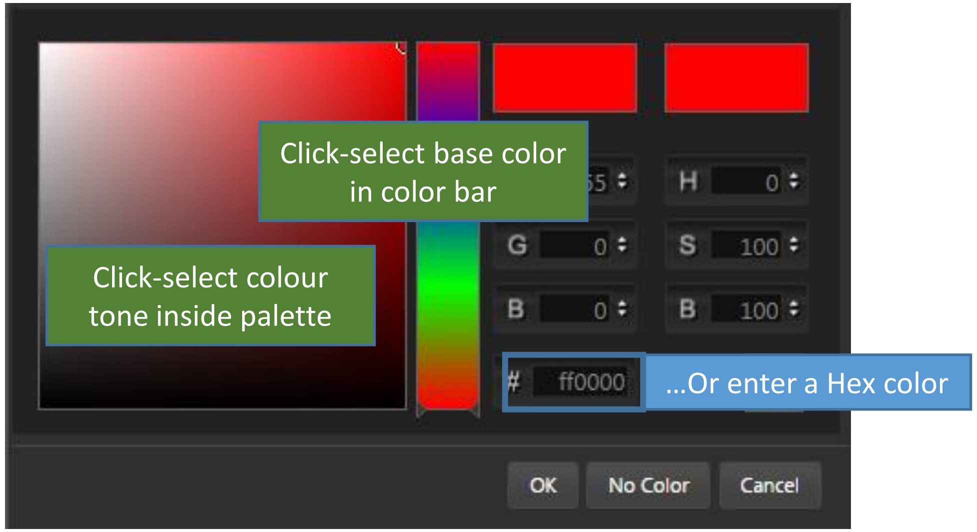
# Set Min / Max value for a numeric column



Min Value Color

Max Value Color

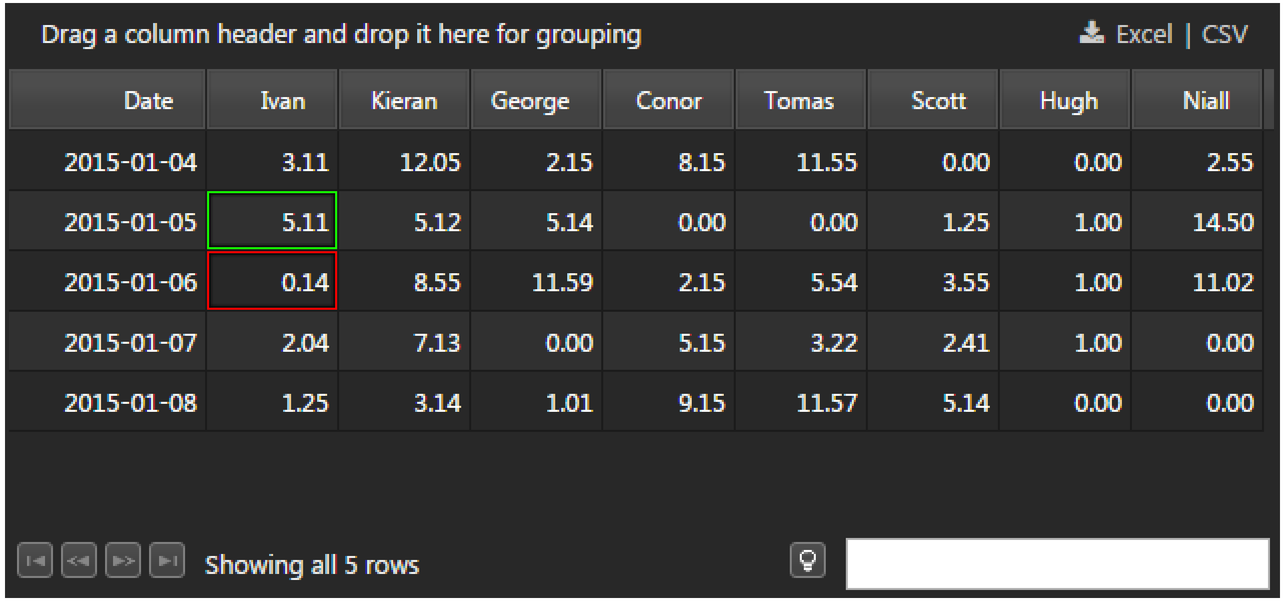
Left-click color bar



Click-select base color in color bar

Click-select colour tone inside palette

...Or enter a Hex color



Drag a column header and drop it here for grouping

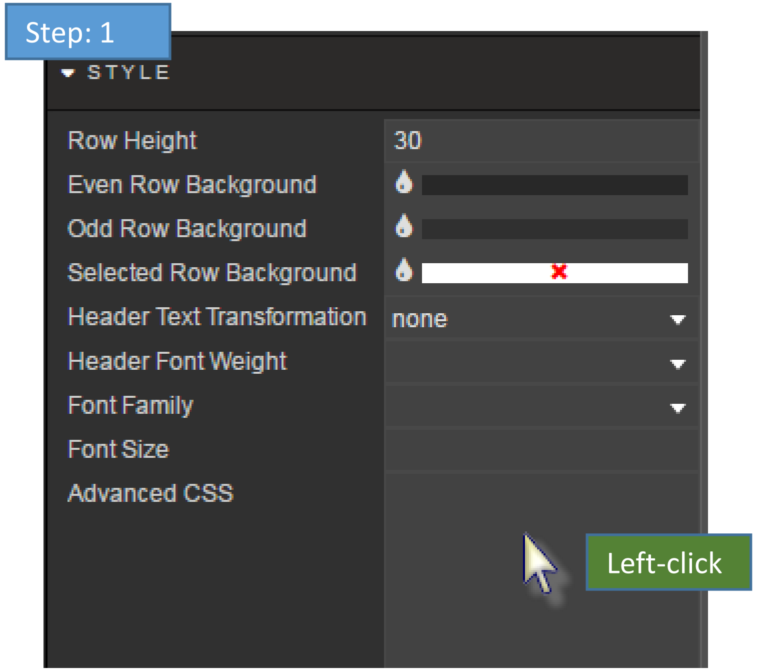
| Date | Ivan | Kieran | George | Conor | Tomas | Scott | Hugh | Niall |
|---|---|---|---|---|---|---|---|---|
| 2015-01-04 | 3.11 | 12.05 | 2.15 | 8.15 | 11.55 | 0.00 | 0.00 | 2.55 |
| 2015-01-05 | 5.11 | 5.12 | 5.14 | 0.00 | 0.00 | 1.25 | 1.00 | 14.50 |
| 2015-01-06 | 0.14 | 8.55 | 11.59 | 2.15 | 5.54 | 3.55 | 1.00 | 11.02 |
| 2015-01-07 | 2.04 | 7.13 | 0.00 | 5.15 | 3.22 | 2.41 | 1.00 | 0.00 |
| 2015-01-08 | 1.25 | 3.14 | 1.01 | 9.15 | 11.57 | 5.14 | 0.00 | 0.00 |

Showing all 5 rows

Set Min/Max Value color for each column

# Apply CSS to Data Grid Header

Step: 1

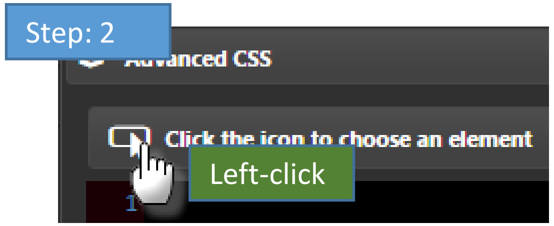


▾ STYLE

| | |
|---|---|
| Row Height | 30 |
| Even Row Background | |
| Odd Row Background | |
| Selected Row Background | |
| Header Text Transformation | none |
| Header Font Weight | |
| Font Family | |
| Font Size | |
| Advanced CSS | |

Left-click

Step: 2

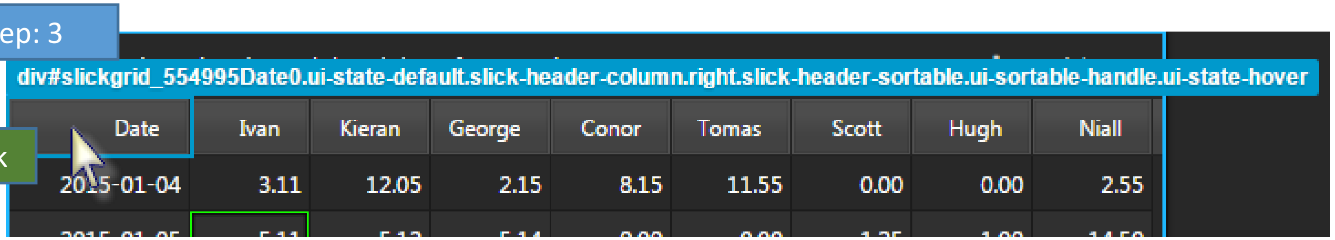


Advanced CSS

Click the icon to choose an element

Left-click

Step: 3

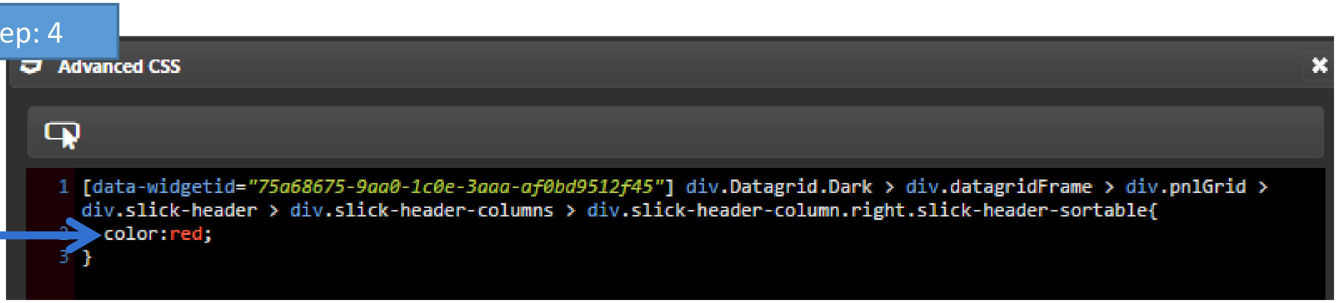


div#slickgrid_554995Date0.ui-state-default.slick-header-column.right.slick-header-sortable.ui-sortable-handle.ui-state-hover

| Date | Ivan | Kieran | George | Conor | Tomas | Scott | Hugh | Niall |
|---|---|---|---|---|---|---|---|---|
| 2015-01-04 | 3.11 | 12.05 | 2.15 | 8.15 | 11.55 | 0.00 | 0.00 | 2.55 |

Left-click

Step: 4

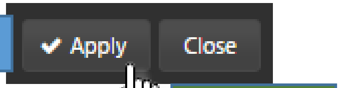


Advanced CSS

```
[data-widgetid="75a68675-9aa0-1c0e-3aaa-af0bd9512f45"] div.Datagrid.Dark > div.datagridFrame > div.pnlGrid >
div.slick-header > div.slick-header-columns > div.slick-header-column.right.slick-header-sortable{
  color:red;
}
```

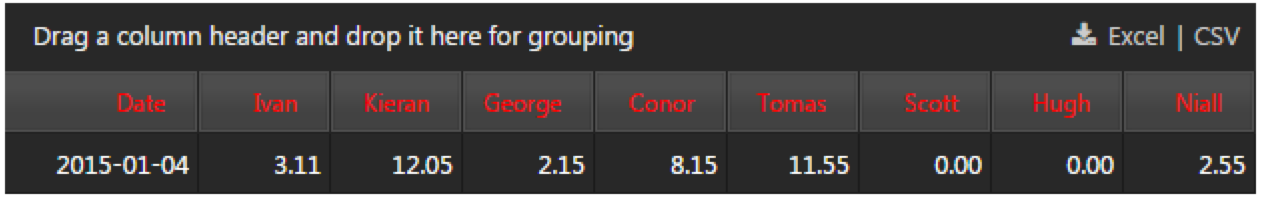
Add CSS: e.g. “color:red;”

Step: 5



Left-click

Step 6: RESULT:

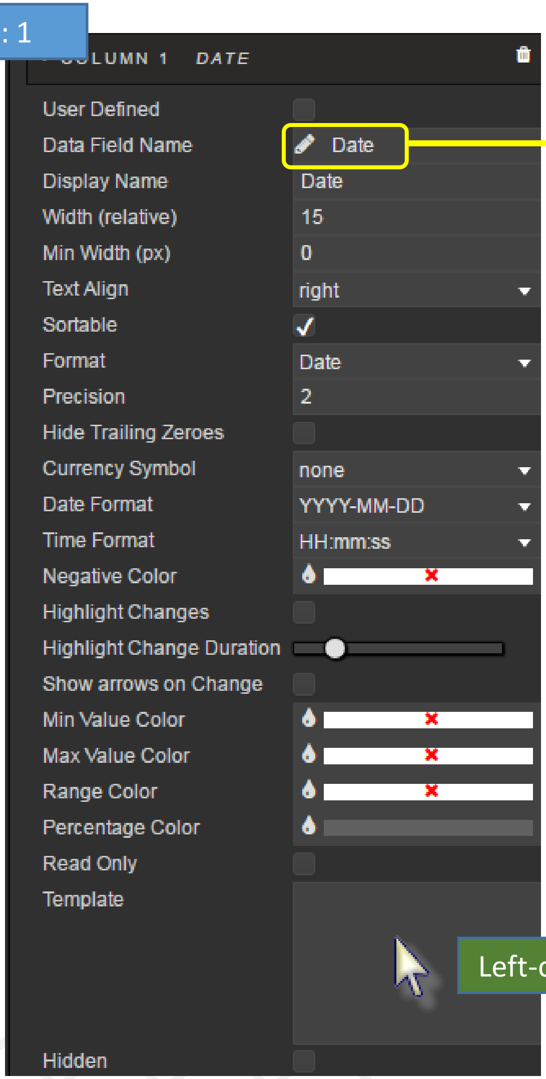


Drag a column header and drop it here for grouping

| Date | Ivan | Kieran | George | Conor | Tomas | Scott | Hugh | Niall |
|---|---|---|---|---|---|---|---|---|
| 2015-01-04 | 3.11 | 12.05 | 2.15 | 8.15 | 11.55 | 0.00 | 0.00 | 2.55 |

# Use Template to Apply Format to Column

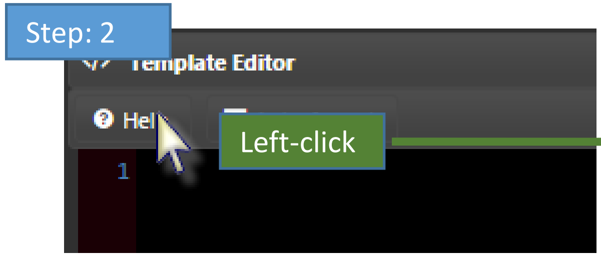
Step: 1



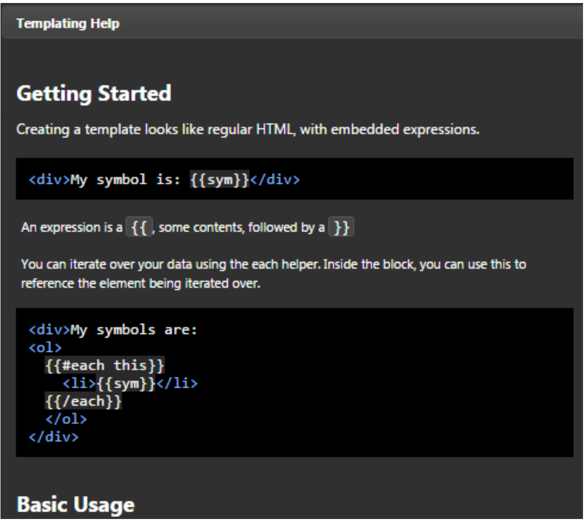
| | |
|---|---|
| User Defined | □ |
| Data Field Name | Date |
| Display Name | Date |
| Width (relative) | 15 |
| Min Width (px) | 0 |
| Text Align | right |
| Sortable | ✓ |
| Format | Date |
| Precision | 2 |
| Hide Trailing Zeroes | □ |
| Currency Symbol | none |
| Date Format | YYYY-MM-DD |
| Time Format | HH:mm:ss |
| Negative Color | |
| Highlight Changes | □ |
| Highlight Change Duration | |
| Show arrows on Change | □ |
| Min Value Color | |
| Max Value Color | |
| Range Color | |
| Percentage Color | |
| Read Only | □ |
| Template | |
| Hidden | □ |

Left-click

Step: 2



Left-click



Templating Help

**Getting Started**

Creating a template looks like regular HTML, with embedded expressions.

```
<div>My symbol is: {{sym}}</div>
```

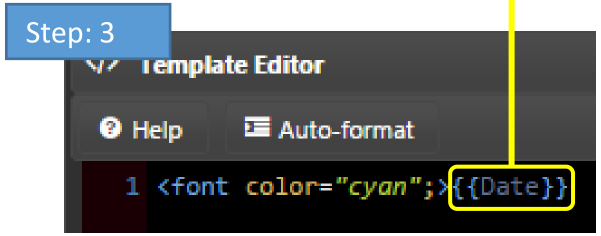
An expression is a {{ , some contents, followed by a }}

You can iterate over your data using the each helper. Inside the block, you can use this to reference the element being iterated over.

```
<div>My symbols are:
<ol>
  {{#each this}}
    <li>{{sym}}</li>
  {{/each}}
  </ol>
</div>
```

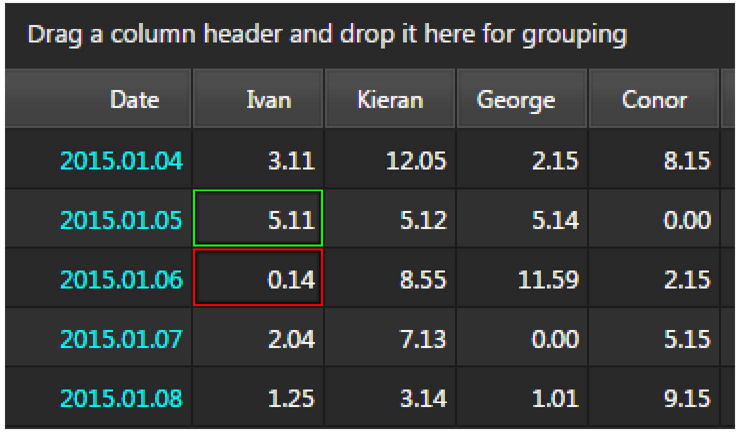
**Basic Usage**

Step: 3



```
<font color="cyan";>{{Date}}
```

Set font color for *Date* column. Remember to enclose variable name inside {{ }} as it appears in database

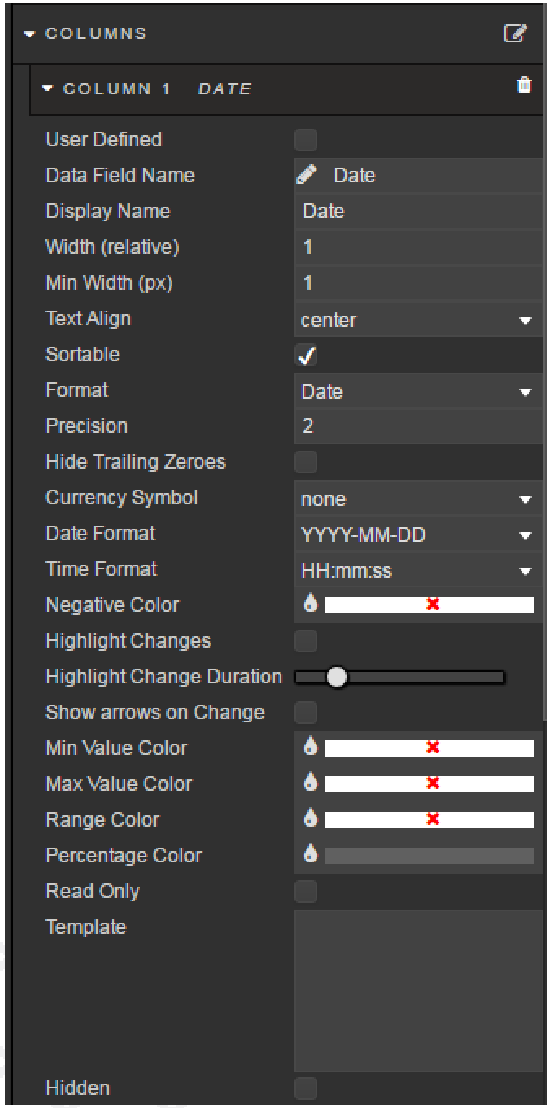


Drag a column header and drop it here for grouping

| Date | Ivan | Kieran | George | Conor |
|---|---|---|---|---|
| 2015.01.04 | 3.11 | 12.05 | 2.15 | 8.15 |
| 2015.01.05 | 5.11 | 5.12 | 5.14 | 0.00 |
| 2015.01.06 | 0.14 | 8.55 | 11.59 | 2.15 |
| 2015.01.07 | 2.04 | 7.13 | 0.00 | 5.15 |
| 2015.01.08 | 1.25 | 3.14 | 1.01 | 9.15 |

⚠ Template formats will overwrite any highlight rules applied to the column

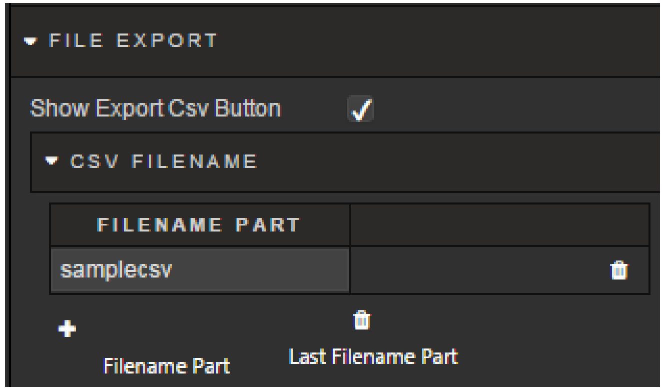
# Check out Data Grid Properties



COLUMNS

COLUMN 1 *DATE*

| | |
|---|---|
| User Defined | ☐ |
| Data Field Name | Date |
| Display Name | Date |
| Width (relative) | 1 |
| Min Width (px) | 1 |
| Text Align | center |
| Sortable | ☑ |
| Format | Date |
| Precision | 2 |
| Hide Trailing Zeroes | ☐ |
| Currency Symbol | none |
| Date Format | YYYY-MM-DD |
| Time Format | HH:mm:ss |
| Negative Color | ✗ |
| Highlight Changes | ☐ |
| Highlight Change Duration | |
| Show arrows on Change | ☐ |
| Min Value Color | ✗ |
| Max Value Color | ✗ |
| Range Color | ✗ |
| Percentage Color | ▨ |
| Read Only | ☐ |
| Template | |
| Hidden | ☐ |

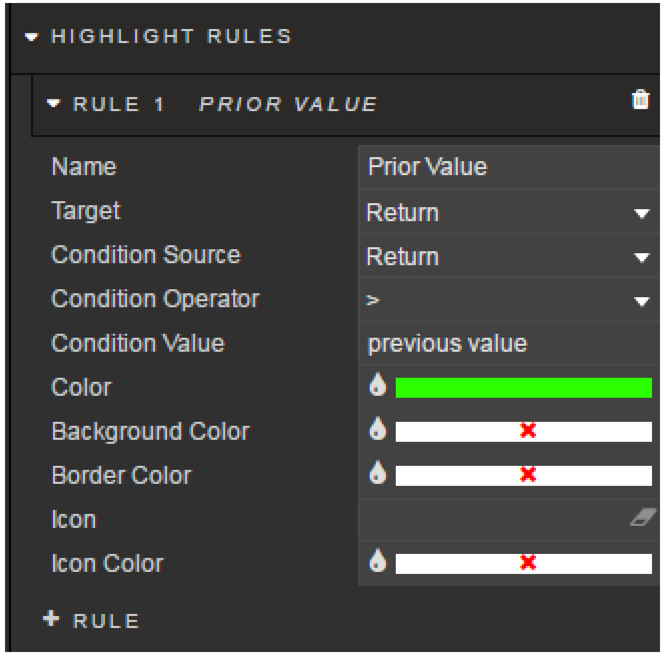


FILE EXPORT

Show Export Csv Button ☑

CSV FILENAME

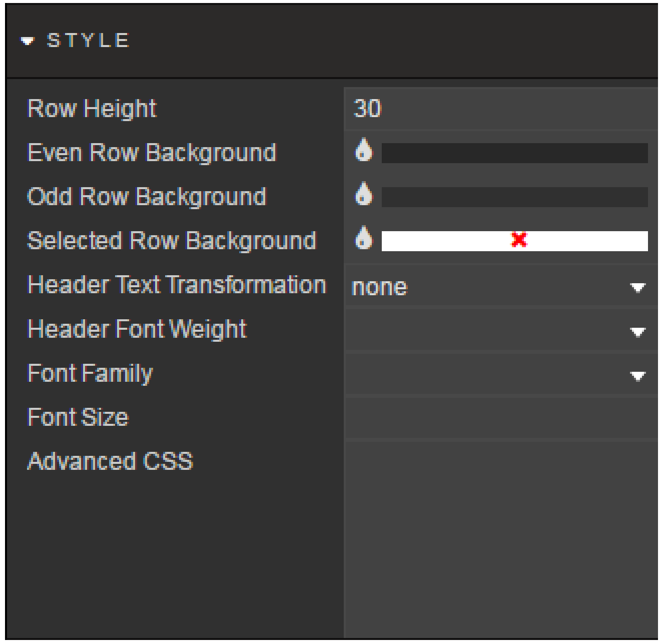
| FILENAME PART | |
|---|---|
| samplecsv | |



HIGHLIGHT RULES

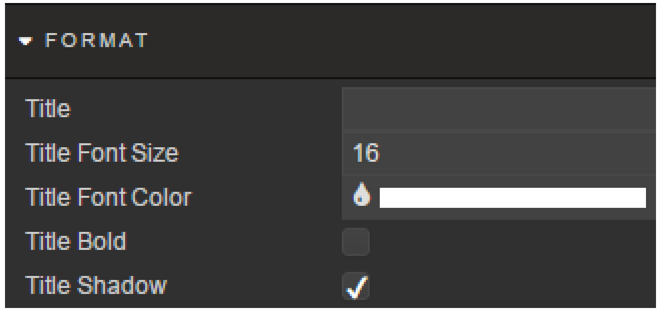
RULE 1 *PRIOR VALUE*

| | |
|---|---|
| Name | Prior Value |
| Target | Return |
| Condition Source | Return |
| Condition Operator | > |
| Condition Value | previous value |
| Color | 🟩 |
| Background Color | ✗ |
| Border Color | ✗ |
| Icon | |
| Icon Color | ✗ |



STYLE

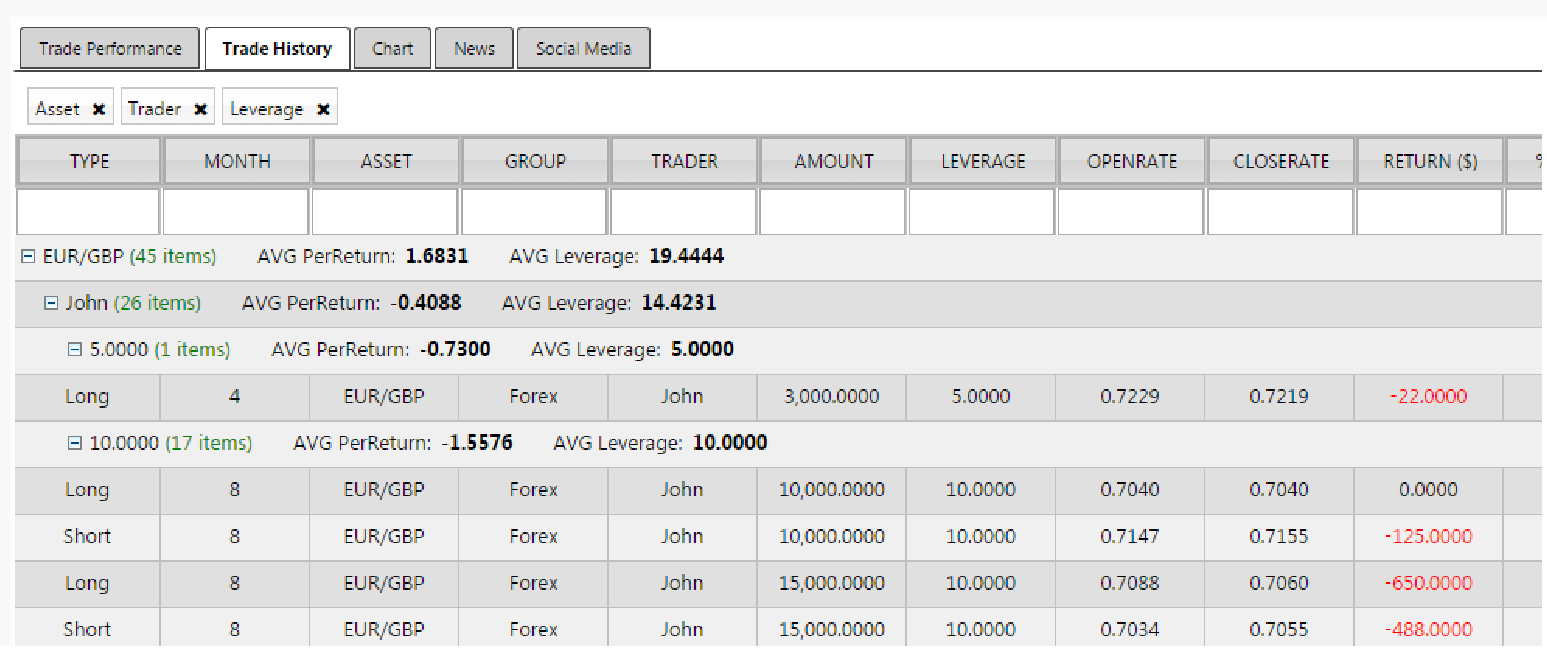
| | |
|---|---|
| Row Height | 30 |
| Even Row Background | ■ |
| Odd Row Background | ■ |
| Selected Row Background | ✗ |
| Header Text Transformation | none |
| Header Font Weight | |
| Font Family | |
| Font Size | |
| Advanced CSS | |



FORMAT

| | |
|---|---|
| Title | |
| Title Font Size | 16 |
| Title Font Color | □ |
| Title Bold | ☐ |
| Title Shadow | ☑ |

# Data Grouping Example



⊟ EUR/GBP (45 items) AVG PerReturn: **1.6831** AVG Leverage: **19.4444**

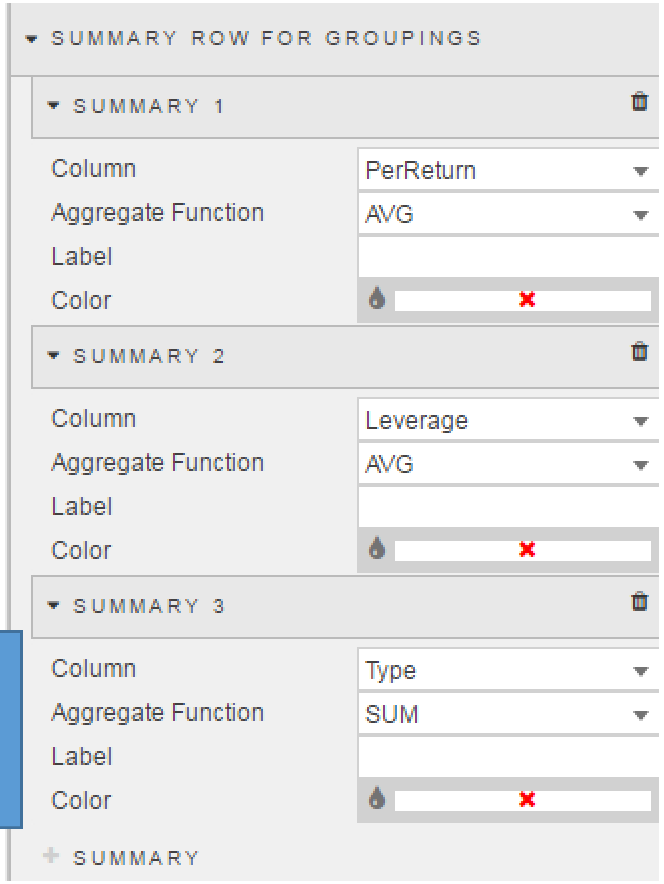
⊟ John (26 items) AVG PerReturn: **-0.4088** AVG Leverage: **14.4231**

⊟ 5.0000 (1 items) AVG PerReturn: **-0.7300** AVG Leverage: **5.0000**

| TYPE | MONTH | ASSET | GROUP | TRADER | AMOUNT | LEVERAGE | OPENRATE | CLOSERATE | RETURN ($) |
|---|---|---|---|---|---|---|---|---|---|
| Long | 4 | EUR/GBP | Forex | John | 3,000.0000 | 5.0000 | 0.7229 | 0.7219 | -22.0000 |

⊟ 10.0000 (17 items) AVG PerReturn: **-1.5576** AVG Leverage: **10.0000**

| TYPE | MONTH | ASSET | GROUP | TRADER | AMOUNT | LEVERAGE | OPENRATE | CLOSERATE | RETURN ($) |
|---|---|---|---|---|---|---|---|---|---|
| Long | 8 | EUR/GBP | Forex | John | 10,000.0000 | 10.0000 | 0.7040 | 0.7040 | 0.0000 |
| Short | 8 | EUR/GBP | Forex | John | 10,000.0000 | 10.0000 | 0.7147 | 0.7155 | -125.0000 |
| Long | 8 | EUR/GBP | Forex | John | 15,000.0000 | 10.0000 | 0.7088 | 0.7060 | -650.0000 |
| Short | 8 | EUR/GBP | Forex | John | 15,000.0000 | 10.0000 | 0.7034 | 0.7055 | -488.0000 |



SUMMARY 1

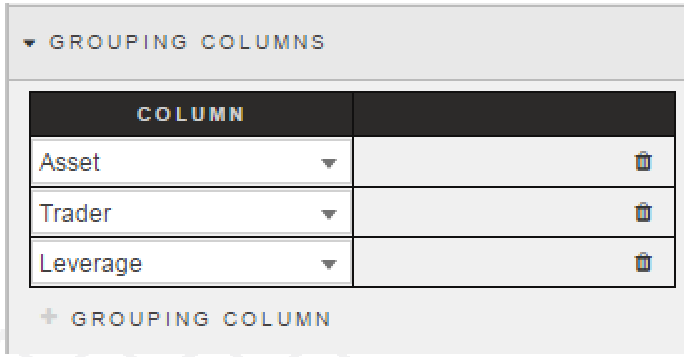
| | |
|---|---|
| Column | PerReturn |
| Aggregate Function | AVG |
| Label | |
| Color | ✖ |

SUMMARY 2

| | |
|---|---|
| Column | Leverage |
| Aggregate Function | AVG |
| Label | |
| Color | ✖ |

SUMMARY 3

| | |
|---|---|
| Column | Type |
| Aggregate Function | SUM |
| Label | |
| Color | ✖ |



| COLUMN | |
|---|---|
| Asset | 🗑 |
| Trader | 🗑 |
| Leverage | 🗑 |

Groupings are done by *Data Source* columns.

Groupings can include pre-determined calculations of dependent data

# Try This: In **DataGridData**

- For: **DataGridData**
- Ensure each column has a valid *Display Name;* e.g. *PerReturn* = “% Return”
- *Format* **Month** and **Return** at *Precision:* 0
- *Format* **PerReturn** as *Percentage*
- Apply a *Range Color* to **Return**
- Set a *Negative Color* to **Return**
- Create a *Summary Grouping* for **Type** and **Group**
- Create a *Summary Row for Grouping* for *Avg* **Return**

it's about time

# Adding Input Parameters

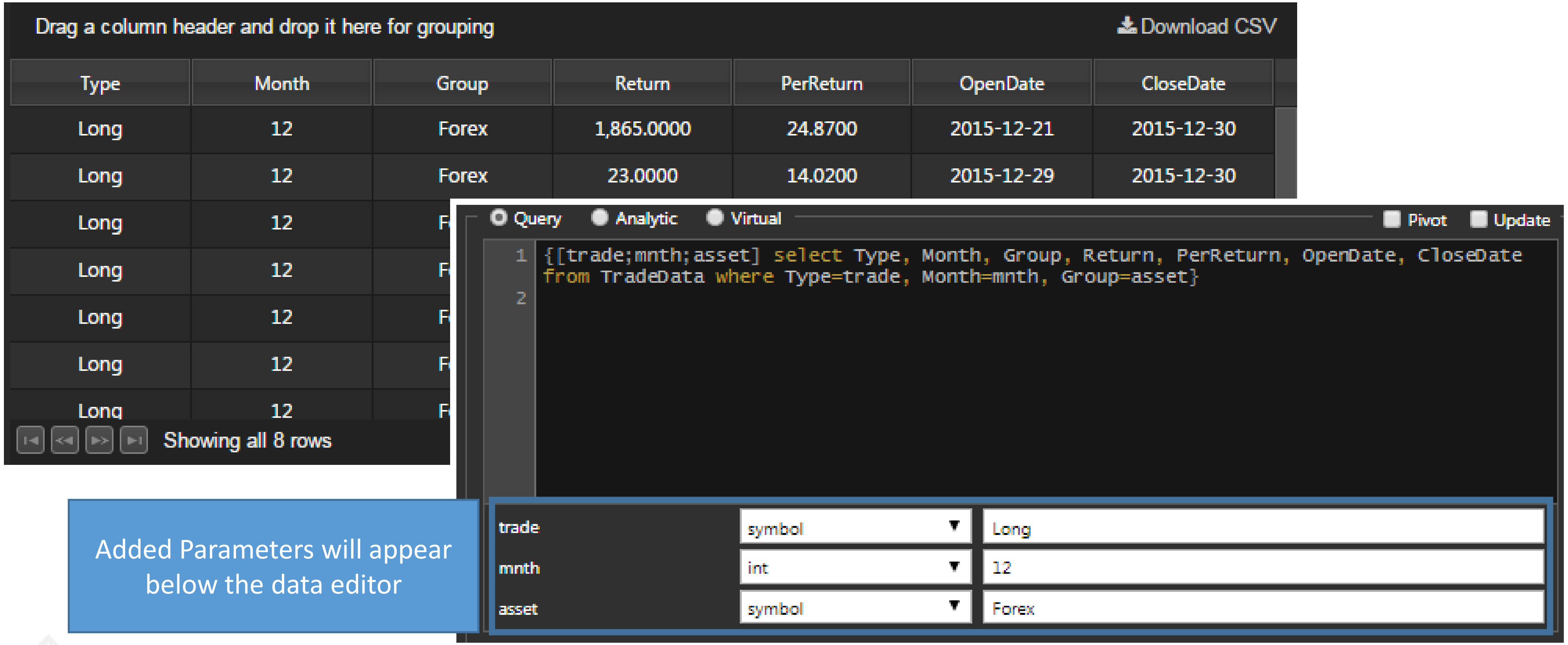
## Dashboards for Kx – "How to" Guide

- Create a new query: **DataGrid2**
- Connect to html5eval_grp (or html5evalcongroup)
- Add the following query

**{[trade;mnth;asset] select Type, Month, Group, Return, PerReturn, OpenDate, CloseDate from TradeData where Type=trade, Month=mnth, Group=asset}**

- Use pre-set values for added query Parameters:
  - ***trade*** is Type symbol and Value = Long
  - ***mnth*** is Type int and Value = 12
  - ***asset*** is Type symbol and Value = Forex
- Execute -> Apply -> Select

# **DataGrid2** : Data Editor View



Drag a column header and drop it here for grouping

| Type | Month | Group | Return | PerReturn | OpenDate | CloseDate |
|---|---|---|---|---|---|---|
| Long | 12 | Forex | 1,865.0000 | 24.8700 | 2015-12-21 | 2015-12-30 |
| Long | 12 | Forex | 23.0000 | 14.0200 | 2015-12-29 | 2015-12-30 |
| Long | 12 | F | | | | |
| Long | 12 | F | | | | |
| Long | 12 | F | | | | |
| Long | 12 | F | | | | |
| Long | 12 | F | | | | |

Showing all 8 rows

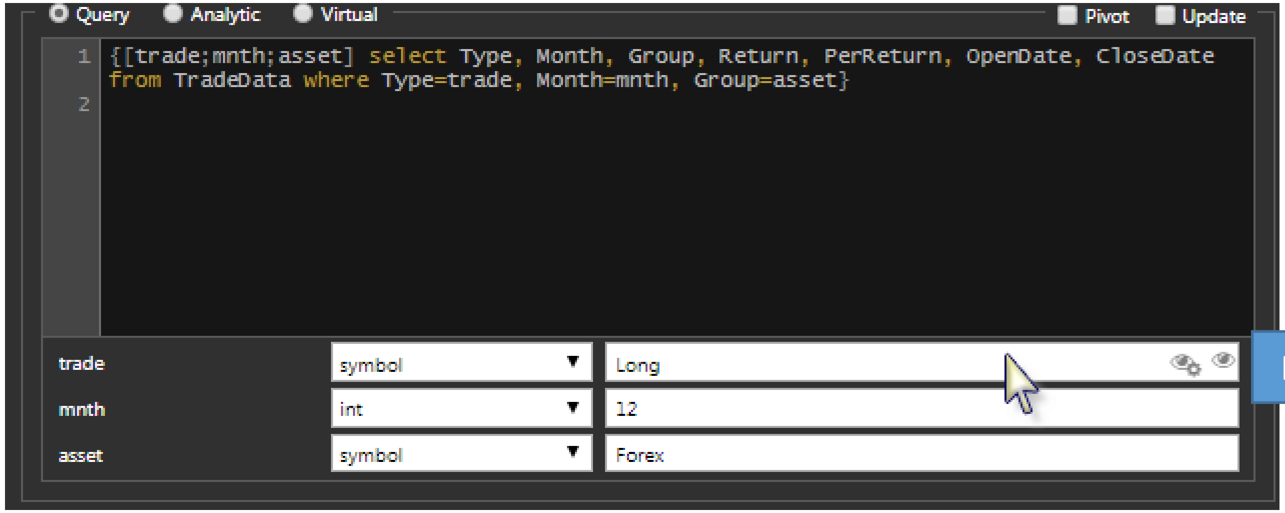
○ Query ○ Analytic ○ Virtual □ Pivot □ Update

```
{[trade;mnth;asset] select Type, Month, Group, Return, PerReturn, OpenDate, CloseDate
from TradeData where Type=trade, Month=mnth, Group=asset}
```

| | | |
|---|---|---|
| trade | symbol | Long |
| mnth | int | 12 |
| asset | symbol | Forex |

Added Parameters will appear below the data editor

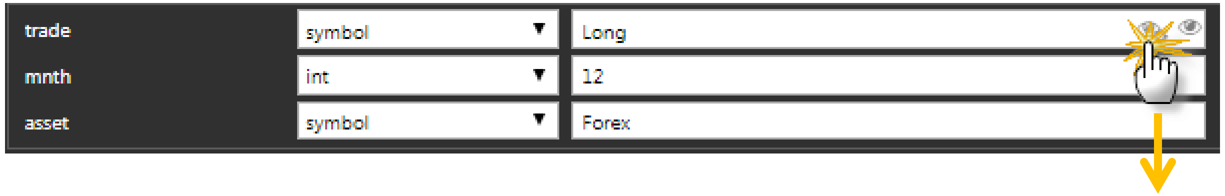
# Map Query Parameters to Dashboard View State Parameters



```
{[trade;mnth;asset] select Type, Month, Group, Return, PerReturn, OpenDate, CloseDate
from TradeData where Type=trade, Month=mnth, Group=asset}
```

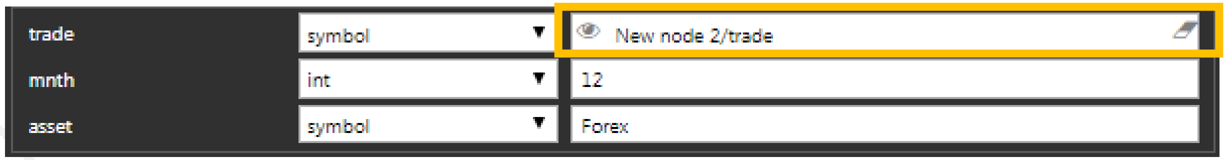
| | | |
|---|---|---|
| trade | symbol ▼ | Long |
| mnth | int ▼ | 12 |
| asset | symbol ▼ | Forex |

Rollover to view Icons



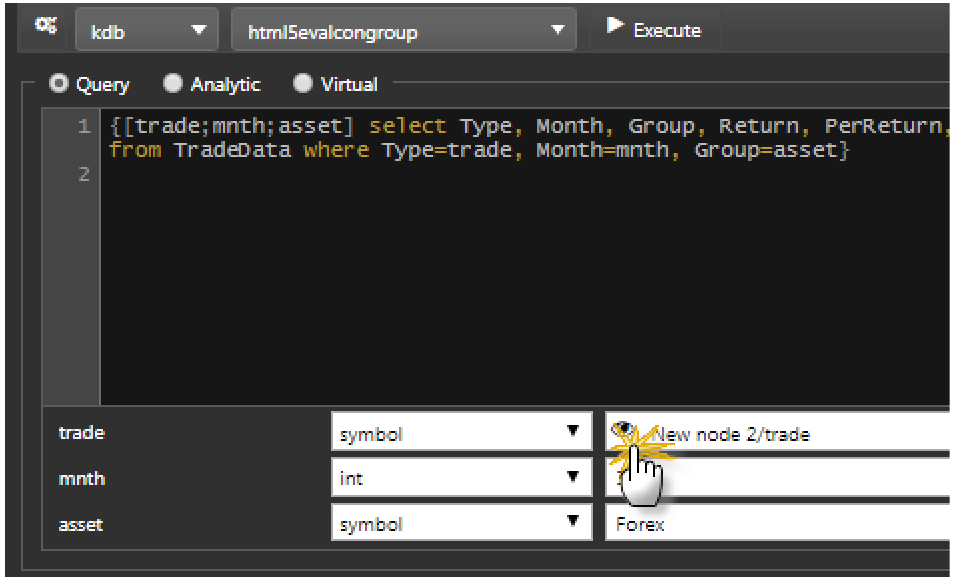
| | | |
|---|---|---|
| trade | symbol ▼ | Long |
| mnth | int ▼ | 12 |
| asset | symbol ▼ | Forex |

Click the Eye icon+ to map the query parameter to a dashboard View State Parameter



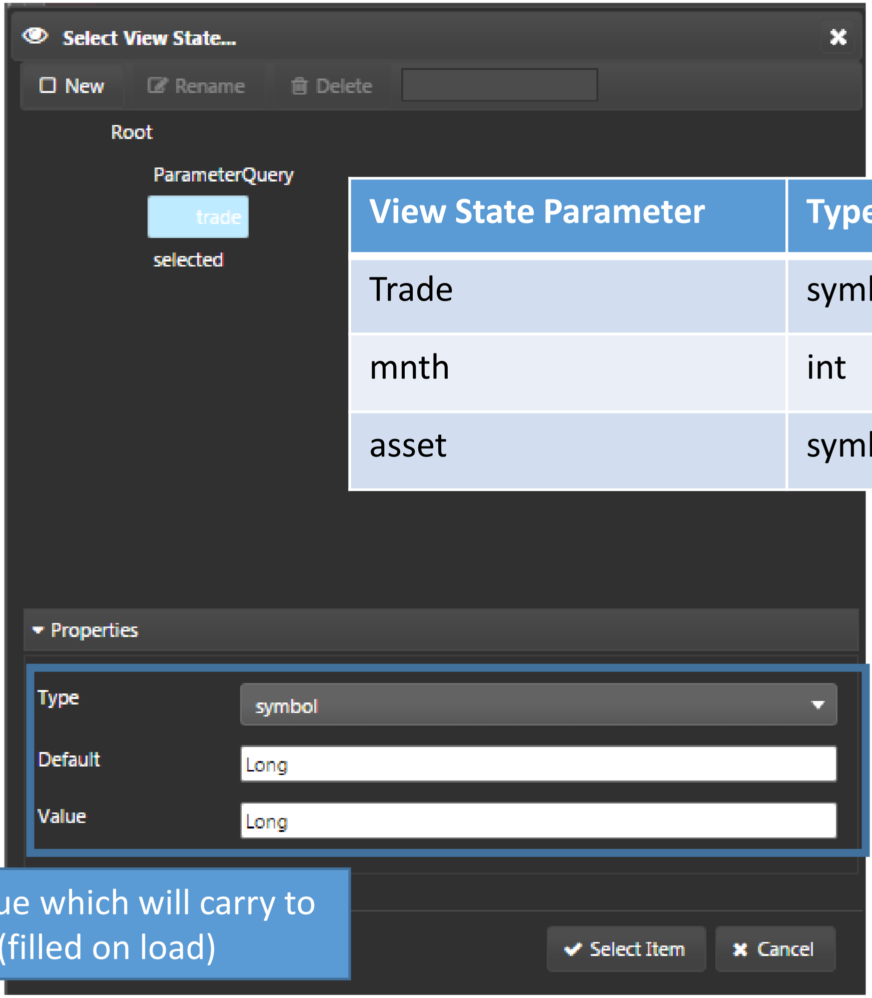
| | | |
|---|---|---|
| trade | symbol ▼ | New node 2/trade |
| mnth | int ▼ | 12 |
| asset | symbol ▼ | Forex |

# Set Dashboard View State Parameter Defaults



```
{[trade;mnth;asset] select Type, Month, Group, Return, PerReturn,
from TradeData where Type=trade, Month=mnth, Group=asset}
```

| | | |
|---|---|---|
| trade | symbol | New node 2/trade |
| mnth | int | |
| asset | symbol | Forex |



| | |
|---|---|
| Type | symbol |
| Default | Long |
| Value | Long |

| View State Parameter | Type | Value |
|---|---|---|
| Trade | symbol | Long |
| mnth | int | 12 |
| asset | symbol | Forex |

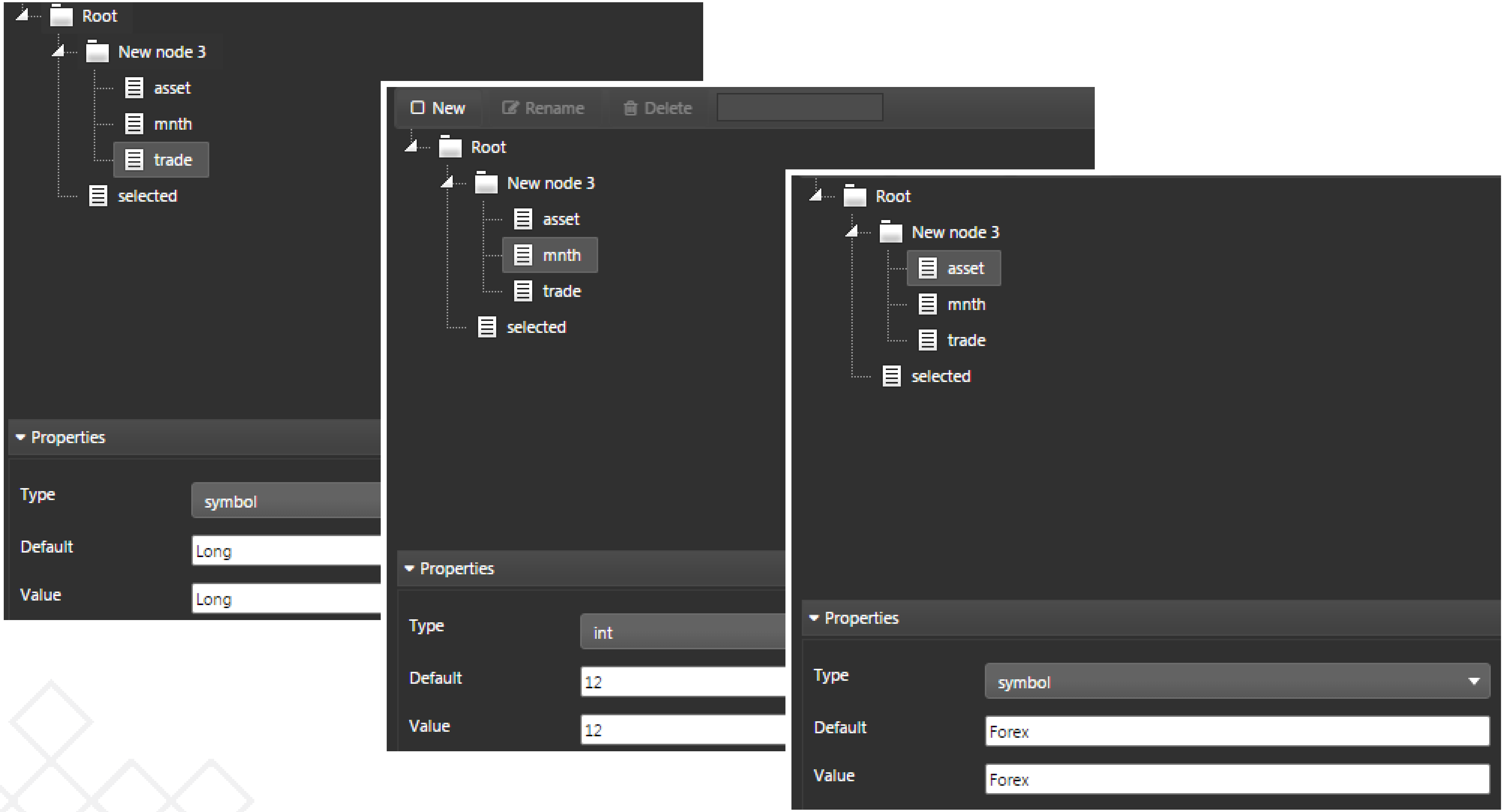
Click the Eye icon open the View State Parameter menu

⚠ Ensure the correct Type is assigned to your data

Set a Default Value which will carry to the Value (filled on load)

Repeat for *mnth* and *asset*

# Assigned View States for **DataGrid2**

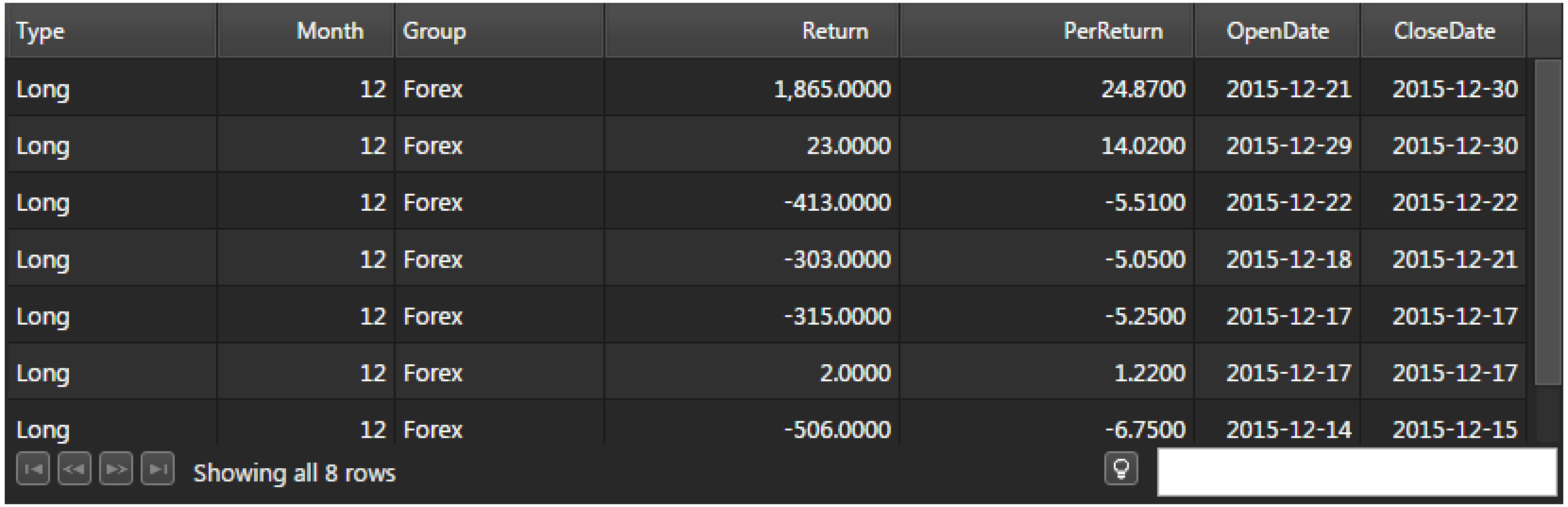


Type: symbol
Default: Long
Value: Long

Type: int
Default: 12
Value: 12

Type: symbol
Default: Forex
Value: Forex

# DataGrid2

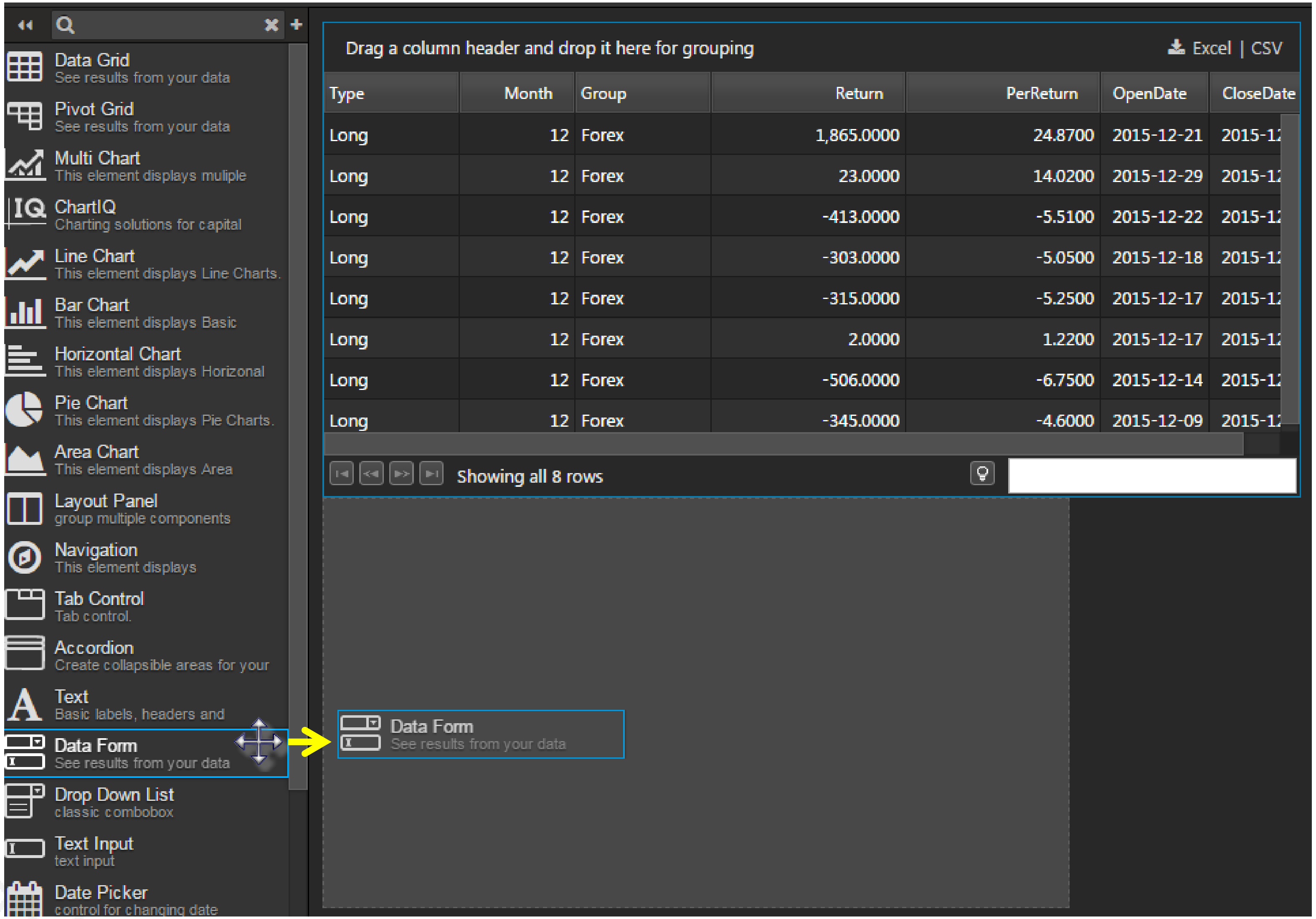


| Type | Month | Group | Return | PerReturn | OpenDate | CloseDate |
|---|---|---|---|---|---|---|
| Long | 12 | Forex | 1,865.0000 | 24.8700 | 2015-12-21 | 2015-12-30 |
| Long | 12 | Forex | 23.0000 | 14.0200 | 2015-12-29 | 2015-12-30 |
| Long | 12 | Forex | -413.0000 | -5.5100 | 2015-12-22 | 2015-12-22 |
| Long | 12 | Forex | -303.0000 | -5.0500 | 2015-12-18 | 2015-12-21 |
| Long | 12 | Forex | -315.0000 | -5.2500 | 2015-12-17 | 2015-12-17 |
| Long | 12 | Forex | 2.0000 | 1.2200 | 2015-12-17 | 2015-12-17 |
| Long | 12 | Forex | -506.0000 | -6.7500 | 2015-12-14 | 2015-12-15 |

Showing all 8 rows

Use Column Formats to improve the visual look of the Data Grid

# Add New Components: **Data Form**



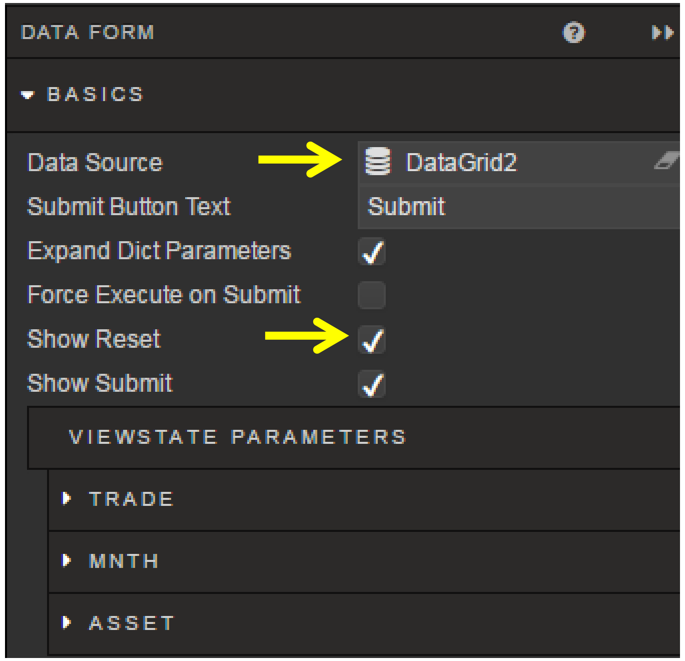
Data Grid
See results from your data
Pivot Grid
See results from your data
Multi Chart
This element displays muliple
ChartIQ
Charting solutions for capital
Line Chart
This element displays Line Charts.
Bar Chart
This element displays Basic
Horizontal Chart
This element displays Horizonal
Pie Chart
This element displays Pie Charts.
Area Chart
This element displays Area
Layout Panel
group multiple components
Navigation
This element displays
Tab Control
Tab control.
Accordion
Create collapsible areas for your
Text
Basic labels, headers and
Data Form
See results from your data
Drop Down List
classic combobox
Text Input
text input
Date Picker
control for changing date
Excel | CSV
Data Form
See results from your data

Drag a column header and drop it here for grouping

| Type | Month | Group | Return | PerReturn | OpenDate | CloseDate |
|---|---|---|---|---|---|---|
| Long | 12 | Forex | 1,865.0000 | 24.8700 | 2015-12-21 | 2015-1 |
| Long | 12 | Forex | 23.0000 | 14.0200 | 2015-12-29 | 2015-1 |
| Long | 12 | Forex | -413.0000 | -5.5100 | 2015-12-22 | 2015-1 |
| Long | 12 | Forex | -303.0000 | -5.0500 | 2015-12-18 | 2015-1 |
| Long | 12 | Forex | -315.0000 | -5.2500 | 2015-12-17 | 2015-1 |
| Long | 12 | Forex | 2.0000 | 1.2200 | 2015-12-17 | 2015-1 |
| Long | 12 | Forex | -506.0000 | -6.7500 | 2015-12-14 | 2015-1 |
| Long | 12 | Forex | -345.0000 | -4.6000 | 2015-12-09 | 2015-1 |

Showing all 8 rows

# Configure: **Data Form**



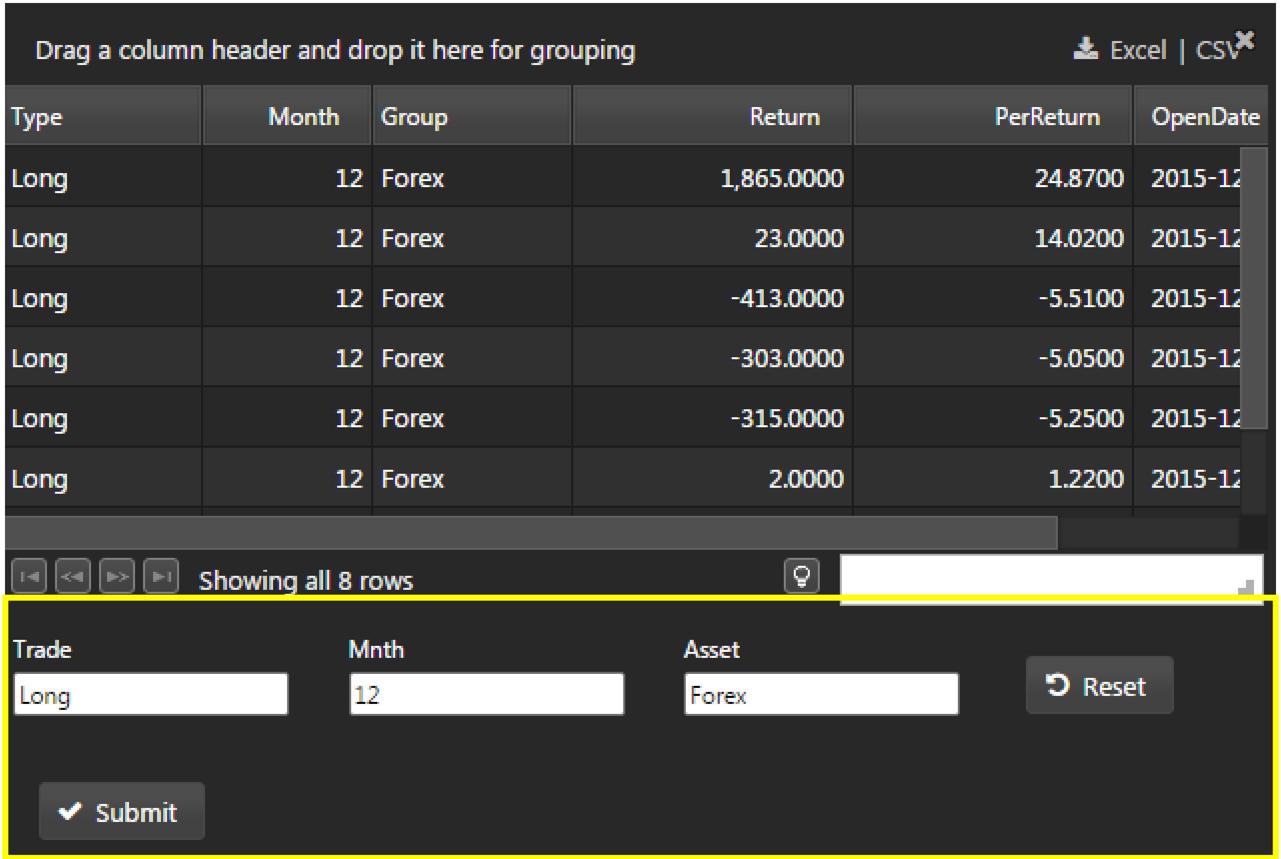
DATA FORM

BASICS

- Data Source: DataGrid2
- Submit Button Text: Submit
- Expand Dict Parameters: ✓
- Force Execute on Submit: □
- Show Reset: ✓
- Show Submit: ✓

VIEWSTATE PARAMETERS

- TRADE
- MNTH
- ASSET



Drag a column header and drop it here for grouping

| Type | Month | Group | Return | PerReturn | OpenDate |
|---|---|---|---|---|---|
| Long | 12 | Forex | 1,865.0000 | 24.8700 | 2015-12 |
| Long | 12 | Forex | 23.0000 | 14.0200 | 2015-12 |
| Long | 12 | Forex | -413.0000 | -5.5100 | 2015-12 |
| Long | 12 | Forex | -303.0000 | -5.0500 | 2015-12 |
| Long | 12 | Forex | -315.0000 | -5.2500 | 2015-12 |
| Long | 12 | Forex | 2.0000 | 1.2200 | 2015-12 |

Showing all 8 rows

| Trade | Mnth | Asset |
|---|---|---|
| Long | 12 | Forex |

DataGrid 2 applied to *Data Source* of ***Data Form***

***Data Form*** will use default values of *View State Parameters* to populate ***Data Form*** input boxes.

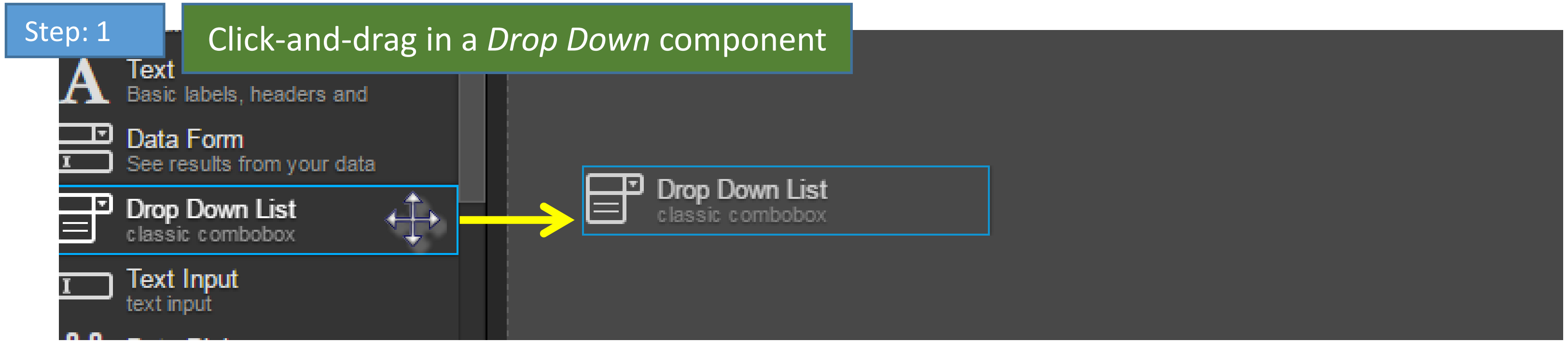
# Test Data Form Inputs

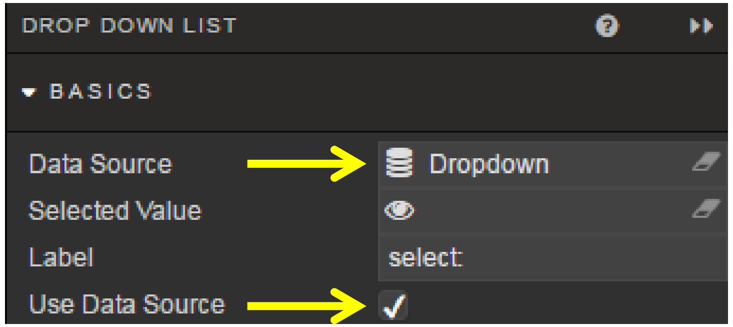
| Asset | Mnth | Trade |
|---|---|---|
| Forex | 12 | Long |
| Commodity | 11 | Short |
| Equity | 10 | |
| | 9 | |
| | 8 | |
| | 7 | |
| | 6 | |
| | 5 | |
| | 4 | |
| | 3 | |
| | 2 | |

# Use **Dropdown** component with a *Data Source*

Step: 1

Click-and-drag in a *Drop Down* component





DROP DOWN LIST

BASICS

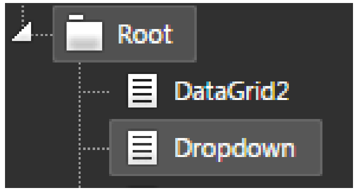
| | |
|---|---|
| Data Source | Dropdown |
| Selected Value | |
| Label | select |
| Use Data Source | ✓ |

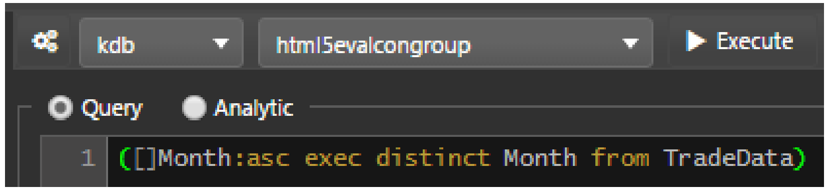
Step: 2

Create a Data Source which parses the individual categories – in this case, Month

Cut-and-paste into **Dropdown** *Data Source* Editor (html5evalcongroup)

**([]Month: asc exec distinct Month from TradeData)**

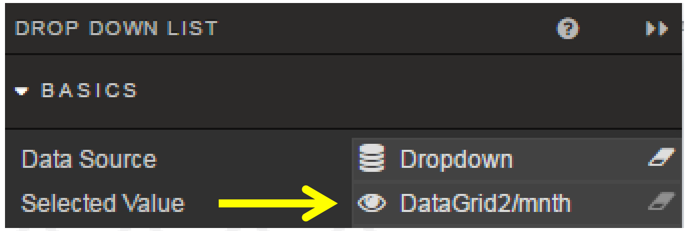




```
([]Month:asc exec distinct Month from TradeData)
```

Step: 3

Map the *Selected Value* of the dropdown to the *View State Parameter* of ***Mnth*** used by **DataGrid2**



DROP DOWN LIST

BASICS

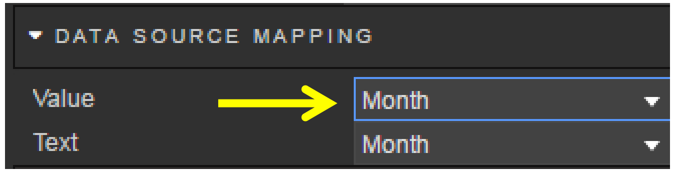
| | |
|---|---|
| Data Source | Dropdown |
| Selected Value | DataGrid2/mnth |

Continued…

# Use **Dropdown** component with a *Data Source*

Step: 4

Set the *Data Source Mapping*



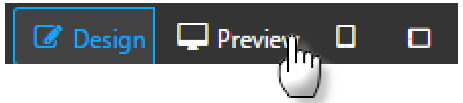
DATA SOURCE MAPPING

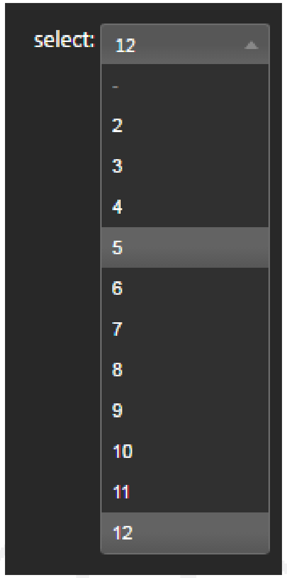
| | |
|---|---|
| Value | Month |
| Text | Month |

*Value* and *Text* can be separate columns if defined in *Data Source*

Step: 5

Preview

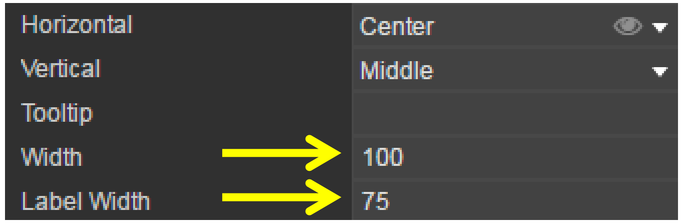




select: 12

-
2
3
4
5
6
7
8
9
10
11
12

Change label size, dropdown width and description



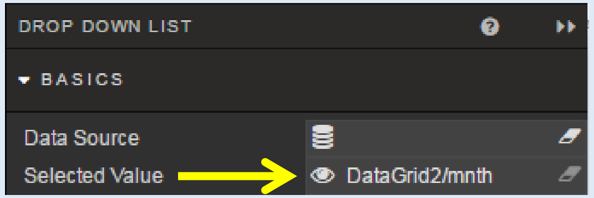
| | |
|---|---|
| Horizontal | Center |
| Vertical | Middle |
| Tooltip | |
| Width | 100 |
| Label Width | 75 |

Use *Selected Row Viewstate Routing* to assign selected value to other view state parameters

## Alternative

Step: 1

Set *Selected Value* to View State Parameter to **DataGrid2** ***Mnth***



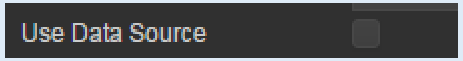
DROP DOWN LIST

BASICS

| | |
|---|---|
| Data Source | |
| Selected Value | DataGrid2/mnth |

Step: 2

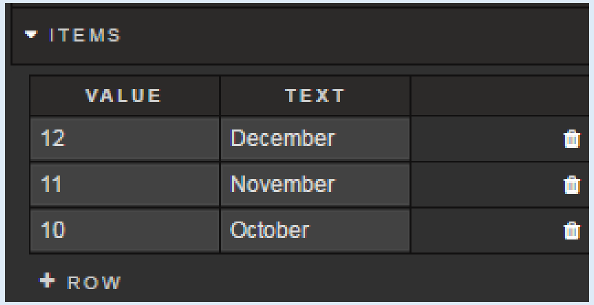
Uncheck *Use Data Source*



Use Data Source □

Step: 3

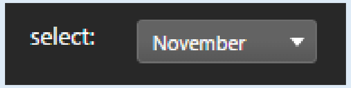
Define *Items: Values* and *Text*



ITEMS

| VALUE | TEXT |
|---|---|
| 12 | December |
| 11 | November |
| 10 | October |

+ ROW

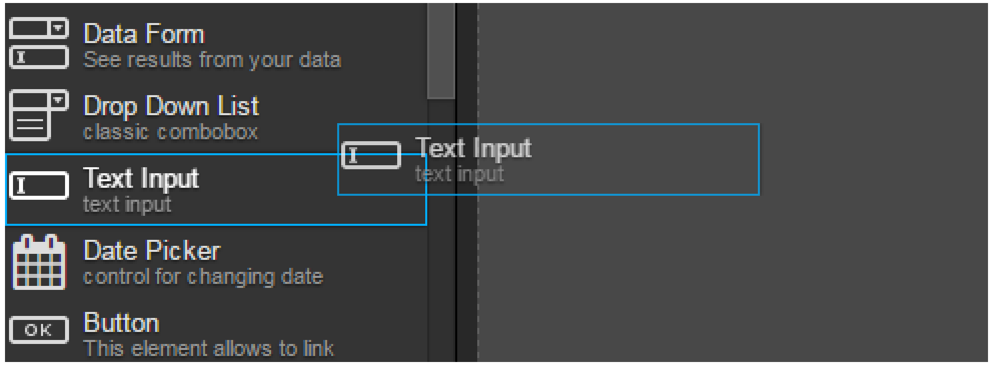


select: November

# Use **Text Input** component with a *View State Parameter*

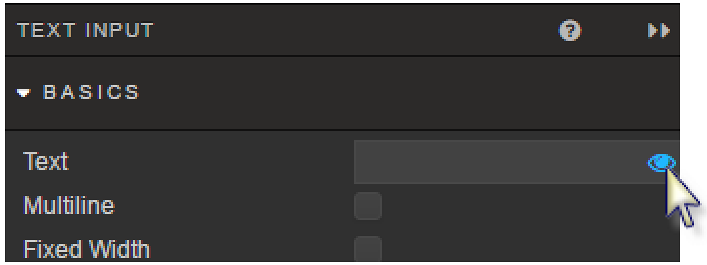
Step: 1

Click-and-drag in a *Text Input* component

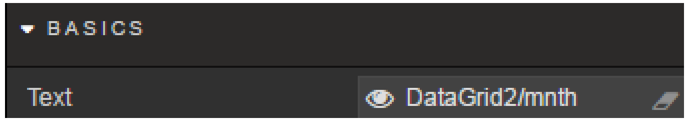


Step: 2

Map the *Selected Value* to the *View State Parameter, **Mnth*** used by **DataGrid2**



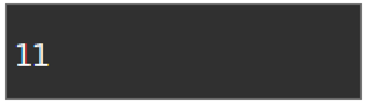
Left-click



Text: DataGrid2/mnth

Step: 3

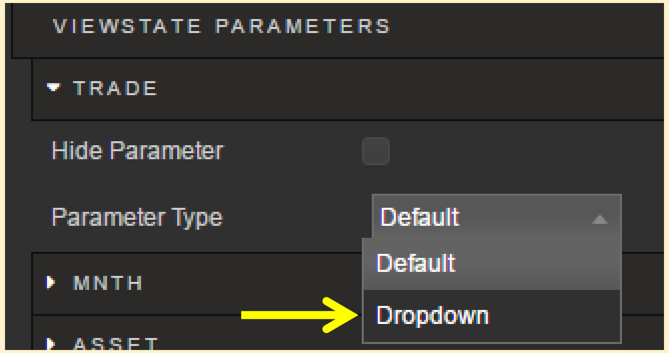
Preview



User has freedom to input any value; out of range values will return a blank Data Grid

# Try This for **DataGrid2**

- Using **DataGrid2** configure a Dataform to use a dropdown for each *View State* of *mnth, asset* and *trade*

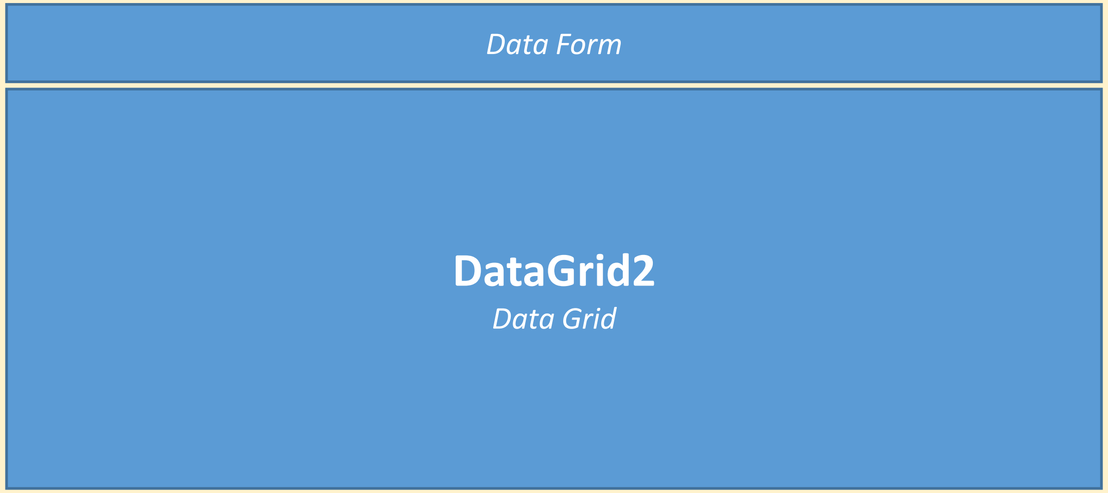


VIEWSTATE PARAMETERS

TRADE

Hide Parameter □

Parameter Type: Default



it's about time

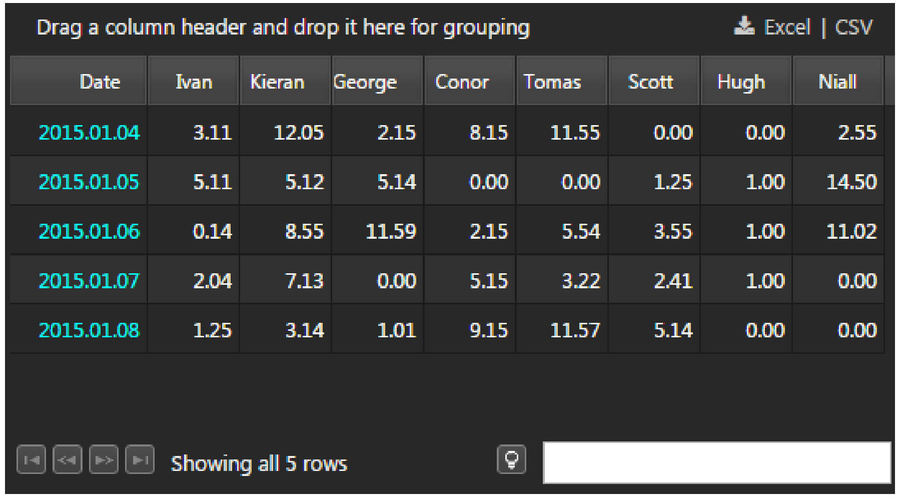
# Row Selection

## Dashboards for Kx – "How to" Guide

# Row Selection: Data Grid populating View State Parameters

Step: 1

Open **GroupName** Data Grid



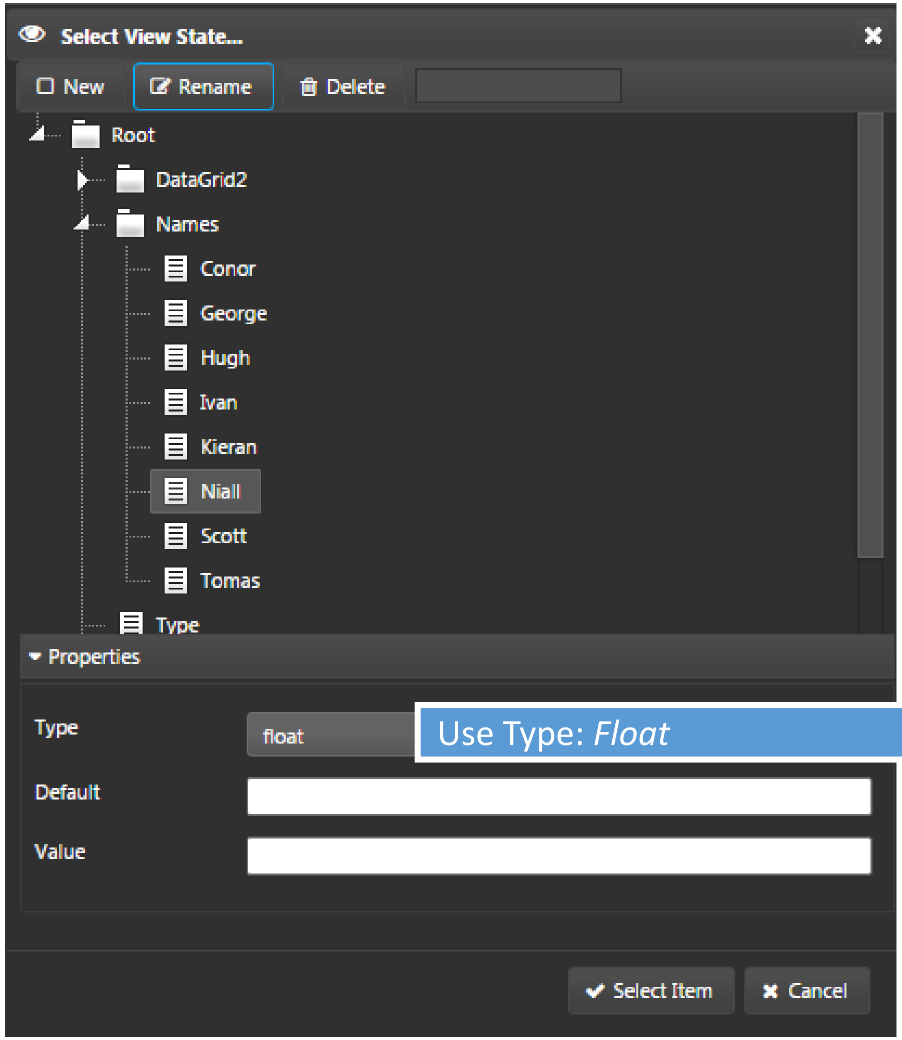
Drag a column header and drop it here for grouping

| Date | Ivan | Kieran | George | Conor | Tomas | Scott | Hugh | Niall |
|---|---|---|---|---|---|---|---|---|
| 2015.01.04 | 3.11 | 12.05 | 2.15 | 8.15 | 11.55 | 0.00 | 0.00 | 2.55 |
| 2015.01.05 | 5.11 | 5.12 | 5.14 | 0.00 | 0.00 | 1.25 | 1.00 | 14.50 |
| 2015.01.06 | 0.14 | 8.55 | 11.59 | 2.15 | 5.54 | 3.55 | 1.00 | 11.02 |
| 2015.01.07 | 2.04 | 7.13 | 0.00 | 5.15 | 3.22 | 2.41 | 1.00 | 0.00 |
| 2015.01.08 | 1.25 | 3.14 | 1.01 | 9.15 | 11.57 | 5.14 | 0.00 | 0.00 |

Showing all 5 rows

Step: 2

Create *View State Parameters* for each user



Select View State...

- Root
  - DataGrid2
  - Names
    - Conor
    - George
    - Hugh
    - Ivan
    - Kieran
    - Niall
    - Scott
    - Tomas
  - Type

Properties

Type: float

Default:

Value:

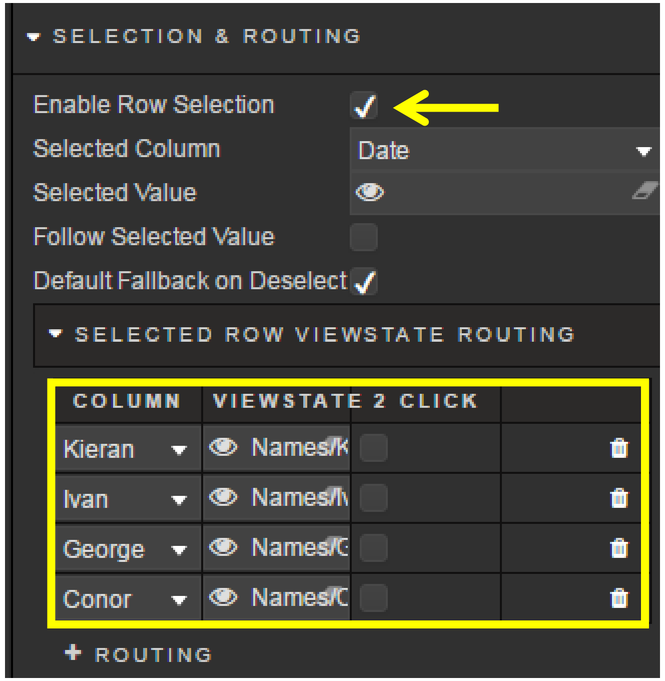
Use Type: *Float*

Continued...

# Row Selection: Data Grid populating View State Parameters, presented in a **Text** component

Step: 3

*Enable Row Selection* in **GroupName** Data Grid



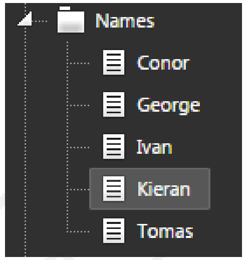
SELECTION & ROUTING

| | |
|---|---|
| Enable Row Selection | ✓ |
| Selected Column | Date |
| Selected Value | 👁 |
| Follow Selected Value | □ |
| Default Fallback on Deselect | ✓ |

SELECTED ROW VIEWSTATE ROUTING

| COLUMN | VIEWSTATE | 2 CLICK |
|---|---|---|
| Kieran | 👁 Names/K | □ |
| Ivan | 👁 Names/Iv | □ |
| George | 👁 Names/G | □ |
| Conor | 👁 Names/C | □ |

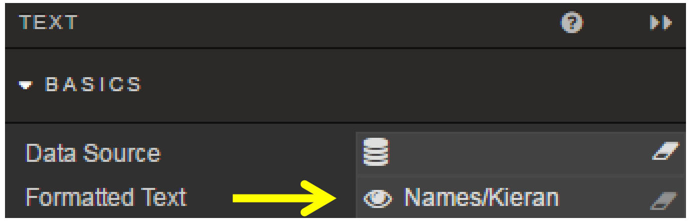
+ ROUTING



In *Selected Row Viewstate Routing* map *Data Source Column* to named *Viewstate*

Step: 4

Add a *Text* component and Link to a named *Viewstate* Text Basic labels, headers and



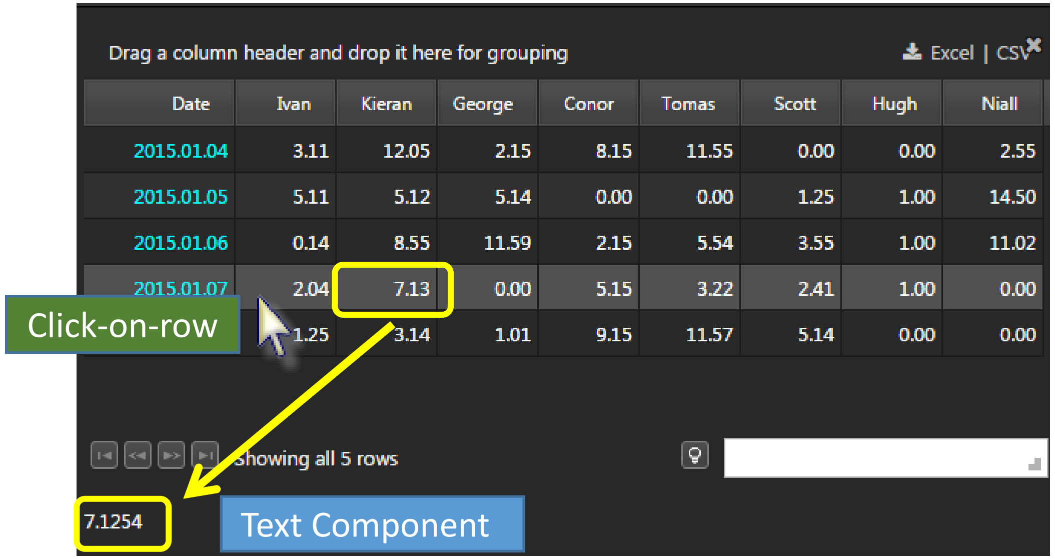
TEXT

BASICS

| | |
|---|---|
| Data Source | |
| Formatted Text | 👁 Names/Kieran |

Step: 5

Preview: Select Row Preview



Drag a column header and drop it here for grouping

| Date | Ivan | Kieran | George | Conor | Tomas | Scott | Hugh | Niall |
|---|---|---|---|---|---|---|---|---|
| 2015.01.04 | 3.11 | 12.05 | 2.15 | 8.15 | 11.55 | 0.00 | 0.00 | 2.55 |
| 2015.01.05 | 5.11 | 5.12 | 5.14 | 0.00 | 0.00 | 1.25 | 1.00 | 14.50 |
| 2015.01.06 | 0.14 | 8.55 | 11.59 | 2.15 | 5.54 | 3.55 | 1.00 | 11.02 |
| 2015.01.07 | 2.04 | 7.13 | 0.00 | 5.15 | 3.22 | 2.41 | 1.00 | 0.00 |
| | 1.25 | 3.14 | 1.01 | 9.15 | 11.57 | 5.14 | 0.00 | 0.00 |

Showing all 5 rows

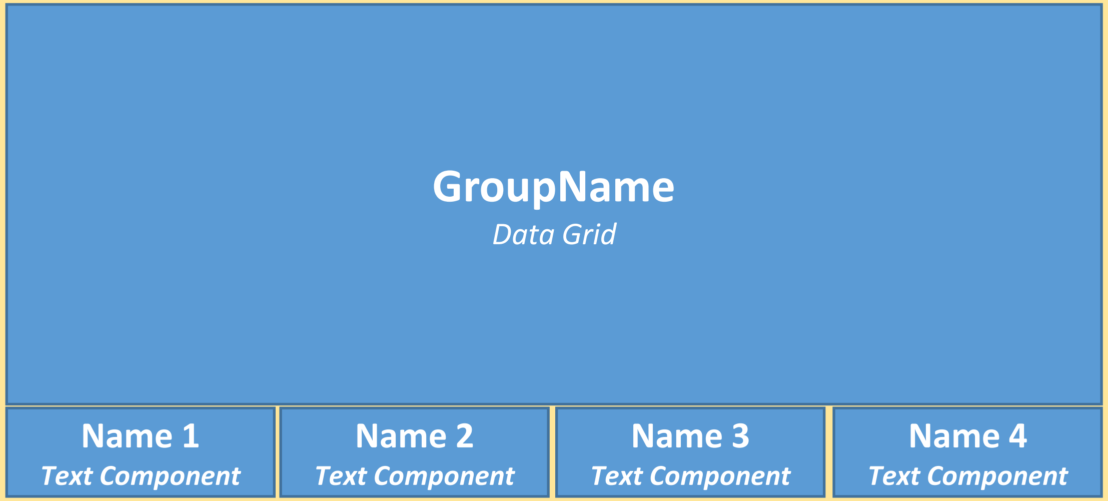
7.1254

Click-on-row

Text Component

# Try This for **GroupName**

- Create a view state routing for each named individual
- Associate each view state with a text output, so user can view values for each user when a row is selected.



**GroupName**
*Data Grid*

**Name 1**
***Text Component***

**Name 2**
***Text Component***

**Name 3**
***Text Component***

**Name 4**
***Text Component***

it's about time

# Date Picker

## Dashboards for Kx – "How to" Guide

# Date Picker: New Queries **SourceDate** and **DateRange**

Support for date picker requires a little modification to the query to support a date range.  Create two new data sources:

- Create Data Source: **SourceDate**
- Connection: html5eval_grp (or html5evalcongroup)

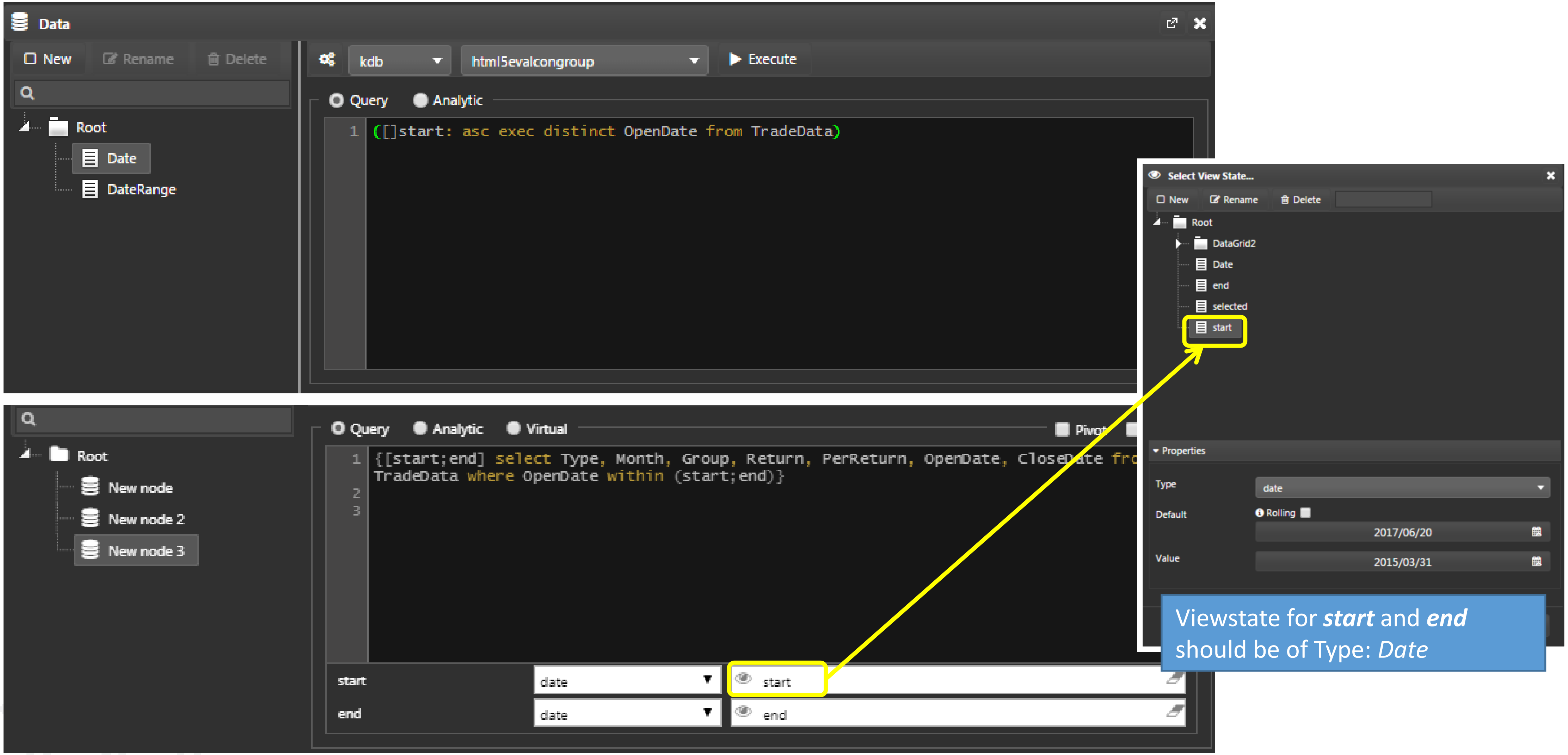
```
([]start: asc exec distinct OpenDate from TradeData)
```

- Create Data Source: **DateRange**
- Connection: html5eval_grp (or html5evalcongroup)

```
{[start;end] select Type, Month, Group, Return, PerReturn, OpenDate, CloseDate from
TradeData where OpenDate within (start;end)}
```

- Map query parameters, ***start*** and ***end*** to dashboard view states: *start* and *end*

# Queries **SourceDate** and **DateRange**



```
([]start: asc exec distinct OpenDate from TradeData)
```

```
{[start;end] select Type, Month, Group, Return, PerReturn, OpenDate, CloseDate fro
TradeData where OpenDate within (start;end)}
```

| | | |
|---|---|---|
| start | date | start |
| end | date | end |

Type: date

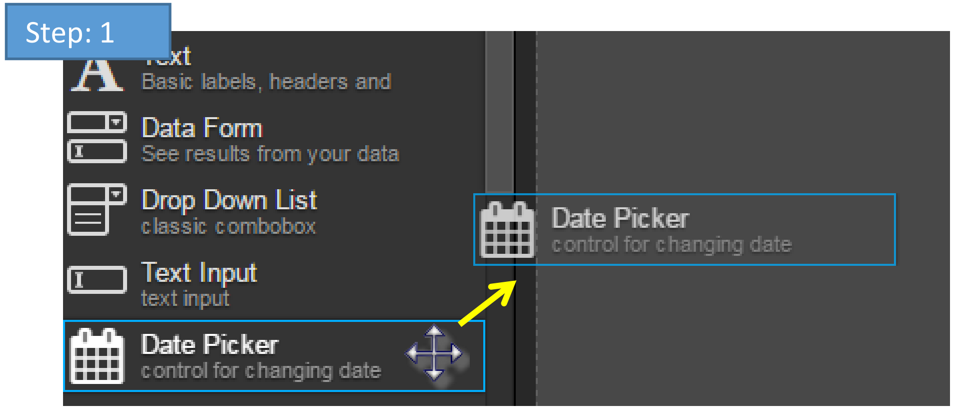
Default: Rolling □ 2017/06/20

Value: 2015/03/31

Viewstate for ***start*** and ***end*** should be of Type: *Date*

# Add a Data Grid and Two Date Pickers

Step: 1

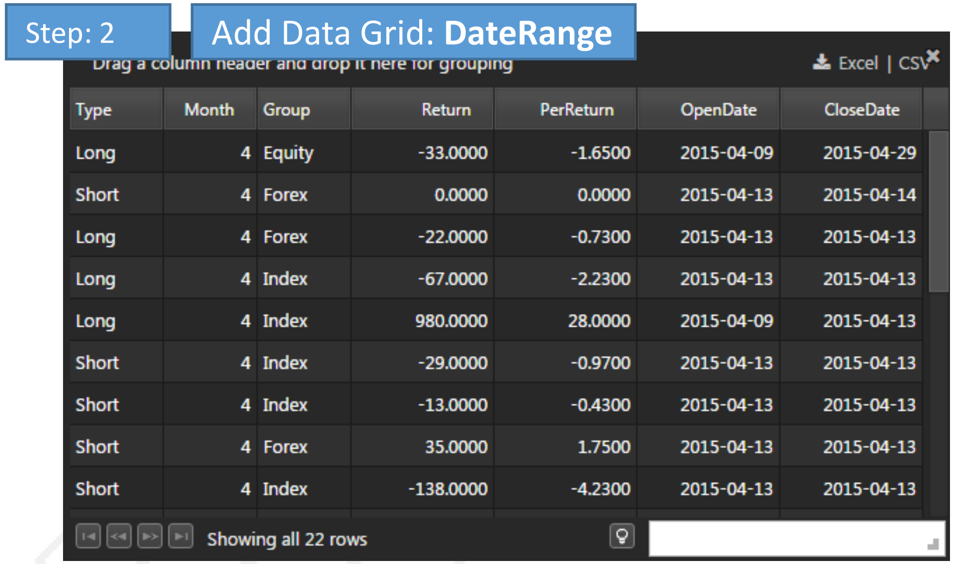


Text
Basic labels, headers and
Data Form
See results from your data
Drop Down List
classic combobox
Text Input
text input
Date Picker
control for changing date
Date Picker
control for changing date

Add Two Date Pickers to the Dashboard

Step: 2

Add Data Grid: **DateRange**



Drag a column header and drop it here for grouping

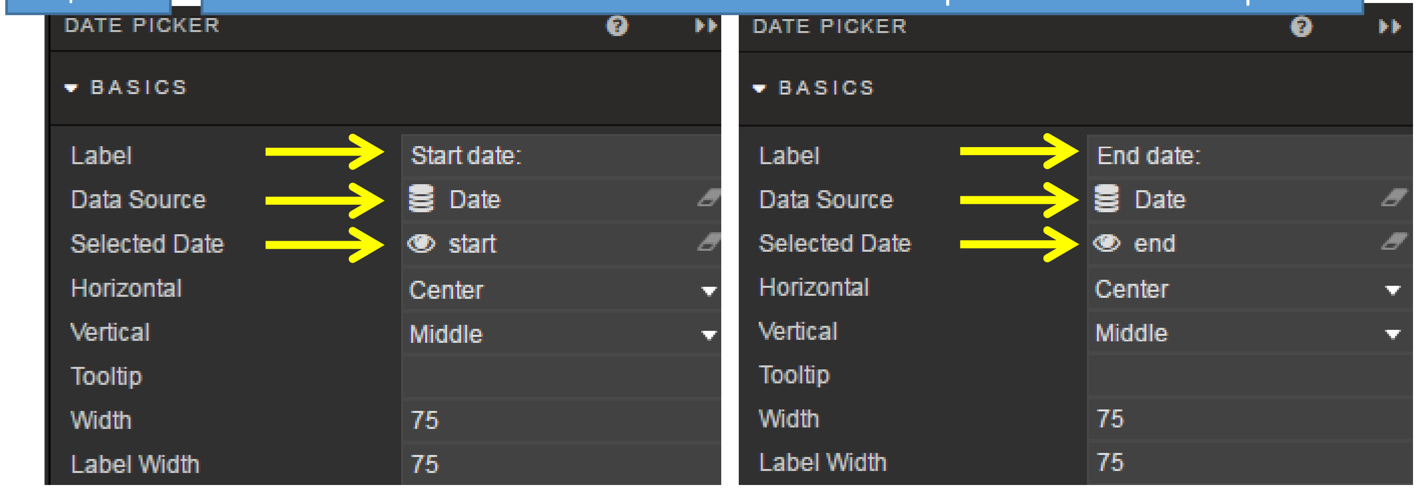
Excel | CSV

| Type | Month | Group | Return | PerReturn | OpenDate | CloseDate |
|---|---|---|---|---|---|---|
| Long | 4 | Equity | -33.0000 | -1.6500 | 2015-04-09 | 2015-04-29 |
| Short | 4 | Forex | 0.0000 | 0.0000 | 2015-04-13 | 2015-04-14 |
| Long | 4 | Forex | -22.0000 | -0.7300 | 2015-04-13 | 2015-04-13 |
| Long | 4 | Index | -67.0000 | -2.2300 | 2015-04-13 | 2015-04-13 |
| Long | 4 | Index | 980.0000 | 28.0000 | 2015-04-09 | 2015-04-13 |
| Short | 4 | Index | -29.0000 | -0.9700 | 2015-04-13 | 2015-04-13 |
| Short | 4 | Index | -13.0000 | -0.4300 | 2015-04-13 | 2015-04-13 |
| Short | 4 | Forex | 35.0000 | 1.7500 | 2015-04-13 | 2015-04-13 |
| Short | 4 | Index | -138.0000 | -4.2300 | 2015-04-13 | 2015-04-13 |

Showing all 22 rows

Step: 3

Connect Data Source and Dashboard Viewstate parameter for each picker



DATE PICKER

BASICS

| Setting | Value |
|---|---|
| Label | Start date: |
| Data Source | Date |
| Selected Date | start |
| Horizontal | Center |
| Vertical | Middle |
| Tooltip | |
| Width | 75 |
| Label Width | 75 |

DATE PICKER

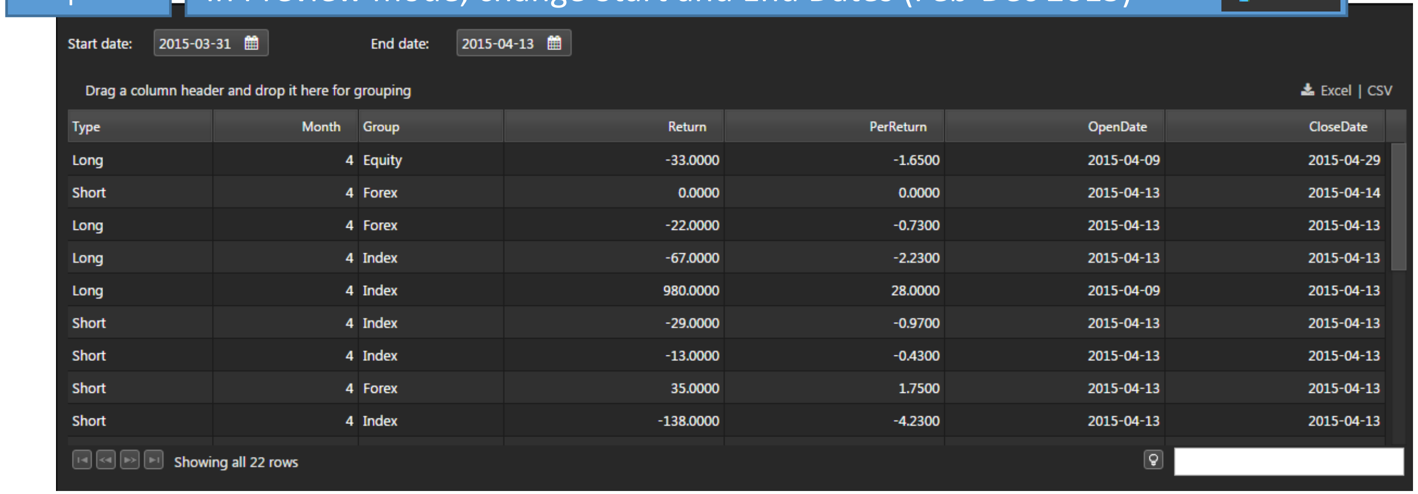
BASICS

| Setting | Value |
|---|---|
| Label | End date: |
| Data Source | Date |
| Selected Date | end |
| Horizontal | Center |
| Vertical | Middle |
| Tooltip | |
| Width | 75 |
| Label Width | 75 |

Step: 4

In Preview mode, change Start and End Dates (Feb-Dec 2015)

Preview



Start date: 2015-03-31 End date: 2015-04-13

Drag a column header and drop it here for grouping

Excel | CSV

| Type | Month | Group | Return | PerReturn | OpenDate | CloseDate |
|---|---|---|---|---|---|---|
| Long | 4 | Equity | -33.0000 | -1.6500 | 2015-04-09 | 2015-04-29 |
| Short | 4 | Forex | 0.0000 | 0.0000 | 2015-04-13 | 2015-04-14 |
| Long | 4 | Forex | -22.0000 | -0.7300 | 2015-04-13 | 2015-04-13 |
| Long | 4 | Index | -67.0000 | -2.2300 | 2015-04-13 | 2015-04-13 |
| Long | 4 | Index | 980.0000 | 28.0000 | 2015-04-09 | 2015-04-13 |
| Short | 4 | Index | -29.0000 | -0.9700 | 2015-04-13 | 2015-04-13 |
| Short | 4 | Index | -13.0000 | -0.4300 | 2015-04-13 | 2015-04-13 |
| Short | 4 | Forex | 35.0000 | 1.7500 | 2015-04-13 | 2015-04-13 |
| Short | 4 | Index | -138.0000 | -4.2300 | 2015-04-13 | 2015-04-13 |

Showing all 22 rows

kx | it's about time

# Pivot Grid & Breadcrumbs

## Dashboards for Kx – "How to" Guide

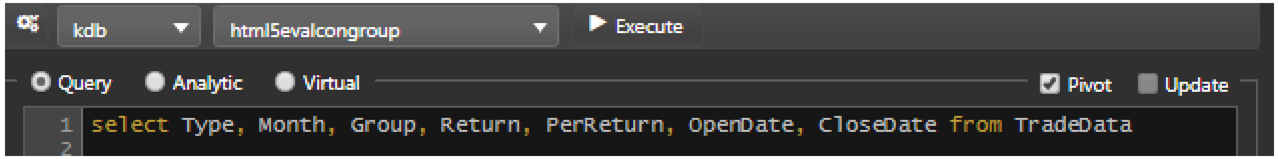
# Pivot Grid: New Query **PivotData**

Pivot Grids create data groupings (independent variables) with summary statistics from dependent variables; e.g. sum, average, count, min and max values. Navigation is controlled using the Breadcrumbs component.

- Create Data Source: **PivotData**
- Connection: html5eval_grp (or html5evalcongroup)

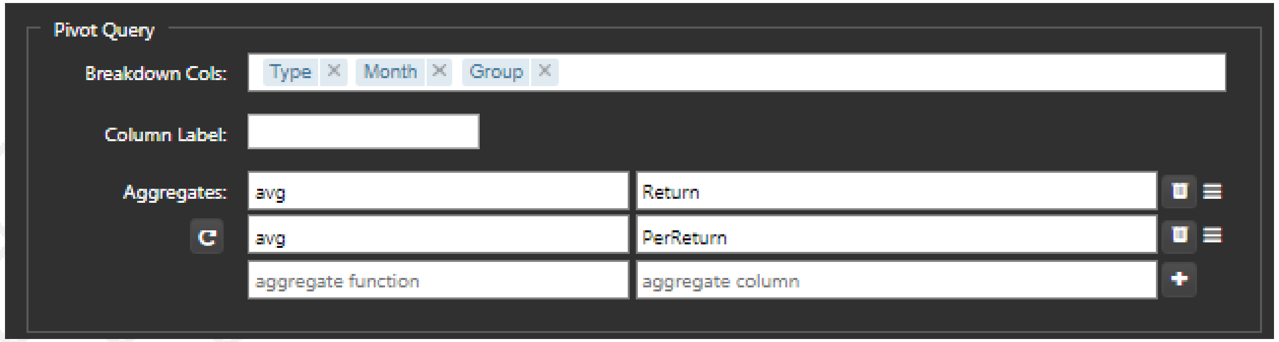
**select Type, Month, Group, Return, PerReturn, OpenDate, CloseDate from TradeData**

- Check Pivot Query



◉ Query ○ Analytic ○ Virtual ☑ Pivot □ Update

```
select Type, Month, Group, Return, PerReturn, OpenDate, CloseDate from TradeData
```



Pivot Query

Breakdown Cols: Type × Month × Group ×

Column Label:

Aggregates: avg | Return

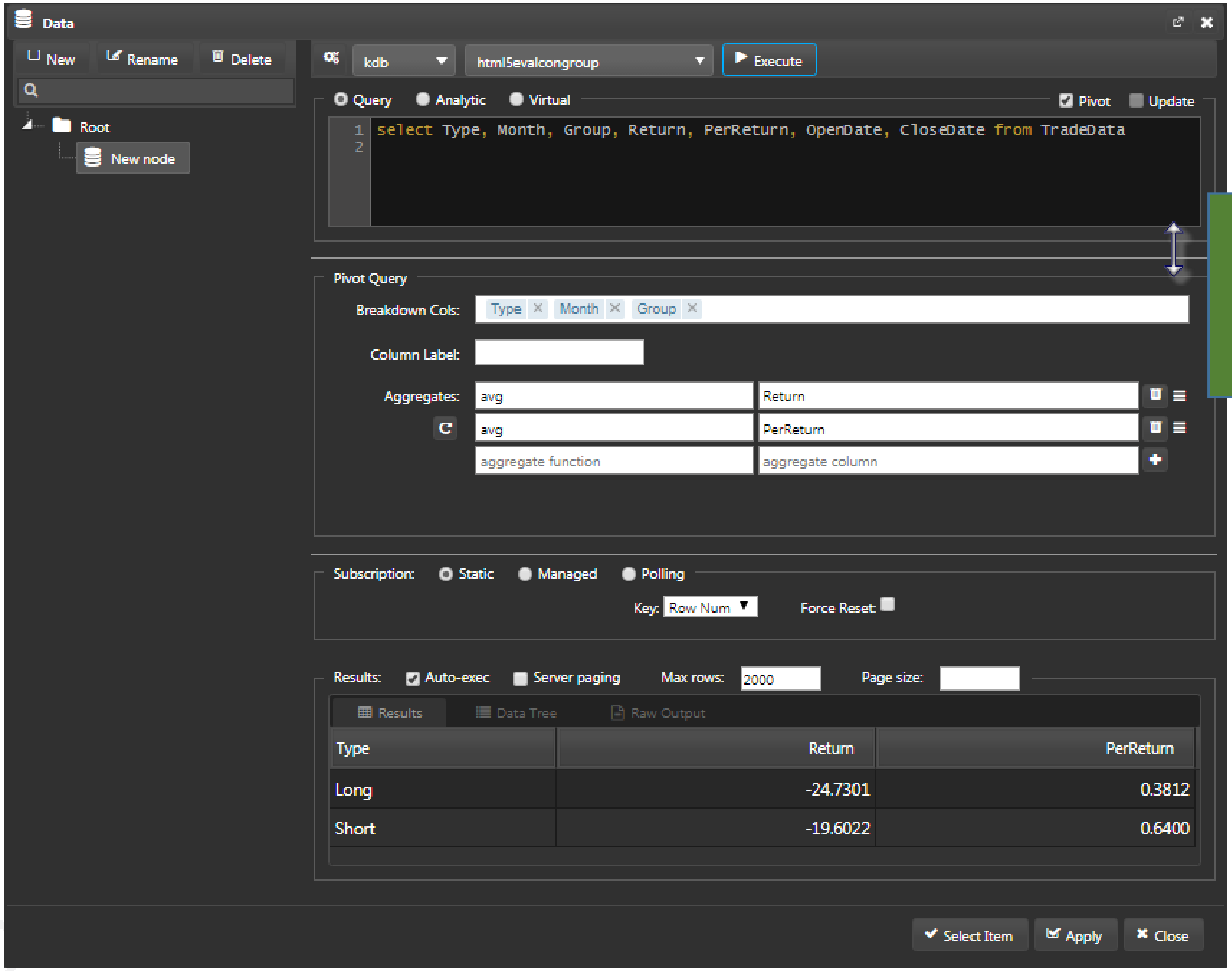
avg | PerReturn

aggregate function | aggregate column

Breakdown Cols: *Type, Month, Group*

Aggregates (avg): *Return, PerReturn*

# Complete Date Editor: **PivotData**



kdb | html5evalcongroup

○ Query ○ Analytic ○ Virtual ☑ Pivot ☐ Update

```
select Type, Month, Group, Return, PerReturn, OpenDate, CloseDate from TradeData
```

Pivot Query

Breakdown Cols: Type ✕ Month ✕ Group ✕

Column Label:

Aggregates:

|  |  |
|---|---|
| avg | Return |
| avg | PerReturn |
| aggregate function | aggregate column |

Subscription: ○ Static ○ Managed ○ Polling

Key: Row Num ▼ Force Reset: ☐

Results: ☑ Auto-exec ☐ Server paging Max rows: 2000 Page size:

Results | Data Tree | Raw Output

| Type | Return | PerReturn |
|---|---|---|
| Long | -24.7301 | 0.3812 |
| Short | -19.6022 | 0.6400 |

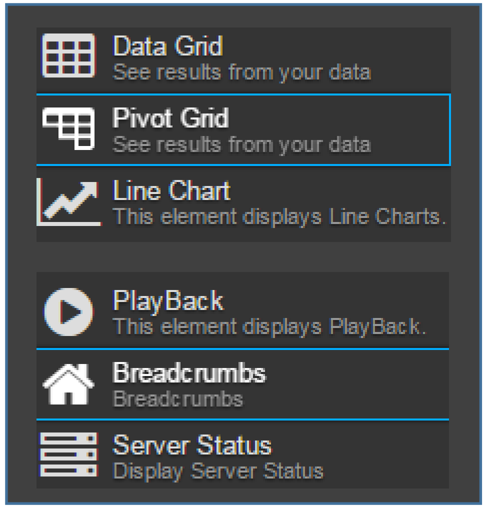
✔ Select Item | Apply | ✖ Close

Left-click-drag to increase viewable area

# Linking Pivot Grid to Breadcrumbs

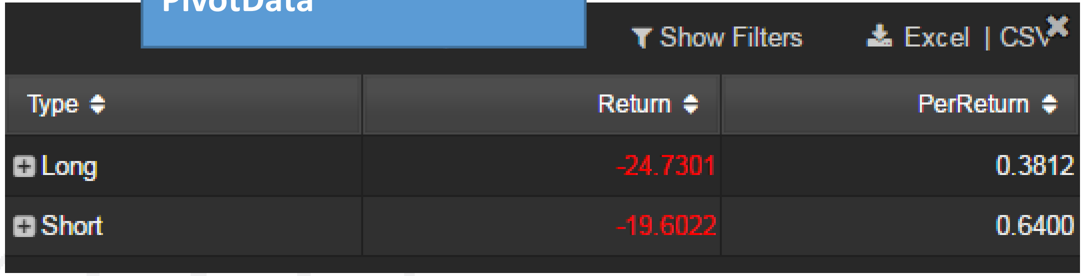
Step: 1

Drag a *Pivot Grid* and *Breadcrumbs* component into dashboard


Data Grid
See results from your data
Pivot Grid
See results from your data
Line Chart
This element displays Line Charts.
PlayBack
This element displays PlayBack.
Breadcrumbs
Breadcrumbs
Server Status
Display Server Status

Step: 2

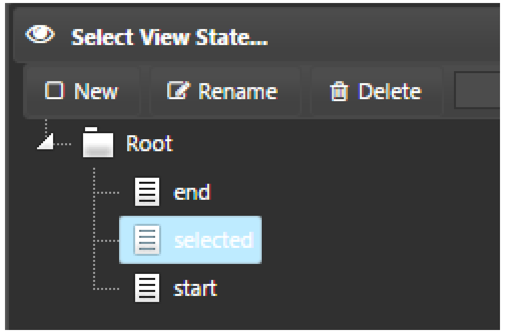
Configure Pivot Grid as **PivotData**

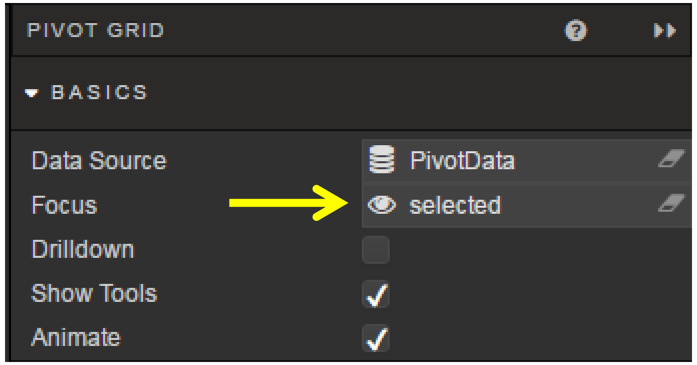


| Type | Return | PerReturn |
|---|---|---|
| ⊞ Long | -24.7301 | 0.3812 |
| ⊞ Short | -19.6022 | 0.6400 |

Step: 3

Create a *Viewstate* called *selected* of type symbol. Map to *Focus* property of Pivot Grid.

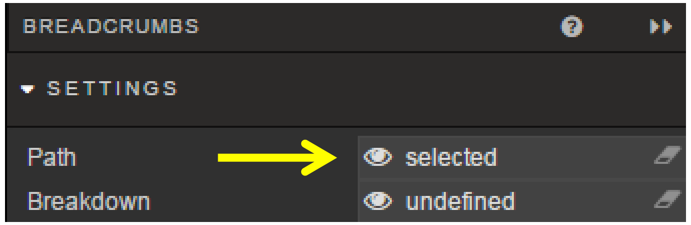




| | |
|---|---|
| Data Source | PivotData |
| Focus | selected |
| Drilldown | □ |
| Show Tools | ✓ |
| Animate | ✓ |

Step: 4

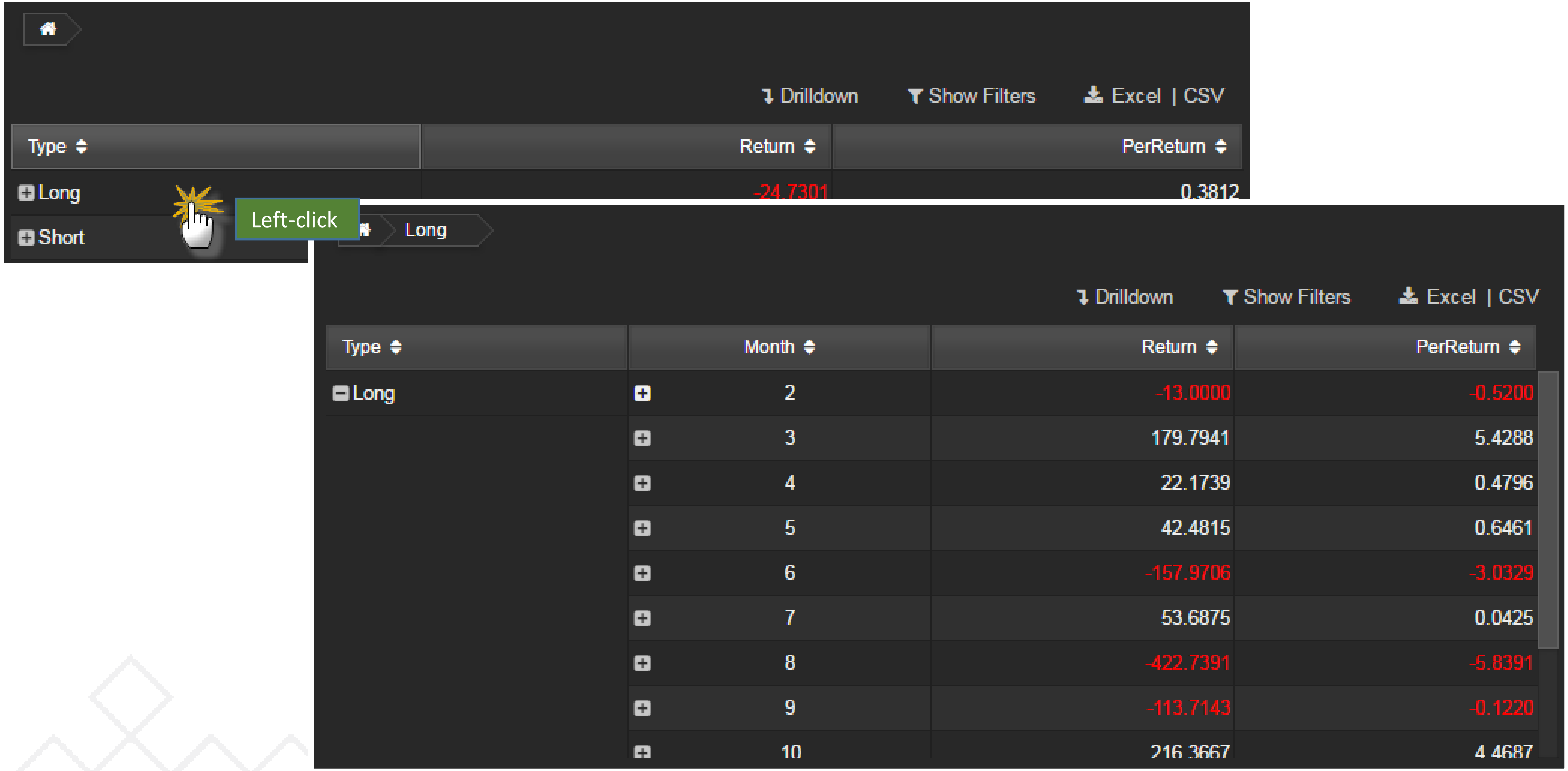
Link *selected* property to *Path* of Breadcrumbs



| | |
|---|---|
| Path | selected |
| Breakdown | undefined |

Continued...

# Preview Pivot Grid and Breadcrumbs Interaction

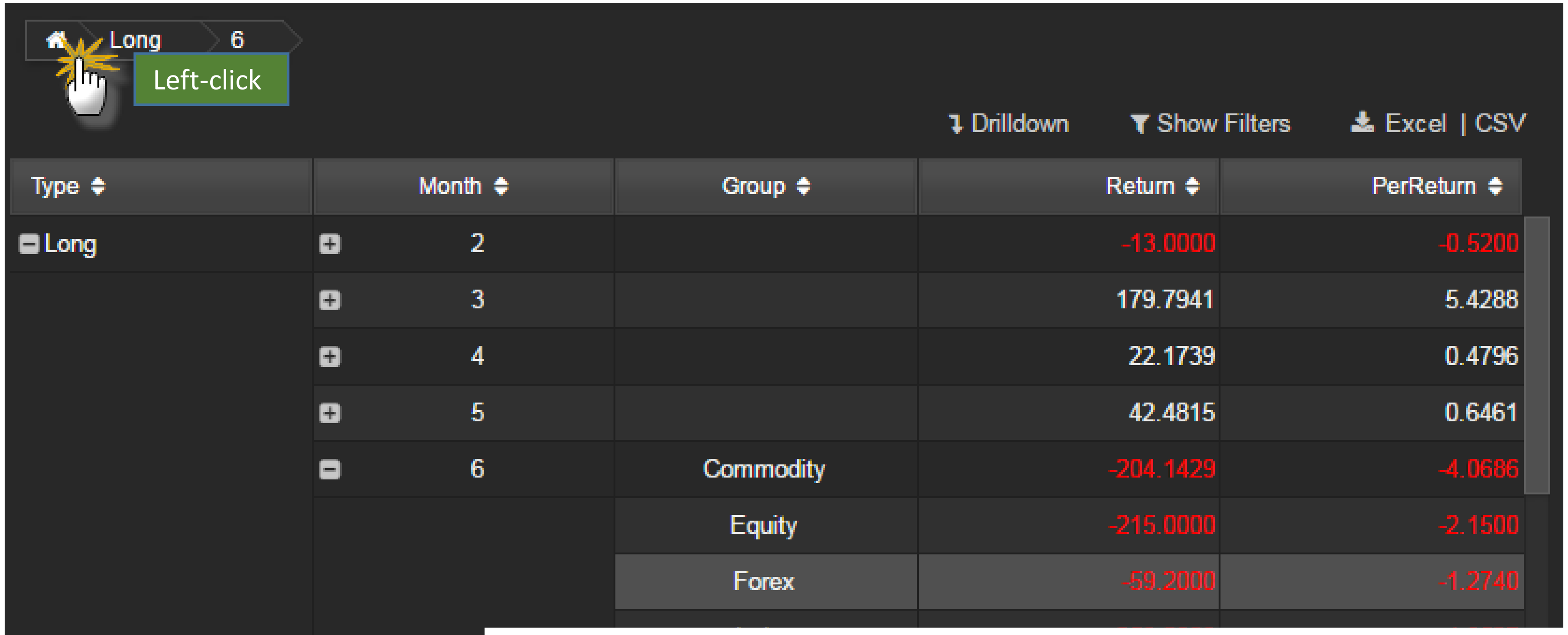


| Type | Return | PerReturn |
|---|---|---|
| Long | -24.7301 | 0.3812 |
| Short | | |

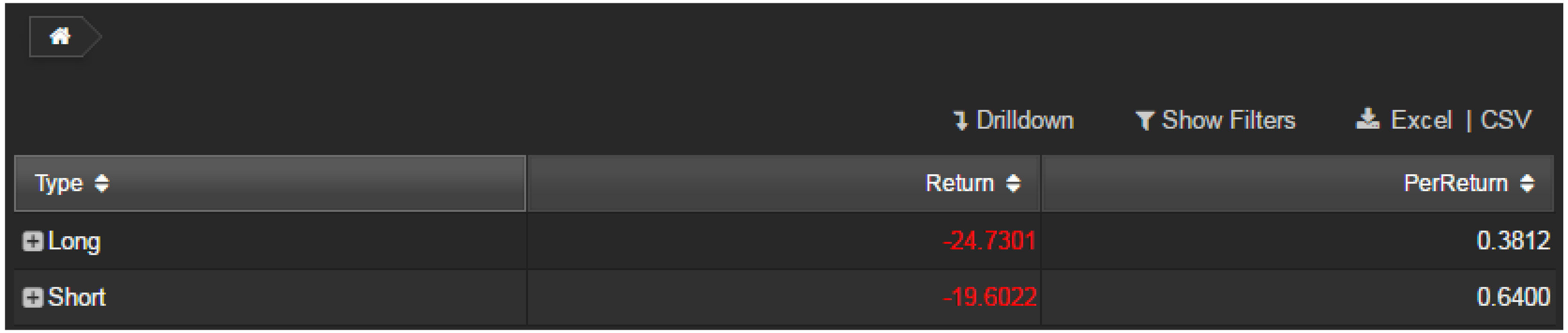
Long

| Type | Month | Return | PerReturn |
|---|---|---|---|
| Long | 2 | -13.0000 | -0.5200 |
| | 3 | 179.7941 | 5.4288 |
| | 4 | 22.1739 | 0.4796 |
| | 5 | 42.4815 | 0.6461 |
| | 6 | -157.9706 | -3.0329 |
| | 7 | 53.6875 | 0.0425 |
| | 8 | -422.7391 | -5.8391 |
| | 9 | -113.7143 | -0.1220 |
| | 10 | 216.3667 | 4.4687 |

# Return to Home



| Type | Month | Group | Return | PerReturn |
|---|---|---|---|---|
| Long | 2 | | -13.0000 | -0.5200 |
| | 3 | | 179.7941 | 5.4288 |
| | 4 | | 22.1739 | 0.4796 |
| | 5 | | 42.4815 | 0.6461 |
| | 6 | Commodity | -204.1429 | -4.0686 |
| | | Equity | -215.0000 | -2.1500 |
| | | Forex | -59.2000 | -1.2740 |
| | 7 | | | |

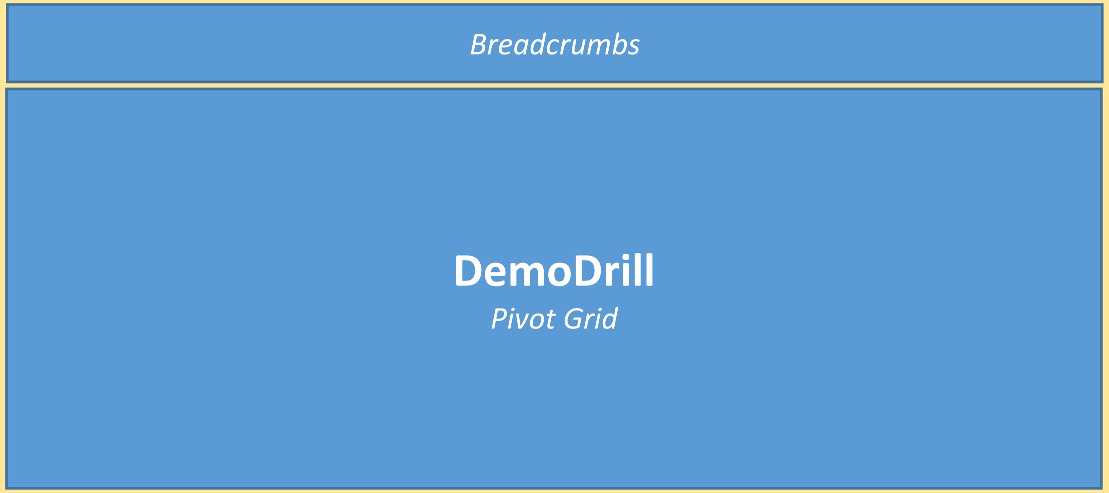


| Type | Return | PerReturn |
|---|---|---|
| Long | -24.7301 | 0.3812 |
| Short | -19.6022 | 0.6400 |

# Try This

- Create a Pivot Query, “**DemoDrill**”, from (connect to **html5evalcongroup**):

```
{`sym`src`hour`minute xcols 0!select quoteCount:count i,quoteSpread:10000*avg
(ask-bid),quoteSize:avg (bsize+asize)%2 by hour:`$string time.hh,minute:`$string 10
xbar time.minute, sym,src from dfxQuote where sym in exec distinct sym from
dfxTrade}[]
```



*Hint: Query from DemoDrillDown dashboard*

kx | it's about time

# Bar Chart

## Dashboards for Kx – “How to” Guide

Add Visuals to Data.

- Use Data Source: **PivotData**
- Connection: html5eval_grp (or html5evalcongroup)

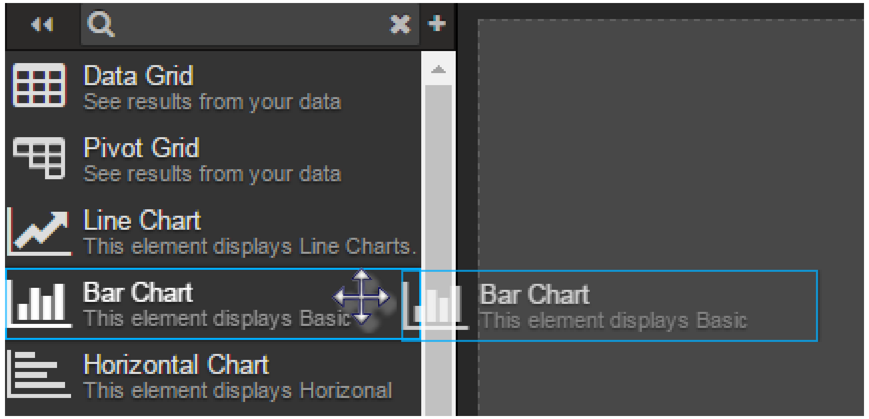
  **select Type, Month, Group, Return, PerReturn, OpenDate, CloseDate from TradeData**

- Breakdown Cols: *Type, Month, Group*
- Aggregates: *Return, PerReturn*
- Keep **Breadcrumbs** component; required for data navigation in Chart

# Visualising **PivotData**

Step: 1

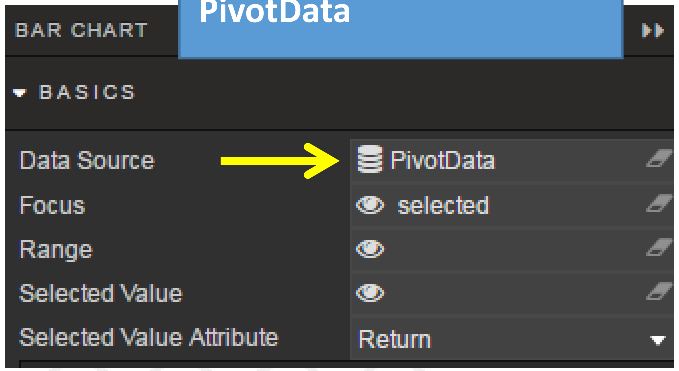
Drag *Bar Chart* inside dashboard



Data Grid
See results from your data
Pivot Grid
See results from your data
Line Chart
This element displays Line Charts.
Bar Chart
This element displays Basic
Horizontal Chart
This element displays Horizonal
Bar Chart
This element displays Basic

Step: 2

Configure Bar Chart as **PivotData**



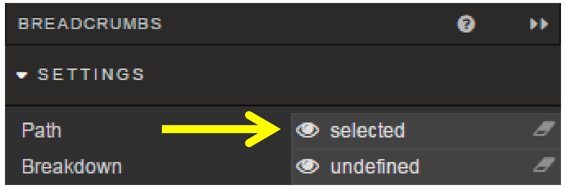
BAR CHART

BASICS

| | |
|---|---|
| Data Source | PivotData |
| Focus | selected |
| Range | |
| Selected Value | |
| Selected Value Attribute | Return |

Step: 3

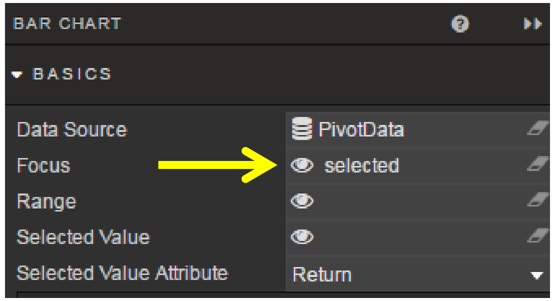
Add *Breadcrumbs* component (if not already). Create a linking viewstate called *selected* (Type: Symbol); share with Bar Chart



BREADCRUMBS

SETTINGS

| | |
|---|---|
| Path | selected |
| Breakdown | undefined |



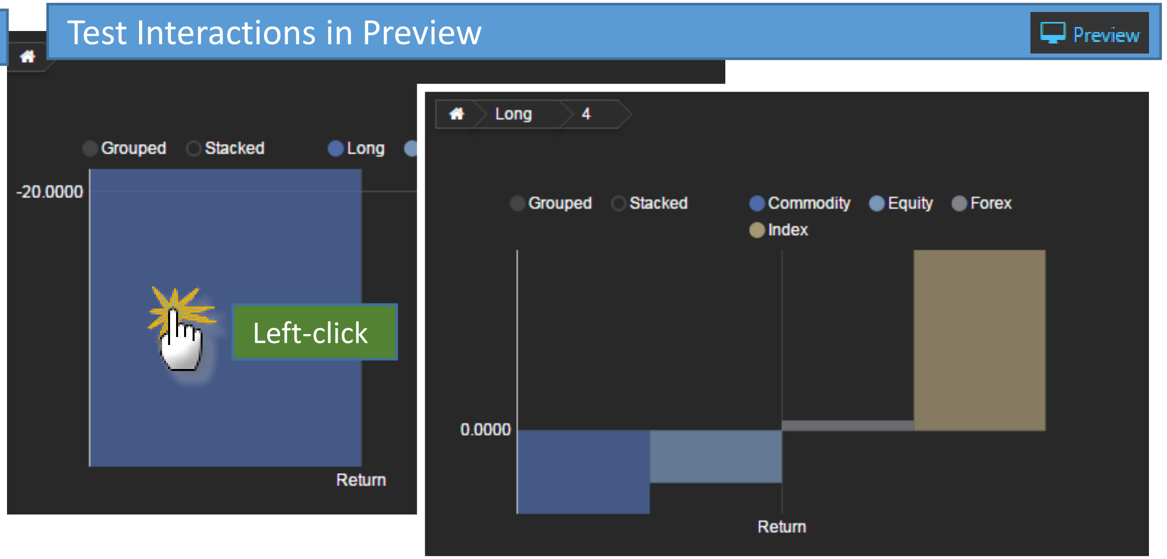
BAR CHART

BASICS

| | |
|---|---|
| Data Source | PivotData |
| Focus | selected |
| Range | |
| Selected Value | |
| Selected Value Attribute | Return |

Step: 4

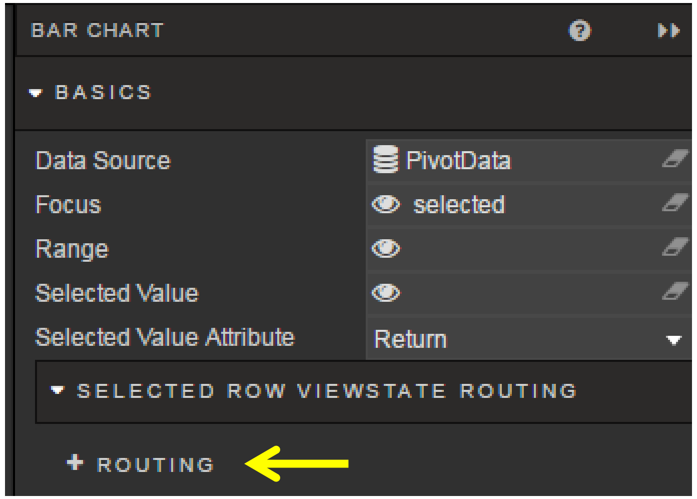
Test Interactions in Preview



# Mapping **Bar Chart** variables to Dashboard *Viewstates*

Step: 1

Click + ROUTING



BAR CHART

BASICS

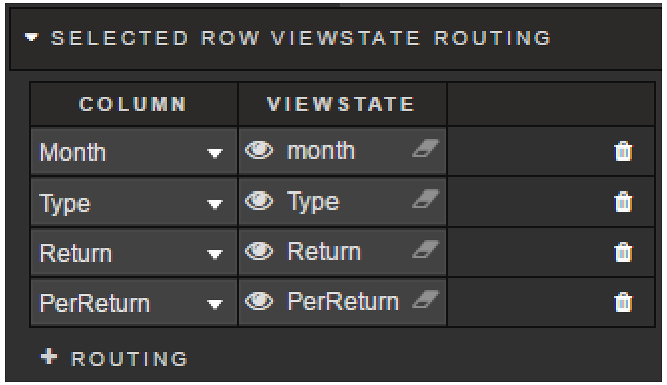
| | |
|---|---|
| Data Source | PivotData |
| Focus | selected |
| Range | |
| Selected Value | |
| Selected Value Attribute | Return |

SELECTED ROW VIEWSTATE ROUTING

+ ROUTING

Step: 2

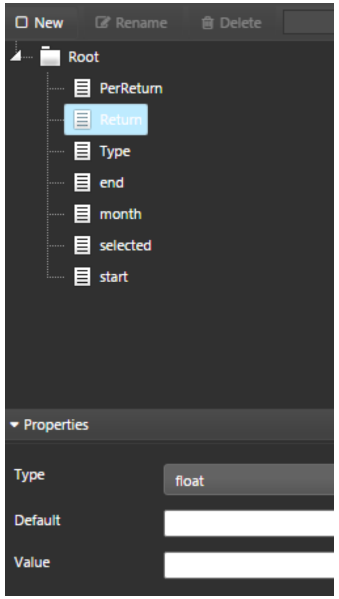
Link Chart *Data Source* variables to Dashboard *Viewstates*



SELECTED ROW VIEWSTATE ROUTING

| COLUMN | VIEWSTATE |
|---|---|
| Month | month |
| Type | Type |
| Return | Return |
| PerReturn | PerReturn |

+ ROUTING

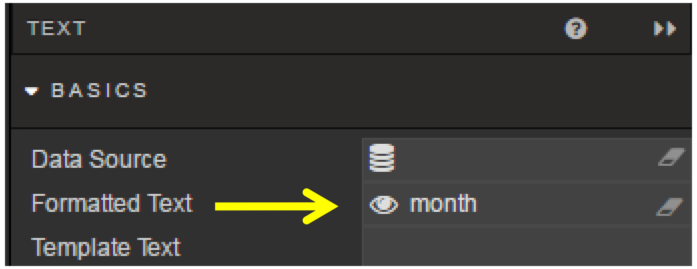


| | |
|---|---|
| Type | float |
| Default | |
| Value | |

| Viewstate | Type |
|---|---|
| Month | Int |
| Type | Symbol |
| Group | Symbol |
| Return | Float |
| PerReturn | Float |

Step: 3

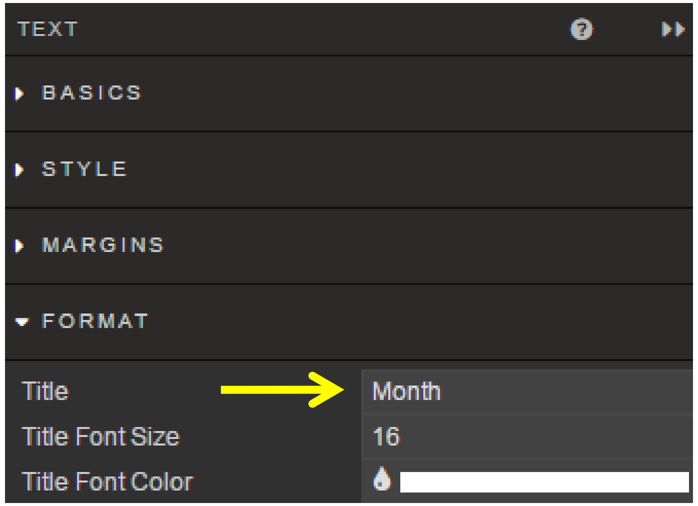
Display *Viewstate* values in a **Text** component; add 5 components.



TEXT

BASICS

| | |
|---|---|
| Data Source | |
| Formatted Text | month |
| Template Text | |



TEXT

BASICS

STYLE

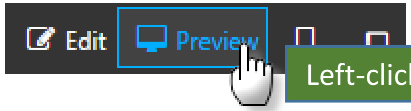
MARGINS

FORMAT

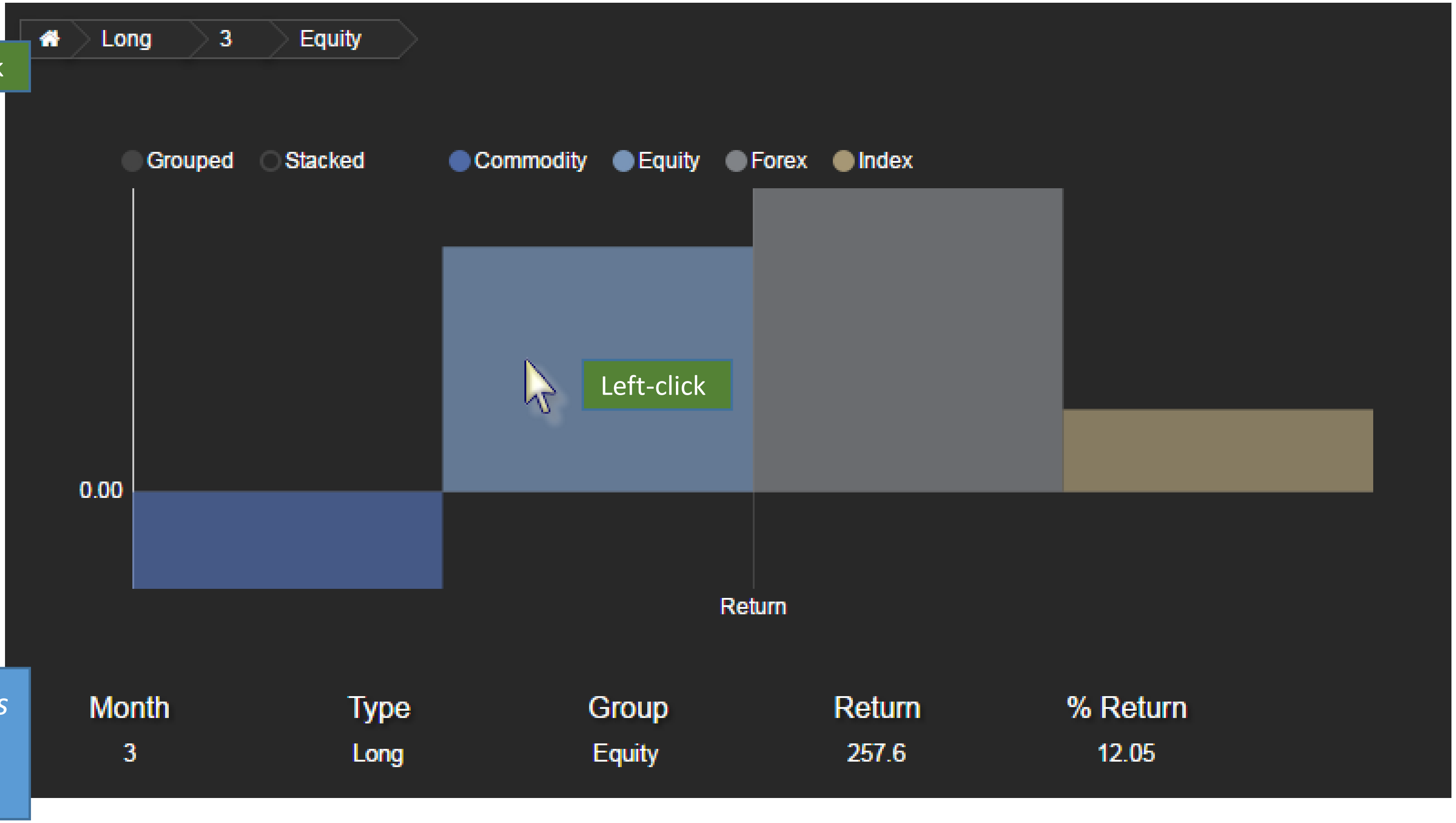
| | |
|---|---|
| Title | Month |
| Title Font Size | 16 |
| Title Font Color | |

Configure a Text component for *viewstates:* Month, Type, Group, Return and PerReturn (% Return)

# Interacting with **Bar Chart**; values assigned to *Viewstates*



*Preview* to test interactions and populate **Text** component



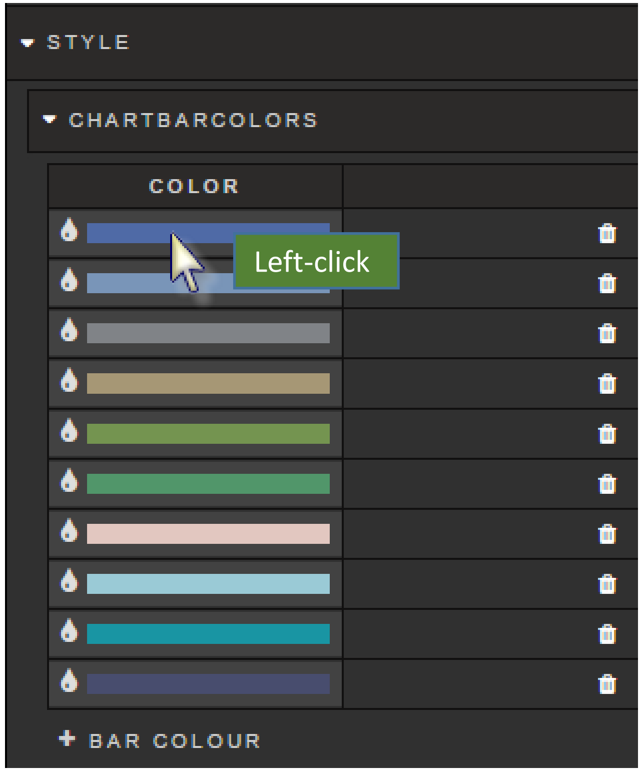
| Month | Type | Group | Return | % Return |
|---|---|---|---|---|
| 3 | Long | Equity | 257.6 | 12.05 |

Use **Text** component *Titles* in *Format* menu to add text header label

# Change **Bar Chart** colors

Step: 1

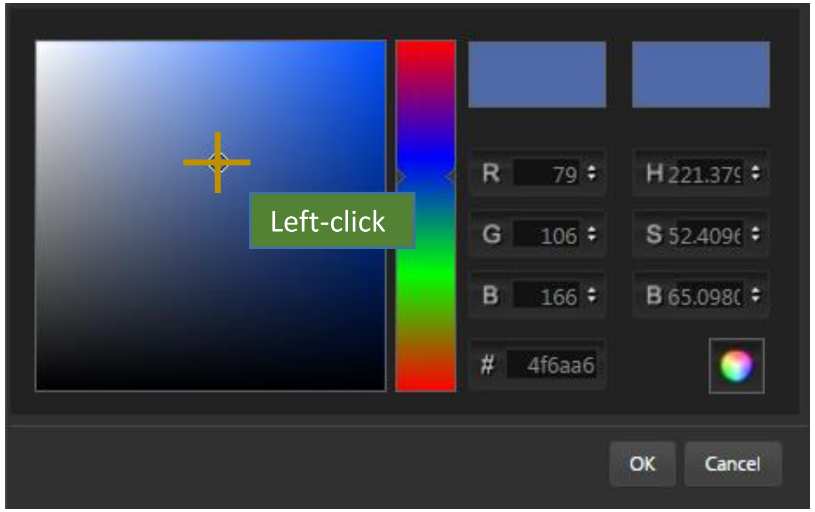
Bar chart colors are displayed in order from top to bottom



Left-click

Step: 2

Left click to bring up palette menu



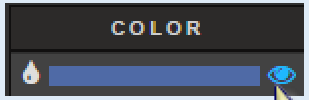
R 79
G 106
B 166
H 221.37
S 52.409
B 65.098
# 4f6aa6

Left-click

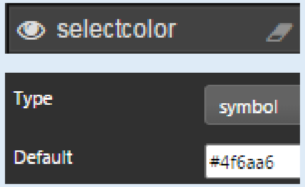
Or enter Hex color reference # 132e69

## Alternative

Assign a color to a *Viewstate;* use Hex color



Left-click

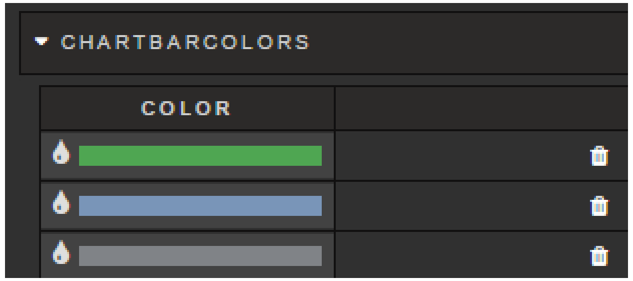


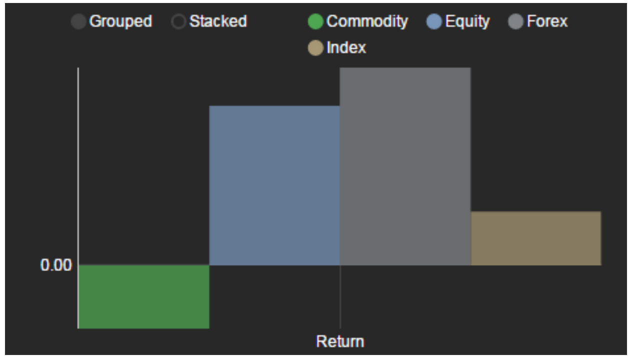
Type symbol

Default #4f6aa6

Step: 3

Color Assigned





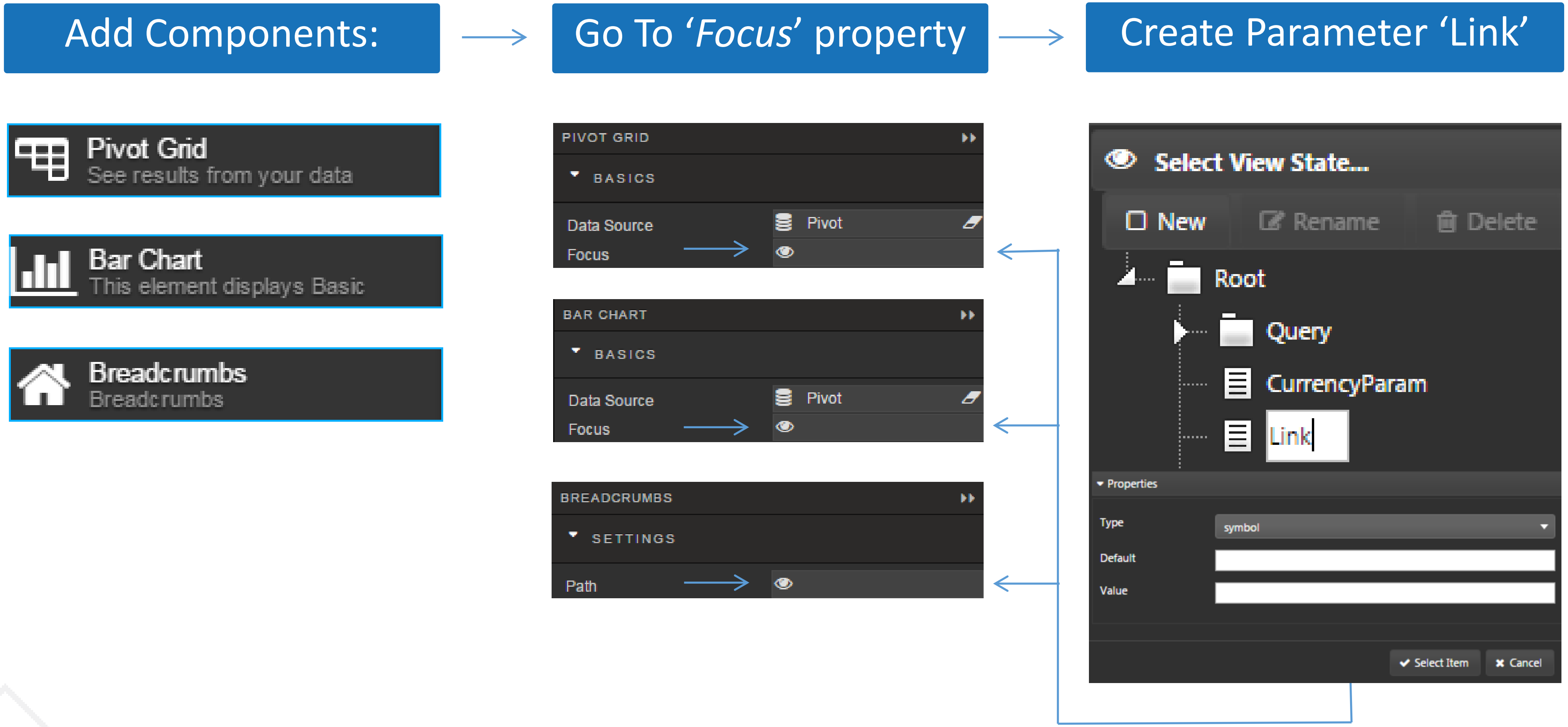
Hex color: #4fa652

kx | it's about time

# Component Linking

## Dashboards for Kx – "How to" Guide

# Link Components to Share Interactions*



Add Components: → Go To ‘*Focus*’ property → Create Parameter ‘Link’

Pivot Grid
See results from your data

Bar Chart
This element displays Basic

Breadcrumbs
Breadcrumbs

PIVOT GRID
BASICS
Data Source: Pivot
Focus →

BAR CHART
BASICS
Data Source: Pivot
Focus →

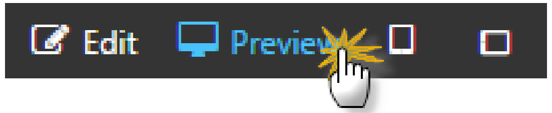
BREADCRUMBS
SETTINGS
Path →

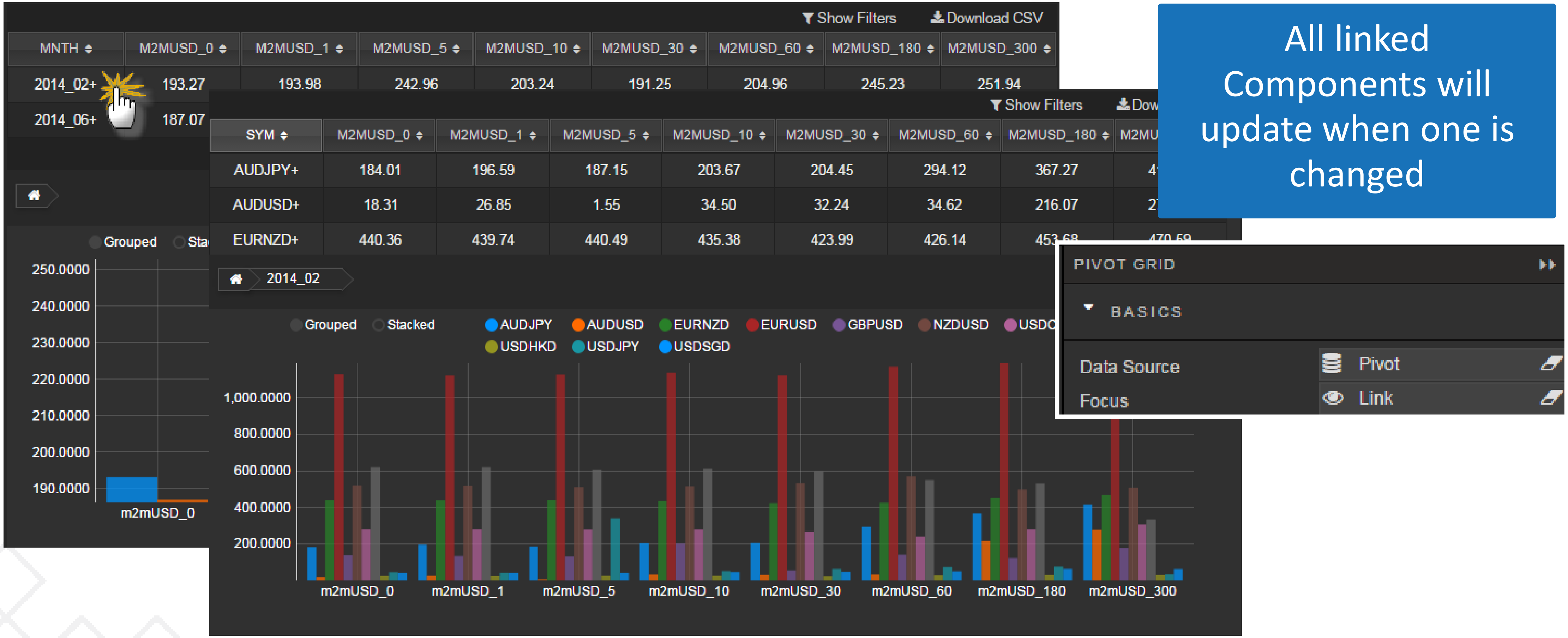
Select View State...
New · Rename · Delete
Root
- Query
- CurrencyParam
- Link

Properties
Type: symbol
Default:
Value:

*Components must share same Data Source; e.g. **PivotData**

Go to '*Preview*' mode and test Dashboard





| MNTH ⬍ | M2MUSD_0 ⬍ | M2MUSD_1 ⬍ | M2MUSD_5 ⬍ | M2MUSD_10 ⬍ | M2MUSD_30 ⬍ | M2MUSD_60 ⬍ | M2MUSD_180 ⬍ | M2MUSD_300 ⬍ |
|---|---|---|---|---|---|---|---|---|
| 2014_02+ | 193.27 | 193.98 | 242.96 | 203.24 | 191.25 | 204.96 | 245.23 | 251.94 |
| 2014_06+ | 187.07 | | | | | | | |

| SYM ⬍ | M2MUSD_0 ⬍ | M2MUSD_1 ⬍ | M2MUSD_5 ⬍ | M2MUSD_10 ⬍ | M2MUSD_30 ⬍ | M2MUSD_60 ⬍ | M2MUSD_180 ⬍ | M2MU |
|---|---|---|---|---|---|---|---|---|
| AUDJPY+ | 184.01 | 196.59 | 187.15 | 203.67 | 204.45 | 294.12 | 367.27 | 4 |
| AUDUSD+ | 18.31 | 26.85 | 1.55 | 34.50 | 32.24 | 34.62 | 216.07 | 2 |
| EURNZD+ | 440.36 | 439.74 | 440.49 | 435.38 | 423.99 | 426.14 | 453.68 | 470.59 |

Data Source: Pivot

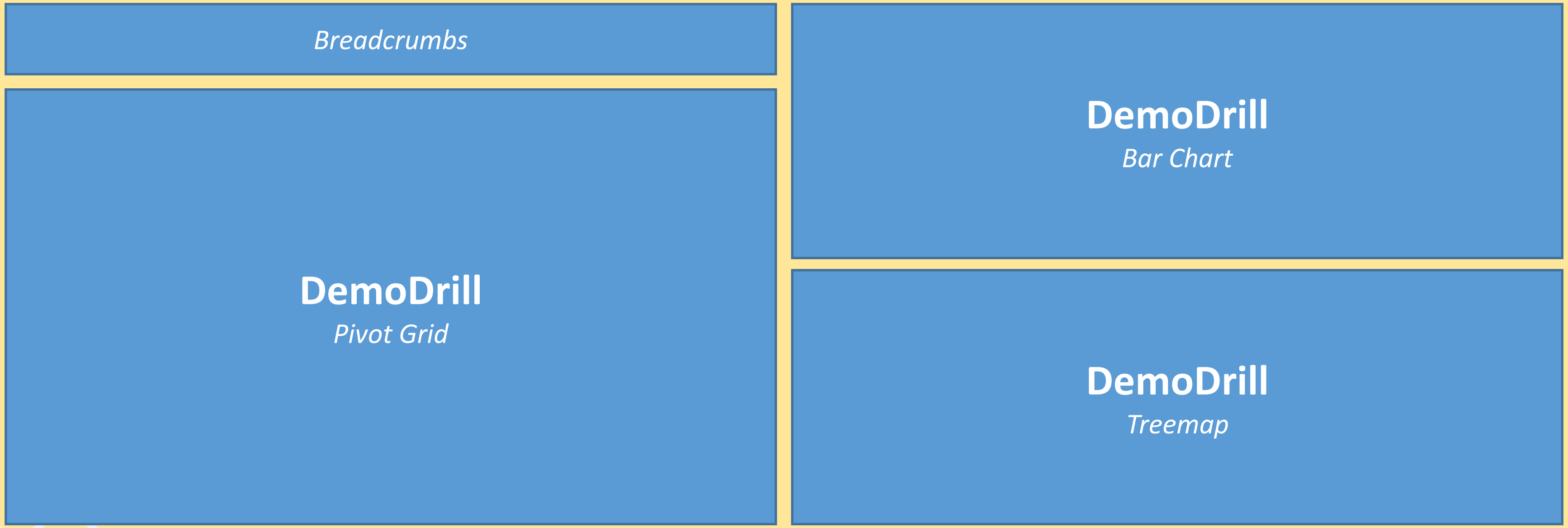
Focus: Link

All linked Components will update when one is changed

# Try This

- Link a **Pivot Grid**, **Bar Chart**, **Breadcrumbs** and **Treemap** for **“DemoDrill”**:

```
{`sym`src`hour`minute xcols 0!select quoteCount:count i,quoteSpread:10000*avg
(ask-bid),quoteSize:avg (bsize+asize)%2 by hour:`$string time.hh,minute:`$string 10
xbar time.minute, sym,src from dfxQuote where sym in exec distinct sym from
dfxTrade}[]
```


*Breadcrumbs*

**DemoDrill**
*Pivot Grid*

**DemoDrill**
*Bar Chart*

**DemoDrill**
*Treemap*

*Hint: Refer to [code.kx.docs ](code.kx.docs)for information on configuring the **Treemap** component*

kx | it's about time

# Highlight Rules

## Dashboards for Kx – "How to" Guide

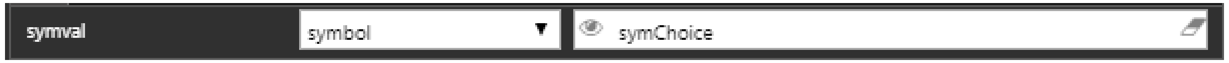
# Highlighting Change in Your Data

Highlight rules help direct users to changes and updates in their data. It’s best used with streaming and polling data

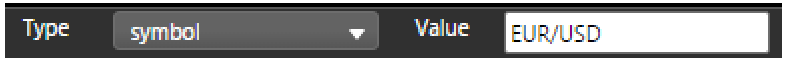
- Use Data Source: **LatestPrices**
- Connection: html5eval_grp (or html5evalcongroup)

```
{[symval] `src xasc select last bsize, last bid, last ask, last asize by src from dfxRandomQuote where sym=symval}
```

- Map *symval* to viewstate, *symChoice.* Set Default *symChoice* to *EUR/USD*

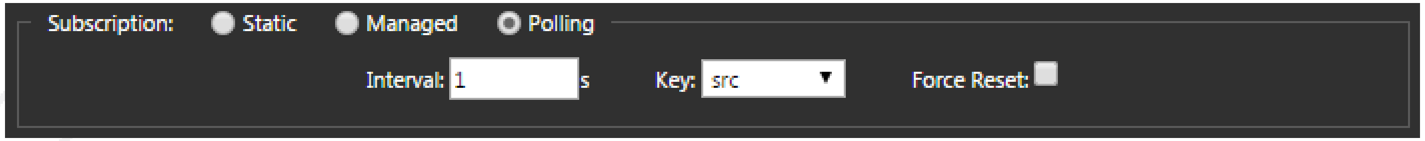


symval | symbol ▼ | symChoice



Type: symbol ▼ Value: EUR/USD

- Set *Subscription* to *Polling, 1 second.* Part of Query Editor.



Subscription: ○ Static ○ Managed ● Polling

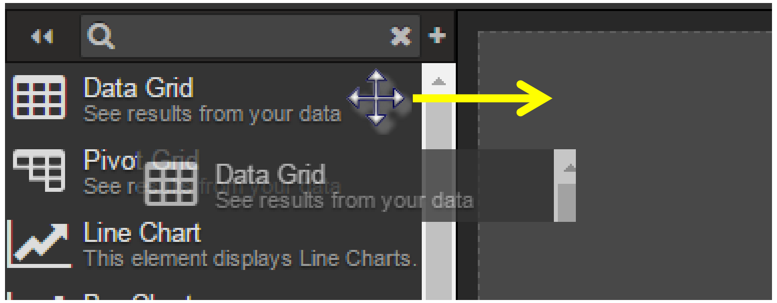
Interval: 1 s Key: src ▼ Force Reset: □

Ensure data is polling, managed or streamed for highlights rule to update

# Highlighting Change in Your Data: Bid

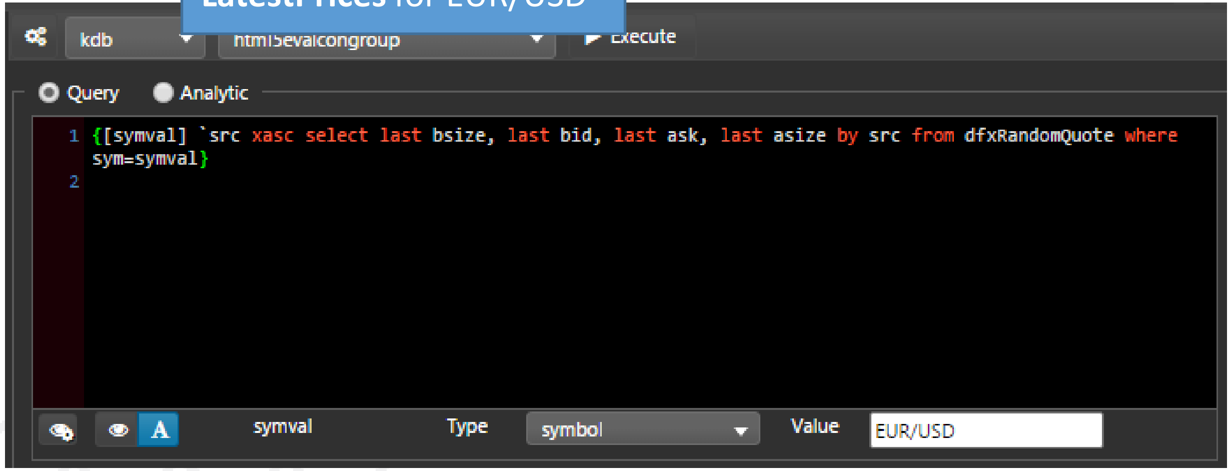
Step: 1

Drag *Data Grid* inside dashboard


Data Grid
See results from your data
Pivot Grid
Data Grid
See results from your data
Line Chart
This element displays Line Charts.

Step: 2

Configure query **LatestPrices** for EUR/USD



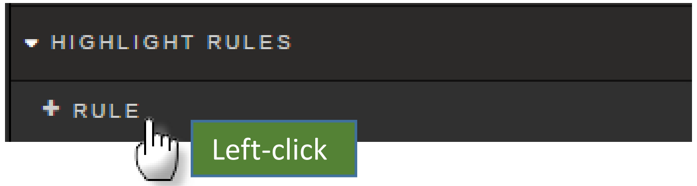
● Query ○ Analytic

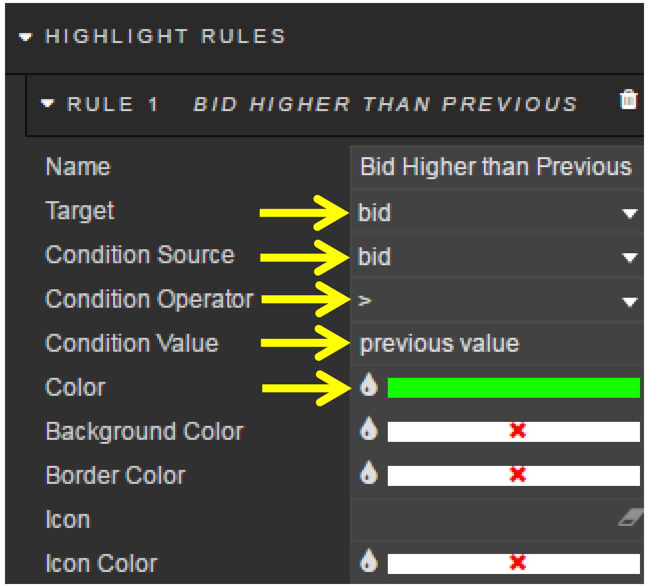
```
{[symval] `src xasc select last bsize, last bid, last ask, last asize by src from dfxRandomQuote where
sym=symval}
```

symval Type symbol Value EUR/USD

Step: 3

Create a Highlight Rule for Bid: Value greater than previous value **+ RULE**





HIGHLIGHT RULES

RULE 1 *BID HIGHER THAN PREVIOUS*

| | |
|---|---|
| Name | Bid Higher than Previous |
| Target | bid |
| Condition Source | bid |
| Condition Operator | > |
| Condition Value | previous value |
| Color | 🟩 |
| Background Color | ✗ |
| Border Color | ✗ |
| Icon | |
| Icon Color | ✗ |

Continued...

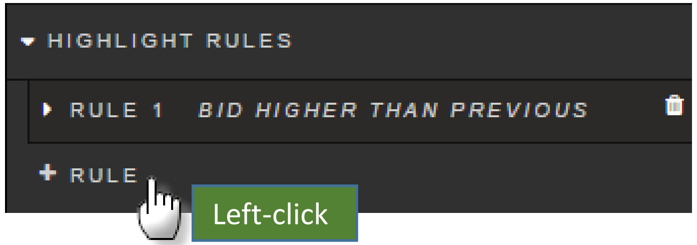
# Highlighting Change in Your Data: Ask

Step: 4

Create a Highlight Rule for Bid: Value less than previous value ✚ RULE

Step: 5

Repeat Rules for Ask

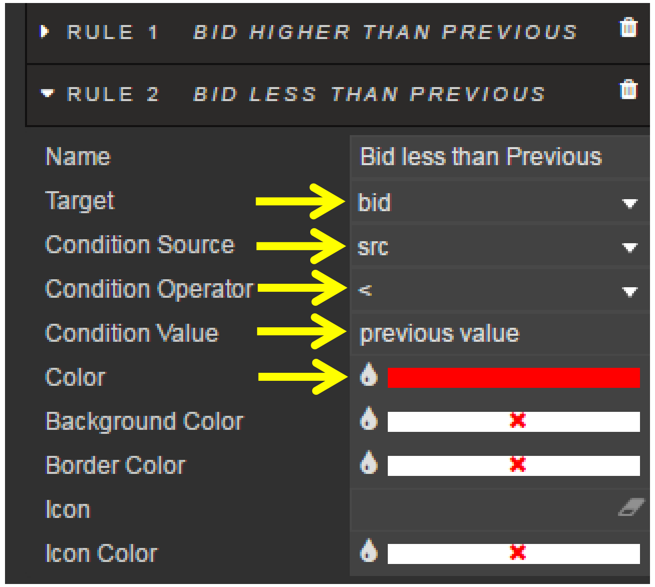


HIGHLIGHT RULES

RULE 1 BID HIGHER THAN PREVIOUS

✚ RULE

Left-click



RULE 1 BID HIGHER THAN PREVIOUS

RULE 2 BID LESS THAN PREVIOUS

| | |
|---|---|
| Name | Bid less than Previous |
| Target | bid |
| Condition Source | src |
| Condition Operator | < |
| Condition Value | previous value |
| Color | 🟥 |
| Background Color | × |
| Border Color | × |
| Icon | |
| Icon Color | × |

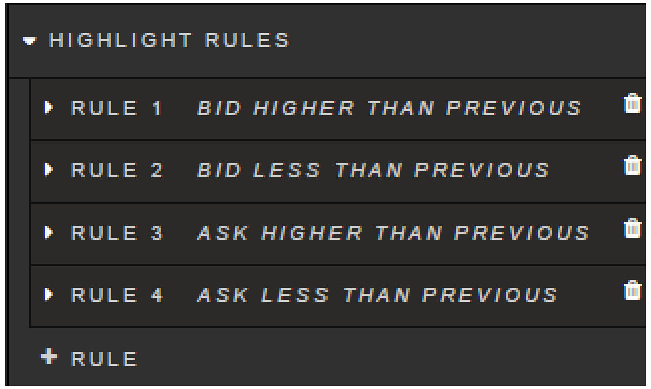
## Alternative

*Background Color*: changes cell color

*Border Color*: changes cell border color

*Icon*: Select from Icon menu, icon will appear when highlight rule is true

*Icon Color*: Select color of icon to appear when rule is true



HIGHLIGHT RULES

RULE 1 BID HIGHER THAN PREVIOUS

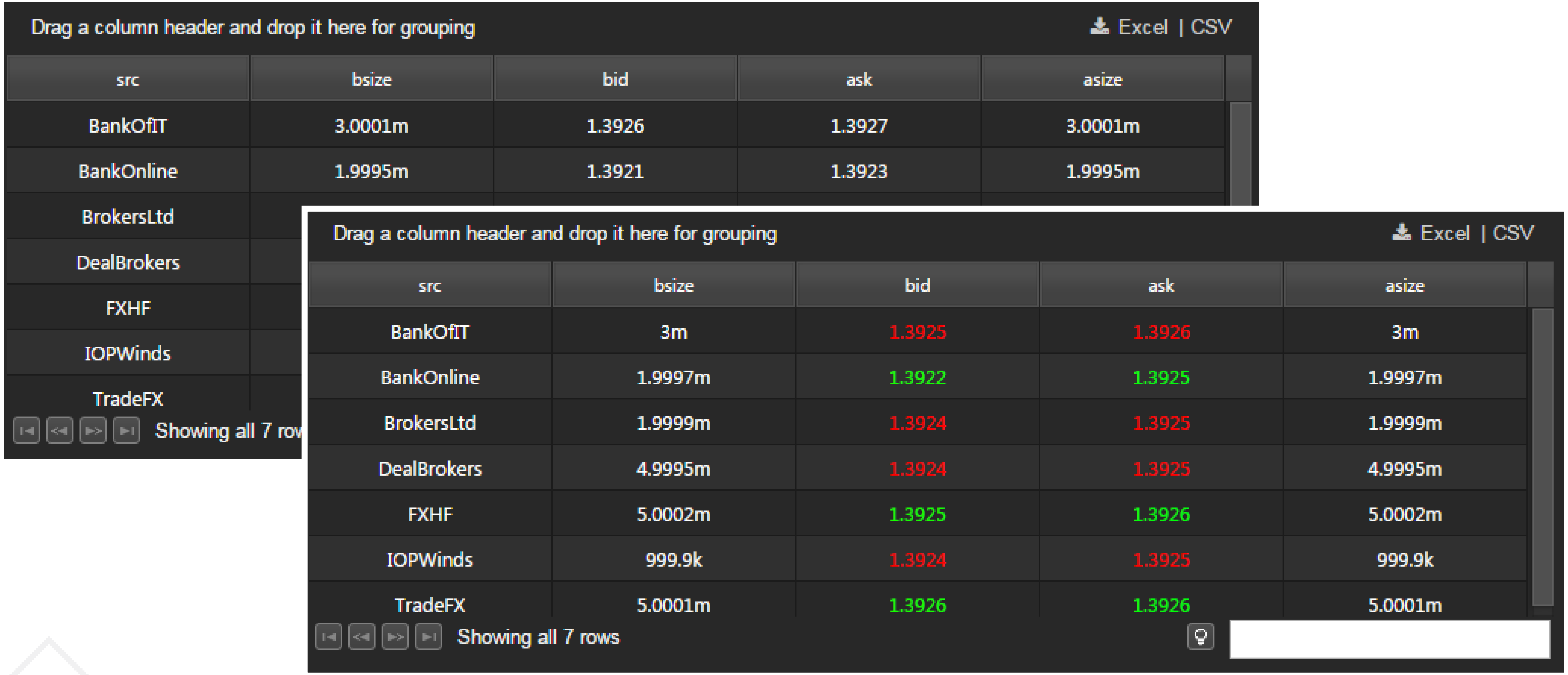
RULE 2 BID LESS THAN PREVIOUS

RULE 3 ASK HIGHER THAN PREVIOUS

RULE 4 ASK LESS THAN PREVIOUS

✚ RULE

# Preview to see Highlight rules in action



Drag a column header and drop it here for grouping

| src | bsize | bid | ask | asize |
|---|---|---|---|---|
| BankOfIT | 3.0001m | 1.3926 | 1.3927 | 3.0001m |
| BankOnline | 1.9995m | 1.3921 | 1.3923 | 1.9995m |
| BrokersLtd | | | | |
| DealBrokers | | | | |
| FXHF | | | | |
| IOPWinds | | | | |
| TradeFX | | | | |

Showing all 7 row

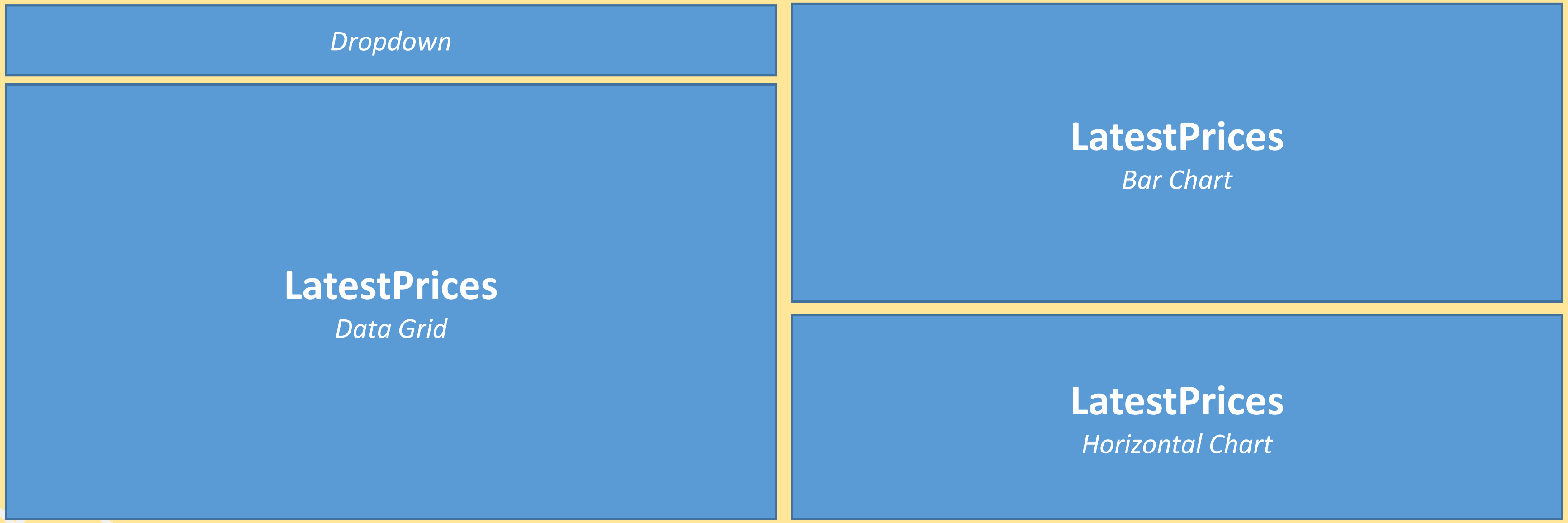
Drag a column header and drop it here for grouping

| src | bsize | bid | ask | asize |
|---|---|---|---|---|
| BankOfIT | 3m | 1.3925 | 1.3926 | 3m |
| BankOnline | 1.9997m | 1.3922 | 1.3925 | 1.9997m |
| BrokersLtd | 1.9999m | 1.3924 | 1.3925 | 1.9999m |
| DealBrokers | 4.9995m | 1.3924 | 1.3925 | 4.9995m |
| FXHF | 5.0002m | 1.3925 | 1.3926 | 5.0002m |
| IOPWinds | 999.9k | 1.3924 | 1.3925 | 999.9k |
| TradeFX | 5.0001m | 1.3926 | 1.3926 | 5.0001m |

Showing all 7 rows

# Try This

- Using **LatestPrices** query, add a **Dropdown** component to feed symbols: EUR/USD, GBP/USD, USD/CAD, USD/CHF and USD/CHF into *symChoice*
- Apply Red and Green Range color to columns A(sk)size and B(id)size respectively
- Apply Red and Green Max color to Ask and Bid columns
- Add **Bar** and **Horizontal** chart; use chart highlight rules to mark change



*Dropdown*

**LatestPrices**
*Data Grid*

**LatestPrices**
*Bar Chart*

**LatestPrices**
*Horizontal Chart*

*Hint: Query from DemoMarketMakers dashboard. See code.kx.docs for more information on configuring Horizontal charts*

it's about time

# Multi-Chart

## Dashboards for Kx – "How to" Guide

# Multiple Chart overlays

Dashboard’s Multi-chart allows for different chart types including bar, bubble, candlesticks and lines to be overlaid on a single chart. Includes dual axis support.

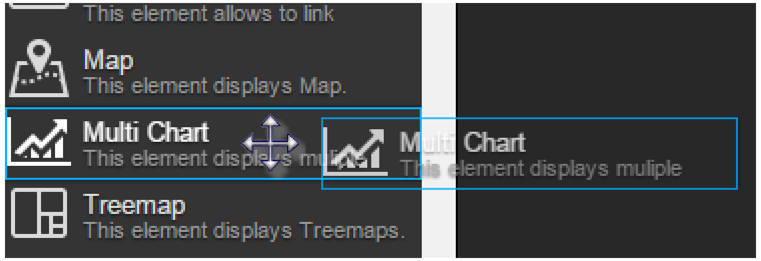
- Use Data Source: **TradingPerformance**
- Connection: html5eval_grp (or html5evalcongroup)

```
`Date xasc select Date, Open, Day0_Close, Day0_Ret, Day0_Cumu, Day0_StepFcast from ChartDNA
```

# Configure Multi-Chart

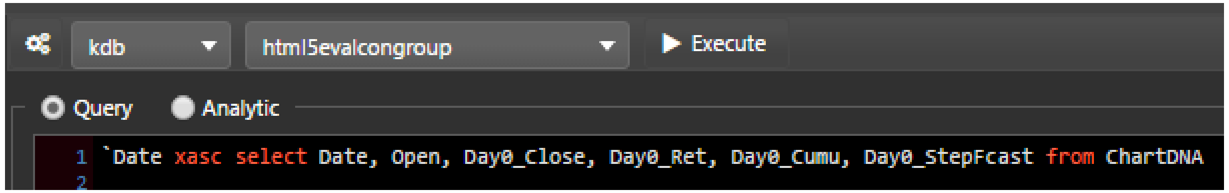
Step: 1

Drag *Multi Chart* inside dashboard

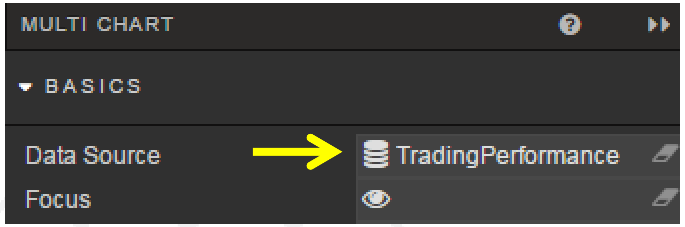

This element allows to link
Map
This element displays Map.
Multi Chart
This element displays mu...
Multi Chart
This element displays muliple
Treemap
This element displays Treemaps.

Step: 2

Configure query **TradingPerformance**



```
`Date xasc select Date, Open, Day0_Close, Day0_Ret, Day0_Cumu, Day0_StepFcast from ChartDNA
```

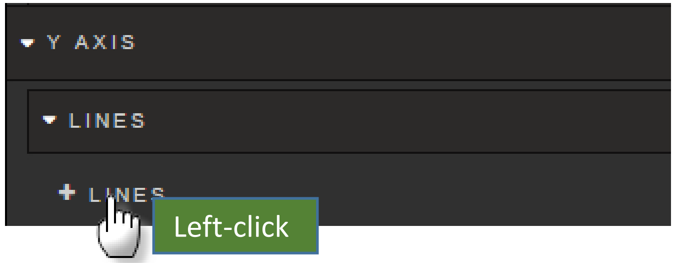


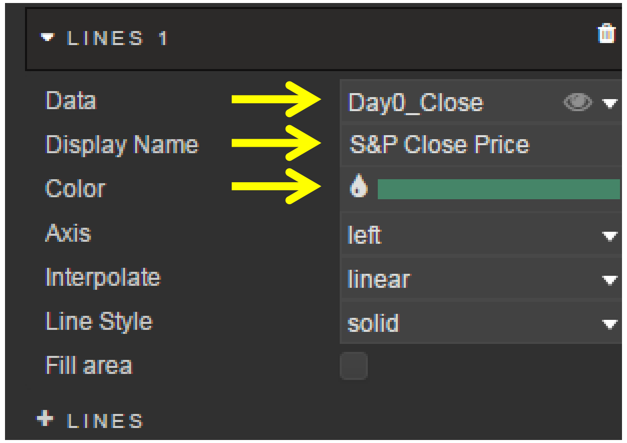
Data Source: TradingPerformance

Focus

Step: 3

Add a Line Chart





Data: Day0_Close

Display Name: S&P Close Price

Color: 🟩

Axis: left

Interpolate: linear

Line Style: solid

Fill area: □

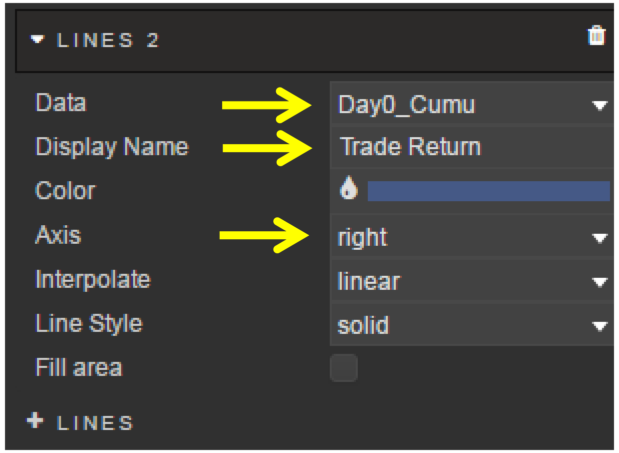
Add prices for S&P Index. Update color to #458568 and add Legend name

Continued...

# Configure Multi-Chart

Step: 4

Add a second Line chart for Trade Return



LINES 2

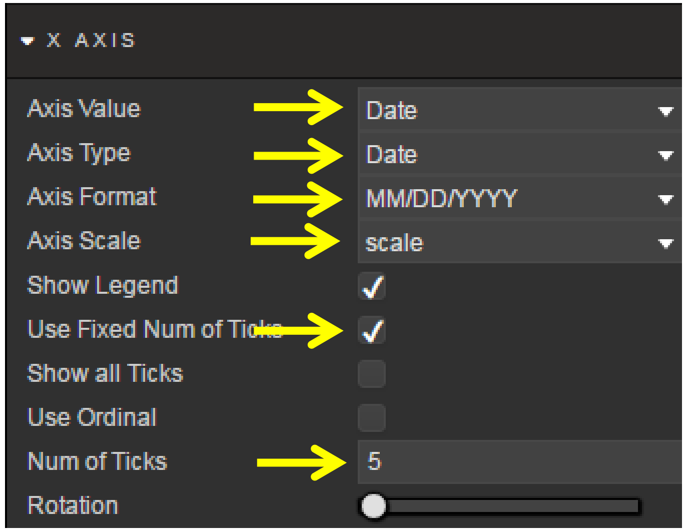
| | |
|---|---|
| Data | Day0_Cumu |
| Display Name | Trade Return |
| Color | |
| Axis | right |
| Interpolate | linear |
| Line Style | solid |
| Fill area | □ |

✚ LINES

Chart Trade Return on the right axis

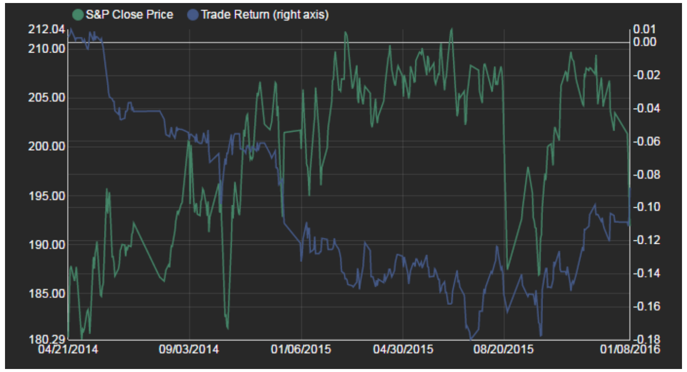
Step: 5

Fix x-axis formatting



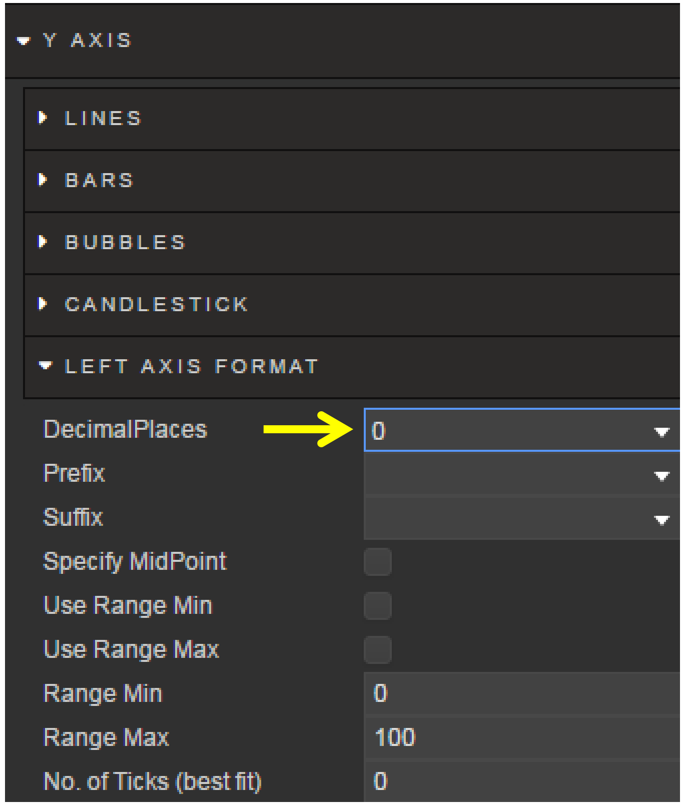
X AXIS

| | |
|---|---|
| Axis Value | Date |
| Axis Type | Date |
| Axis Format | MM/DD/YYYY |
| Axis Scale | scale |
| Show Legend | ✓ |
| Use Fixed Num of Ticks | ✓ |
| Show all Ticks | □ |
| Use Ordinal | □ |
| Num of Ticks | 5 |
| Rotation | |



Step: 6

Fix y-axis formatting: left



Y AXIS

LINES

BARS

BUBBLES

CANDLESTICK

LEFT AXIS FORMAT

| | |
|---|---|
| DecimalPlaces | 0 |
| Prefix | |
| Suffix | |
| Specify MidPoint | □ |
| Use Range Min | □ |
| Use Range Max | □ |
| Range Min | 0 |
| Range Max | 100 |
| No. of Ticks (best fit) | 0 |

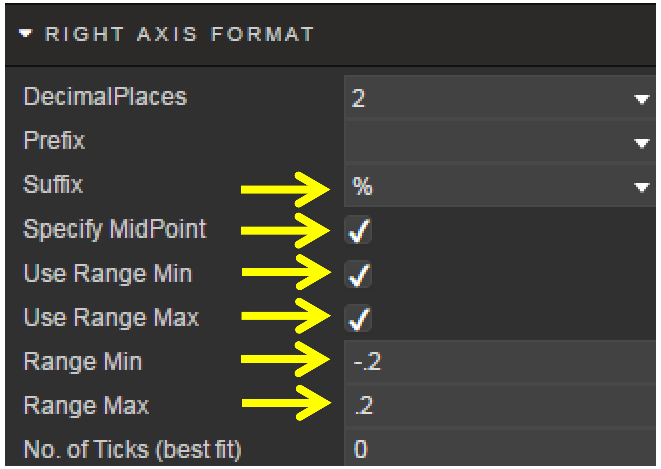
Single adjustment of left decimal places

Continued…

# Configure Multi-Chart

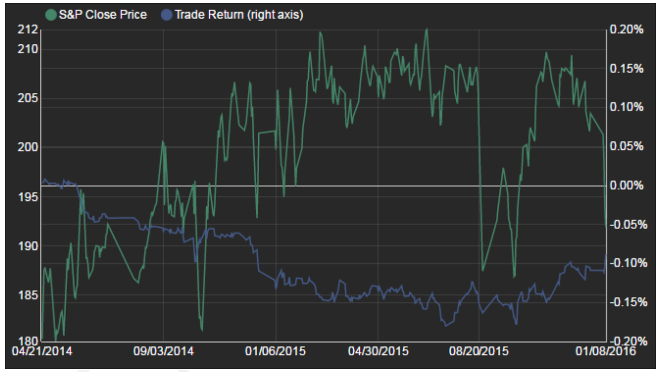
Step: 7

## Fix y-axis formatting: right



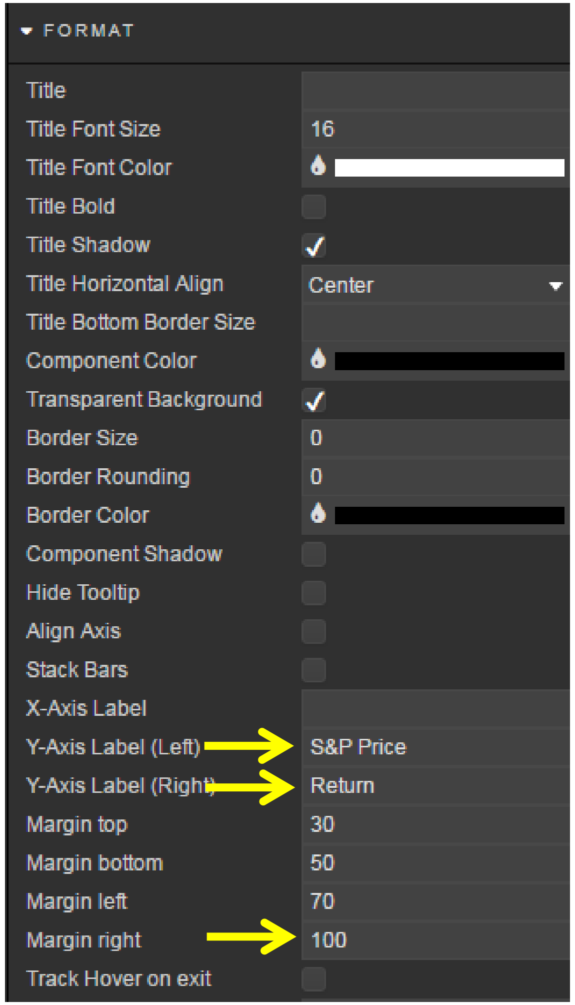
▾ RIGHT AXIS FORMAT

| | |
|---|---|
| DecimalPlaces | 2 ▾ |
| Prefix | ▾ |
| Suffix | % ▾ |
| Specify MidPoint | ✓ |
| Use Range Min | ✓ |
| Use Range Max | ✓ |
| Range Min | -.2 |
| Range Max | .2 |
| No. of Ticks (best fit) | 0 |



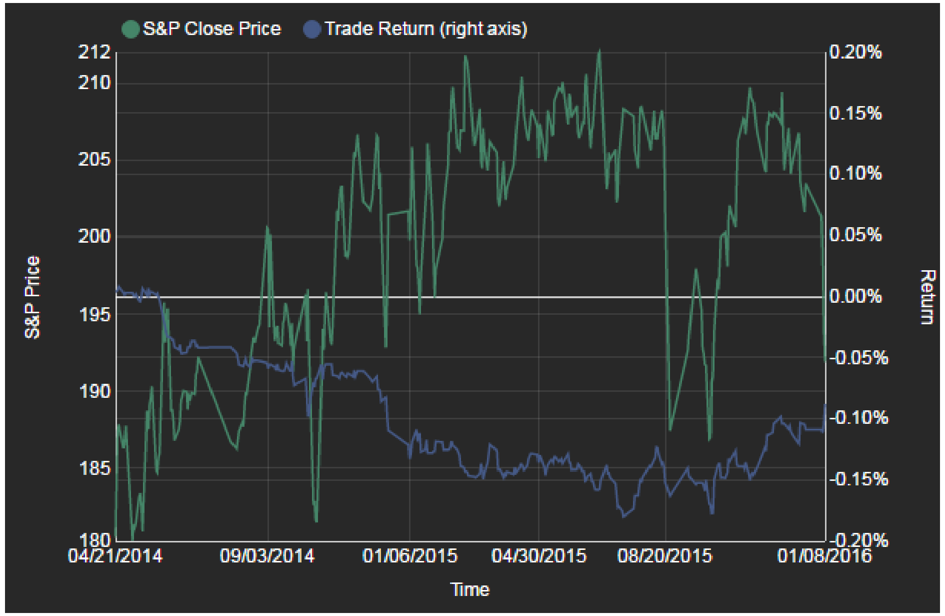
Step: 8

## Format Chart



▾ FORMAT

| | |
|---|---|
| Title | |
| Title Font Size | 16 |
| Title Font Color | ⬜ |
| Title Bold | ☐ |
| Title Shadow | ✓ |
| Title Horizontal Align | Center ▾ |
| Title Bottom Border Size | |
| Component Color | ■ |
| Transparent Background | ✓ |
| Border Size | 0 |
| Border Rounding | 0 |
| Border Color | ■ |
| Component Shadow | ☐ |
| Hide Tooltip | ☐ |
| Align Axis | ☐ |
| Stack Bars | ☐ |
| X-Axis Label | |
| Y-Axis Label (Left) | S&P Price |
| Y-Axis Label (Right) | Return |
| Margin top | 30 |
| Margin bottom | 50 |
| Margin left | 70 |
| Margin right | 100 |
| Track Hover on exit | ☐ |



- Add an *Overlay* data source (connection: ***html5evalcongroup***)

```
`Date xasc select Date, Day4_Cumu from ChartDNA
```

- Change the chart type for Day0_Cumu to show fill while not obscuring S&P prices
- Apply a date range filter to Multi-chart

it's about time

# Zoom and Pan

## Dashboards for Kx – "How to" Guide

Zoom and Pan provides built-in navigation focus.  The zoom-and-pan range can be paired with dashboard viewstates and used as filters in other dashboard queries.

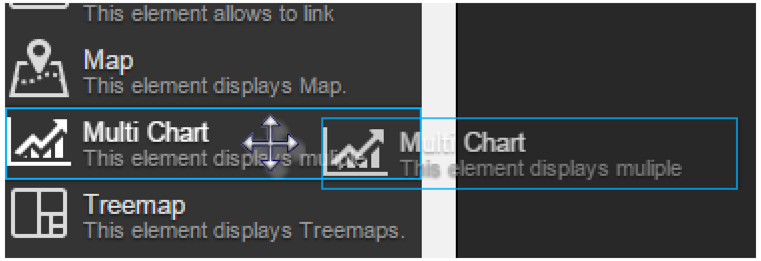
- Create Data Source: **ZoomandPan**
- Connection: html5eval_grp (or html5evalcongroup)

```
`Date xasc select Date, Open from ChartDNA
```

# Configure Zoom and Pan

Step: 1

Drag *Multi Chart* inside dashboard


This element allows to link

Map
This element displays Map.

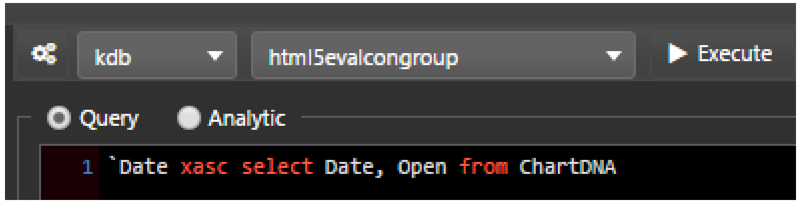
Multi Chart
This element displays multiple

Multi Chart
This element displays muliple

Treemap
This element displays Treemaps.

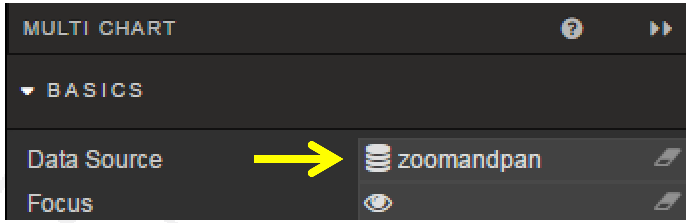
Step: 2

Configure query **ZoomandPan**



● Query ○ Analytic

```
`Date xasc select Date, Open from ChartDNA
```


MULTI CHART

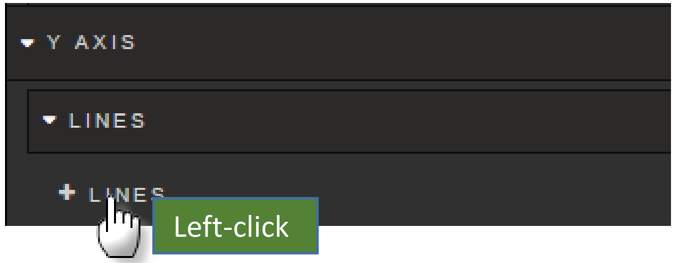
BASICS

Data Source → zoomandpan

Focus

Step: 3

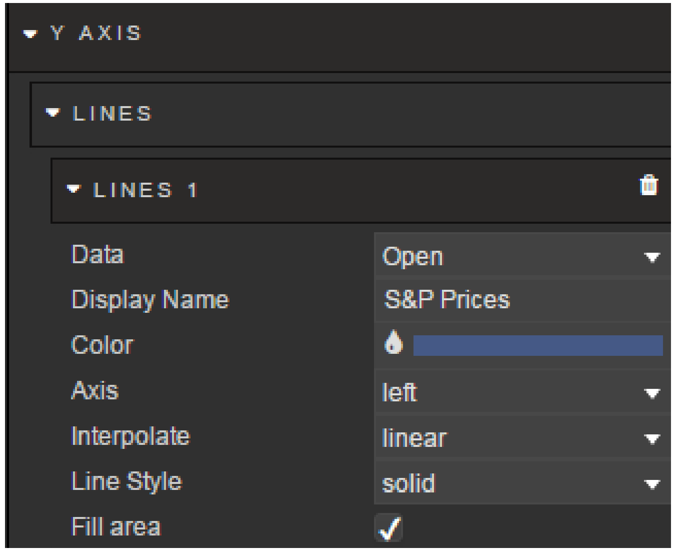
Add a Line Chart


Y AXIS

LINES

+ LINES

Left-click


Y AXIS

LINES

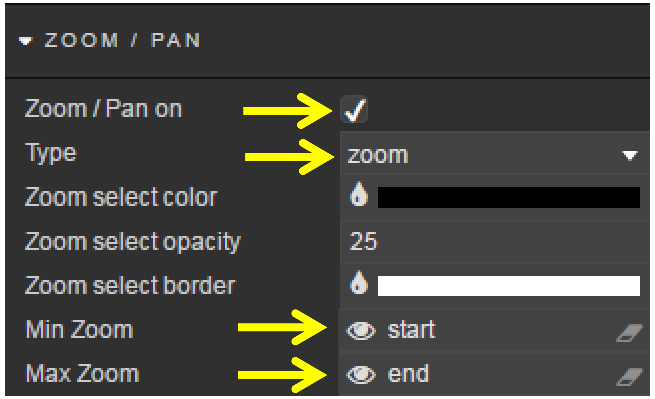
LINES 1

| | |
|---|---|
| Data | Open |
| Display Name | S&P Prices |
| Color | |
| Axis | left |
| Interpolate | linear |
| Line Style | solid |
| Fill area | ✓ |

# Configure Zoom and Pan

Step: 4

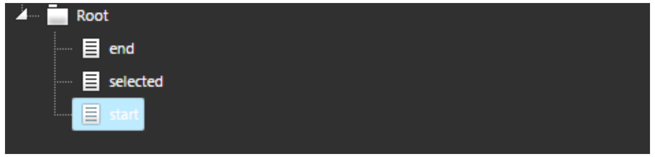
Switch on *Zoom / Pan*



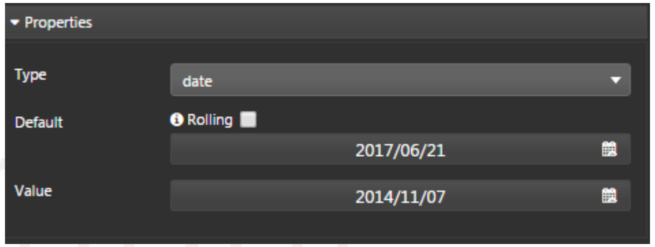
▾ ZOOM / PAN

| | |
|---|---|
| Zoom / Pan on | ✓ |
| Type | zoom ▾ |
| Zoom select color | ■ |
| Zoom select opacity | 25 |
| Zoom select border | □ |
| Min Zoom | start |
| Max Zoom | end |

Create *Viewstate* parameters for *Min Zoom* and *Max Zoom*

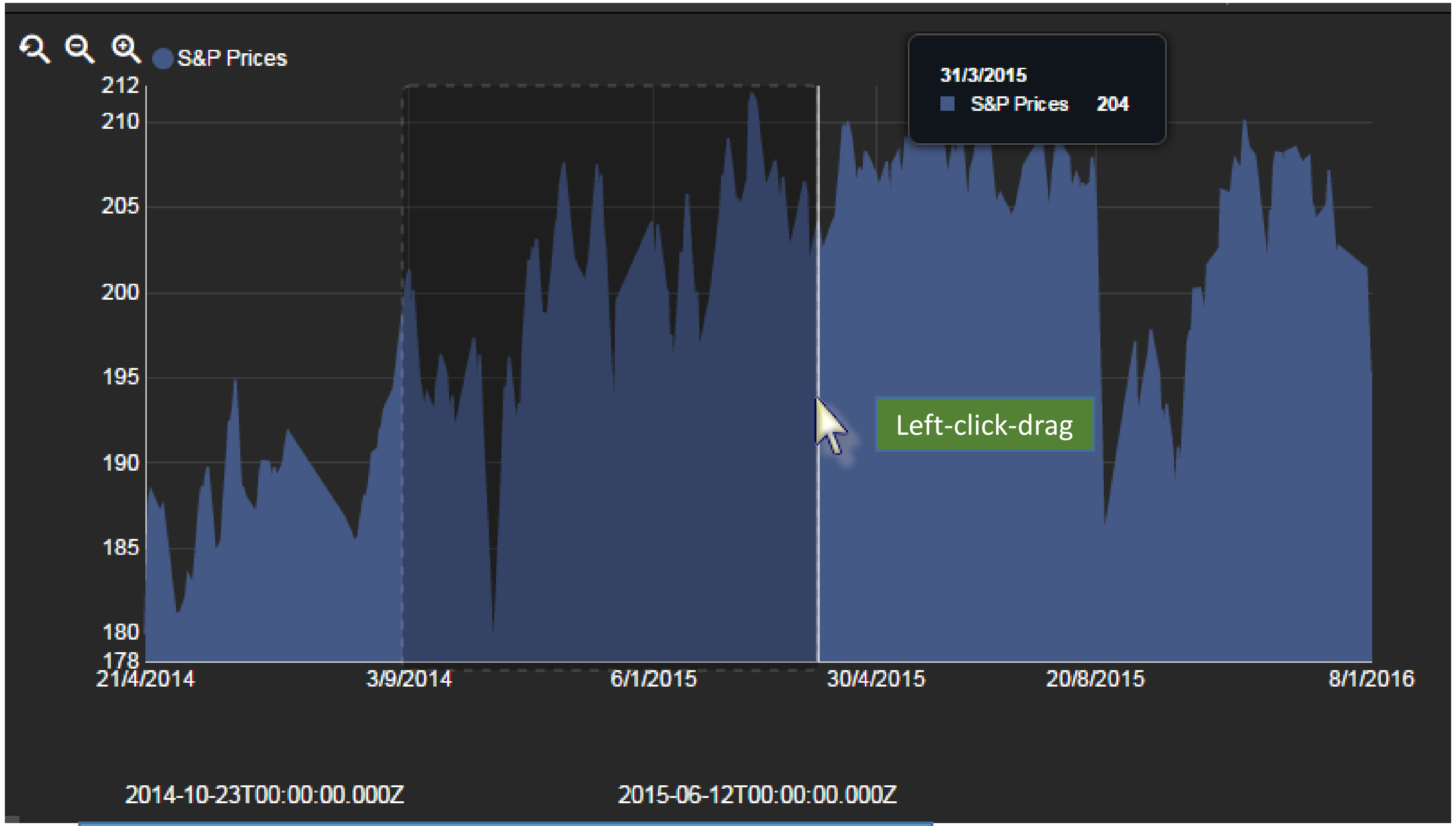


Root
- end
- selected
- start



▾ Properties

| | |
|---|---|
| Type | date ▾ |
| Default | Rolling □<br>2017/06/21 |
| Value | 2014/11/07 |



2014-10-23T00:00:00.000Z 2015-06-12T00:00:00.000Z

Text component displaying *viewstates, start* and *end*

it's about time

# Range Slider

## Dashboards for Kx – "How to" Guide

# Breadcrumbs for Charts: Range Slider

Similar to Pan-and-Zoom, Range Slider is a separate component which is paired with a chart and can be used to control the amount of data to display.  It's paired with a chart similar to how Breadcrumbs works.

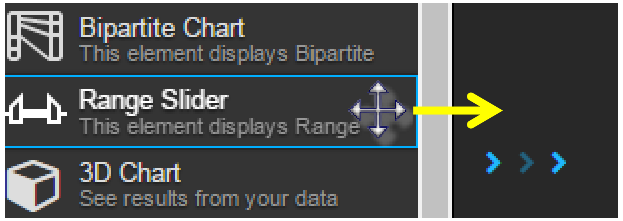
- Use Data Source: **ZoomandPan**
- Connection: html5eval_grp (or html5evalcongroup)

```
`Date xasc select Date, Open from ChartDNA
```

# Breadcrumbs for Charts: Range Slider

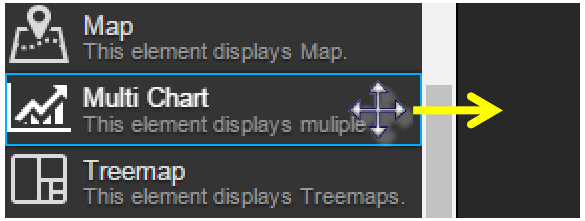
Step: 1

Drag *Range Slider* inside dashboard


Bipartite Chart
This element displays Bipartite
Range Slider
This element displays Range
3D Chart
See results from your data

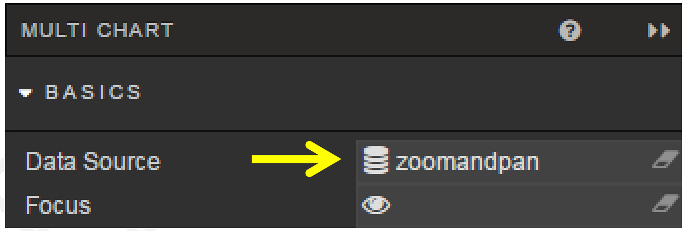
Step: 2

Add a *Multi-Chart* to the dashboard


Map
This element displays Map.
Multi Chart
This element displays muliple
Treemap
This element displays Treemaps.

Step: 3

Configure the *Multi-Chart* to use **ZoomandPan** data source; create a line chart for Open price



MULTI CHART

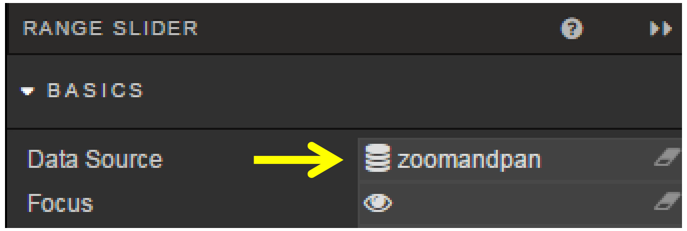
BASICS

Data Source: zoomandpan

Focus

Step: 4

Likewise, configure the *Range Slider* to use **ZoomandPan** data source.



RANGE SLIDER

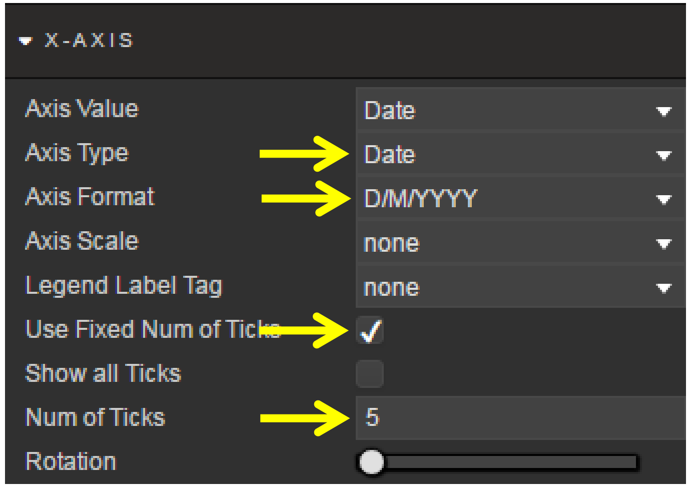
BASICS

Data Source: zoomandpan

Focus

Step: 5

Set X-Axis for Date for *Range Slider*



X-AXIS

Axis Value: Date

Axis Type: Date

Axis Format: D/M/YYYY

Axis Scale: none

Legend Label Tag: none

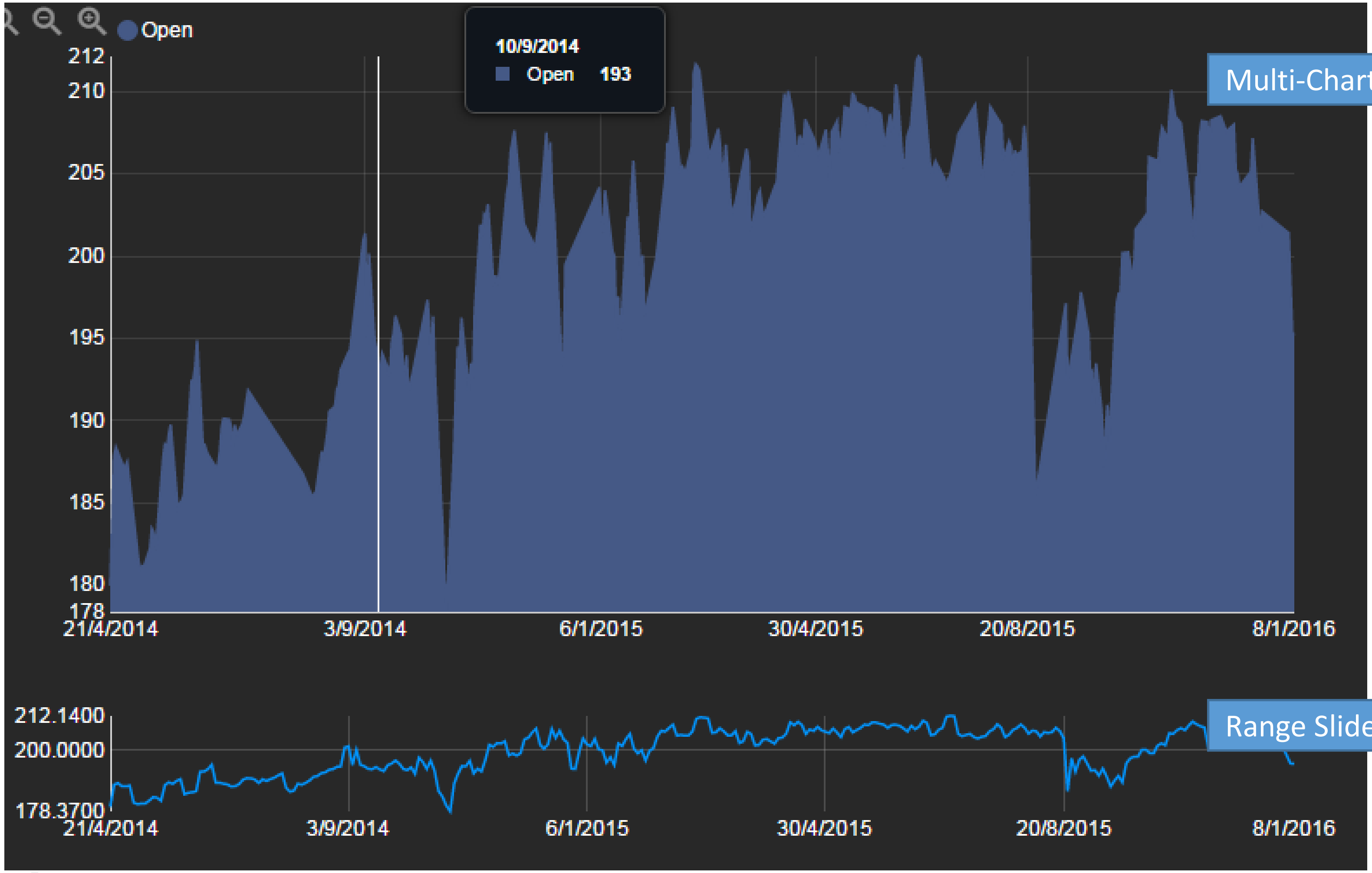
Use Fixed Num of Ticks: ✓

Show all Ticks: □

Num of Ticks: 5

Rotation

# How It Looks So Far



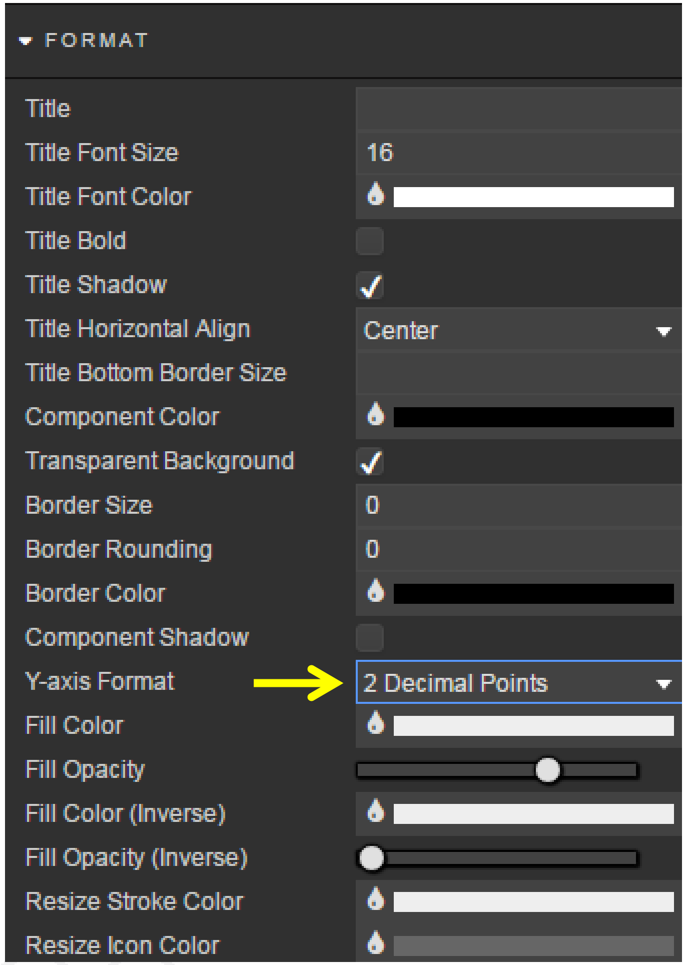
Multi-Chart

Range Slider

# Link Range Slider to Multi-Chart

Step: 6

## Format Y-Axis

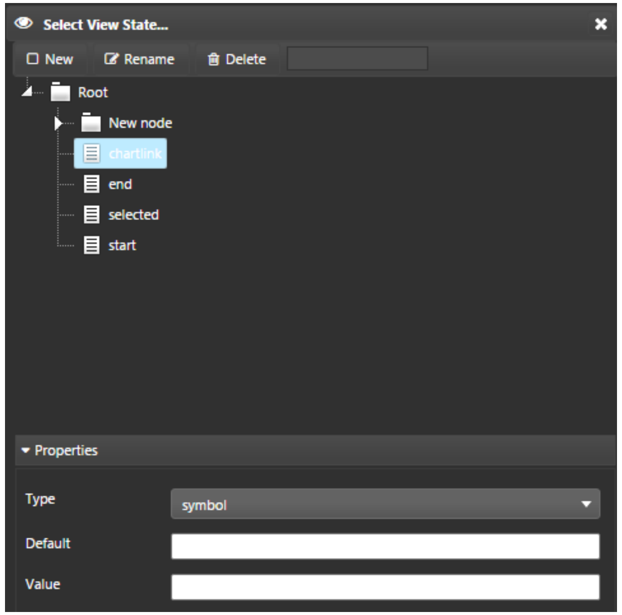


FORMAT

| | |
|---|---|
| Title | |
| Title Font Size | 16 |
| Title Font Color | ⬜ |
| Title Bold | ☐ |
| Title Shadow | ☑ |
| Title Horizontal Align | Center |
| Title Bottom Border Size | |
| Component Color | ⬛ |
| Transparent Background | ☑ |
| Border Size | 0 |
| Border Rounding | 0 |
| Border Color | ⬛ |
| Component Shadow | ☐ |
| Y-axis Format | 2 Decimal Points |
| Fill Color | ⬜ |
| Fill Opacity | |
| Fill Color (Inverse) | ⬜ |
| Fill Opacity (Inverse) | |
| Resize Stroke Color | ⬜ |
| Resize Icon Color | |

Step: 7

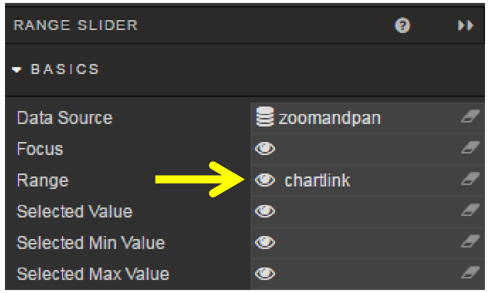
## Create *viewstate, "chartlink"* of type Symbol



| | |
|---|---|
| Type | symbol |
| Default | |
| Value | |

Step: 8

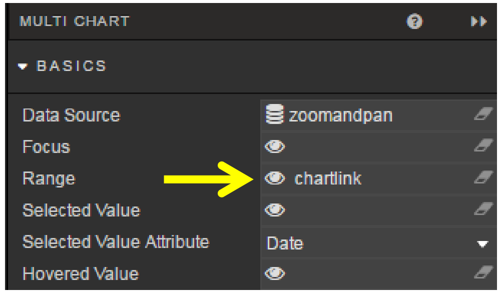
## Link *Range Slider* to *Multi-Chart* using *Range* property



RANGE SLIDER

BASICS

| | |
|---|---|
| Data Source | zoomandpan |
| Focus | |
| Range | chartlink |
| Selected Value | |
| Selected Min Value | |
| Selected Max Value | |



MULTI CHART

BASICS

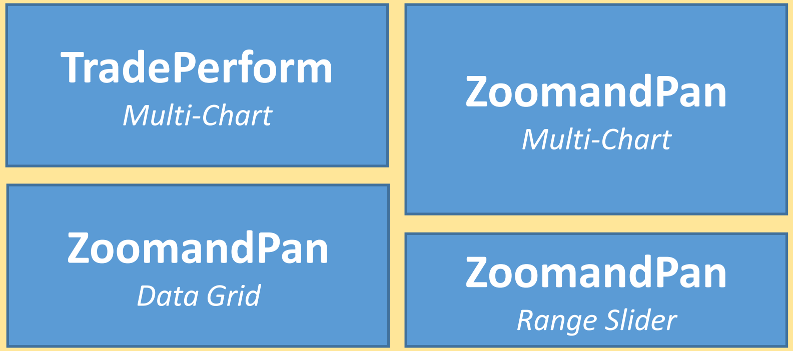
| | |
|---|---|
| Data Source | zoomandpan |
| Focus | |
| Range | chartlink |
| Selected Value | |
| Selected Value Attribute | Date |
| Hovered Value | |

Try This

- Set Viewstate for *Selected Min* and *Selected Max* property
- Add a **Data Grid**. Configure it to display the values of the Range Slider
- Add a second **Multi-chart**; have it use the data source (**TradePerform**):

```
`Date xasc select Date, Day4_Cumu, Day3_Cumu, Day2_Cumu, Day1_Cumu, Day0_Cumu from
ChartDNA
```

- Configure the second Multi-chart to display Day4_Cumu, Day3_Cumu, Day2_Cumu, Day1_Cumu, Day0_Cumu ("Trade Performance for days 0 to 4"). Connect it to the Range Slider



# Try This

- Create a set of Highlight Rules to Show Changes in VWAP

kx | it's about time

# Navigation

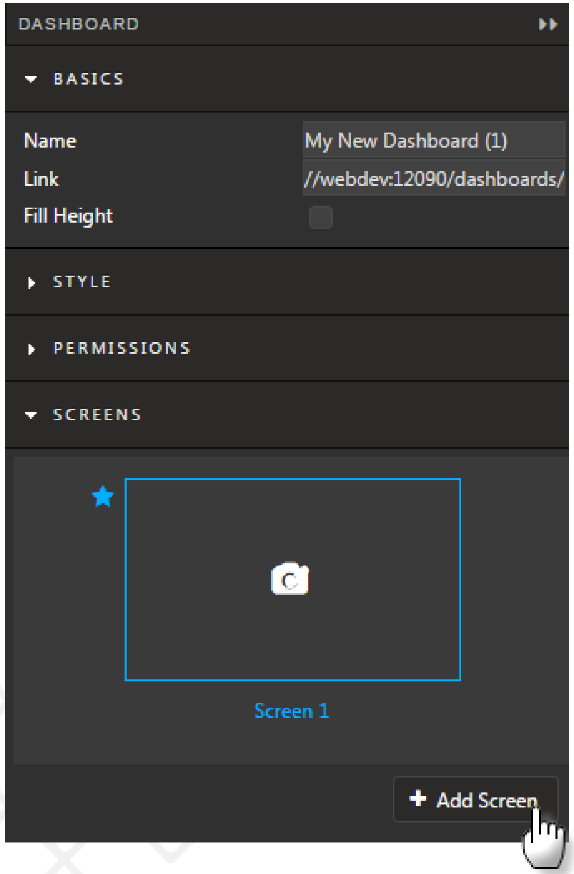
## Dashboards for Kx – "How to" Guide

The Navigation component allows users to navigated between different dashboards, and different screens in the same dashboard.  Viewstates can be shared across dashboards so information from one can be passed into another.

# Add Screens to support multiple dashboard views in a single dashboard

Step: 1

First create additional screens within the parent dashboard; these can be populate with earlier examples (optional)



DASHBOARD

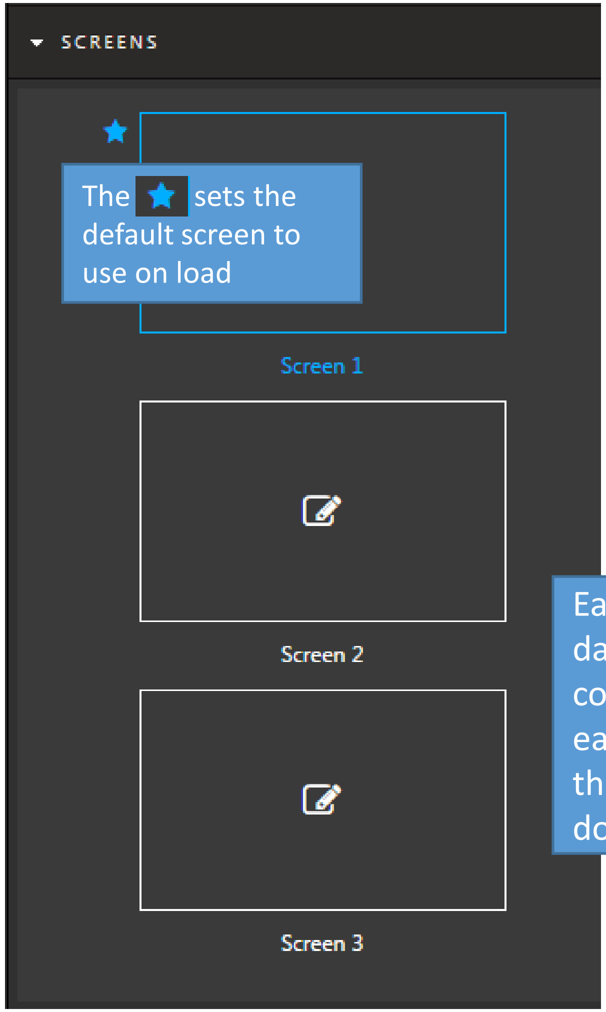
BASICS

Name: My New Dashboard (1)

Link: //webdev:12090/dashboards/

Fill Height: □

Left-click

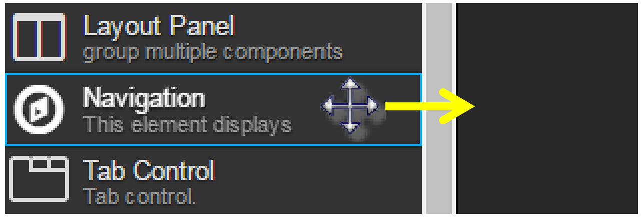


The ★ sets the default screen to use on load

Each Screen can be its own dashboard with components. For example, each “Try Me” tutorial in this document could be done as a screen

Step: 2

Left click-and-drag the *Navigation* component into Screen 1

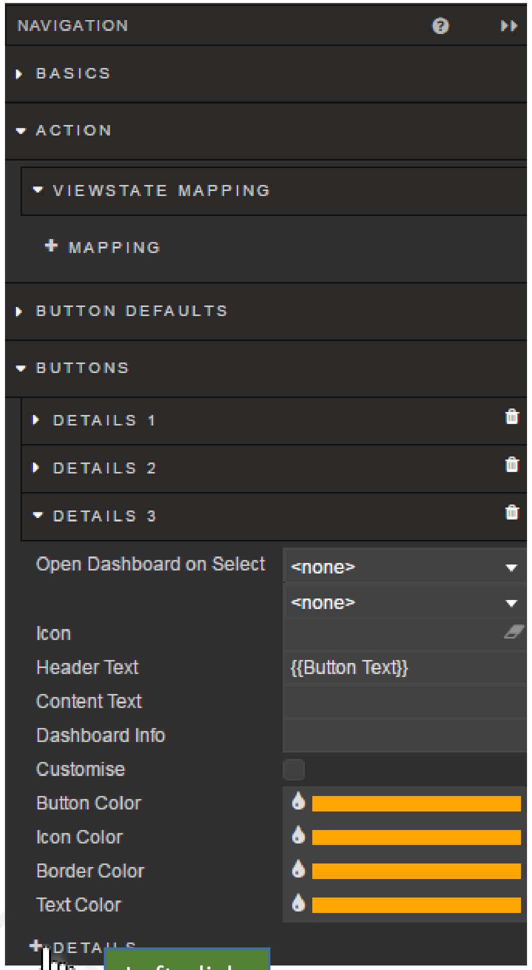


Continued…

# Configure Button Style

Step: 3

Add three Navigation buttons



NAVIGATION

BASICS

ACTION

VIEWSTATE MAPPING

+ MAPPING

BUTTON DEFAULTS

BUTTONS

DETAILS 1

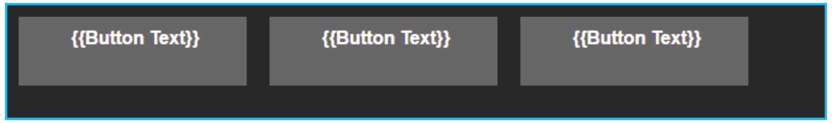
DETAILS 2

DETAILS 3

| | |
|---|---|
| Open Dashboard on Select | <none><br><none> |
| Icon | |
| Header Text | {{Button Text}} |
| Content Text | |
| Dashboard Info | |
| Customise | □ |
| Button Color | |
| Icon Color | |
| Border Color | |
| Text Color | |

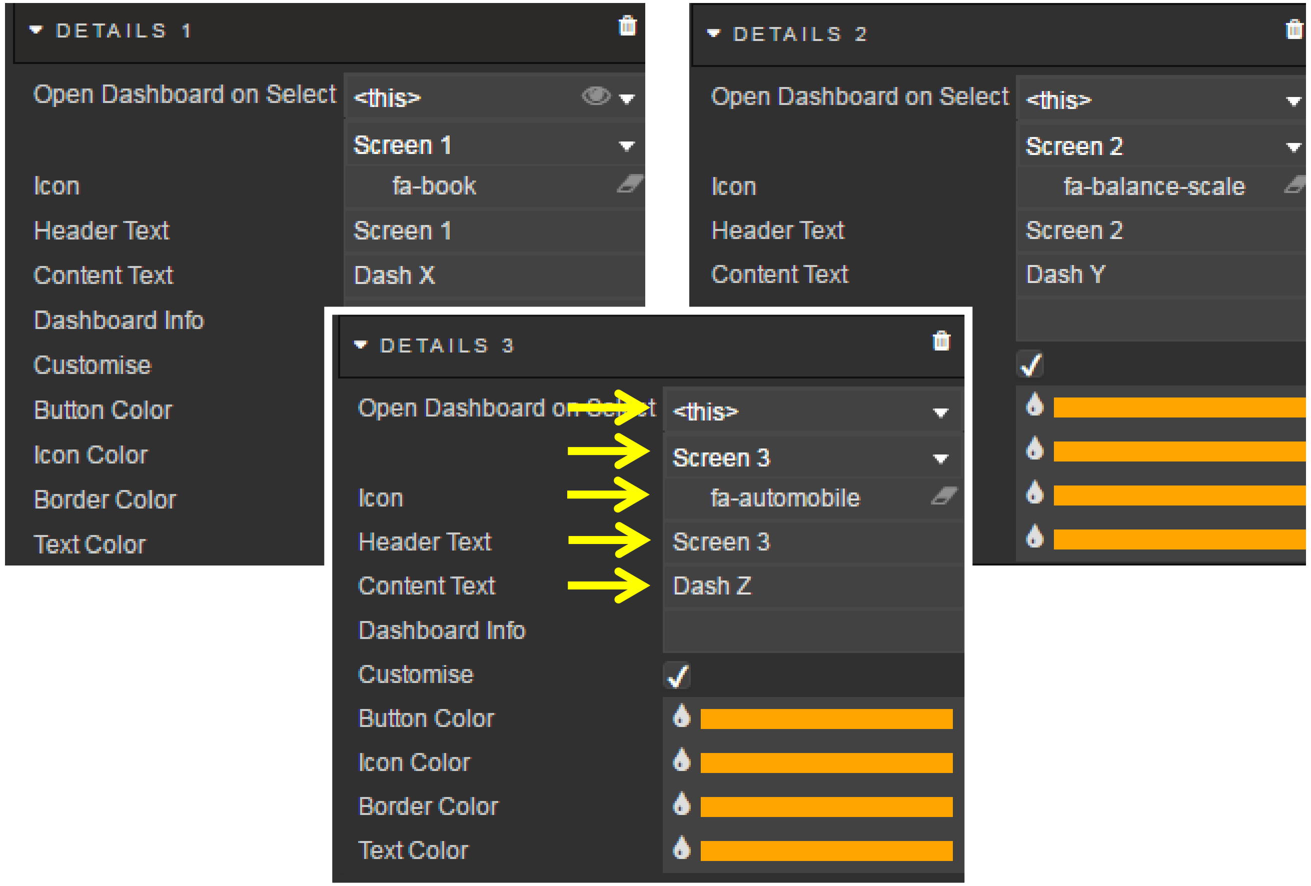
+ DETAILS

Left-click



Step: 4

Configure individual button links



DETAILS 1

| | |
|---|---|
| Open Dashboard on Select | <this><br>Screen 1 |
| Icon | fa-book |
| Header Text | Screen 1 |
| Content Text | Dash X |
| Dashboard Info | |
| Customise | |
| Button Color | |
| Icon Color | |
| Border Color | |
| Text Color | |

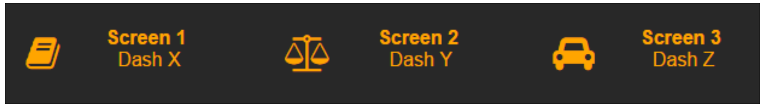
DETAILS 2

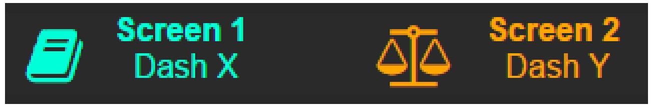
| | |
|---|---|
| Open Dashboard on Select | <this><br>Screen 2 |
| Icon | fa-balance-scale |
| Header Text | Screen 2 |
| Content Text | Dash Y |

DETAILS 3

| | |
|---|---|
| Open Dashboard on Select | <this><br>Screen 3 |
| Icon | fa-automobile |
| Header Text | Screen 3 |
| Content Text | Dash Z |
| Dashboard Info | |
| Customise | ✓ |
| Button Color | |
| Icon Color | |
| Border Color | |
| Text Color | |

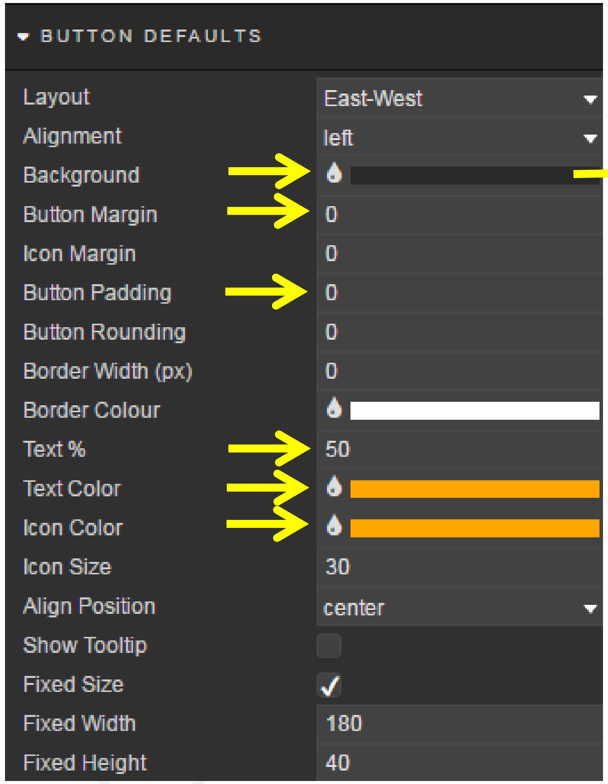
# Global vs Individual Button Styles





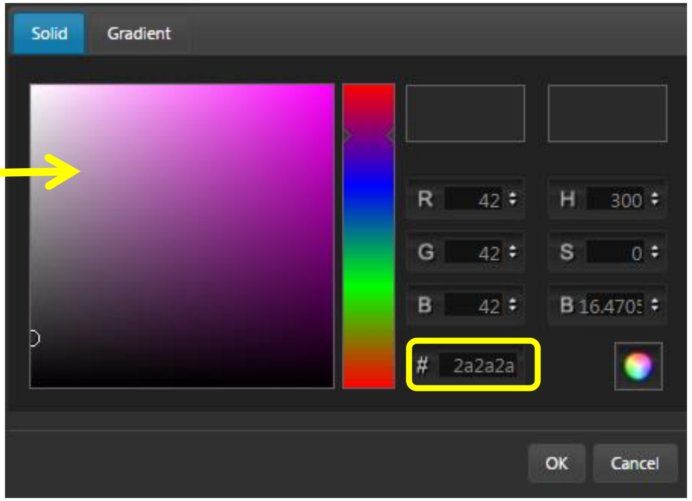
Step: 5

Change macro button styles; applied across all buttons



BUTTON DEFAULTS

| | |
|---|---|
| Layout | East-West |
| Alignment | left |
| Background | ⬛ |
| Button Margin | 0 |
| Icon Margin | 0 |
| Button Padding | 0 |
| Button Rounding | 0 |
| Border Width (px) | 0 |
| Border Colour | ⬜ |
| Text % | 50 |
| Text Color | 🟧 |
| Icon Color | 🟧 |
| Icon Size | 30 |
| Align Position | center |
| Show Tooltip | □ |
| Fixed Size | ✓ |
| Fixed Width | 180 |
| Fixed Height | 40 |

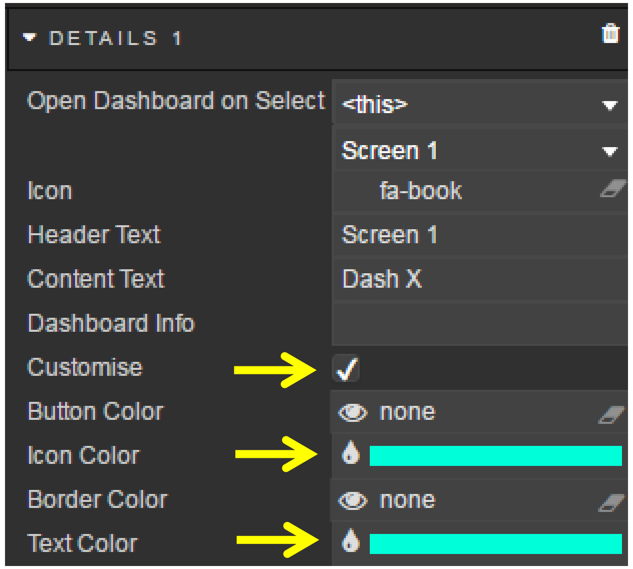


| | | | |
|---|---|---|---|
| R | 42 | H | 300 |
| G | 42 | S | 0 |
| B | 42 | B | 16.4705 |

# 2a2a2a

Step: 6

Style Individual Buttons



DETAILS 1

| | |
|---|---|
| Open Dashboard on Select | <this><br>Screen 1 |
| Icon | fa-book |
| Header Text | Screen 1 |
| Content Text | Dash X |
| Dashboard Info | |
| Customise | ✓ |
| Button Color | none |
| Icon Color | 🟦 |
| Border Color | none |
| Text Color | 🟦 |

Add a Navigation Component to each screen and dashboard

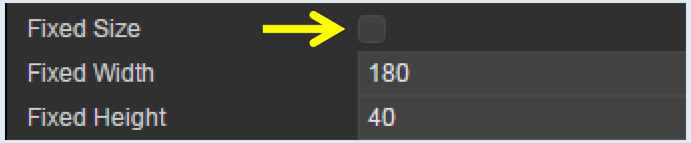
Use the keyboard cut-and-paste shortcut (CTRL+C / CTRL+V) to copy components within and across Dashboard screens.

# Alternatives

## Alternative
Flexible Button Width

Step: 1

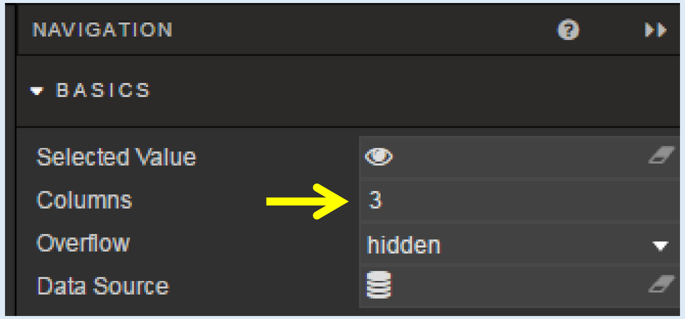
Uncheck *Fixed Size*



Fixed Size □
Fixed Width 180
Fixed Height 40

Step: 2

Set *Columns* to number of buttons to display



NAVIGATION

BASICS

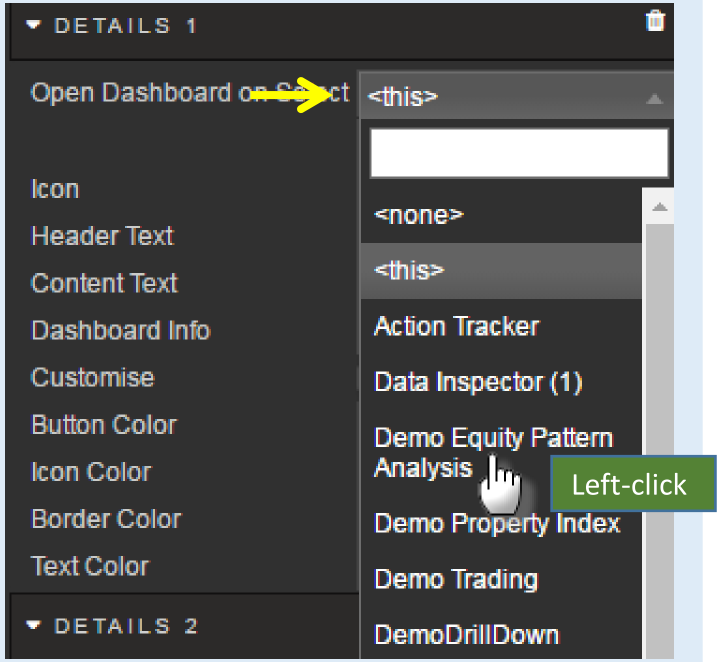
Selected Value
Columns 3
Overflow hidden
Data Source

A *Column* value of ‘1’ will stack buttons

## Alternative
Navigate to other Dashboards

Step: 1

Select from Dropdown



DETAILS 1

Open Dashboard on Select <this>

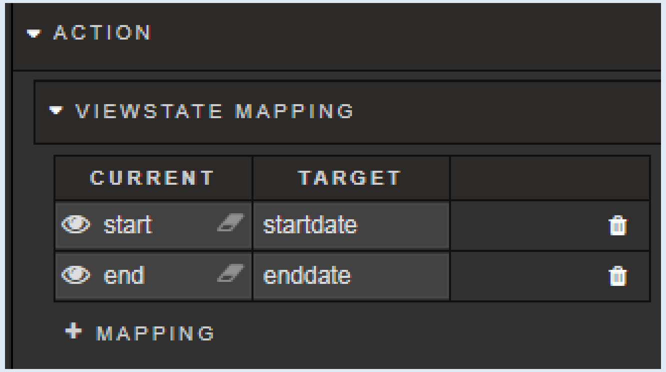
Icon
Header Text
Content Text
Dashboard Info
Customise
Button Color
Icon Color
Border Color
Text Color

DETAILS 2

Left-click

## Alternative
Share View States Across Dashboards

Match dashboard *viewstates* to destination dashboard *viewstates*



ACTION

VIEWSTATE MAPPING

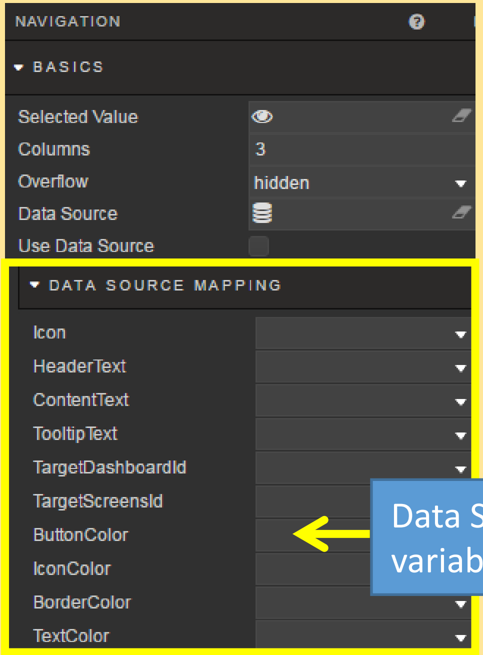
| CURRENT | TARGET |
|---|---|
| start | startdate |
| end | enddate |

+ MAPPING

⚠ It’s necessary to create in the destination dashboard *viewstates* which match the name of the *Target*

- Create a Navigation Panel using a **Data Source**, and map a View State so a value from one dashboard is displayed in another.

There is no example in Html5evalcongroup which has a Navigation panel. This may require a local kdb connection for the **Data Source**



NAVIGATION

BASICS

Selected Value
Columns: 3
Overflow: hidden
Data Source
Use Data Source

DATA SOURCE MAPPING

Icon
HeaderText
ContentText
TooltipText
TargetDashboardId
TargetScreensId
ButtonColor
IconColor
BorderColor
TextColor

Data Source should include variables for items listed

kx | it's about time

# Tabs, Accordions & Layout Panels

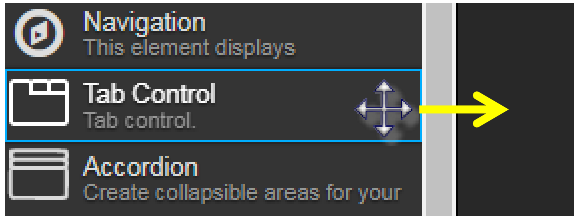
## Dashboards for Kx – "How to" Guide

Tabs, Accordion and Layout Panel allow for additional functionality and space inside a single Dashboard screen

# Adding a Tab

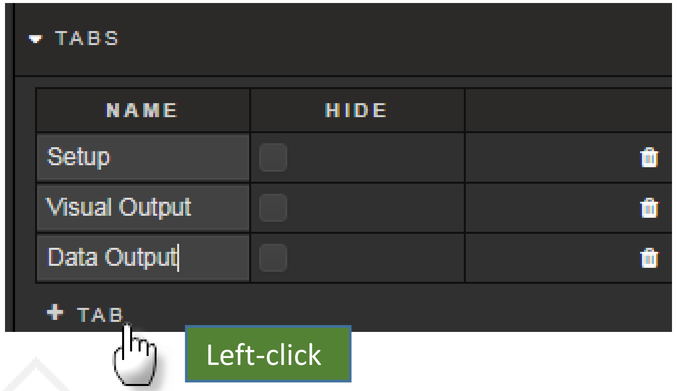
Step: 1

Drag the Tab component into the dashboard



Step: 2

Add a new tab, and rename existing tabs + TAB

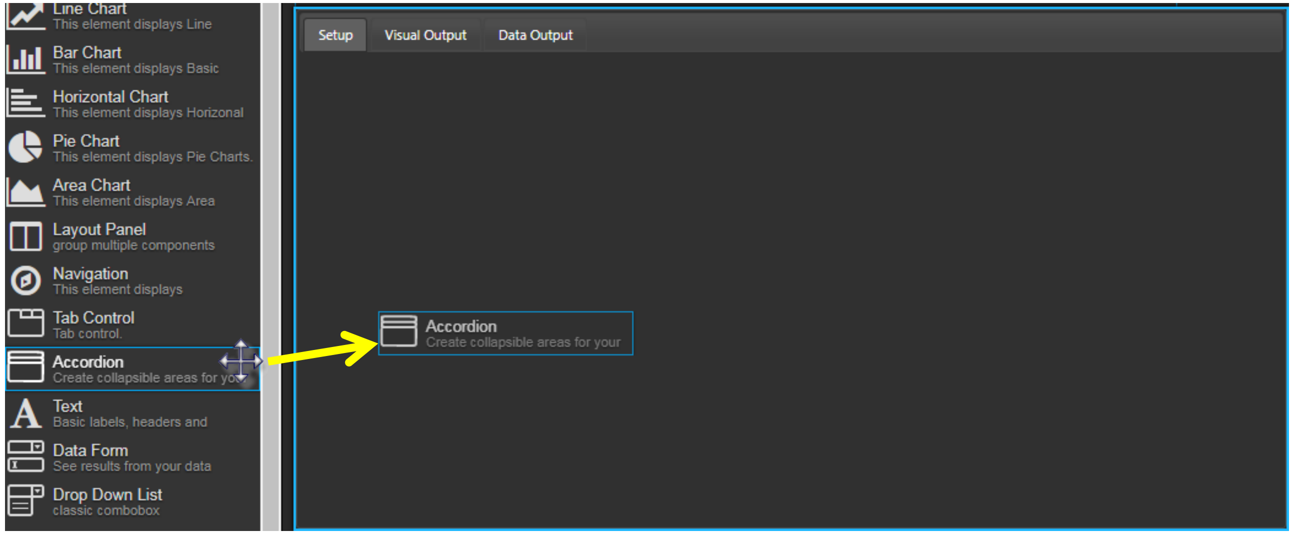


| NAME | HIDE | |
|---|---|---|
| Setup | □ | |
| Visual Output | □ | |
| Data Output | □ | |

Left-click

Step: 3

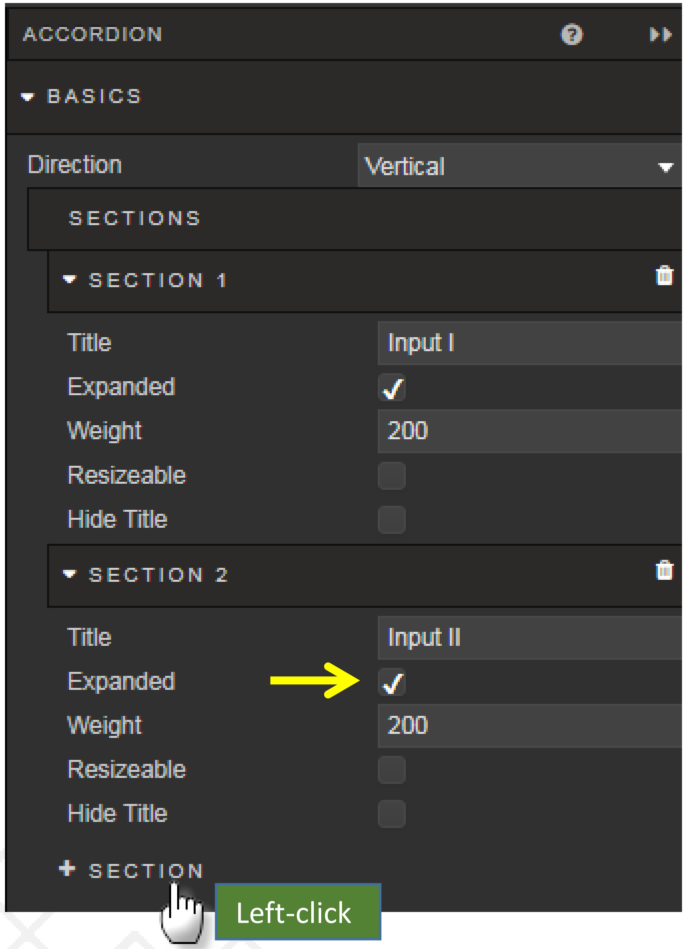
Select the first tab ("Setup") and drag inside the Accordion component


Line Chart
This element displays Line
Bar Chart
This element displays Basic
Horizontal Chart
This element displays Horizonal
Pie Chart
This element displays Pie Charts.
Area Chart
This element displays Area
Layout Panel
group multiple components
Navigation
This element displays
Tab Control
Tab control.
Accordion
Create collapsible areas for yo
Text
Basic labels, headers and
Data Form
See results from your data
Drop Down List
classic combobox
Setup
Visual Output
Data Output
Accordion
Create collapsible areas for your

# Adding an Accordion inside a Tab

Step: 4

Add a second section to the Accordion panel: ✚ SECTION



ACCORDION

BASICS

Direction: Vertical

SECTIONS

SECTION 1

Title: Input I

Expanded: ✓

Weight: 200

Resizeable: □

Hide Title: □

SECTION 2

Title: Input II

Expanded: ✓

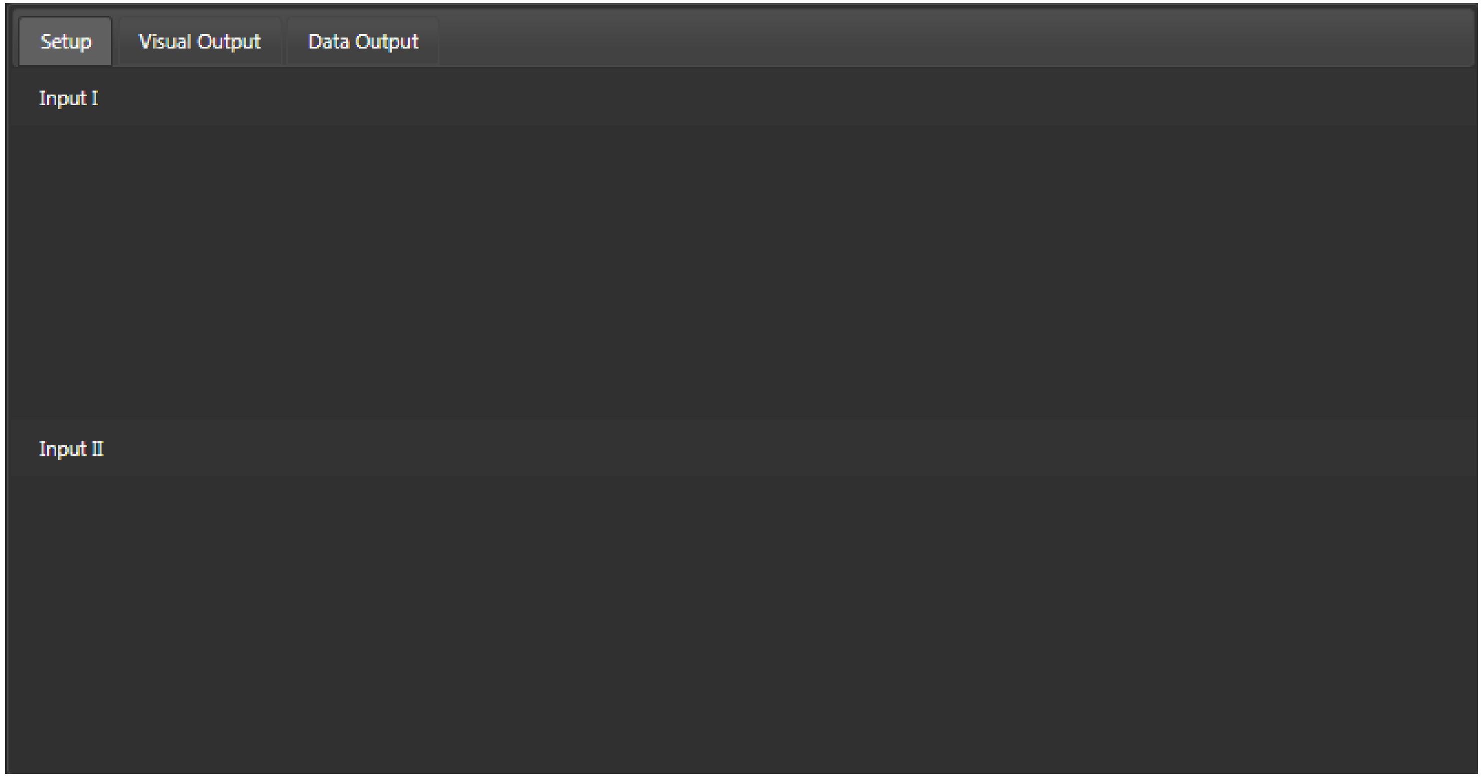
Weight: 200

Resizeable: □

Hide Title: □

✚ SECTION

Left-click

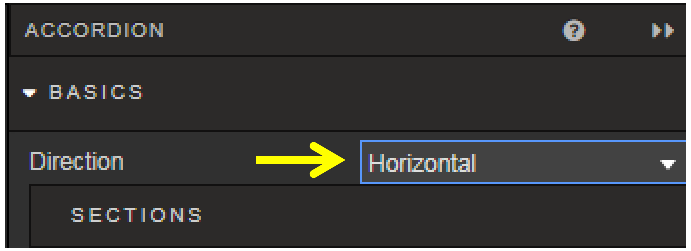


When working with Accordion sections, it's best to keep them in expanded mode so layout panel or components can be added to them. Expanded ✓

# Adding an Layout Panel inside an Accordion section

Step: 5

Switch Accordion view to Horizontal for vertical controlled accordion slider

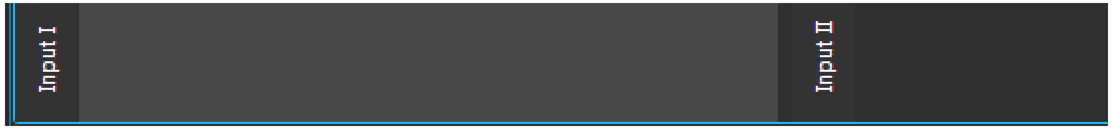


ACCORDION

BASICS

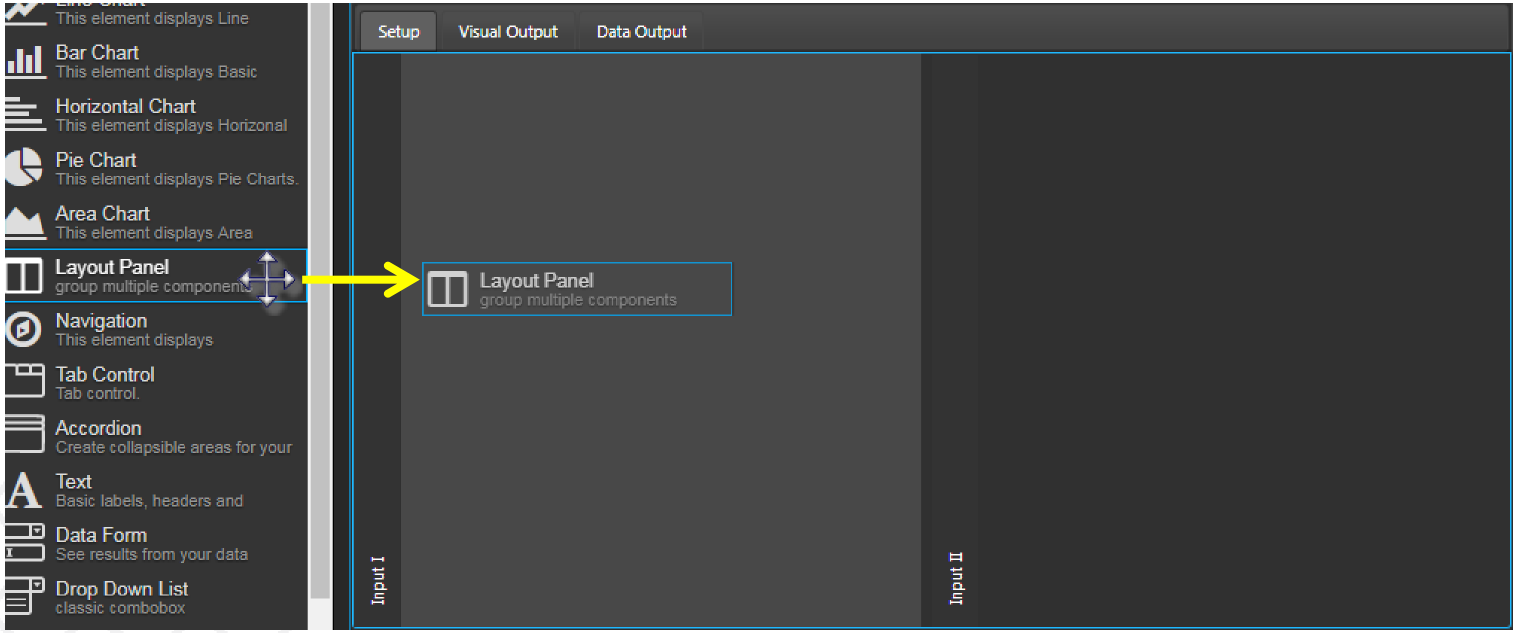
Direction: Horizontal

SECTIONS



Step: 6

Add a Layout panel to each Accordion section

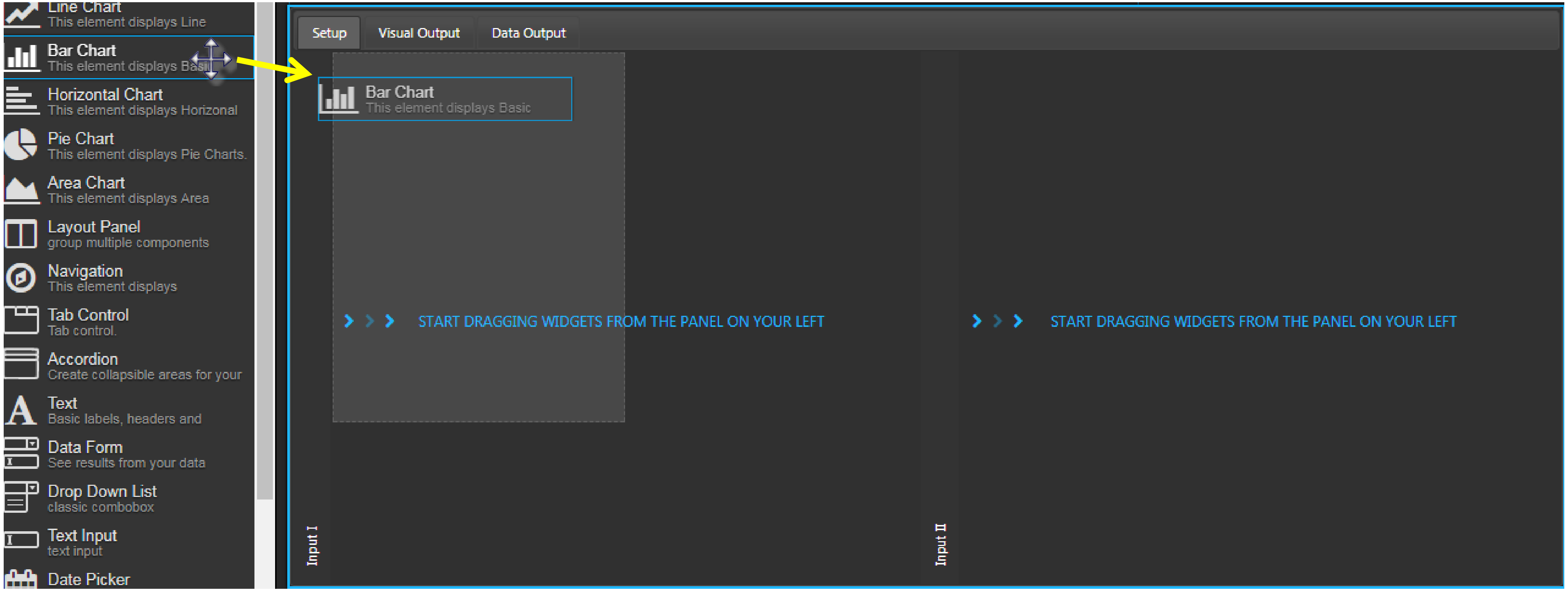


This element displays Line
Bar Chart
This element displays Basic
Horizontal Chart
This element displays Horizonal
Pie Chart
This element displays Pie Charts.
Area Chart
This element displays Area
Layout Panel
group multiple components
Navigation
This element displays
Tab Control
Tab control.
Accordion
Create collapsible areas for your
Text
Basic labels, headers and
Data Form
See results from your data
Drop Down List
classic combobox
Setup
Visual Output
Data Output
Layout Panel
group multiple components
Input I
Input II

# Add components inside the Layout Panel

Step: 7

Add components to your Layout section


Line Chart
This element displays Line
Bar Chart
This element displays Basi
Horizontal Chart
This element displays Horizonal
Pie Chart
This element displays Pie Charts.
Area Chart
This element displays Area
Layout Panel
group multiple components
Navigation
This element displays
Tab Control
Tab control.
Accordion
Create collapsible areas for your
Text
Basic labels, headers and
Data Form
See results from your data
Drop Down List
classic combobox
Text Input
text input
Date Picker
Setup
Visual Output
Data Output
Bar Chart
This element displays Basic
START DRAGGING WIDGETS FROM THE PANEL ON YOUR LEFT
START DRAGGING WIDGETS FROM THE PANEL ON YOUR LEFT
Input I
Input II

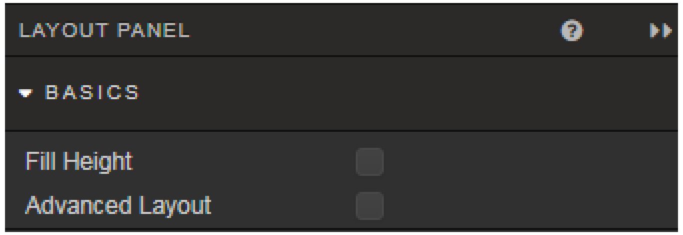
tip!

No Layout Panel is required if a Tab or Accordion section is only to house one component; simply drag-and-drop the component inside the Tab or Accordion section and the component will automatically resize to fill the Tab or Accordion panel.

# Have added components fill out layout panel

Step: 8

In Layout Panel, check *Fill Height* for components to fill the space of the Layout panel

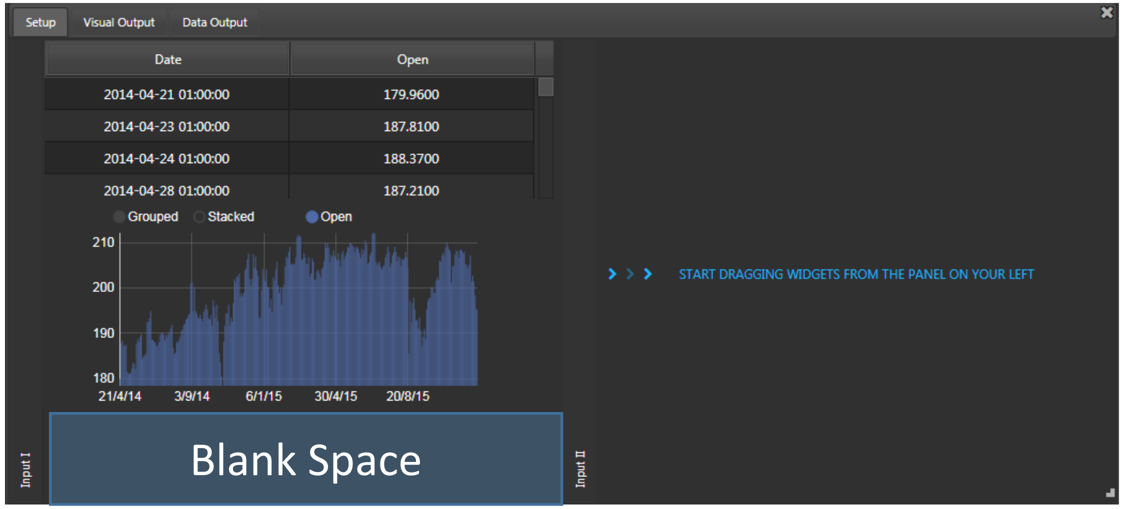


LAYOUT PANEL

BASICS

Fill Height ☐

Advanced Layout ☐

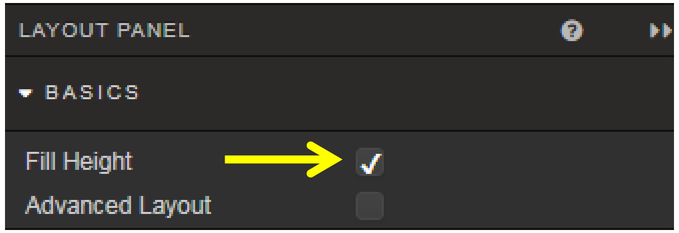


| Date | Open |
|---|---|
| 2014-04-21 01:00:00 | 179.9600 |
| 2014-04-23 01:00:00 | 187.8100 |
| 2014-04-24 01:00:00 | 188.3700 |
| 2014-04-28 01:00:00 | 187.2100 |

Blank Space

Alternative

*Advanced Layout* will stack components; good for dashboards configured for mobile use

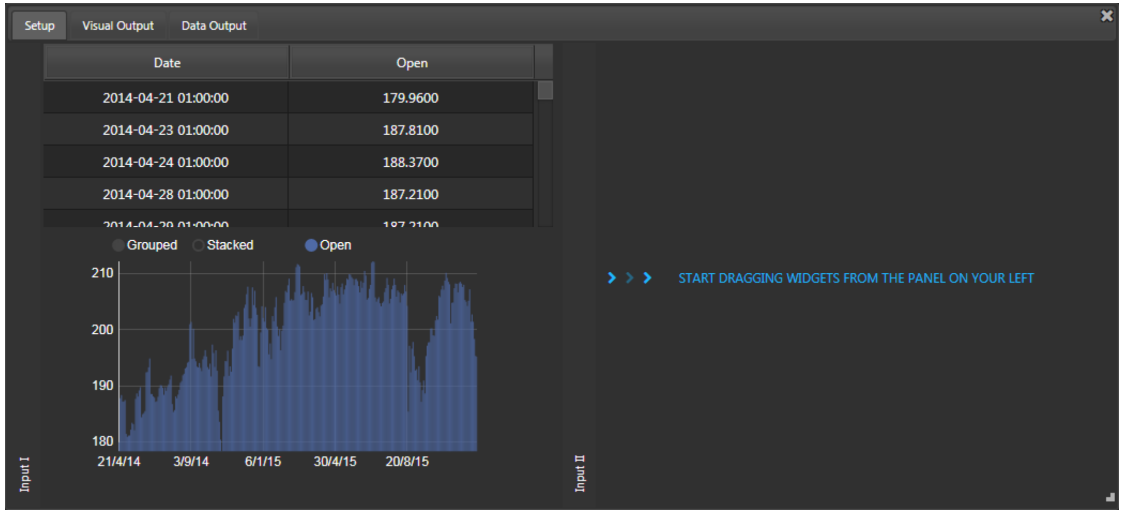


LAYOUT PANEL

BASICS

Fill Height ☑

Advanced Layout ☐



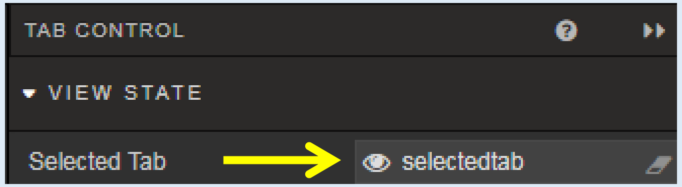
| Date | Open |
|---|---|
| 2014-04-21 01:00:00 | 179.9600 |
| 2014-04-23 01:00:00 | 187.8100 |
| 2014-04-24 01:00:00 | 188.3700 |
| 2014-04-28 01:00:00 | 187.2100 |

Once checked, it can be difficult to re-select the layout panel to make changes (1 px width selection area). Therefore, check this box as a last step when building dashboards

# Tab and Accordion Alternatives

## Alternative
*Viewstate* shared control of Tabs

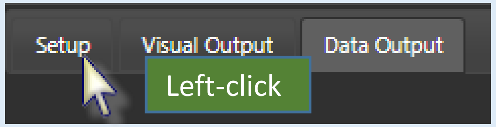
Step: 1 – Assign a *viewstate* to Selected Tab


TAB CONTROL
VIEW STATE
Selected Tab: selectedtab

Step: 2 – Add a Second Tab Component

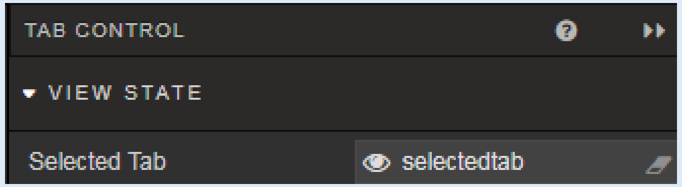
Step: 3 – Assign the second tab component *Selected Tab* the same *viewstate* selectedtab

Step: 4 – Switch between tabs of either Tab component

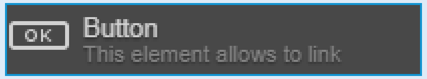


## Alternative
Button control to open a particular Tab

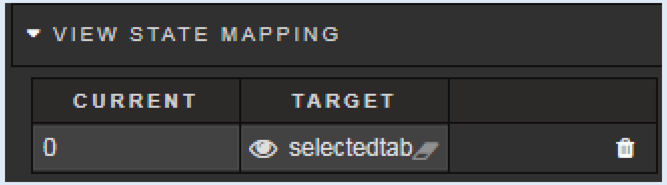
Step: 1 – Assign a *viewstate* to Selected Tab


TAB CONTROL
VIEW STATE
Selected Tab: selectedtab

Step: 2 – Add a Button Component anywhere to the dashboard



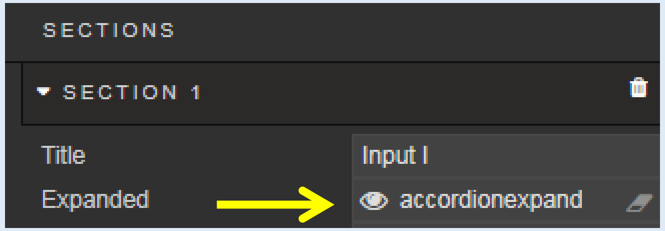
Step: 3 – Set Button View State Mapping; ‘0’ = Tab 1


VIEW STATE MAPPING

| CURRENT | TARGET |
|---|---|
| 0 | selectedtab |

## Alternative
Accordion Expand control bound to *Viewstate*

Step: 1 – Assign a *viewstate* to Accordion expand


SECTIONS
SECTION 1
Title: Input I
Expanded: accordionexpand

Step: 2 – Add a second Accordion

Step: 3 – In the second Accordion, assign *Expanded* the viewstate accordionexpand

Step: 4 – Expand paired section

A Button can also be used to control expansion. In Button *View State Mapping* set *Target* to accordionexpand and *Current* to “True”

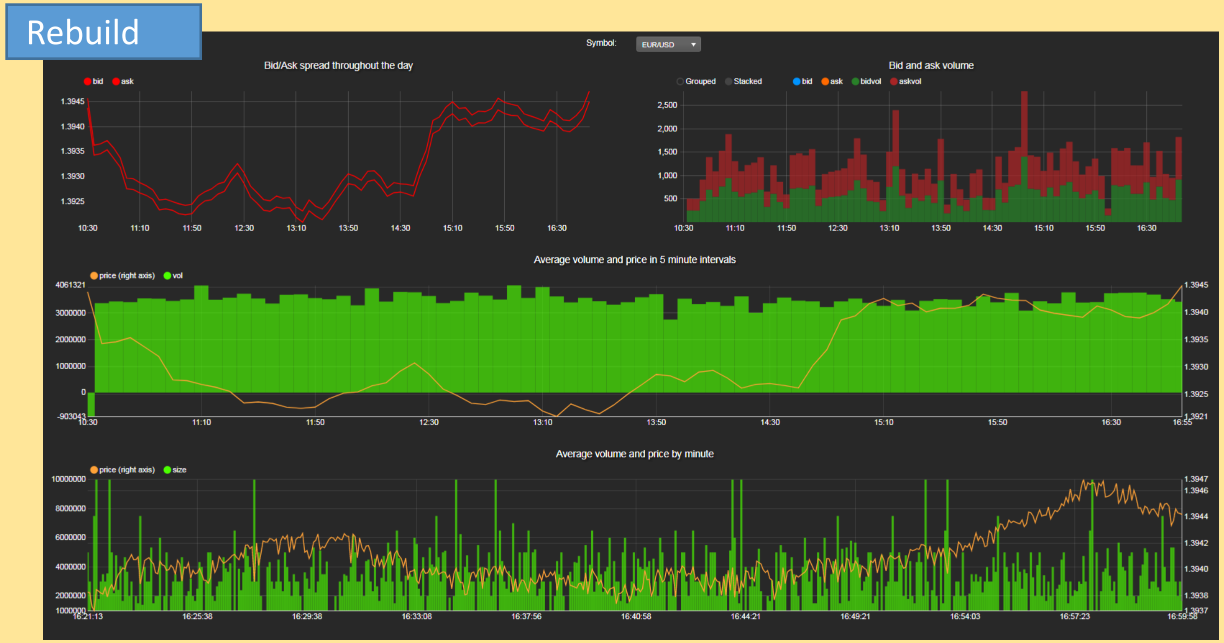
it's about time

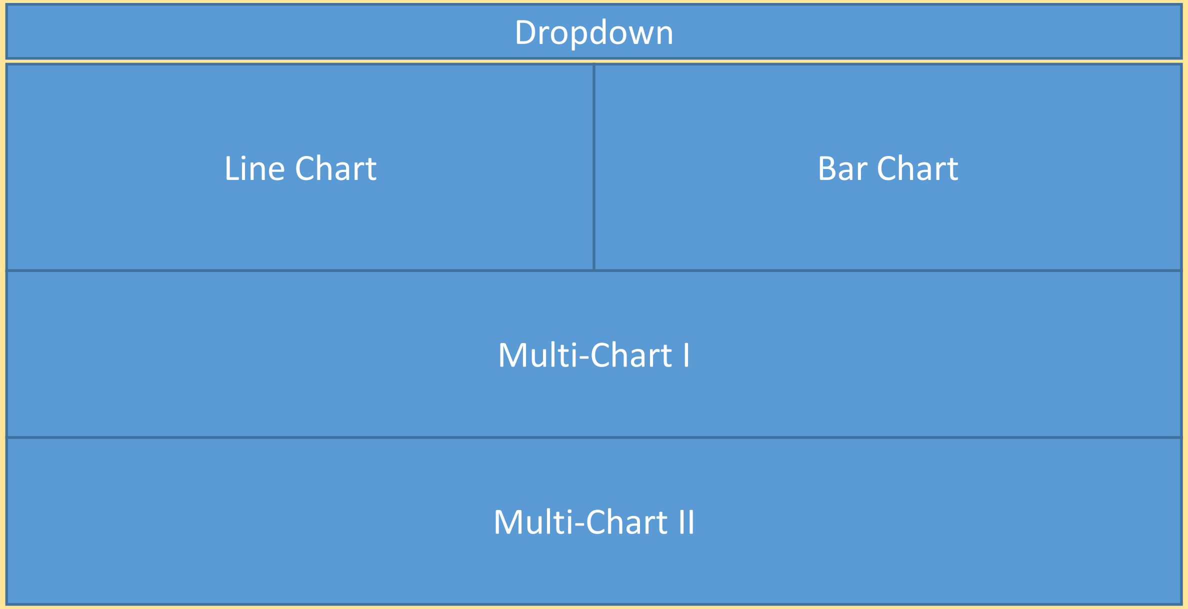
# Bring it Together

## Dashboards for Kx – "How to" Guide

# Try This: DemoLiquidity

Rebuild





Dropdown

Line Chart

Bar Chart

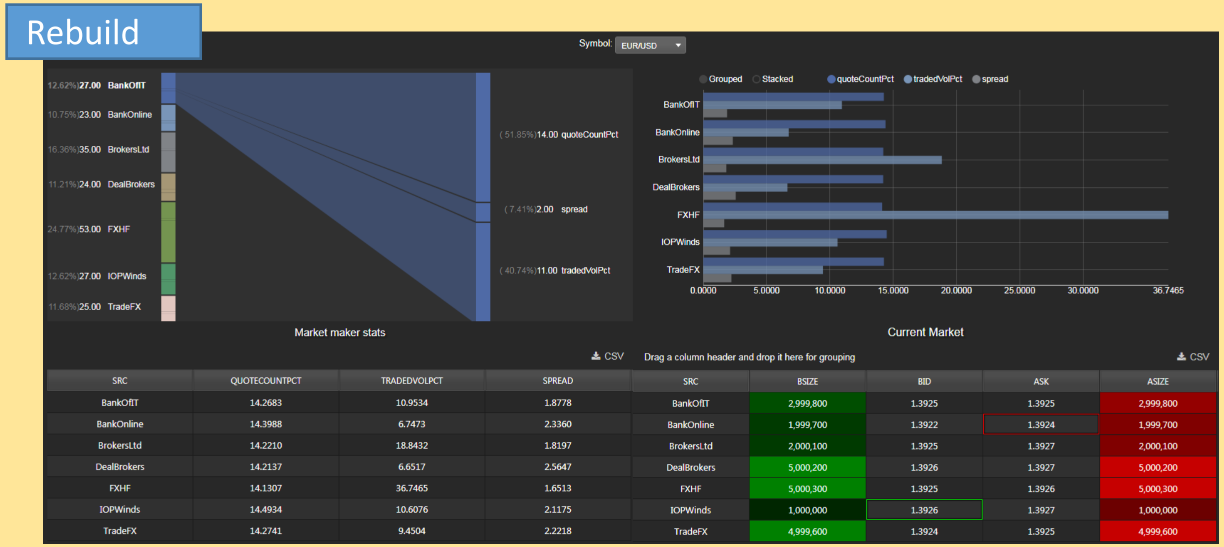
Multi-Chart I

Multi-Chart II

| Component | All connect to *html5evalcongroup* |
|---|---|
| Dropdown | ([]sym:asc exec distinct sym from dfxTrade) |
| Multi-Chart | {[symval]  `minute xasc select avg bid,avg ask, bidvol:max(0;sum bsize)%1e06, askvol:max(0;sum asize)%1e06 by 5 xbar time.minute from dfxQuote where sym=symval} |
| Bar Chart | |
| Multi-Chart I | {[symval] `minute xasc select price:avg bid,vol:(avg bsize) by 5 xbar time.minute from dfxQuote where sym=symval} |
| Multi-Chart II | {[symval;side] -500 sublist $[side=`bid;select price:avg bid,size:avg bsize by time.second from dfxQuote where sym=symval;select price:avg ask,size:avg asize by time.minute from dfxQuote where sym=symval]} |

# Try This: DemoMarketMakers

Rebuild



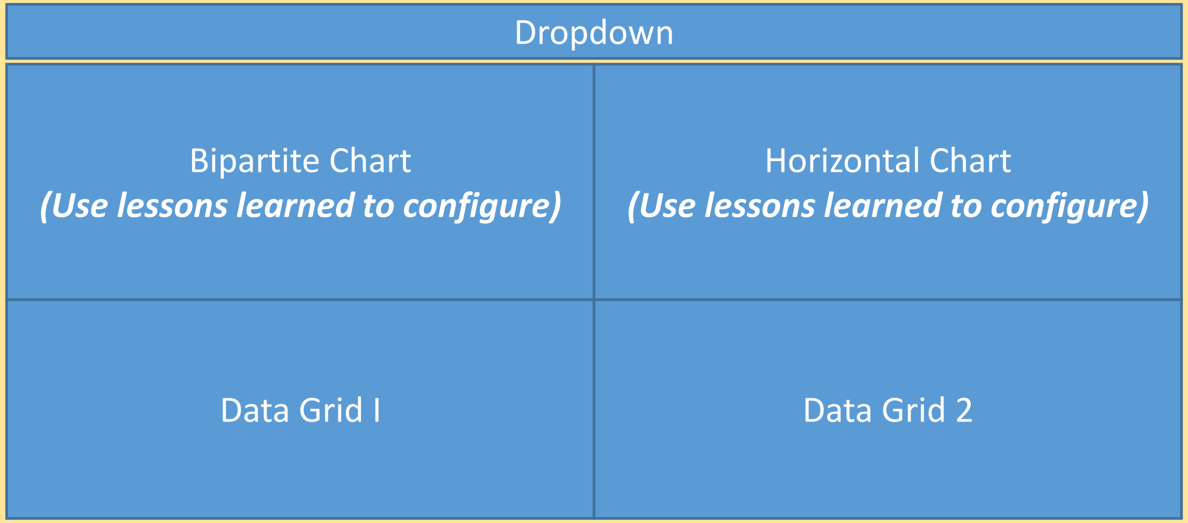
Market maker stats

| SRC | QUOTECOUNTPCT | TRADEDVOLPCT | SPREAD |
|---|---|---|---|
| BankOfIT | 14.2683 | 10.9534 | 1.8778 |
| BankOnline | 14.3988 | 6.7473 | 2.3360 |
| BrokersLtd | 14.2210 | 18.8432 | 1.8197 |
| DealBrokers | 14.2137 | 6.6517 | 2.5647 |
| FXHF | 14.1307 | 36.7465 | 1.6513 |
| IOPWinds | 14.4934 | 10.6076 | 2.1175 |
| TradeFX | 14.2741 | 9.4504 | 2.2218 |

Current Market

Drag a column header and drop it here for grouping

| SRC | BSIZE | BID | ASK | ASIZE |
|---|---|---|---|---|
| BankOfIT | 2,999,800 | 1.3925 | 1.3925 | 2,999,800 |
| BankOnline | 1,999,700 | 1.3922 | 1.3924 | 1,999,700 |
| BrokersLtd | 2,000,100 | 1.3925 | 1.3927 | 2,000,100 |
| DealBrokers | 5,000,200 | 1.3926 | 1.3927 | 5,000,200 |
| FXHF | 5,000,300 | 1.3925 | 1.3926 | 5,000,300 |
| IOPWinds | 1,000,000 | 1.3926 | 1.3927 | 1,000,000 |
| TradeFX | 4,999,600 | 1.3924 | 1.3925 | 4,999,600 |



Dropdown

Bipartite Chart
*(Use lessons learned to configure)*

Horizontal Chart
*(Use lessons learned to configure)*

Data Grid I

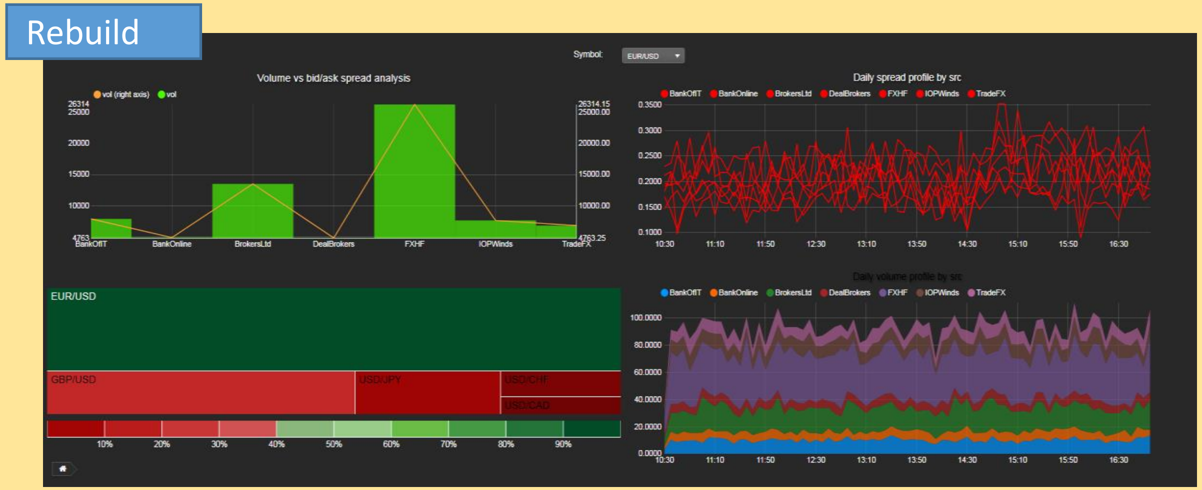
Data Grid 2

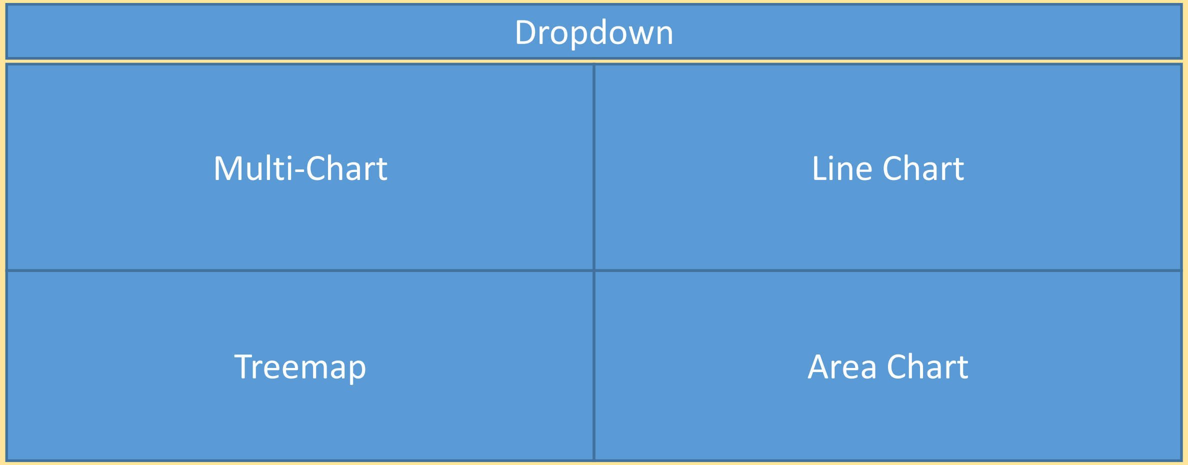
| Component | All connect to *html5evalcongroup* |
|---|---|
| Dropdown | ([]sym:asc exec distinct sym from dfxTrade) |
| Bipartite Chart | {[symval] select src,quoteCountPct:100*numQuotes%sum numQuotes, tradedVolPct:100*size%sum size,spread from  t:(select numQuotes:sum i,size:sum "f"$size by src from dfxTrade where sym=symval) lj select spread:10000*avg (ask-bid) by src from dfxQuote where sym=symval} |
| Horizontal Chart | |
| Data Grid I | |
| Data Grid II* | {[symval] `src xasc select last bsize,last bid,last ask,last asize by src from dfxRandomQuote where sym=symval} |

* Poll data at 1 second intervals

# Try This: DemoMarketShare

Rebuild





Dropdown

Multi-Chart

Line Chart

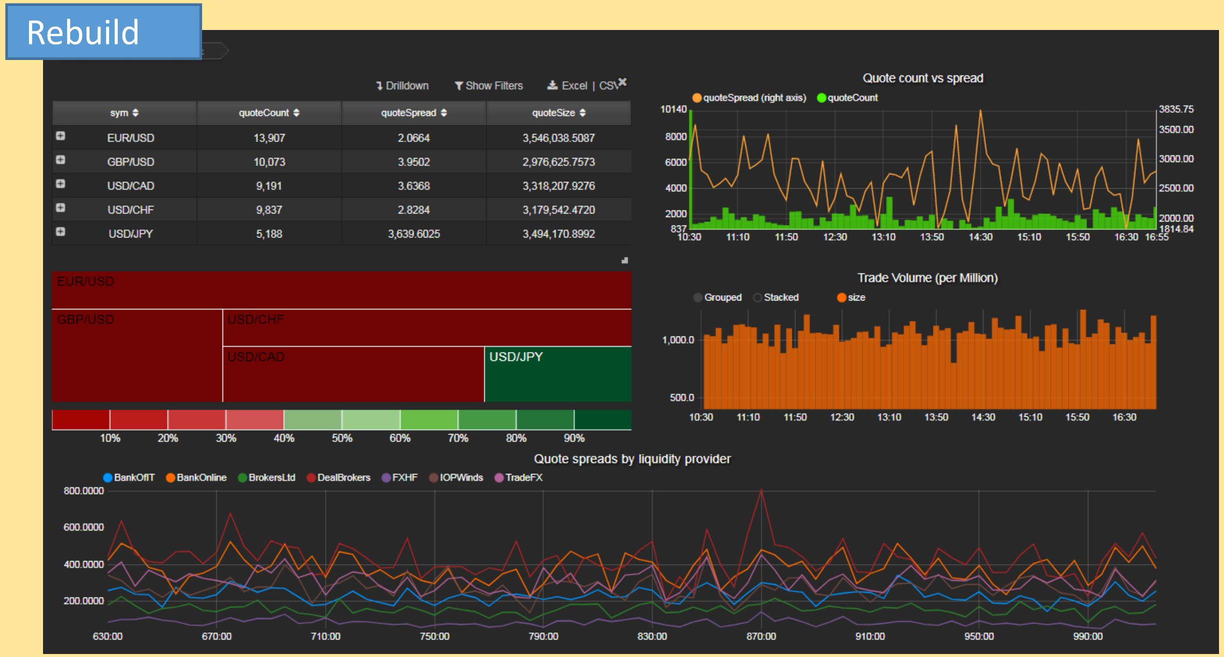
Treemap

Area Chart

| **Component** | **All connect to *html5evalcongroup*** |
|---|---|
| Dropdown | ([]sym:`All,asc exec distinct sym from dfxTrade) |
| Multi-Chart | {[symval] select vol:sum (`float$size)%1e6,avg price,avg spread by src from $[symval=`All;dfxTrade;select from dfxTrade where sym=symval] lj select spread:1000*avg(ask-bid) by sym,src from dfxQuote} |
| Line Chart | {[symval] `minute xasc (select distinct minute from t) pj/ {[t;x] ?[select from t where src=x;();(enlist `minute)!enlist `minute;(enlist x)!enlist (last;`spread)]}[t;] each exec distinct src from t:select spread:1000*avg (ask-bid) by src,5 xbar time.minute from $[symval=`All;dfxQuote;select from dfxQuote where sym=symval]} |
| Treemap | dfxTrade  (Breakdown: sym and src; avg-> size and avg -> price) |
| Area Chart | {[symval] `minute xasc (select distinct minute from t) pj/ {[t;x] ?[select from t where src=x;();(enlist `minute)!enlist `minute;(enlist x)!enlist (last;`vol)]}[t;] each exec distinct src from t:select vol:sum (`float$size)%10e6 by src,5 xbar time.minute from $[symval=`All;dfxTrade;select from dfxTrade where sym=symval]} |

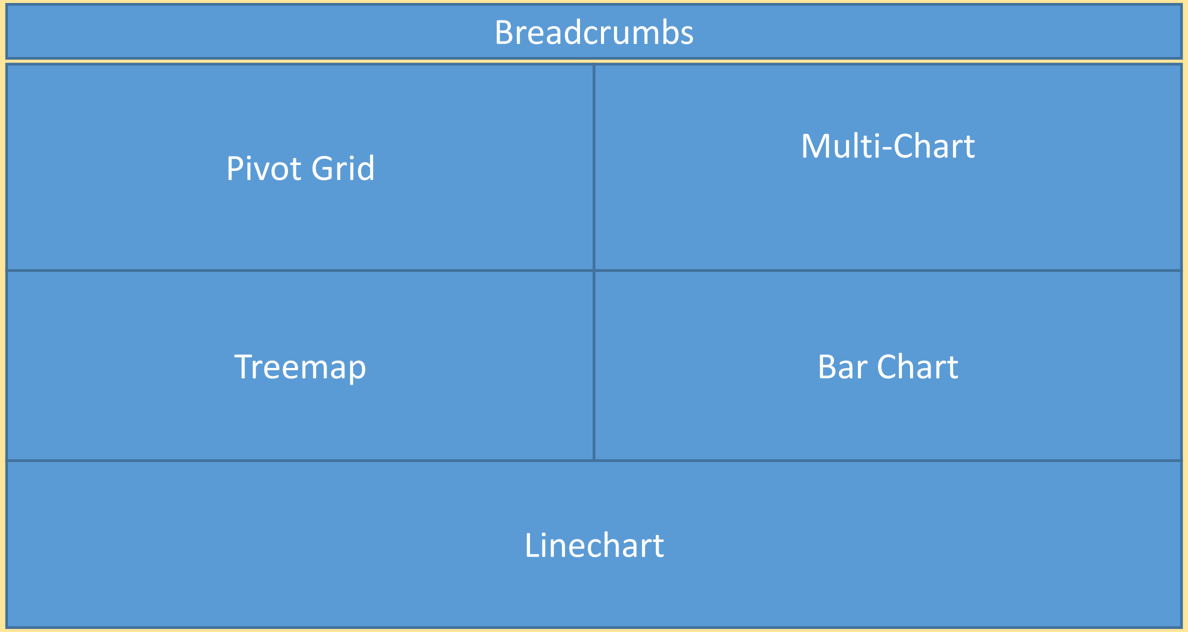
# Try This: DemoDrillDown

Rebuild



↧ Drilldown  ▼ Show Filters  ⤓ Excel | CSV

| sym | quoteCount | quoteSpread | quoteSize |
|---|---|---|---|
| ⊞ EUR/USD | 13,907 | 2.0664 | 3,546,038.5087 |
| ⊞ GBP/USD | 10,073 | 3.9502 | 2,976,625.7573 |
| ⊞ USD/CAD | 9,191 | 3.6368 | 3,318,207.9276 |
| ⊞ USD/CHF | 9,837 | 2.8284 | 3,179,542.4720 |
| ⊞ USD/JPY | 5,188 | 3,639.6025 | 3,494,170.8992 |



Breadcrumbs

Pivot Grid

Multi-Chart

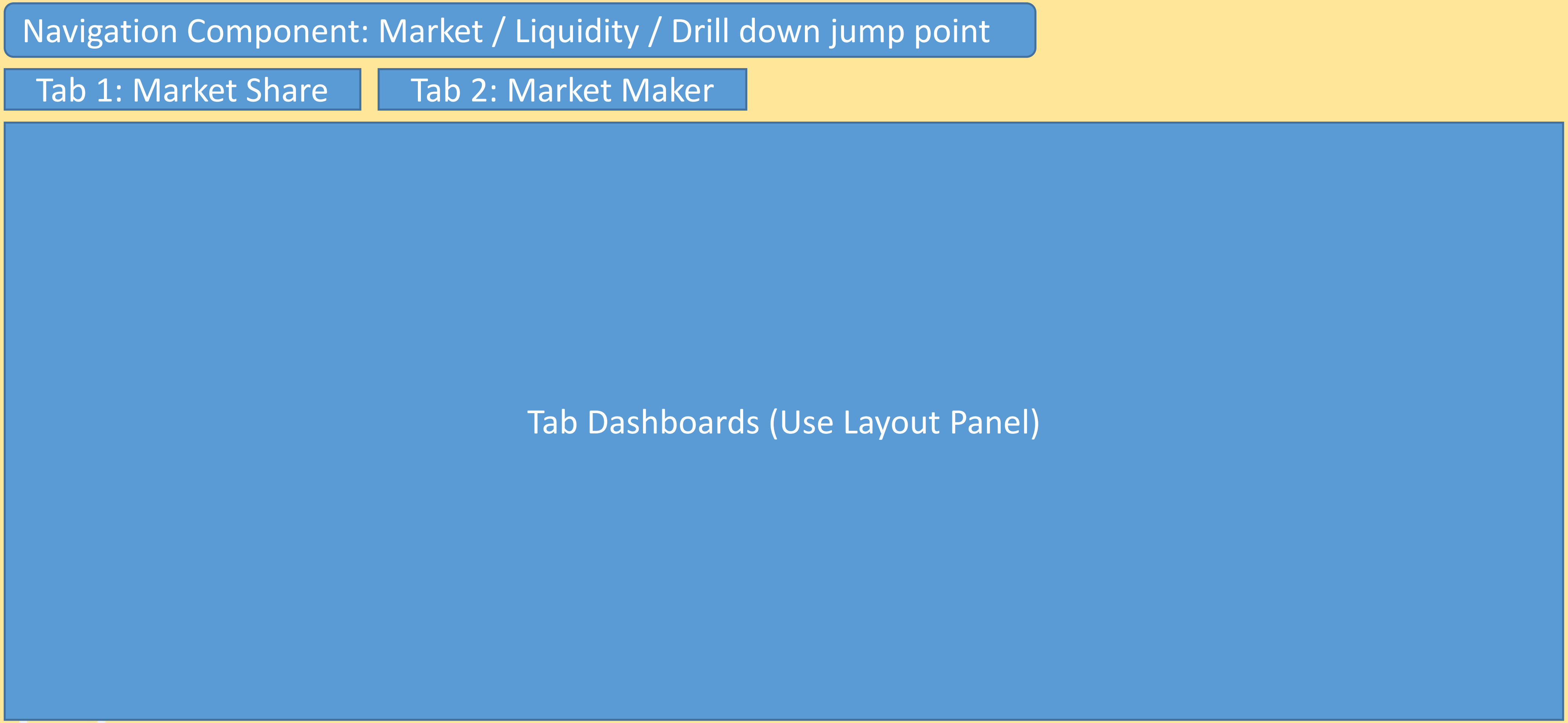
Treemap

Bar Chart

Linechart

| Component | All connect to *html5evalcongroup* |
|---|---|
| Pivot Grid | {`sym`src`hour`minute xcols 0!select quoteCount:count i,quoteSpread:10000*avg (ask-bid),quoteSize:avg (bsize+asize)%2 by hour:`$string time.hh,minute:`$string 10 xbar time.minute, sym,src from dfxQuote where sym in exec distinct sym from dfxTrade}[] |
| Multi-Chart | {[filters] filters:raze `$"," vs string .c.f:filters; :.c.res:select quoteCount:count i,quoteSpread:10000*avg (ask-bid),quoteSize:avg (bsize+asize)%2e6 by 5 xbar time.minute from $[0=count filters;dfxQuote;1=count filters;$[null first filters;dfxQuote;select from dfxQuote where sym=first filters];2=count filters;select from dfxQuote where sym=filters[0],time.hh="I"$string filters[1];3=count filters;select from dfxQuote where sym=filters[0],time.hh="I"$string filters[1],src=filters[2];()]} |
| Horizontal Chart | {[filters] filters:raze `$"," vs string .c.ttf:filters; :select avg price,sum size%1e6 by 5 xbar time.minute from .c.t:$[0=count filters;dfxTrade;1=count filters;$[null first filters;dfxTrade;select from dfxTrade where sym=first filters];2=count filters;select from dfxTrade where sym=filters[0],time.hh="I"$string filters[1];3=count filters;select from dfxTrade where sym=filters[0],time.hh="I"$string filters[1],src=filters[2];()]} |
| Treemap | {`sym`src`hour`minute xcols 0!select quoteCount:count i,quoteSpread:10000*avg (ask-bid),quoteSize:avg (bsize+asize)%2 by hour:`$string time.hh,minute:`$string 10 xbar time.minute, sym,src from dfxQuote where sym in exec distinct sym from dfxTrade}[] |
| Data Grid II* | {[symval] `src xasc select last bsize,last bid,last ask,last asize by src from dfxRandomQuote where sym=symval} |

# Try This: Multi-screen & Tab



Navigation Component: Market / Liquidity / Drill down jump point

Tab 1: Market Share

Tab 2: Market Maker

Tab Dashboards (Use Layout Panel)

it's about time

# Demo POC presentation

## Dashboards for Kx – “How to” Guide

kx

it’s about time

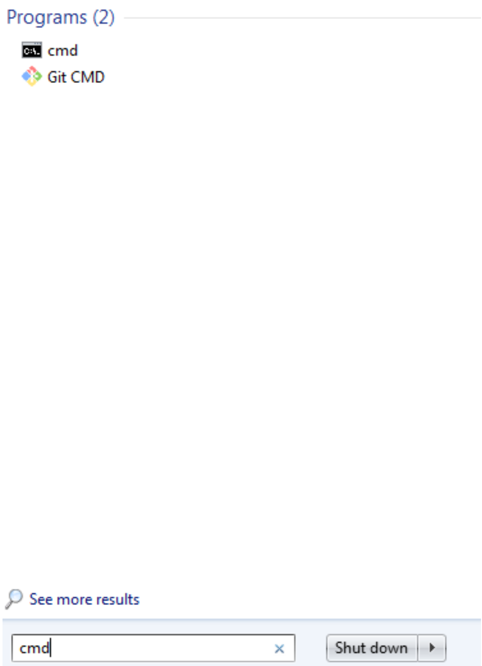
kx | it's about time

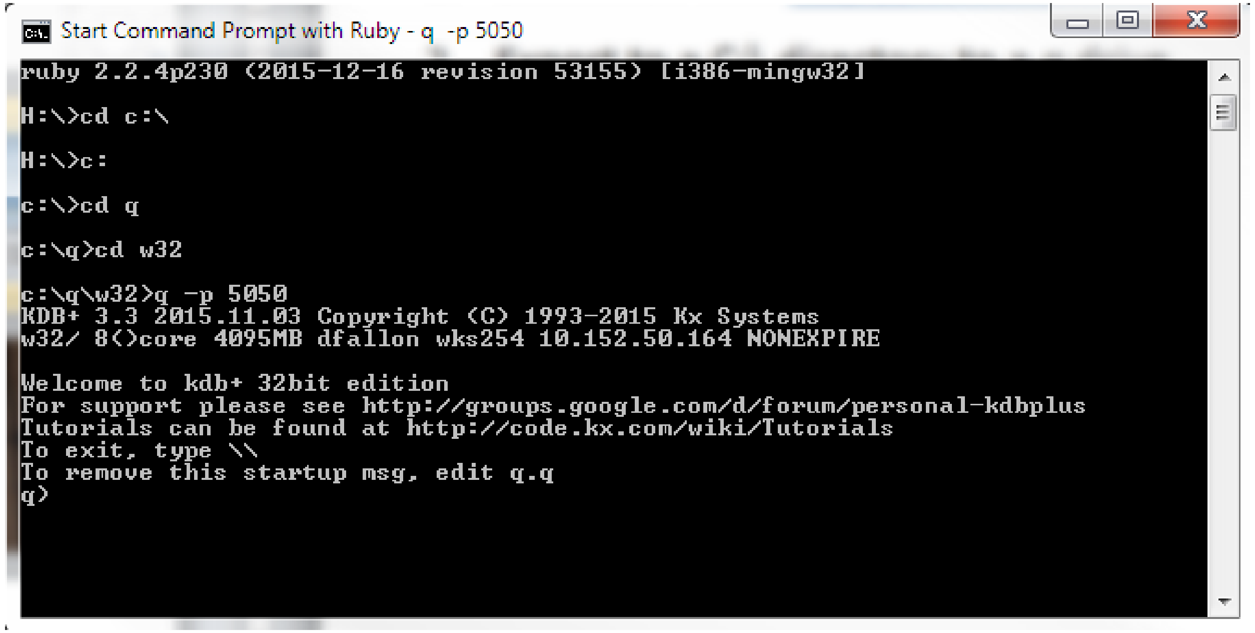
# q install

## Dashboards for Kx – "How to" Guide

# Install local kdb

1. Download kdb: [http://kx.com/software.php](http://kx.com/software.php)
2. Export to a C:\ directory to a q drive
3. Open CMD Prompt
    - cd c:\
    - Cd q
    - Cd w32
    - q –p 5050

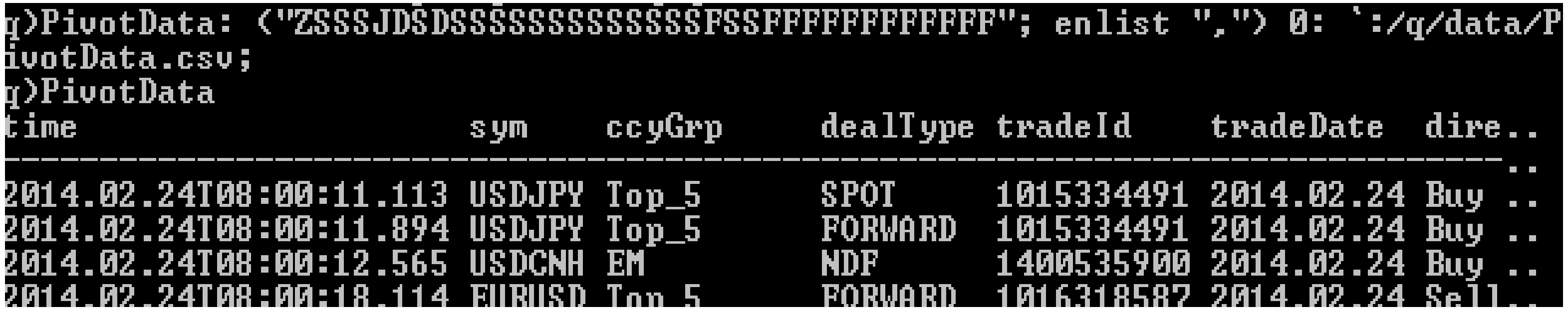




```
ruby 2.2.4p230 (2015-12-16 revision 53155) [i386-mingw32]

H:\>cd c:\

H:\>c:

c:\>cd q

c:\q>cd w32

c:\q\w32>q -p 5050
KDB+ 3.3 2015.11.03 Copyright (C) 1993-2015 Kx Systems
w32/ 8()core 4095MB dfallon wks254 10.152.50.164 NONEXPIRE

Welcome to kdb+ 32bit edition
For support please see http://groups.google.com/d/forum/personal-kdbplus
Tutorials can be found at http://code.kx.com/wiki/Tutorials
To exit, type \\
To remove this startup msg, edit q.q
q)
```

# Create a Kdb file

1. Create a directory [Data] in q directory
2. Save or copy csv files to this directory
3. At q> prompt (in CMD window), run the data import; for example
   - PivotData: ("ZSSSJDSDSSSSSSSSSSSSFSSFFFFFFFFFFFF"; **enlist** ",") 0: `:/q/data/PivotData.csv;

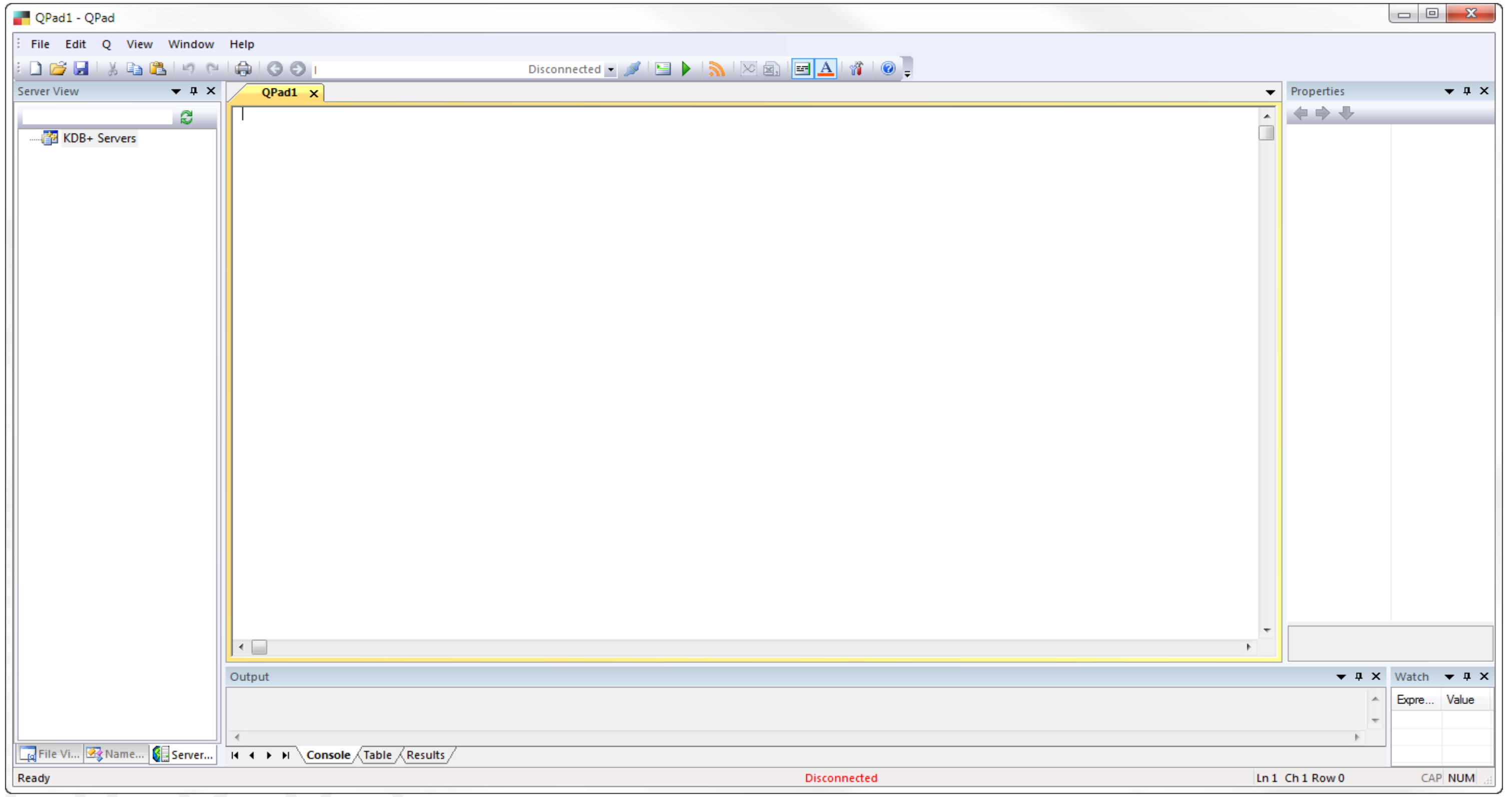


```
q)PivotData: ("ZSSSJDSDSSSSSSSSSSSSFSSFFFFFFFFFFFF"; enlist ",") 0: `:/q/data/P
ivotData.csv;
q)PivotData
time                    sym    ccyGrp     dealType tradeId    tradeDate  dire..
-----------------------------------------------------------------------------..
2014.02.24T08:00:11.113 USDJPY Top_5      SPOT     1015334491 2014.02.24 Buy ..
2014.02.24T08:00:11.894 USDJPY Top_5      FORWARD  1015334491 2014.02.24 Buy ..
2014.02.24T08:00:12.565 USDCNH EM         NDF      1400535900 2014.02.24 Buy ..
2014.02.24T08:00:18.114 EURUSD Top_5      FORWARD  1016318587 2014.02.24 Sell..
```

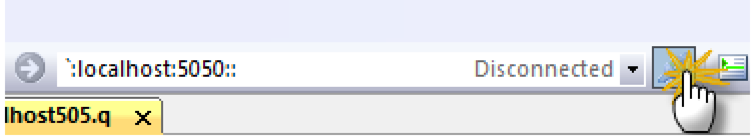
*Dashboard file name*: (“column formats”; enlist “,”) 0: ‘:/*source file name & address*;

1. Install 64-bit QPad from http://www.qinsightpad.com/

2. Run

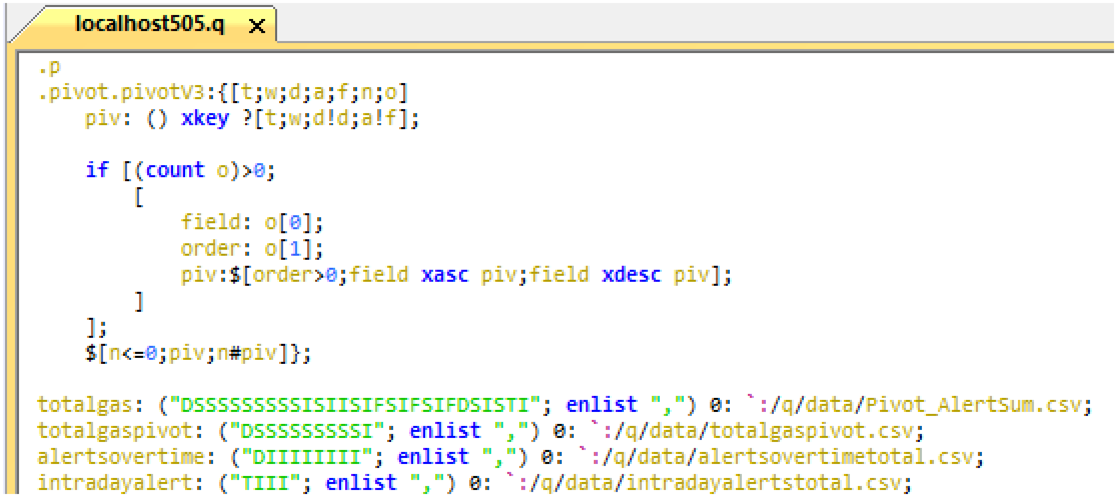


1. Connect to Server:localhost on port 5050



2. Write into the editor

```
.pivot.pivotV3:{[t;w;d;a;f;n;o]
piv: () xkey ?[t;w;d!d;a!f];
if [(count o)>0;
      [
         field: o[0];
         order: o[1];
         piv:$[order>0;field xasc piv;field xdesc piv];
      ]
  ];
  $[n<=0;piv;n#piv]};
```

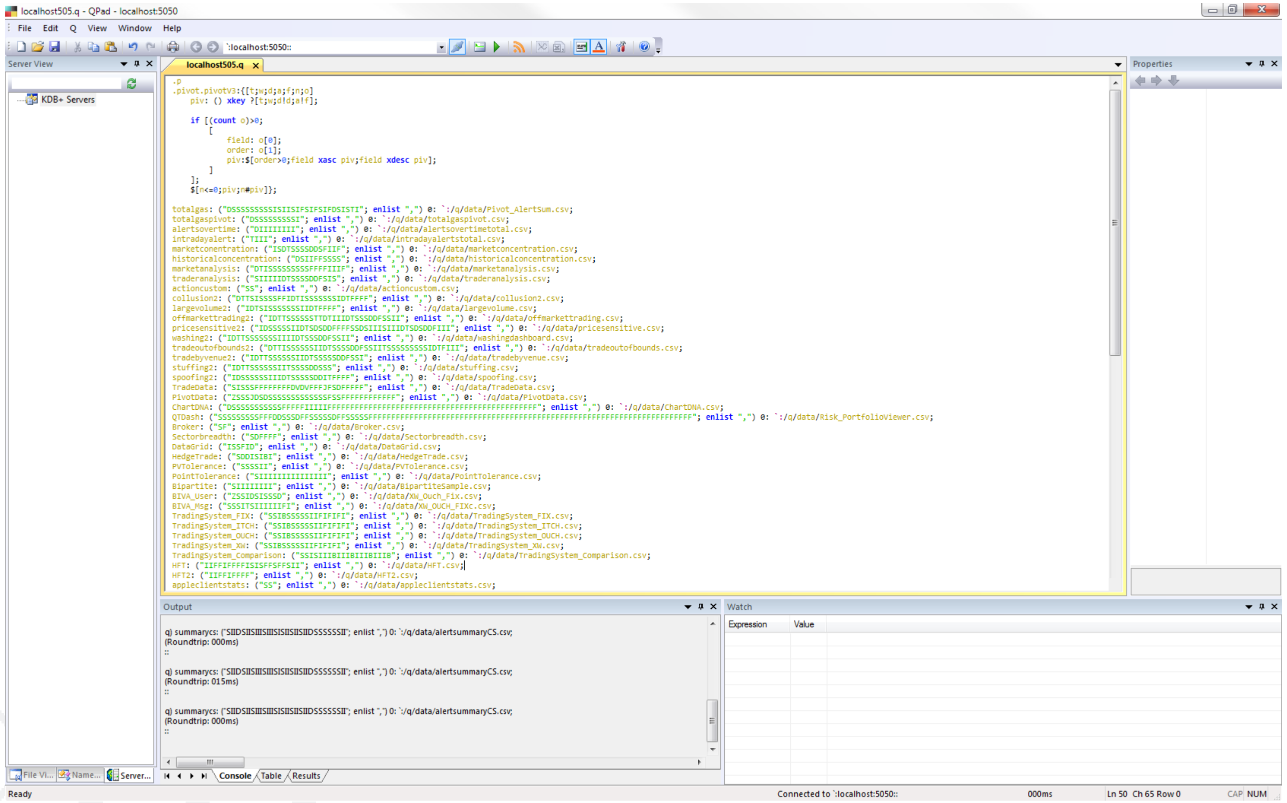


```
.p
.pivot.pivotV3:{[t;w;d;a;f;n;o]
    piv: () xkey ?[t;w;d!d;a!f];

    if [(count o)>0;
        [
            field: o[0];
            order: o[1];
            piv:$[order>0;field xasc piv;field xdesc piv];
        ]
    ];
    $[n<=0;piv;n#piv]};

totalgas: ("DSSSSSSSSSISIISIFSIFSIFDSISTI"; enlist ",") 0: `:/q/data/Pivot_AlertSum.csv;
totalgaspivot: ("DSSSSSSSSSI"; enlist ",") 0: `:/q/data/totalgaspivot.csv;
alertsovertime: ("DIIIIIIII"; enlist ",") 0: `:/q/data/alertsovertimetotal.csv;
intradayalert: ("TIII"; enlist ",") 0: `:/q/data/intradayalertstotal.csv;
```

# Configure QPad

1. Run ▶
2. Add Data with; for example:

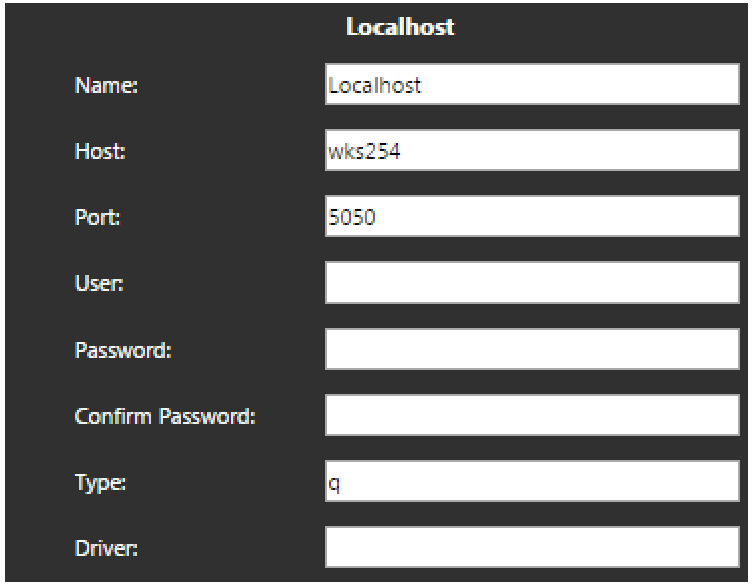
   PivotData: ("ZSSSJDSDSSSSSSSSSSSSSFSSFFFFFFFFFFFF"; **enlist** ",") 0: `:/q/data/PivotData.csv;



```
.p
.pivot.pivotV3:{[t;w;d;a;f;n;o]
    piv: () xkey ?[t;w;d!d;a!f];

    if [(count o)>0;
        [
            field: o[0];
            order: o[1];
            piv:$[order>0;field xasc piv;field xdesc piv];
        ]
    ];
    $[n<=0;piv;n#piv]};

totalgas: ("DSSSSSSSSSISIISIFSIFSIFDSISTI"; enlist ",") 0: `:/q/data/Pivot_AlertSum.csv;
totalgaspivot: ("DSSSSSSSSSI"; enlist ",") 0: `:/q/data/totalgaspivot.csv;
alertsovertime: ("DIIIIIIII"; enlist ",") 0: `:/q/data/alertsovertimetotal.csv;
intradayalert: ("TIII"; enlist ",") 0: `:/q/data/intradayalertstotal.csv;
marketconentration: ("ISDTSSSSDDSFIIF"; enlist ",") 0: `:/q/data/marketconcentration.csv;
historicalconcentration: ("DSIIFFSSSS"; enlist ",") 0: `:/q/data/historicalconcentration.csv;
marketanalysis: ("DTISSSSSSSSSFFFFIIIF"; enlist ",") 0: `:/q/data/marketanalysis.csv;
traderanalysis: ("SIIIIIDTSSSSDDFSIS"; enlist ",") 0: `:/q/data/traderanalysis.csv;
actioncustom: ("SS"; enlist ",") 0: `:/q/data/actioncustom.csv;
collusion2: ("DTTSISSSSFFIDTISSSSSSSIDTFFFF"; enlist ",") 0: `:/q/data/collusion2.csv;
largevolume2: ("IDTSISSSSSSSIIDTFFFF"; enlist ",") 0: `:/q/data/largevolume.csv;
offmarkettrading2: ("IDTTSSSSSSTTDTIIIDTSSSDDFSSII"; enlist ",") 0: `:/q/data/offmarkettrading.csv;
pricesensitive2: ("IDSSSSSIIDTSDSDDFFFFSSDSIIISIIIDTSDSDDFIII"; enlist ",") 0: `:/q/data/pricesensitive.csv;
washing2: ("IDTTSSSSSSSIIIIDTSSSDDFSSII"; enlist ",") 0: `:/q/data/washingdashboard.csv;
tradeoutofbounds2: ("DTTISSSSSSIIDTSSSSDDFSSIITSSSSSSSSSIDTFIII"; enlist ",") 0: `:/q/data/tradeoutofbounds.csv;
tradebyvenue2: ("IDTTSSSSSSIIDTSSSSSDDFSSI"; enlist ",") 0: `:/q/data/tradebyvenue.csv;
stuffing2: ("IDTTSSSSSSSIITSSSSDDSSS"; enlist ",") 0: `:/q/data/stuffing.csv;
spoofing2: ("IDSSSSSSIIIDTSSSSSDDITFFFF"; enlist ",") 0: `:/q/data/spoofing.csv;
TradeData: ("SISSSFFFFFFFFDVDVFFFJFSDFFFFF"; enlist ",") 0: `:/q/data/TradeData.csv;
PivotData: ("ZSSSJDSDSSSSSSSSSSSSSFSSFFFFFFFFFFFF"; enlist ",") 0: `:/q/data/PivotData.csv;
ChartDNA: ("DSSSSSSSSSSSFFFFFIIIIIFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF"; enlist ",") 0: `:/q/data/ChartDNA.csv;
QTDash: ("SSSSSSSSSFFFDDSSSDFFSSSSSDFFSSSSSFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF"; enlist ",") 0: `:/q/data/Risk_PortfolioViewer.csv;
Broker: ("SF"; enlist ",") 0: `:/q/data/Broker.csv;
Sectorbreadth: ("SDFFFF"; enlist ",") 0: `:/q/data/Sectorbreadth.csv;
DataGrid: ("ISSFID"; enlist ",") 0: `:/q/data/DataGrid.csv;
HedgeTrade: ("SDDISIBI"; enlist ",") 0: `:/q/data/HedgeTrade.csv;
PVTolerance: ("SSSSII"; enlist ",") 0: `:/q/data/PVTolerance.csv;
PointTolerance: ("SIIIIIIIIIIIIIII"; enlist ",") 0: `:/q/data/PointTolerance.csv;
Bipartite: ("SIIIIIIII"; enlist ",") 0: `:/q/data/BipartiteSample.csv;
BIVA_User: ("ZSSIDSISSSD"; enlist ",") 0: `:/q/data/XW_Ouch_Fix.csv;
BIVA_Msg: ("SSSITSIIIIIIFI"; enlist ",") 0: `:/q/data/XW_OUCH_FIXc.csv;
TradingSystem_FIX: ("SSIBSSSSSIIFIFIFI"; enlist ",") 0: `:/q/data/TradingSystem_FIX.csv;
TradingSystem_ITCH: ("SSIBSSSSSIIFIFIFI"; enlist ",") 0: `:/q/data/TradingSystem_ITCH.csv;
TradingSystem_OUCH: ("SSIBSSSSSIIFIFIFI"; enlist ",") 0: `:/q/data/TradingSystem_OUCH.csv;
TradingSystem_XW: ("SSIBSSSSSIIFIFIFI"; enlist ",") 0: `:/q/data/TradingSystem_XW.csv;
TradingSystem_Comparison: ("SSISIIIBIIIBIIIBIIIB"; enlist ",") 0: `:/q/data/TradingSystem_Comparison.csv;
HFT: ("IIFFIFFFFISISFFSFFSII"; enlist ",") 0: `:/q/data/HFT.csv;
HFT2: ("IIFFIFFFF"; enlist ",") 0: `:/q/data/HFT2.csv;
appleclientstats: ("SS"; enlist ",") 0: `:/q/data/appleclientstats.csv;
```

```
q) summarycs: ("SIIDSIISIIISIIISISIISIISIIDSSSSSSII"; enlist ",") 0: `:/q/data/alertsummaryCS.csv;
(Roundtrip: 000ms)
::

q) summarycs: ("SIIDSIISIIISIIISISIISIISIIDSSSSSSII"; enlist ",") 0: `:/q/data/alertsummaryCS.csv;
(Roundtrip: 015ms)
::

q) summarycs: ("SIIDSIISIIISIIISISIISIISIIDSSSSSSII"; enlist ",") 0: `:/q/data/alertsummaryCS.csv;
(Roundtrip: 000ms)
::
```

1. In Dashboards, create a connection for localhost



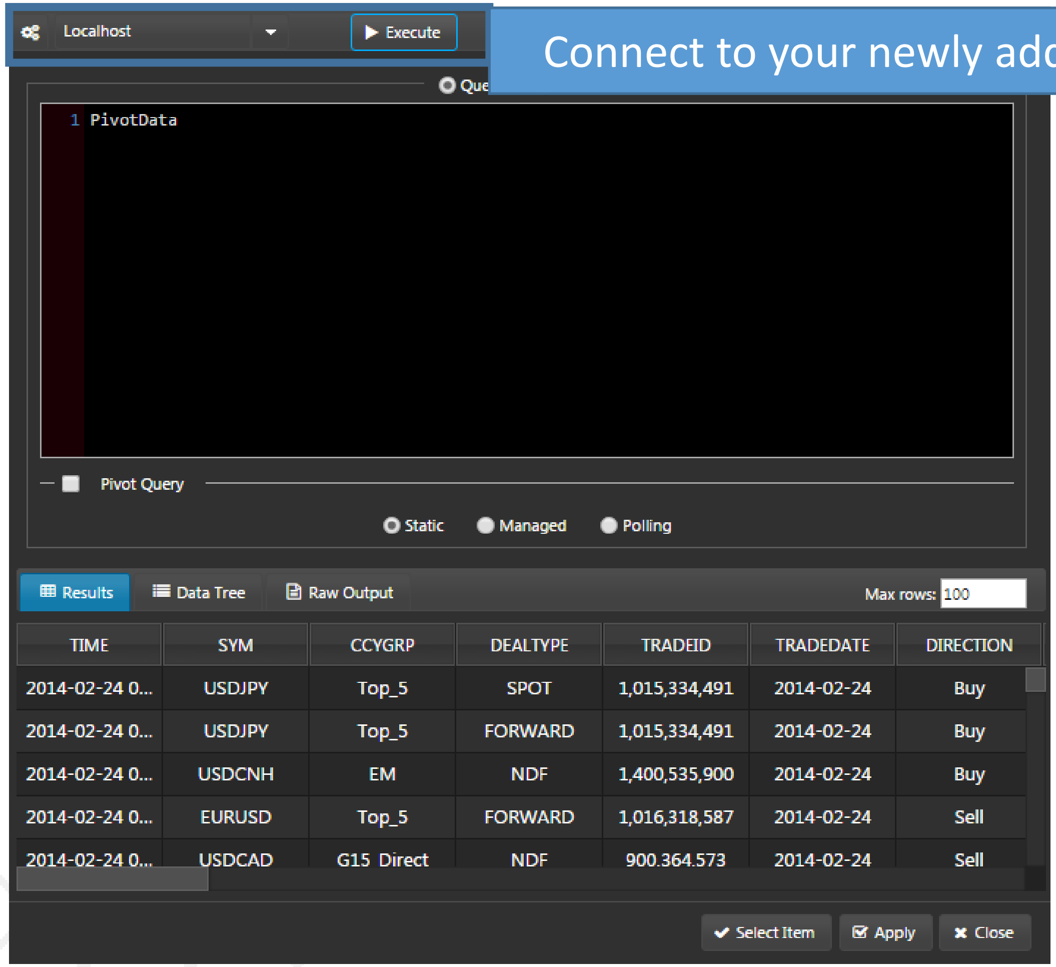
**Localhost**

| | |
|---|---|
| Name: | Localhost |
| Host: | wks254 |
| Port: | 5050 |
| User: | |
| Password: | |
| Confirm Password: | |
| Type: | q |
| Driver: | |

**Host:** *is PC name (e.g. wks254). Do not use "localhost" for* Host

# Set dropdown query connection to Localhost

1. In the Dropdown, select database connection



```
PivotData
```

□ Pivot Query

○ Static ○ Managed ○ Polling

Max rows: 100

| TIME | SYM | CCYGRP | DEALTYPE | TRADEID | TRADEDATE | DIRECTION |
|---|---|---|---|---|---|---|
| 2014-02-24 0... | USDJPY | Top_5 | SPOT | 1,015,334,491 | 2014-02-24 | Buy |
| 2014-02-24 0... | USDJPY | Top_5 | FORWARD | 1,015,334,491 | 2014-02-24 | Buy |
| 2014-02-24 0... | USDCNH | EM | NDF | 1,400,535,900 | 2014-02-24 | Buy |
| 2014-02-24 0... | EURUSD | Top_5 | FORWARD | 1,016,318,587 | 2014-02-24 | Sell |
| 2014-02-24 0... | USDCAD | G15_Direct | NDF | 900,364,573 | 2014-02-24 | Sell |

Connect to your newly added (local) kdb server

If using a local connection, Control for Kx specific apps, **Action Tracker**, and **Order Book Replay** won't be configurable