



WHAT'S NEW AND RELEASE NOTES

Release: 2020.9



IN THIS RELEASE

NEW FEATURES AND IMPROVEMENTS..... 2

DATA PREPARATION..... 2

- Apply number formatting..... 2
- Apply date & time formatting 4
- Rename all columns in a table..... 6
- Dataset columns structured in Tableau..... 7
- Consistent naming for aggregations..... 8
- Rename columns used in calculations..... 10
- TM1 Cube View connection details 11

STORYBOARDS..... 12

- Date & number formatting..... 12
- HTML code widget 14
- HTML input widget 15
- HTML button widget..... 16
- Add CSS and JavaScript to a storyboard page 17
- View storyboards’ recipients in List view 19
- Filter data by one column in Table 20
- Clear filter on cell toggle in Table..... 21
- Use additional columns for visualization queries..... 22
- Other enhancements in Table 23
- Column search in Filter & Highlighter 24
- Text & date formats in Single Value 25
- Consistent naming for aggregations..... 26
- Preview a script code..... 27
- The default size for some widgets..... 28
- Cancel visualizing a widget 29
- Percentage sign in Pareto charts..... 30

DATA SCIENCE 31

- Deploy custom Python libraries 31

INSTALLATIONS & CONFIGURATION 32

- Deploy the platform behind the external SSL balancer..... 32
- Create a tenant in Access Manager..... 33

NEW FEATURES AND IMPROVEMENTS

DATA PREPARATION

Apply number formatting

In this release, dataset modelers can control how to display values in numeric columns. The column format defines how values are displayed in a dataset and in visualizations on a storyboard. Number formats only determine the appearance of the values but do not change how the actual values are stored.

You can format numeric columns as number, currency, or percentage. Each format has its own set of properties. For example, you can define how many decimals to show, what separator symbols to use for decimals and thousands, select the appearance of negative numbers, or whether to use abbreviations for large numbers such as K, M, G, B, or T.

You can define a number format in Step 2 and Step 3 of the dataset preparation process. For example, in the **Data preview** table, point to a numeric column, click **column options**, and then select **Format**.

The screenshot shows a 'Data preview' table for 'Sample - World Sales'. The table has columns: Quantity, Unit Price, Discount, Gross Sales, Planned Gross Sales, Manufacturing Cost, and Sales Cost. A context menu is open over the 'Unit Price' column, with the 'Format' option highlighted.

rs	Quantity	Unit Price	Discount	Gross Sales	Planned Gross Sales	Manufacturing Cost	Sales Cost
	#	#	#	#	#	#	#
	15		7%	167.40	190.8400	3.60	8.35
	30		3%	1978.80	1.7018K	40.80	44.58
	25		5%	1021.25	919.1300	21.50	25.43
	48		8%	1324.80	1.5103K	28.80	31.50
	20		5%	323	293.9300	6.80	11.46
	15		5%	1425	1.3538K	30	33.50
	10	\$200	4%	1920	1.7664K	40	43.40
	15	\$12	2%	176.40	201.1000	3.60	8.53
	7	\$112	6%	736.96	751.7000	15.68	19.74

In the **Column format** dialog, select which format type to apply: **Number**, **Currency**, or **Percentage**. Then, you can customize the individual format properties as needed.

No of Transactions	No of Customers	Quantity	Unit Price	Discount	Gross Sales	Planned Gross Sales
#					#	#
10					1425	1353.75
6					1920	1766.40
10					27648	24053.76
3					1187.50	1246.88
9					1425	1539
7					1920	1843.20
3					1187.50	1330.00
10					1380	1435.20
9					14332.50	13902.52
8					4185	3975.75
2					3880	3918.80
10					1426.88	1312.73
4					2184	1943.76
4					2250	2250
8	64	30	60	5.00%	1710	1949.40

COLUMN FORMAT ✕

Specify a number format for **Unit Price**

<p>Format as</p> <div style="border: 1px solid #ccc; padding: 2px; width: 100%;">Currency ▼</div>	<p>Negative values</p> <div style="border: 1px solid #ccc; padding: 2px; width: 100%;">-1234 ▼</div>
<p>Decimal places</p> <div style="border: 1px solid #ccc; padding: 2px; width: 100%;">2 ▲▼</div>	<p>Display units</p> <div style="border: 1px solid #ccc; padding: 2px; width: 100%;">Auto ▼</div>
<p>Decimal separator</p> <div style="border: 1px solid #ccc; padding: 2px; width: 100%;">Period (.) ▼</div>	<p>Currency</p> <div style="border: 1px solid #ccc; padding: 2px; width: 100%;">US dollar (\$ / USD) ▼</div>
<p>Thousands separator</p> <div style="border: 1px solid #ccc; padding: 2px; width: 100%;">Comma (,) ▼</div>	<p>Display currency</p> <div style="border: 1px solid #ccc; padding: 2px; width: 100%;">Symbol ▼</div>
<p>Prefix</p> <div style="border: 1px solid #ccc; padding: 2px; width: 100%;">\$</div>	<p>Suffix</p> <div style="border: 1px solid #ccc; padding: 2px; width: 100%;"></div>

CANCEL
OK

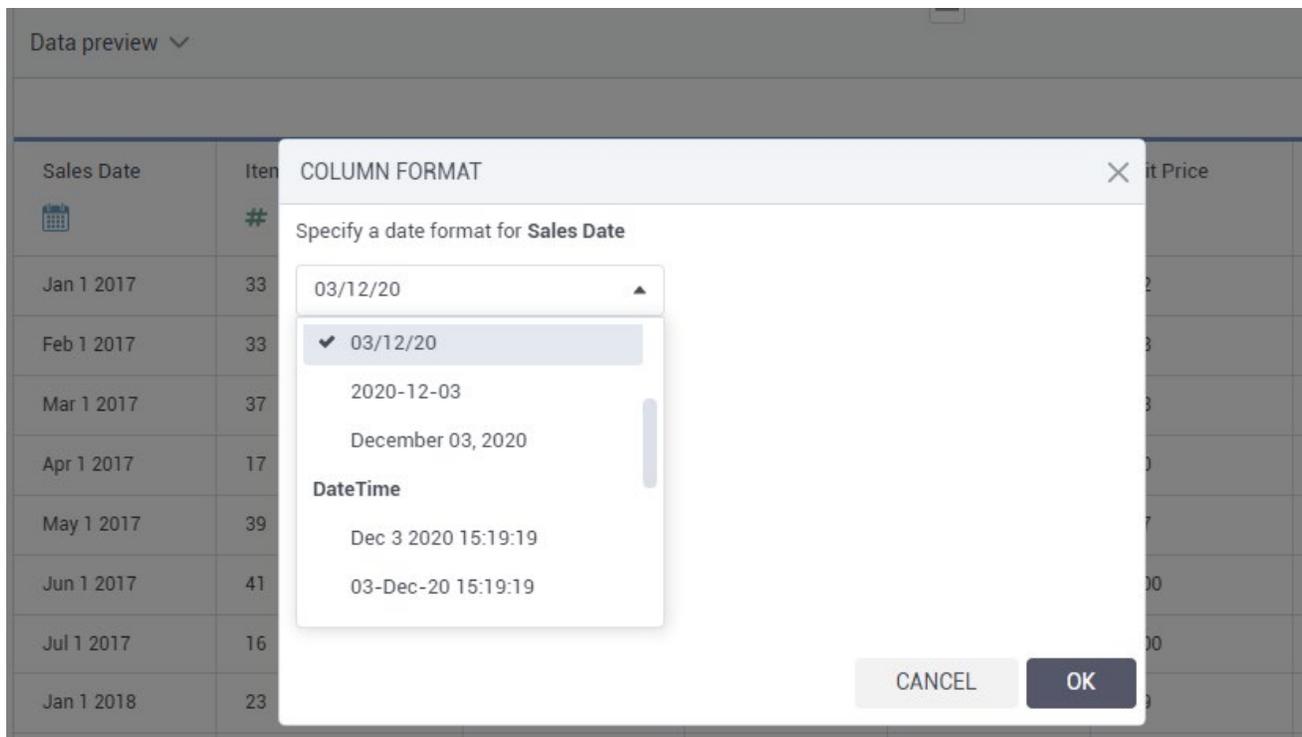
Apply date & time formatting

This release also allows you to control the appearance of the columns containing dates, date-time, or time information. The column format defines how values are displayed in a dataset and in visualizations on a storyboard. Date formats only determine the appearance of the values but do not change how the actual values are stored.

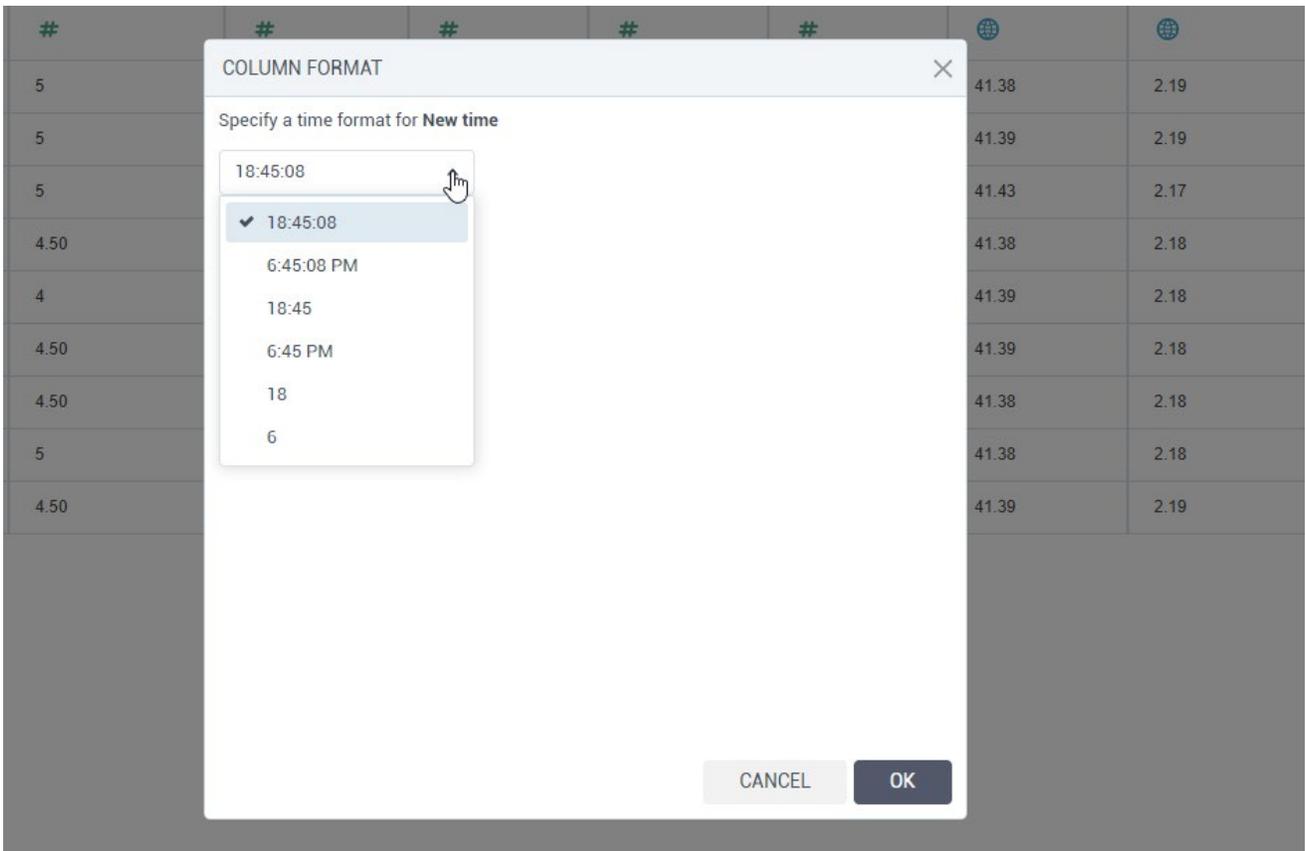
For a date column, you can select a format from a predefined list. For example, you can choose to display a long date, a short date, include a weekday, or spell out month names:

- 03-Dec-20
- 12/03/20
- 03/12/20
- 03/12/2020
- 2020-12-03
- December 03, 2020
- 03-Dec-20 13:30:00
- 03-Dec-20 1:30:00 PM
- Thursday, December 3, 2020
- Thu, December 3, 2020

You can select a date format in Step 2 and Step 3 of the dataset preparation process. For example, in Step 3, go to the **Data preview** table, point to a date column, click **column options**, and select **Format**. In the **Column format** dialog, select a format and click **OK**.



For a time column, you can also select one of the predefined formats.



Rename all columns in a table

Before, you could bulk rename columns only in the last step of the data preparation process. The user experience has been improved, and now, you can rename all table columns right after you select a data source for a dataset.

In the **Selected items** section, hover over a selected table and click **Rename all columns**. Then, in the **Rename all columns** dialog, you can choose how to rename the columns:

- **Add prefix** – Add a data source name or a custom string in front of the column name.
- **Add suffix** – Add a data source name or a custom string after the column name.
- **Format** – Capitalize the column names: only first letter, all uppercase, or all lowercase.
- **Replace** – Enter the text that you want to replace and specify a new text. For example, you can substitute the underscore with the space character.

The screenshot shows the 'CHOOSE TABLES TO IMPORT' interface. On the left, under 'TABLES', there is a search bar and a tree view showing 'dbo' > 'Customers' selected. On the right, under 'SELECTED ITEMS', there is a search bar and a tree view showing 'dbo' > 'Customers' selected. A blue box highlights the 'Customers' table in the 'SELECTED ITEMS' section, with a 'Rename all columns' button visible. A dialog box titled 'RENAME ALL COLUMNS' is open, showing the following options:

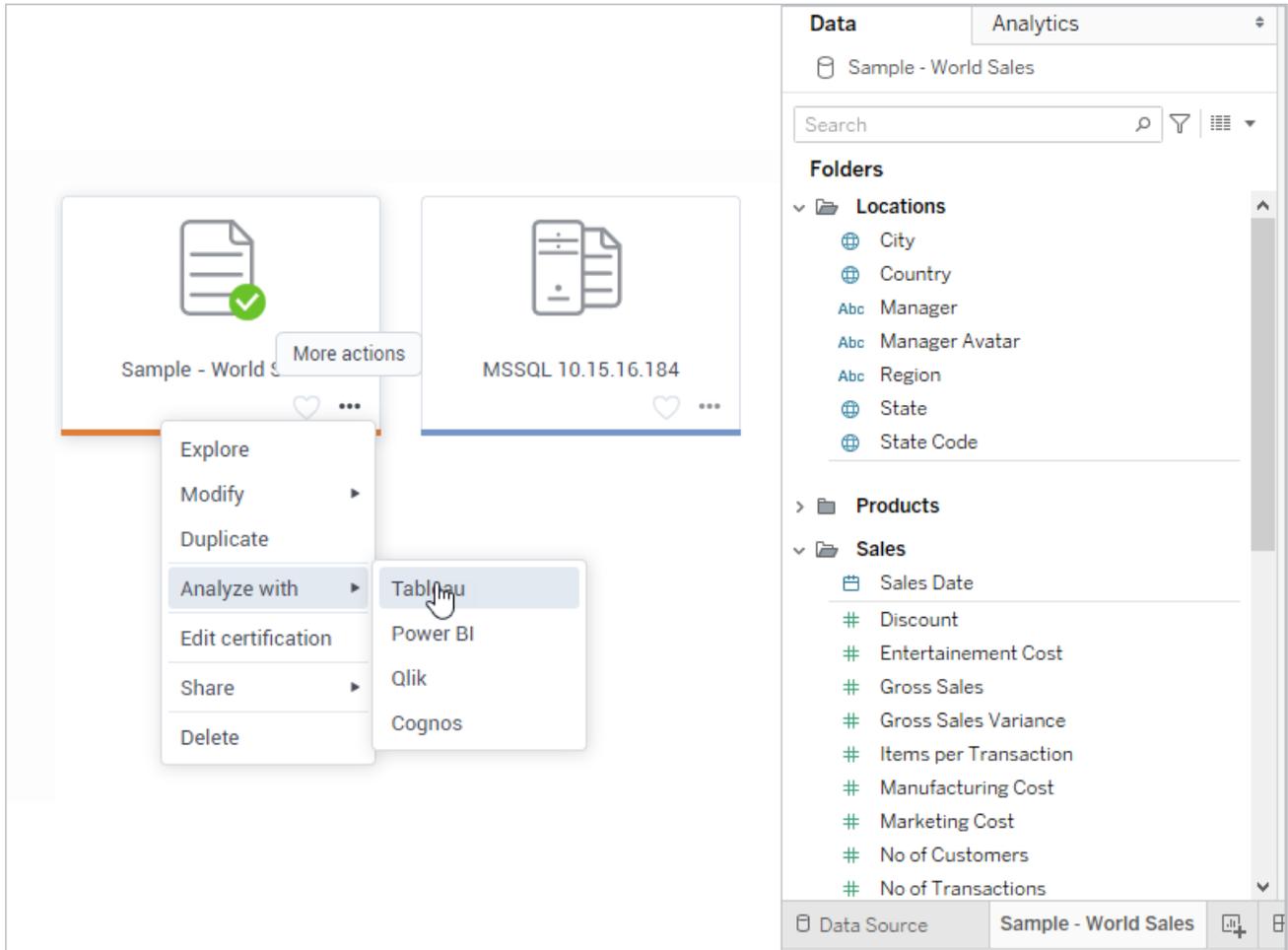
- Add prefix (Source name dropdown)
- Add suffix (Source name dropdown)
- Format (Capitalize first letter dropdown)
- Replace (Old value and Replace with text boxes)

Example: Sales.Sales_Date

CANCEL OK

Dataset columns structured in Tableau

When you work with DataClarity datasets in Tableau, the dataset columns are now organized by data sources. If you open a Tableau workbook file generated for a dataset, the columns are grouped in folders by data sources the same way as in Data Preparation.



Consistent naming for aggregations

The aggregation options are now using the same naming conventions in Step 2 and Step 3 of the data preparation process, as follows:

- Sum (instead of Total)
- Average (instead of Avg)
- Minimum (instead of Min)
- Maximum (instead of Max)

Items per Transaction	No of Transactions	No of Customers	Quantity	Unit Price
#	#	#	#	#
33			30	68
17			48	30
41	10		15	100
16	6		10	200
11	10	9	12	2400
26	3	9	5	250
24	10	12	48	87
33	9	14	15	100

In the **Explore dataset** pages, the aggregation options are also using non-abbreviated names consistently.

↓	Items per Tran...	No of Transactions	Quantity
☰	☹	☰	
📌 Pin Column		3	10
Σ Value Aggregation		3	10
Autosize This Column			8
Autosize All Columns			25
☰ Group by Items per Transaction			8
Reset Columns			10
Expand All			15
Collapse All			25
	44	2	25

Rename columns used in calculations

Now, when creating dataset calculations, you can rename any column that is used in a calculation formula. If you do so, the dataset calculations are automatically updated to use the new column names. Note that you still need to review the calculations created in storyboards if any of the input columns have been renamed in Data Preparation. This constraint will be removed in the upcoming release.

Sales	Store Types	Locations	Products		
Supply Cost	Profit	Planned Profit	Profit Variance	Gross Sales Variance	
#	#	#	fx	fx	
1.70	1820.46	1543.43	277.03	277.03	
1.51	1214.63	1400.10	-185.47	-185.47	
1.35	1308.27	1237.02	71.25	71.25	
1.77	1765.63	1612.03	153.60	153.60	
24.05	25520.31	21926.07	3594.24	3594.24	
1.25	1088.94	1148.32	-59.38	-59.38	
4.00	3536.10	3689.78	-153.68	-153.68	

TM1 Cube View connection details

When connecting to the TM1 Cube View, the **Server** and **Port** fields have been merged into a single **URL** field, where you can provide the full address (including the protocol and port) to the server with TM1 cube view.

ADD DATA CONNECTION ✕

Connection details Caching



TM1 Cube View data source connection

Username and Password ▼

URL
http://10.12.12.225:8001

Username
george.becker|

Password
.....

Cube Name ▼ Load list

Ask user for credentials
Off

Connection name
TM1 Cube View 10.12.12.225

Test your connection

After you fill in the data connection details, test your connection. If something goes wrong, an error message will be displayed.

TEST CONNECTION

STORYBOARDS

Date & number formatting

In this release, users can visualize the data columns with the number and date formatting applied in Data Preparation. For example, if a dataset modeler formatted a numeric column with price information as currency, the column will use this format by default.

The legacy local widget formatting has been renamed to **Default number format** and **Default date format**. By default, the local widget formatting is turned off and applies only to the columns with no defined formatting in Data Preparation.



If you select **On** for **Override for all columns**, all the columns will use local widget formatting. For example, the discount value of 7% (formatted as a percentage in Data Preparation) will be displayed as a number, 0.07.



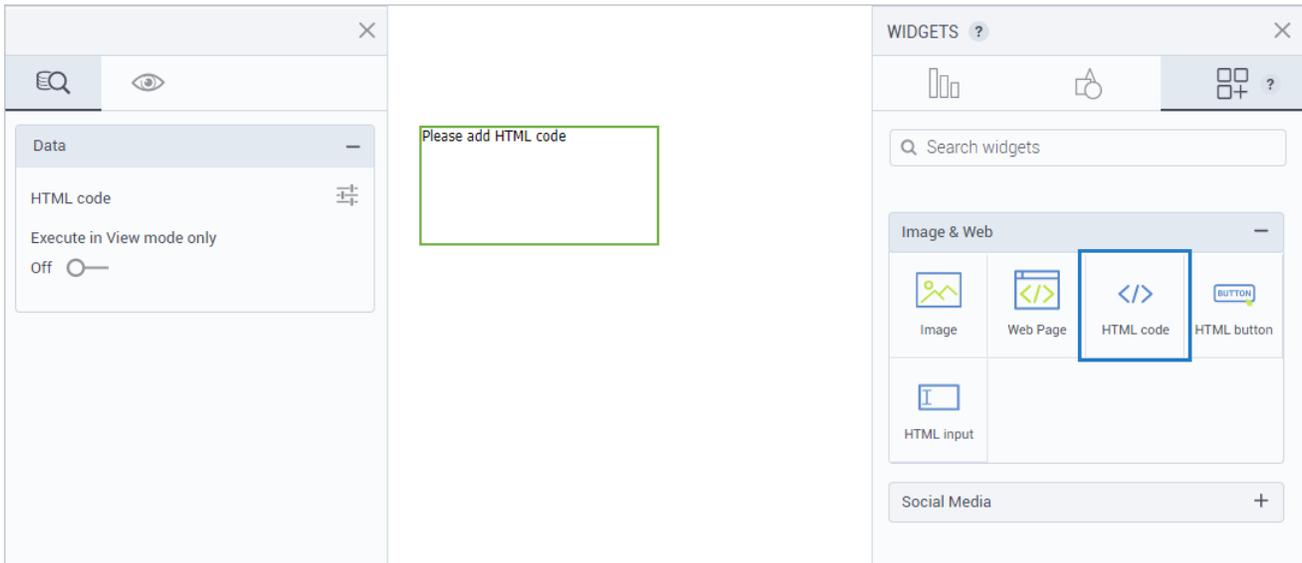
If a dataset modeler changes some formats after you create visualizations for a given dataset, you will need to re-visualize your widgets to update the columns with the new formats.

HTML code widget

In this release, power users can use the **HTML code** widget to add an HTML code and execute it on a storyboard page. For example, you can add some new interface elements, customize pages, and use the **HTML button** and **HTML input** widgets to execute more complex scenarios.

To add your code in the widget, click  **manage html code**, paste the code, and click **Save**. If you do not want to run the code when your storyboard is in Edit mode, turn on **Execute in View mode only**. This way, switching a storyboard to View mode will be a trigger to execute the code.

You can find the new widget in the **Widgets** pane, on the **Other widgets** tab.



HTML input widget

With the new **HTML input** widget, you can add a text field to enter some input data directly from the storyboard. The input data can be used in a script added in the **HTML code** widget or in the **Custom JavaScript code** section added for a storyboard page.

On the **data** tab, you define the field's reference data, add a default value and a placeholder to let users know what is expected in the text field for further processing.

You can find the new widget in the **Widgets** pane, on the **Other widgets** tab.

The screenshot displays the DataClarity interface with three main panels:

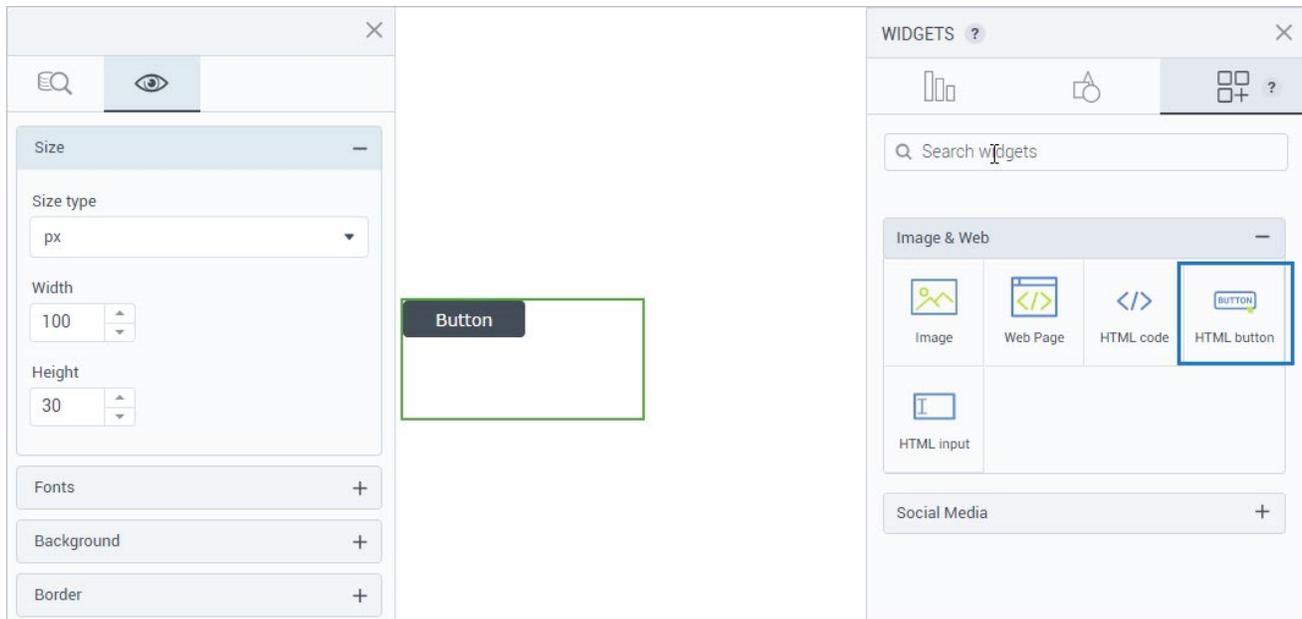
- Left Panel (Data Tab):** A configuration panel for the HTML input widget. It includes fields for:
 - Name:** Store ID
 - ID:** storeid
 - Default value:** (empty field)
 - Placeholder:** Enter the store ID
- Center Panel (Storyboard):** A visual representation of the widget on a storyboard. A text input field with the placeholder text "Enter the store ID" is highlighted with a green border.
- Right Panel (WIDGETS):** A sidebar containing various widget categories. The "Image & Web" category is expanded, showing icons for Image, Web Page, HTML code, and HTML button. The "HTML input" widget is highlighted with a blue border. Below this category is a "Social Media" section with a plus sign.

HTML button widget

With the new **HTML button** widget, you can add and style a button on a storyboard page. The button can be used to run the script provided in the **HTML code** widget or in the **Custom JavaScript code** section for a storyboard page.

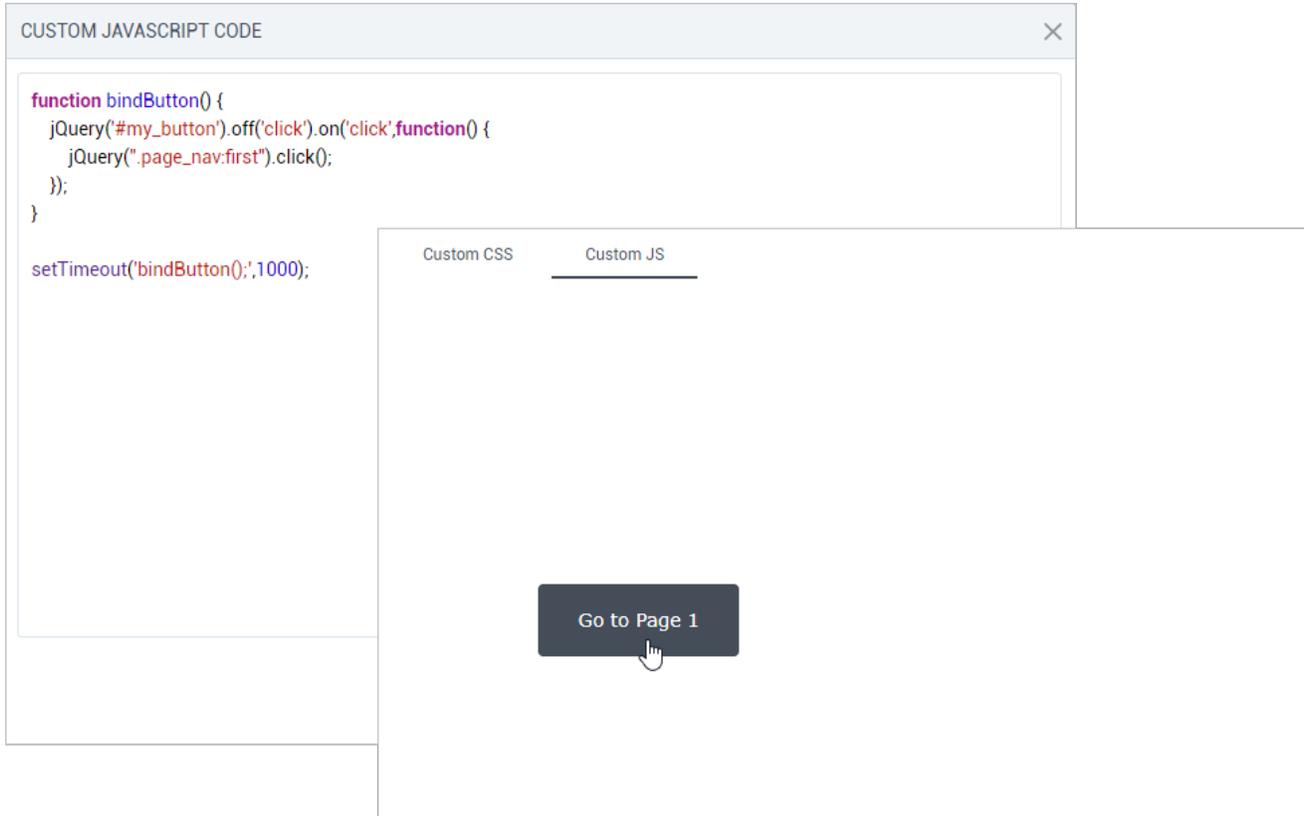
On the **data** tab, you define the button's reference data. On the **appearance** tab, you can specify its size, font, background color, and borders.

You can find the new widget in the **Widgets** pane, on the **Other widgets** tab.



Add CSS and JavaScript to a storyboard page

In this release, advanced users can add JavaScript code and CSS to customize storyboard pages or to design additional elements on a page.

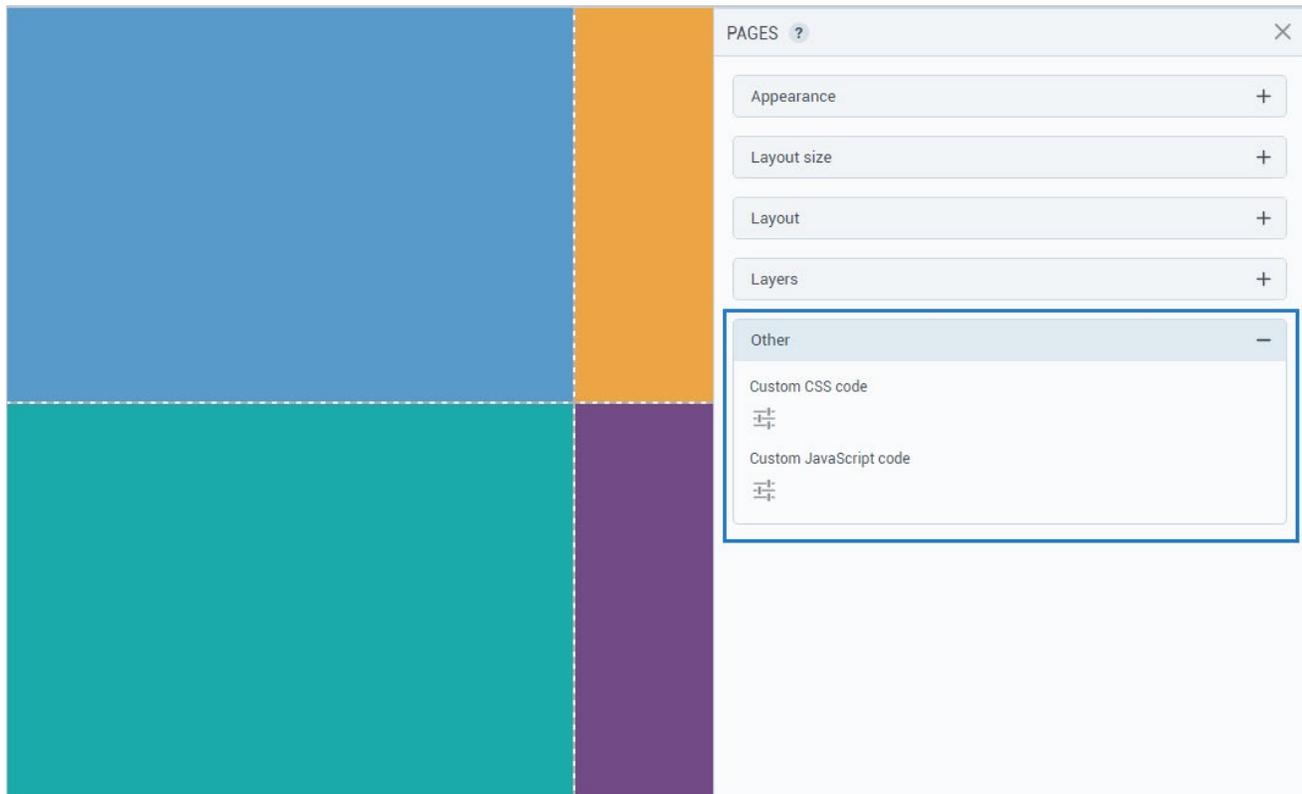


The image shows a software interface with two main components. On the left is a window titled "CUSTOM JAVASCRIPT CODE" with a close button (X) in the top right corner. The code editor contains the following JavaScript code:

```
function bindButton() {  
  jQuery("#my_button").off('click').on('click',function() {  
    jQuery(".page_nav:first").click();  
  });  
}  
  
setTimeout('bindButton();',1000);
```

On the right is a preview window with two tabs: "Custom CSS" and "Custom JS". The "Custom JS" tab is selected and underlined. The preview area shows a dark grey rectangular button with the text "Go to Page 1" in white. A white mouse cursor is positioned over the button, indicating it is interactive.

The new options are placed in the **Other** section of the **Pages** pane.



To customize a page's appearance or some elements on a page with your custom CSS code, click  **Manage custom CSS code**, and define the styles. For example, you can color each cell with a different color.

To add code to a storyboard page, click  **Manage custom JavaScript code**. For example, you can add code for a button (widget) to open a certain storyboard page.

View storyboards' recipients in List view

Previously, the **Subscribers** column appeared as a shortcut to the **Share** dialog for each storyboard, even if it was not shared. The column has been renamed to **Shared with**, and the corresponding icon appears only if a storyboard is shared. Moreover, you can quickly see how many users received the storyboard: the first number represents group recipients, and the second one represents user recipients.

The screenshot shows the 'My storyboards' section of the DataClarity interface. At the top, there are tabs for 'My storyboards' and 'Shared with me', along with a search bar and a '+ NEW STORYBOARD' button. Below the search bar is a table of storyboards. The table has columns for 'Name', 'Last updated', and 'Shared with'. The 'Shared with' column is highlighted with a blue box. The table contains the following data:

Name	Last updated	Shared with
Table	July 8, 2020, 2:59:43 PM	
Sample - Profitability	October 2, 2020, 11:42:31 AM	(1 2)
Sample - Product Segmentation	October 2, 2020, 11:42:31 AM	
Sample - Multiple BI	October 2, 2020, 11:42:31 AM	(1 2)
Sample - Monthly Sales	October 2, 2020, 11:42:31 AM	(0 1)
Sample - Geospatial Routing	October 2, 2020, 11:42:30 AM	
Sample - Geospatial	October 2, 2020, 11:42:30 AM	

Filter data by one column in Table

If you use filtering actions with the **Table** widget, you can now choose which filtering value to use for the target widget. By default, clicking a cell will filter the data by the value in that cell. With the new **Filter by first column only** option, the data is filtered by a specific column no matter where a user clicks in the table row. In this case, the column used for filtering is the first column selected in the **Columns** field on the **data** tab.

You can find the new option on the **advanced** tab of the widget settings pane, in the **Actions** section.

The screenshot shows the 'ACTIONS' settings pane for a widget. On the left, the 'Actions' section is expanded, showing a list of actions. The 'Filter by first column only' option is highlighted with a blue box and is currently turned 'On'. Below it, the 'Clear filter on cell toggle' option is turned 'Off'. On the right, the 'As source (1)' tab is selected, and a table lists the actions. The table has columns for 'Name', 'Event', and 'Action'. The first row is 'Filter by country', 'Left click', and 'Filter'.

Name	Event	Action
<input checked="" type="checkbox"/> Filter by country	Left click	Filter

Clear filter on cell toggle in Table

In this release, you can choose how to interact with the filtering actions in the **Table** widget. Previously, clicking a cell for the second time always cleared the applied filter. If you set the new **Clear filter on cell toggle** setting to **Off**, the target widget will stay filtered even after the second consecutive click on the same table cell.

You can find the new option on the **advanced** tab of the widget settings pane, in the **Actions** section.

The screenshot shows the 'ACTIONS' settings pane for a widget. The left sidebar contains a list of actions, with 'Clear filter on cell toggle' highlighted in a blue box. The main area shows a table of actions with columns for Name, Event, and Action.

Name	Event	Action
<input checked="" type="checkbox"/> Filter by country	Left click	Filter

Use additional columns for visualization queries

Now, users have a dedicated field to add the columns that are not used directly in the visualization layout but are included in the query scope. For example, you create a complex script calculation based on some columns that should not be plotted on the visualization itself. In this case, you add these columns in the **Other** field on the **data** tab of the widget settings pane.

The image shows a settings pane for a visualization widget. It is divided into several sections: Measures, Tooltip, Trellis, Color, and Other. The 'Measures' section contains two items: 'Items per Transaction (Sum)' and 'MyScript (Sum)'. The 'Other' section is highlighted with a blue border and contains four items: 'Gross Sales (Sum)', 'Discount (Sum)', 'Unit Price (Sum)', and 'Country'. Below the 'Color' section, there are two radio buttons: 'Stacked' and 'Clustered', with 'Clustered' being selected.

Measures

- Items per Transaction (Sum) ×
- MyScript (Sum) ⚙ ×

Tooltip

Trellis

Color

Stacked Clustered

Other

- Gross Sales (Sum) ×
- Discount (Sum) ×
- Unit Price (Sum) ×
- Country ×

Other enhancements in Table

The **Table** widget has been enhanced to improve the user experience:

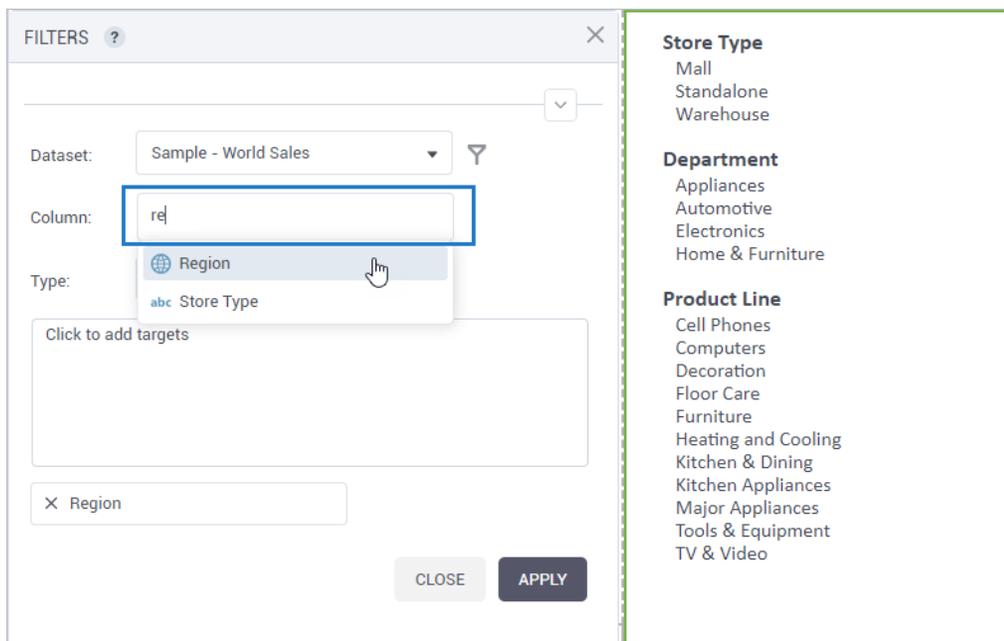
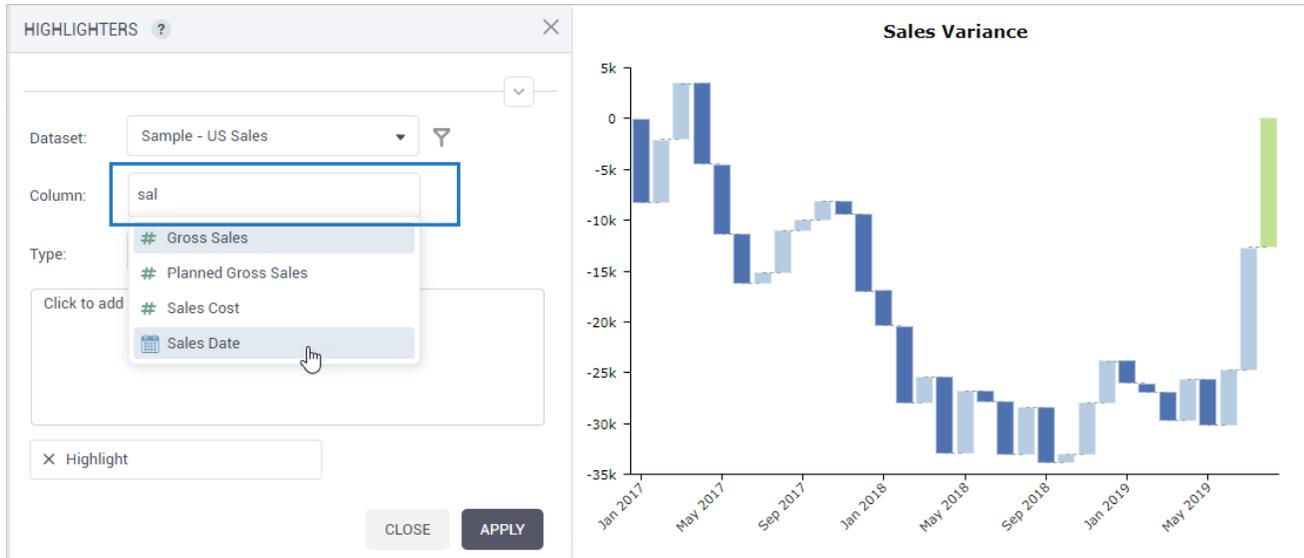
- When you change the **Group data** switch to **On**, each measure's aggregation automatically changes from NONE to the measure's default aggregation.
- When you hover over the column header, the divider line is shown to help you resize the column more quickly.

Country	Appliances			Automotive			Electronics		
	No of Customers	Unit Price	Profit	No of Custom...	Unit Pr...	Profit	No of Custom...		
Chile (4)	196.00	1.09K	10.86K	51.00	150.00	1.79K	205.00		
Kosovo (4)	78.00	737.00	5.03K	17.00	54.00	582.52	131.00		
Uzbekistan (4)	115.00	1.34K	15.13K	29.00	60.00	792.09	194.00		
Mexico (4)	7.81K	1.68K	195.96K	4.24K	2.63K	187.88K	12.31K		
Senegal (4)	145.00	1.1K	12.43K	33.00	150.00	1.79K	216.00		
Germany (4)	5.31K	3.68K	363.67K	9.57K	6K	584.61K	17.49K		
Switzerland (4)	1.72K	2.64K	102.27K	1.3K	6.04K	190.3K	2.96K		
United Kingdom (4)	5.75K	3.38K	333.4K	6.64K	2.23K	168.66K	15.66K		
Sierra Leone (4)	175.00	1.1K	8.39K	51.00	158.00	1.37K	176.00		
Honduras (4)	221.00	714.00	8.73K	33.00	328.00	4.16K	239.00		
Albania (4)	587.00	1.3K	31.04K	83.00	90.00	2.34K	474.00		
Tunisia (4)	259.00	1.12K	14.35K	33.00	328.00	4.16K	308.00		
Portugal (4)	2.19K	2.86K	65.34K	1.61K	5.81K	61.86K	3.12K		
Thailand (4)	306.00	1.1K	11.39K	51.00	150.00	1.79K	455.00		
Zambia (4)	244.00	1.02K	10.7K	31.00	328.00	3.66K	269.00		

1 to 50 of 138 | < Page 1 of 3 > >|

Column search in Filter & Highlighter

Now, selecting a column in the **Filter** and **Highlighter** widgets has become faster. Users can now search for a column directly in the **Column** field, and the column list is filtered as you type. Moreover, the columns are sorted alphabetically.



Text & date formats in Single Value

The **Single Value** widget can now display text and dates. The **Format** drop-down list now includes **Text** and **Date**. To use these display formats, you should create an aggregated calculation. For example, to view the earliest sales date, you can use a formula similar to *min([Sales Date])*.

The screenshot shows a configuration panel on the left and a visualization on the right. The configuration panel includes options for Background, Fonts, Title, and Default number format. The 'Format' dropdown menu is open, showing options: Text (selected), Currency, Date, Number, and Percentage. The visualization displays sales data for four categories: Appliances (\$9,730,642), Automotive (\$7,050,961), Electronics (\$20,861,215), and Home & Furniture (\$9,629,928). Below the data is a large text box containing the word 'Oscar'.

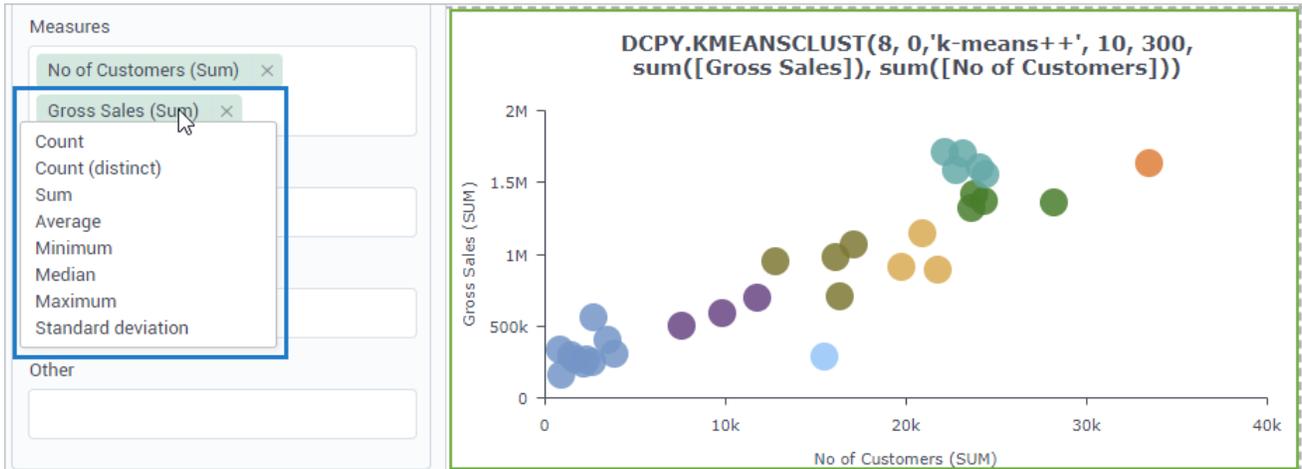
Category	Sales Value
Appliances	\$9,730,642
Automotive	\$7,050,961
Electronics	\$20,861,215
Home & Furniture	\$9,629,928

Oscar

Consistent naming for aggregations

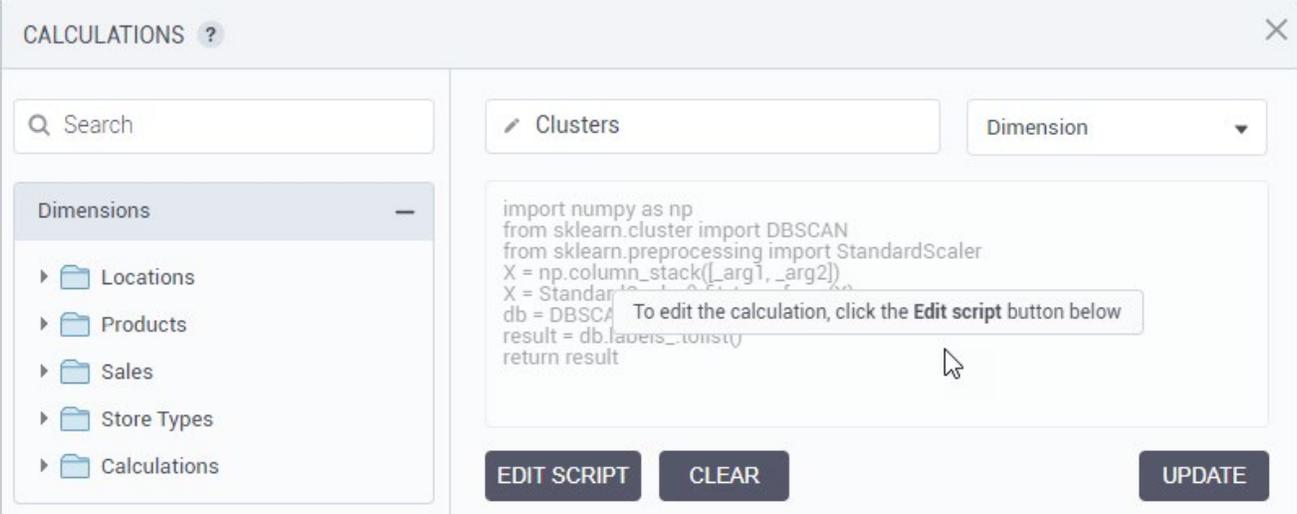
The aggregation options are now using the same naming conventions in visualizations, script calculations, and in the **Table** widget. The “Total” aggregation has been renamed to “Sum.”

When selecting aggregation for a column, you will see the full aggregation names like *Average* or *Maximum* on the menu. However, the visualizations’ data settings and layout use the abbreviated options, such as *(Avg)* or *(Max)*, which improve the readability of the labels.



Preview a script code

Previously, you could only preview the connector ID, output data type, and the columns selected for the script. Now, after adding code in the script editor, the code also appears in the **Calculations** pane as a preview. Additionally, when hovering over the preview area, you can see a message instructing how to open the script editor.



The screenshot shows the 'CALCULATIONS' pane in the DataClarity interface. On the left, there is a search bar and a 'Dimensions' list with folders for 'Locations', 'Products', 'Sales', 'Store Types', and 'Calculations'. The main area is titled 'Clusters' and has a 'Dimension' dropdown menu. A code preview is displayed in a text area, showing Python code for DBSCAN clustering. A tooltip message is overlaid on the code, stating: 'To edit the calculation, click the **Edit script** button below'. At the bottom of the pane, there are three buttons: 'EDIT SCRIPT', 'CLEAR', and 'UPDATE'.

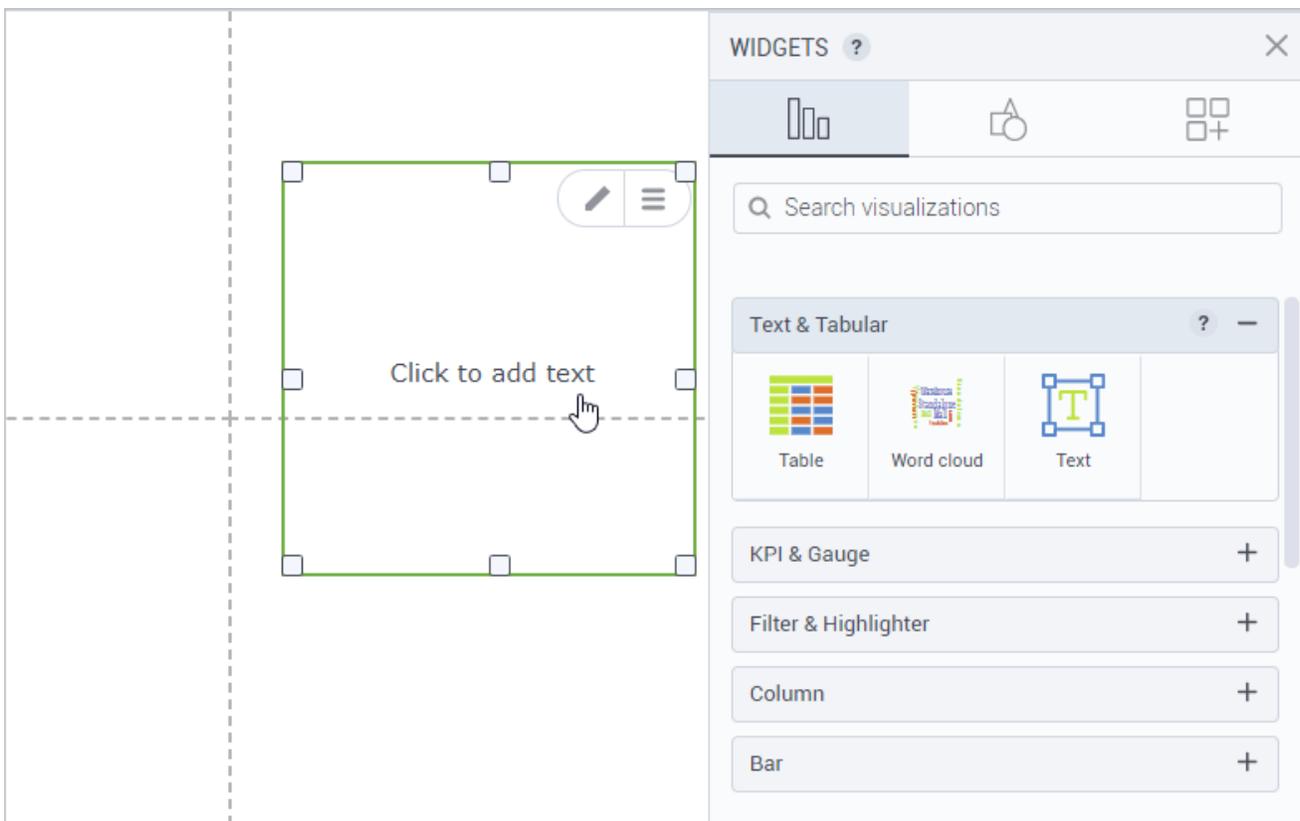
```
import numpy as np
from sklearn.cluster import DBSCAN
from sklearn.preprocessing import StandardScaler
X = np.column_stack([_arg1, _arg2])
X = StandardScaler().fit_transform(X)
db = DBSCAN(eps=0.5, min_samples=5)
result = db.fit_predict(X)
return result
```

The default size for some widgets

Now, instead of automatic snapping to a cell, some widgets have certain default sizes if you drag them from the **Widgets** pane to the page area:

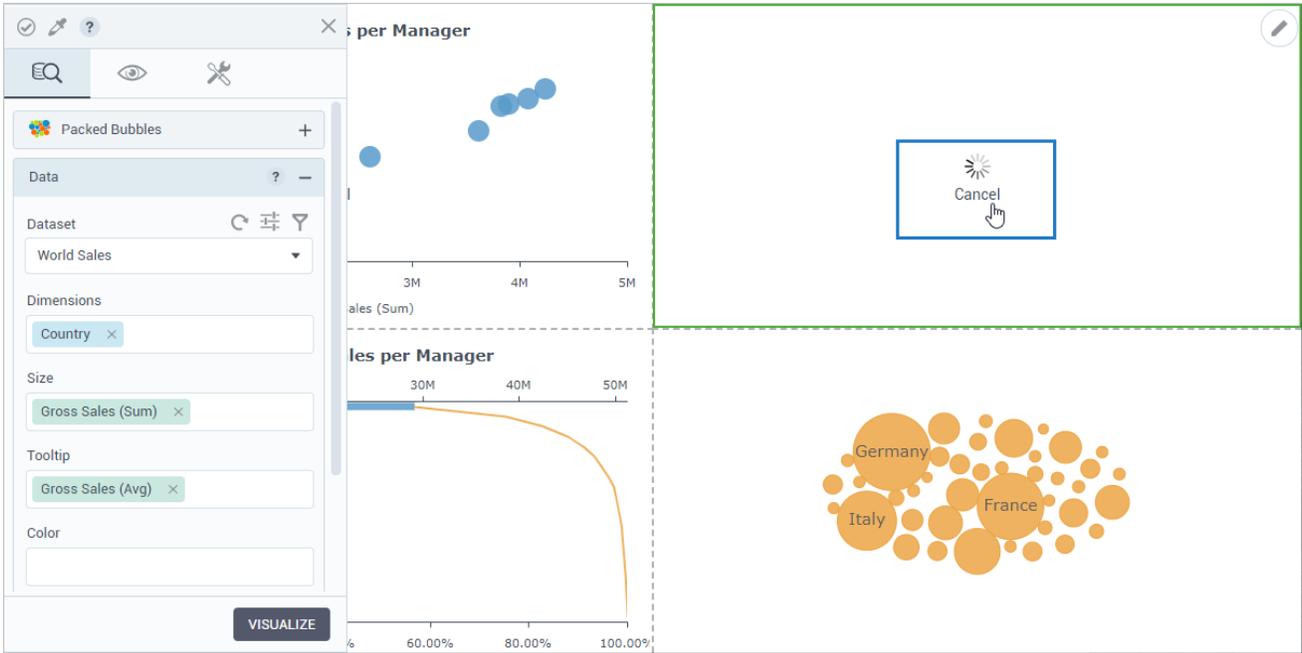
- Single Value (default size 250x250px)
- Image (default size 250x250px)
- Text (default size 250x250px)
- Html button (default size 200x100)
- Html code (default size 200x100)
- Html input (default size 300x100)

If you want any of these widgets to be automatically snapped, add them from within a cell.



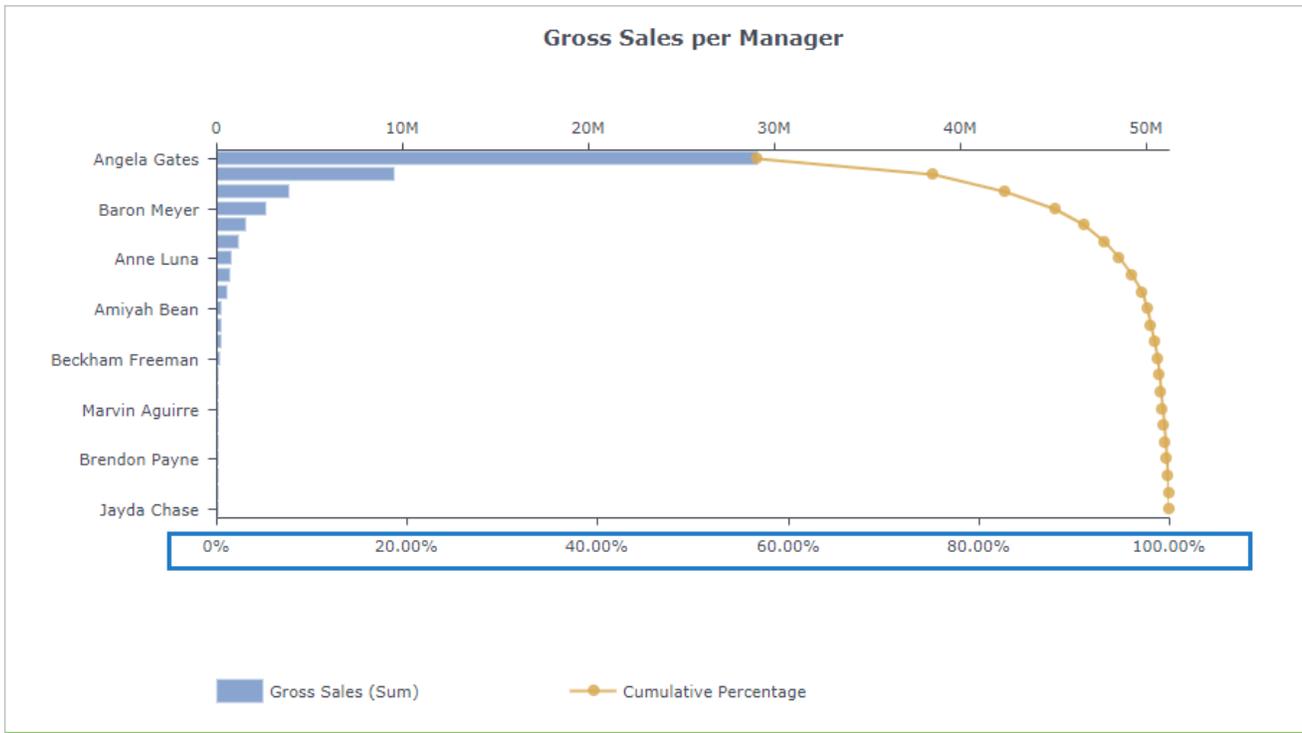
Cancel visualizing a widget

Previously, if a widget's data could not be visualized for some reason, the spinner was displayed infinitely with no possibility to cancel the query. Now, you can use the **Cancel** button to stop visualizing data when needed.



Percentage sign in Pareto charts

The Pareto charts now display a percentage sign after the numbers representing cumulative percentage on the respective axis and in tooltips.



DATA SCIENCE

Deploy custom Python libraries

Now, administrators can deploy the company's own custom Python libraries. The Python libraries location is now persistent, **python_server/scripts/**, where you can add any Python libraries needed for AI projects.

The libraries' location can be found in Kubernetes, under **Config and Storage > Persistent Volume Claims**. Click the **custom-python-script-pvc** and then open the volume name link. The **Path** is specified under the **Source** box.

INSTALLATIONS & CONFIGURATION

Deploy the platform behind the external SSL balancer

Administrators can now deploy the platform behind an external SSL balancer while the internal connection does not use SSL. The installation script now includes two new options:

- **Disable HTTPS** – This option allows to select whether to enable HTTPS.
- **Disable built-in SSL termination** – If an administrator enabled HTTPS, this option allows to place a gateway (reverse proxy) that will handle SSL offloading in front of the platform.

```
devops@      :~/distribution$ ./install.sh
Channel [production, beta] : production
Public entry point (IP address, domain name or hostname) : yourdomain
Add host alias to resolve "yourdomain" [y/n] : y
IP address for "yourdomain" : 12.97.253.46
Disable HTTPS [y/n] : n
Disable built-in SSL termination [y/n] : n
No SSL Certificate provided. Creating Self-Signed SSL Certificate...

Generating a RSA private key
.....+++++
.....+++++
writing new private key to 'certs/cert.key'
-----
Show current admin password of Kubernetes [y/n] : n
Change admin password of Kubernetes [y/n] : n
Apply to Kubernetes now [y/n] : █
```

Create a tenant in Access Manager

Now, master administrators can create a fully configured tenant directly from the Access Manager interface. Under the realms drop-down list, click **Add realm** and then adjust realm configuration settings as needed.

