



WHAT'S NEW AND RELEASE NOTES

Release: 2022.2

IN THIS RELEASE

- NEW FEATURES AND IMPROVEMENTS2**
- DATA PREPARATION2
 - New column options when refining a dataset.....2
 - Specify a Cognos connection name for specific Cognos data sources3
 - Optimization of Cognos calculations4
- STORYBOARDS.....5
 - Code editor improvements.....5
 - Smart layout guides for arranging widgets on a storyboard page9
 - Undo and redo authoring actions in a storyboard10
- INSTALLATION & CONFIGURATION11
 - Installation script improvements.....11
 - Python server enhancements and optimizations11

NEW FEATURES AND IMPROVEMENTS

DATA PREPARATION

New column options when refining a dataset

Starting with this release, when editing dataset columns on the **Step 2 – Refine** page, you can use the following additional column options:

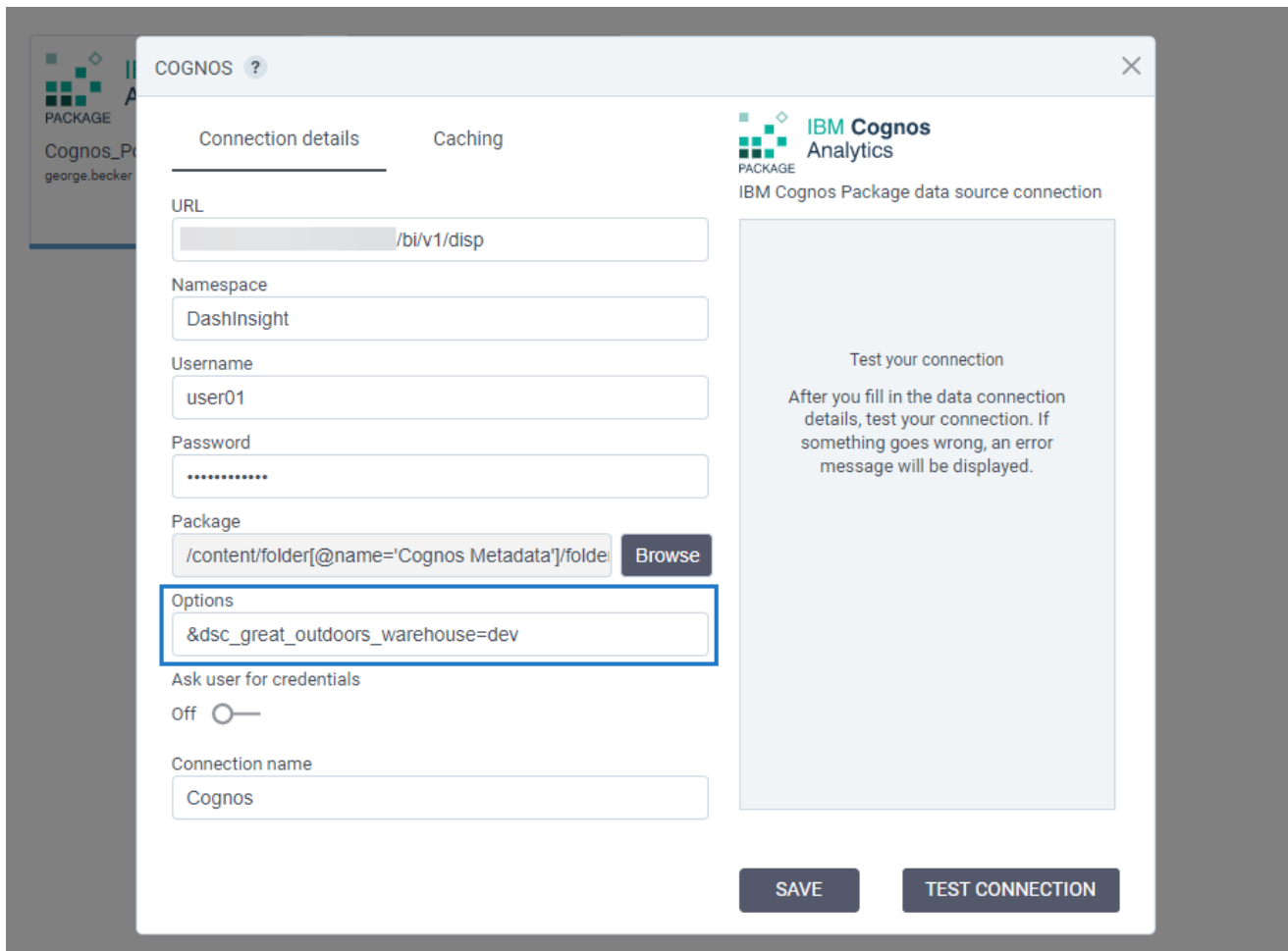
- **Hide** – To hide a column in a dataset. You can create calculations and filters based on the hidden columns. Hidden columns are not available in visualizations.
- **Column info** – To review various details for a column, such as aggregation, data type, whether it is used as a security filter, and description.

The **Step 2 – Refine** page now also includes the **Show hidden columns** checkbox to quickly preview the columns you decided to hide.

The screenshot displays the 'STEP 2 - REFINE' interface in DataClarity. The main table shows columns: Sales Date, Store Type ID, Location ID, Product ID, Items per Transaction, No of Transactions, No of C, column options, and entity. The 'Store Type ID' column is highlighted with a blue box. A context menu is open over the 'column options' header, listing actions: Rename, Exclude, Hide, Column info, Aggregation, Clean, and #% Format. The 'Hide' and 'Column info' options are highlighted with a blue box. A 'Show hidden columns' checkbox is checked and highlighted with a blue box. A blue arrow points from the 'Show hidden columns' checkbox to the 'Store Type ID' column. A 'Preview data' button is visible at the bottom center.

Specify a Cognos connection name for specific Cognos data sources

In some cases, Cognos packages can use a data source with multiple Cognos connections attached. When this happens, Cognos prompts a user to select a connection that will be used to query the data. DataClarity has extended the Cognos data driver to support specifying a Cognos connection name for a specific Cognos data source when it is needed. You can specify the mentioned data in the **Options** field. Moreover, you can set what date and time formats to be used for a connection.



Optimization of Cognos calculations

In this release, the query plan for calculations in Cognos data sources has been significantly optimized. As a result, all the related queries are executed more efficiently, allowing you to work with a significant number of complex calculations.

STORYBOARDS

Code editor improvements

The code editor has been improved to help you better manage the code that you need to add in DataClarity. You can now view the line numbering and color-coding for the keywords.

The code editor improvements are now available in the following places:

- Code editor of the **Python code** widget
- Code editor of the **HTML code** widget
- **Code manager** of the **Custom** widget
- **Script** editor when adding a new script or scalar calculation
- CSS and JavaScript custom code editor of a storyboard's page

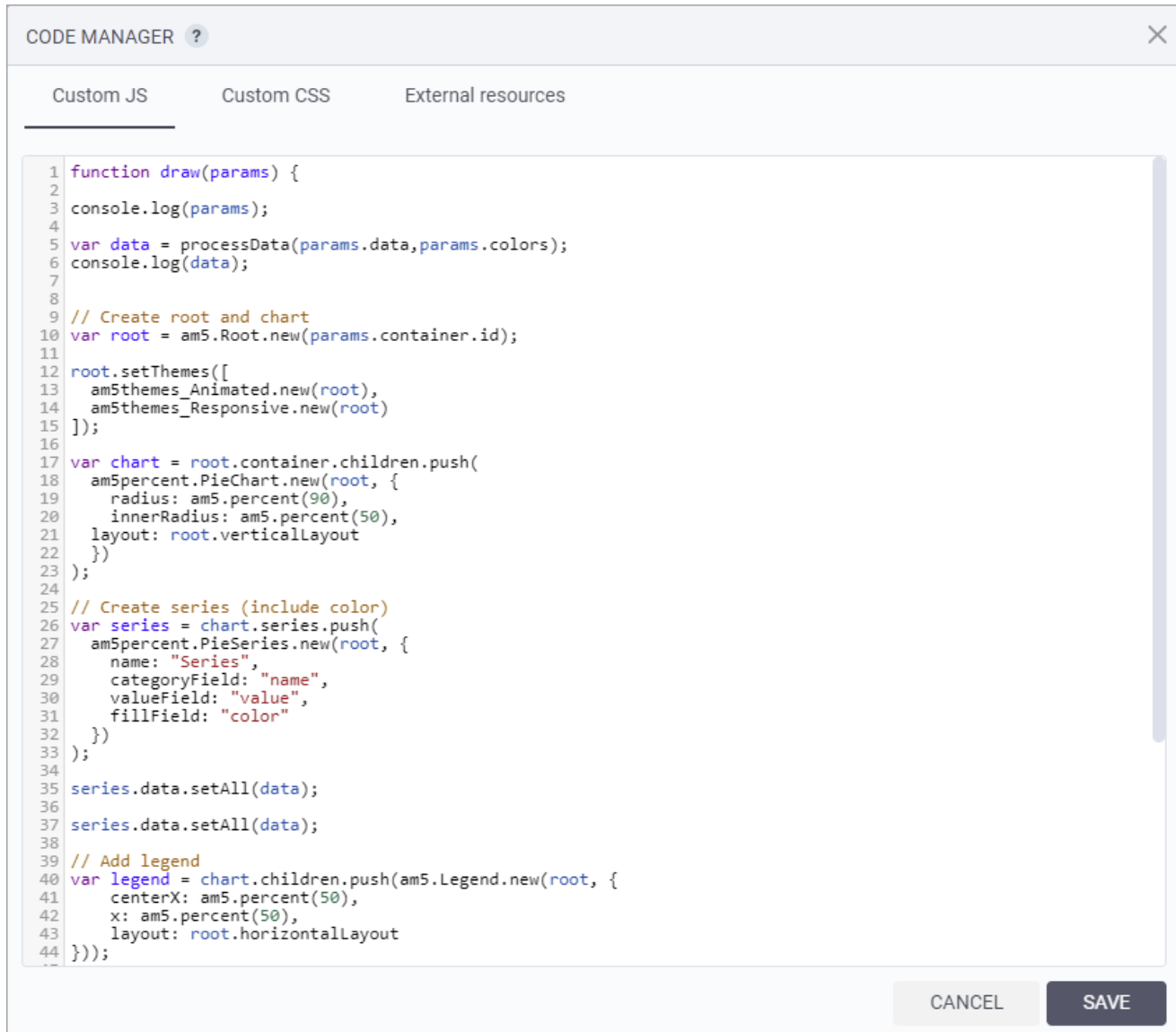
HTML code widget – Adding HTML code in the Code editor



The screenshot shows a window titled "HTML CODE" with a close button in the top right corner. The code editor contains the following code:

```
1 <script>
2   function Demo(){
3     alert("Good morning");
4   };
5 </script>
6 <div style="background:red; color:blue" onclick="Demo();">Click me</div>
```

At the bottom right of the window, there are two buttons: "CANCEL" and "SAVE".

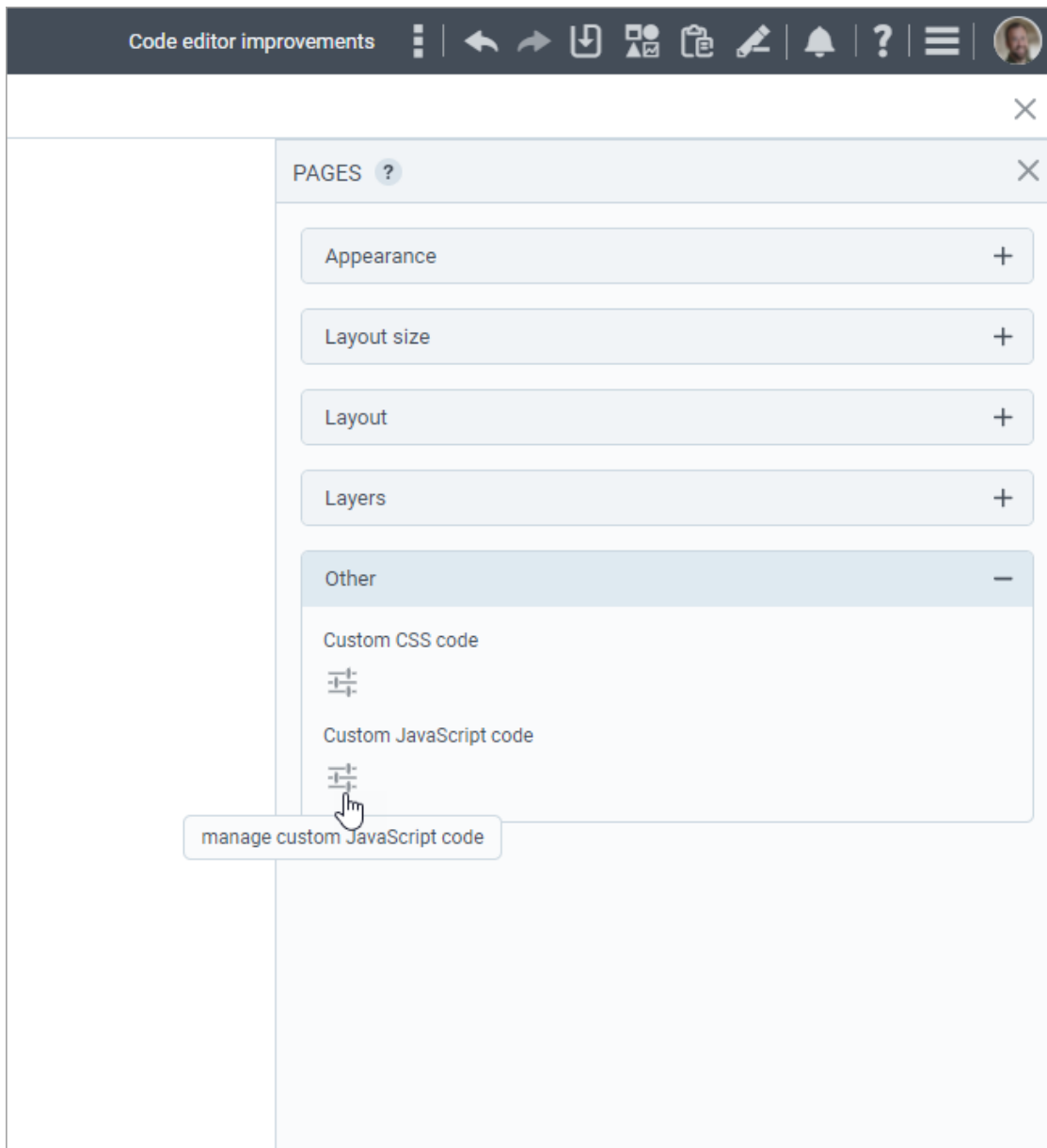
Custom widget – Adding a custom JS code in the Code manager

The screenshot shows a dialog box titled "CODE MANAGER" with a close button (X) in the top right corner. Below the title bar are three tabs: "Custom JS" (selected), "Custom CSS", and "External resources". The main area contains a text editor with the following JavaScript code:

```
1 function draw(params) {
2
3 console.log(params);
4
5 var data = processData(params.data,params.colors);
6 console.log(data);
7
8
9 // Create root and chart
10 var root = am5.Root.new(params.container.id);
11
12 root.setThemes([
13   am5themes_Animated.new(root),
14   am5themes_Responsive.new(root)
15 ]);
16
17 var chart = root.container.children.push(
18   am5percent.PieChart.new(root, {
19     radius: am5.percent(90),
20     innerRadius: am5.percent(50),
21     layout: root.verticalLayout
22   })
23 );
24
25 // Create series (include color)
26 var series = chart.series.push(
27   am5percent.PieSeries.new(root, {
28     name: "Series",
29     categoryField: "name",
30     valueField: "value",
31     fillField: "color"
32   })
33 );
34
35 series.data.setAll(data);
36
37 series.data.setAll(data);
38
39 // Add legend
40 var legend = chart.children.push(am5.Legend.new(root, {
41   centerX: am5.percent(50),
42   x: am5.percent(50),
43   layout: root.horizontalLayout
44 }));
```

At the bottom right of the dialog box, there are two buttons: "CANCEL" and "SAVE".

Pages pane – Customizing a storyboard's page



Script editor – Adding a script calculation

SCRIPT

Connector: Python (built-in) Calculation: Vector Result: Double

Data frame columns: No of Customers (Sum) Profit (Sum)

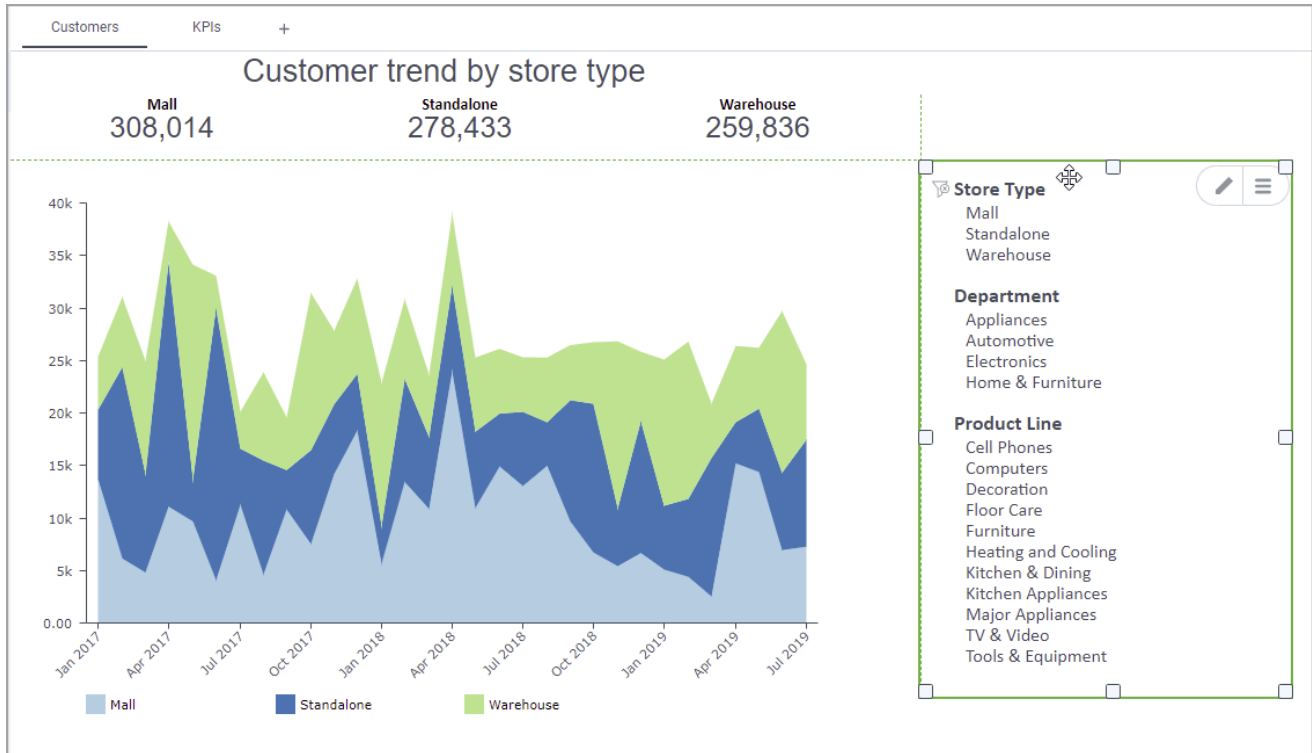
Script

```
1 import numpy as np
2 from sklearn.cluster import DBSCAN
3 from sklearn.preprocessing import StandardScaler
4 X = np.column_stack([_arg1, _arg2])
5 X = StandardScaler().fit_transform(X)
6 db = DBSCAN(eps=1, min_samples=3).fit(X)
7 result = db.labels_.tolist()
8 return result
```

PREVIEW RESULT APPLY

Smart layout guides for arranging widgets on a storyboard page

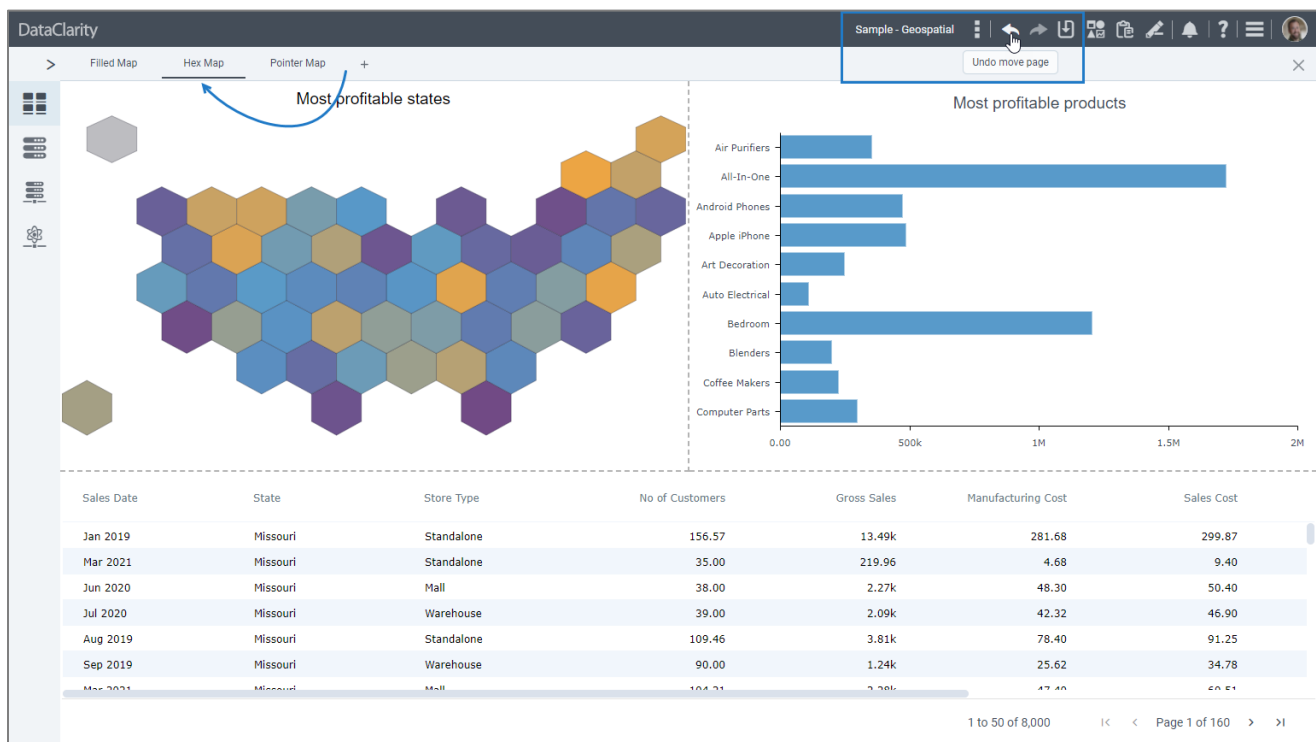
Starting from this release, it's easier to create pixel-perfect storyboard pages by using the new smart layout guides. When moving or resizing widgets on a storyboard page, you can notice the green dashed lines—smart guides—that help you align widgets vertically and horizontally relative to other widgets. Smart guides also appear between objects or near the edges of the page cells so that you can space out widgets evenly.



Undo and redo authoring actions in a storyboard

Previously, you could undo and redo most of the authoring actions performed in a storyboard. Starting with this release, the list of supported actions has been extended, and now, you can also undo or redo the following actions:

- Customize the appearance of page tabs (such as background, fonts, and colors)
- Reorder the page tabs
- Add a new page
- Change settings for the widgets that do not use a dataset (such as the Text or Image widgets)
- Add, edit, delete, and select actions
- Add, edit, delete, and apply filters for the Filter widget
- Add, edit, delete, and apply the highlighting for the Highlight widget
- Change widget's content actions and preferences (such as sorting or filtering the top performers)
- Set the current widget's style as default
- Apply the copied appearance style to a widget



INSTALLATION & CONFIGURATION

Installation script improvements

The commands and prompts in the installation script (install.sh) now include short explanations to clarify the available configuration options. Moreover, where applicable, the script provides default values to streamline the installation process.

```
devops@dc-ubun-vm08:~/repo/DevOps/misc/on-prem/k8s/distribution$ ./install.sh
Select the software package [develop, beta] (Default: develop)
Enter 'develop' to install the latest stable version or 'beta' to install the pre-release version: develop
Enter the public entry point (IP, host, FQDN) (Default: dc-ubun-vm08): 10.10.10.211
Disable HTTPS? [y/n] (Default: n): n
Disable built-in SSL termination? [y/n] (Default: n)
By choosing 'y', the platform will use an insecure connection to communicate internally. Use this option if you want to place a gateway, which will handle SSL offloading, in front of the platform.
No SSL Certificate provided. Creating Self-Signed SSL Certificate...
Generating a RSA private key
.....+++++
.....+++++
writing new private key to 'certs/cert.key'
-----
Enter the Postgres server address: 10.15.16.211
Disable authentication for the Kubernetes Dashboard (not recommended if you are using a public entry point)? [y/n] (Default: n):
Display the access token for the Kubernetes Dashboard? [y/n] (Default: n):
Install DataClarity now? [y/n]: y
```

Python server enhancements and optimizations

The Python server has been enhanced and upgraded to use Python v3.9 and TabPy v2.5.