Introducing Db2 Web Query

Release 2.3

Contents

Chapter 1

Welcome to Db2 Web Query

Db2 Web Query is the reporting engine that turns corporate data into meaningful information. It enables any executive, manager, analyst, employee, partner, or customer to execute smarter decision-making. The platform delivers many rich new capabilities and integrates the four critical elements that any enterprise needs: business intelligence, performance management, advanced analytics, and data integrity.

Depending on the role you play in your organization, using Db2 Web Query, you can:

- Connect and edit data
- Author reports and visualizations
- Perform predictive and location analytics
- Build applications
- Run and schedule procedures
- And more

In this chapter:

Signing in to Db2 Web Query

You can launch Db2 Web Query from any web browser, using the address created during the installation. The sign-in page and default Home Page may be different depending on the configuration requirements of your organization.

Procedure: How to Launch Db2 Web Query

1. In your browser, go to the following page:

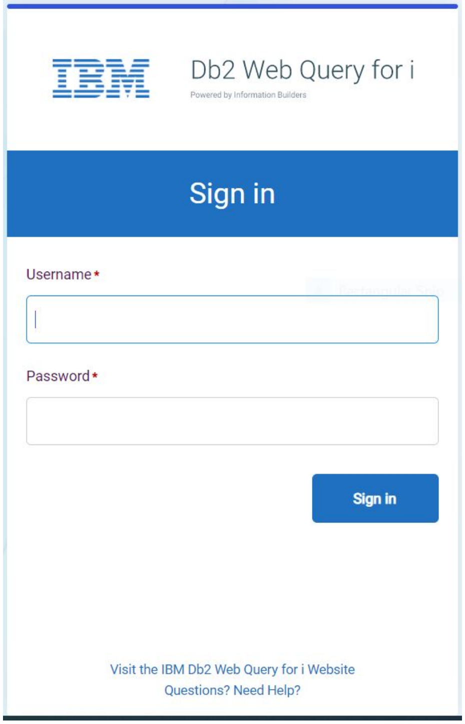
   `http[s]://`*`hostname`*`:`*`port`*`/webquery`

where:

hostname:*port*

Are the host name and HTTP port of the Db2 Web Query web server.

The default Db2 Web Query Sign In page opens, as shown in the following image.



2. Type your user name and password.
3. Click *Sign in*.

   The Home Page loads and shows all the content that you are authorized to see.

Navigating the Db2 Web Query Home Page

The Home Page is the default landing page that opens when you first sign in to Db2 Web Query. It serves as a centralized place for working with your data, creating and organizing content, scheduling reports and procedures, and performing administrative tasks.

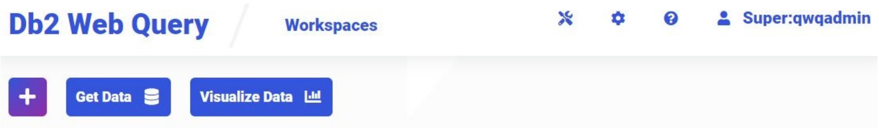
The main elements of the Home Page are:

❑ Banner

- Home view
- Workspaces view

Banner

The banner, shown in the following image, is located on the top of the Home Page. It is accessible from every view of the Home Page.



The Db2 Web Query logo is located in the left corner of the banner. By clicking the logo, you return to the default view of the Home Page. The Workspaces view provides the hierarchical view of the repository and gives access to the Action Bar, where you can upload files, access the Db2 Web Query Server, and create standalone pages, InfoAssist content, URLs, and more.

The three buttons underneath the Workspace view provide access to your data and Db2 Web Query Designer, where you can visualize this data.

The plus button opens a menu with two sections:

- **Visualize Data.** Gives you the choice of two visualization modes.
  - **Create New Visualization.** Allows you to start creating content for your visualization.
  - **Assemble Visualizations.** Opens Db2 Web Query Designer in a page where you can assemble a visualization from existing content.
- **Manage Data.** Provides access to the following options.
  - **Get Data.** Opens the Get Data dialog box, where you can choose a source for your data.
  - **Prepare and Manage Data.** Navigates to the Db2 Web Query Server in a new browser window, where you can view and manage your existing data.

Next to the plus button are the two easy-access buttons:

- **Get Data.** Opens the Get Data dialog box, where you can choose a source for your data.
- **Visualize Data.** Opens Db2 Web Query Designer in the content mode, where you can start creating visuals for your visualization.

In the right corner of the banner, you can access the following buttons and menus.

Utilities

The Utilities button provides access to the following menus.

- **FOC Message Lookup.** Displays the text or explanation for any error message.
- **Stop Requests.** Terminates outstanding interactive requests that are running.
- **Session Viewer.** Opens the Session Viewer in a new browser window, which shows the status of the current session.
- **Runtime Environment.** Provides a way for DB2 Web Query users to quickly, easily, and dynamically change the user portion of a library list prior to retrieving the data from the database.
- **Report Broker Explorer.** Opens the Report Broker Explorer, where you can view the status of your scheduled jobs and log reports.
- **Report Broker Status.** Allows authorized users to view the status of the Report Broker Distribution Server and perform configuration and administration tasks.
- **Change Management.** Provides access to the Import and Export facilities within the Change Management feature, which is used for moving application components between Db2 Web Query environments.

Settings

The Settings button provides access to the following menus.

- **Security Center.** Opens the Security Center where you can define users and groups.
- **Administration Console.** Launches the Administration Console in the new browser window, where you configure, administer, and monitor the components of the Db2 Web Query environment.
- **Db2 Web Query Server.** Launches the Db2 Web Query Server in the new browser window, where you can view and modify data sources, view logs and statistics, create and modify applications, upload raw data, and more.
- **Manage private resources.** Opens the Manage Private Resources window, where you can view and administer private resources for groups and users.
- **Normal view.** The view of the repository that allows you to see and manage your own private content. This setting is enabled by default.

- **Administration view.** The view of the repository that allows you to see and manage the private content of other users.

Help

The Help button provides access to the following menus.

- **Db2 Web Query Online Help.** Opens the Help plugin in a new browser window.
- **Technical Resources.** Opens the KnowledgeBase in a new browser window.
- **Community.** Launches the Customer Community page, where you can access Focal Point forums, announcements, and user groups information.
- **About Db2 Web Query.** Opens the About Db2 Web Query window, which contains information about the current release of the product.

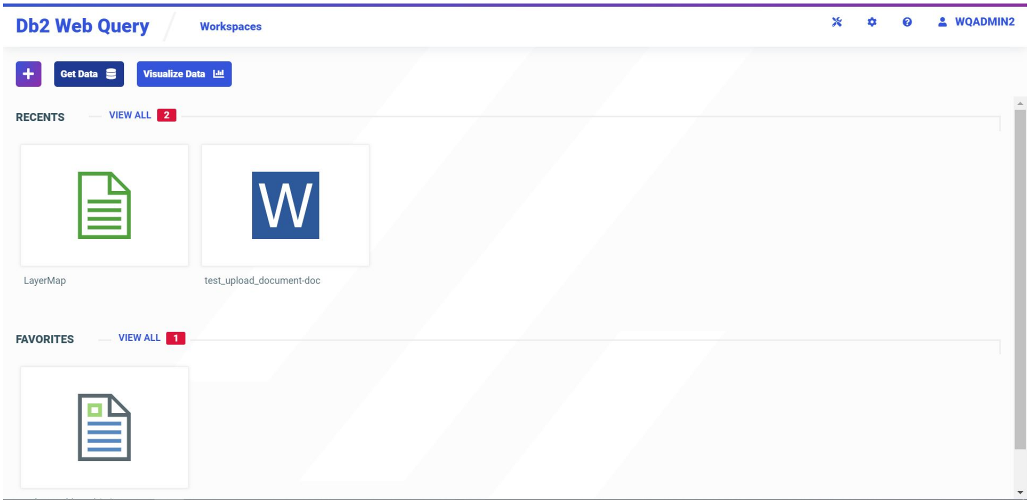
User Menu

The User Menu contains the following action menus.

- **Preferences.** Opens the My Preferences window, where you can change the language of the Home Page.
- **Change Password.** Opens the Change Password window, where you can reset a password for your account.
- **Sign Out.** Signs you out of Db2 Web Query.
- **Legacy Home Page.** Opens the Db2 Web Query legacy Home Page.

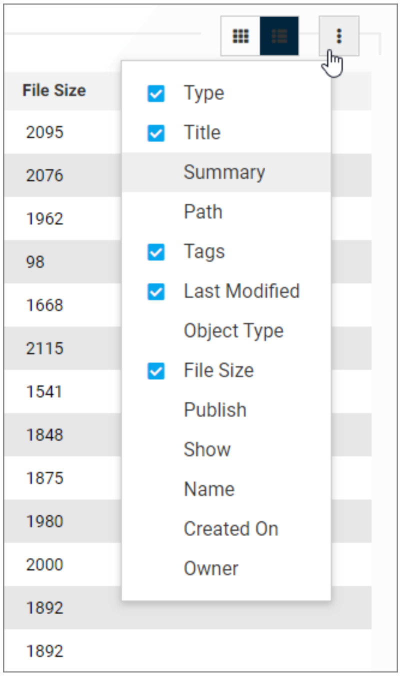
Home View

The home view of the Home Page is what you see when you first login to Db2 Web Query, and later when you click the Db2 Web Query logo. This view provides quick access to the items that you worked on and your favorite content that is currently available to you in the repository. The home view is shown in the following image.



When you create a new item, this item appears in the Recents section. You can scroll through the carousel to view additional items, or click *View All* to access a complete list of recent items. Saving content or items as favorites, automatically adds them to the Favorites section of the Home Page. To save an item as a Favorites, right-click the item and then click *Add to Favorites*. To remove an item from Recents or Favorites, right-click the item and then click *Remove from Recents* or *Remove from Favorites*.

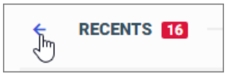
You can cycle through each of these sections using the carousel controls that appear on the edge of each group of slides when you hover your cursor over it. You can also click the *View All* link, which opens the entire list of items in a carousel. You can sort the list by clicking any column title. You can also specify which columns display in the list by clicking the *Choose Columns* button and selecting the desired columns from the list. An example of an expanded columns menu is shown in the following image.



File Size
2095
2076
1962
98
1668
2115
1541
1848
1875
1980
2000
1892
1892

- ☑ Type
- ☑ Title
- Summary
- Path
- ☑ Tags
- ☑ Last Modified
- Object Type
- ☑ File Size
- Publish
- Show
- Name
- Created On
- Owner

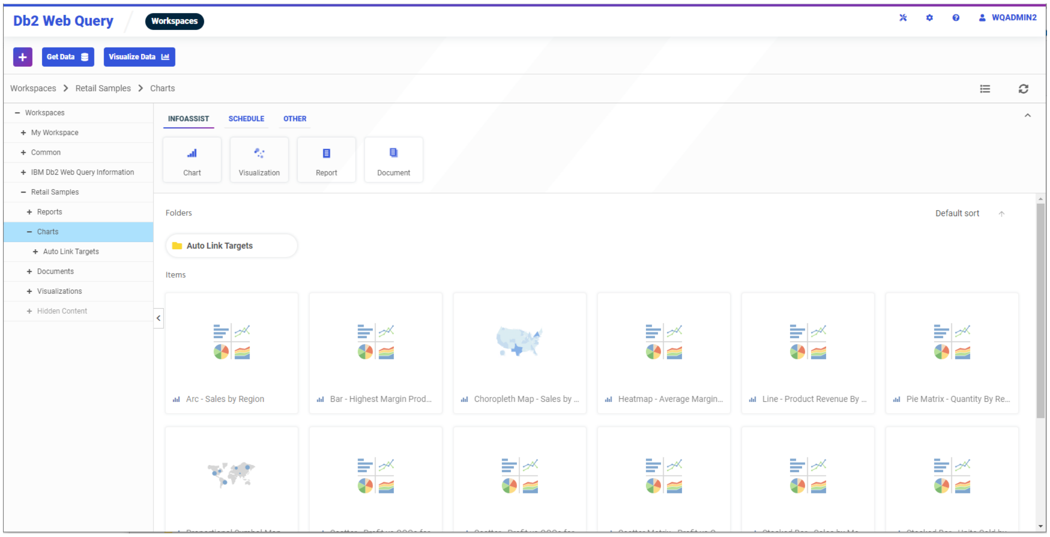
To return to the home view, click the back arrow next to the title of the list, as shown in the following image, or click the Db2 Web Query logo.



Additionally, you can right-click any item in the home view to access shortcut menu options that are available to you, based on your role and permissions.

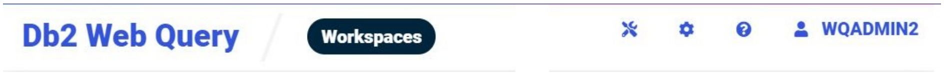
Workspaces View

The Workspaces view is the most comprehensive component of the Home Page. Here, you can organize your repository structure, create and interact with content, create sample content, work with data, and perform other tasks. The Workspaces view is shown in the following image.



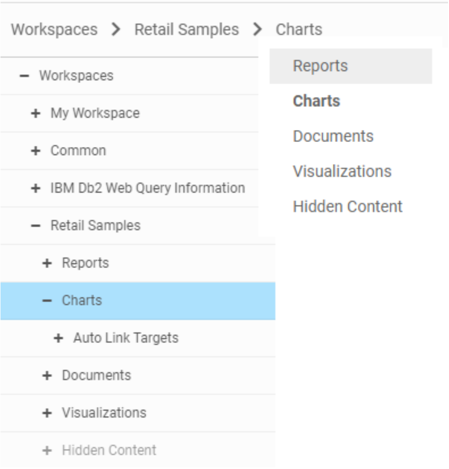
Navigation Bar

The navigation bar is shown in the following image.



You can use the navigation bar to move between folders in your repository, search the contents of a workspace or folder, change the view of items in the content area, and refresh content in a workspace or folder you select.

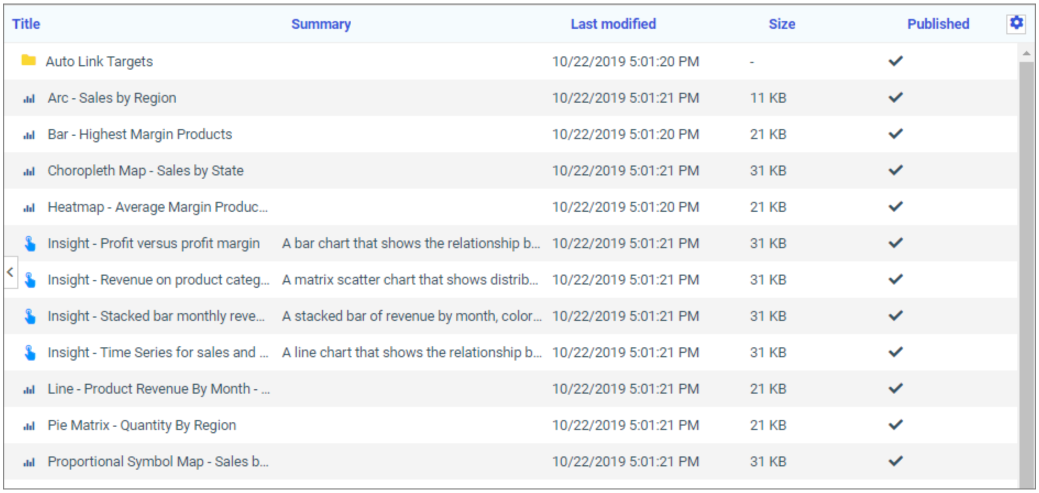
When you drill into the folders in your repository, the navigation bar generates a breadcrumb trail that you can use to move between folders. You can move back one folder or multiple folders in a single click. You can click a folder name to view the contents of that folder. You can also click arrows between the folders to see the file structure that is currently displaying, as shown in the following image. In this example, the selected folder appears in bold.



You also can toggle the view of your content between Grid view and List view and refresh all content in a workspace or folder. By default, content displays in Grid view. Click the *Grid view* icon ▦ or *List view* icon ☰ to change your view.

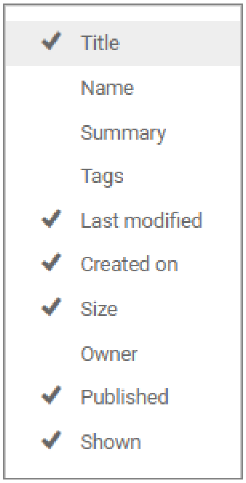
Grid view allows you to see the thumbnails of your items, which you can customize to add transparency to your display. When you hover over a thumbnail image in Grid view that contains a summary, the image fades and the summary displays.

The List view is useful when you need to see a large number of items at a glance. You can sort your items by various criteria. An example of a List view is shown in the following image.



Title	Summary	Last modified	Size	Published
Auto Link Targets		10/22/2019 5:01:20 PM	-	✓
Arc - Sales by Region		10/22/2019 5:01:21 PM	11 KB	✓
Bar - Highest Margin Products		10/22/2019 5:01:20 PM	21 KB	✓
Choropleth Map - Sales by State		10/22/2019 5:01:21 PM	31 KB	✓
Heatmap - Average Margin Produc...		10/22/2019 5:01:20 PM	21 KB	✓
Insight - Profit versus profit margin	A bar chart that shows the relationship b...	10/22/2019 5:01:21 PM	31 KB	✓
Insight - Revenue on product categ...	A matrix scatter chart that shows distrib...	10/22/2019 5:01:21 PM	31 KB	✓
Insight - Stacked bar monthly reve...	A stacked bar of revenue by month, color...	10/22/2019 5:01:21 PM	31 KB	✓
Insight - Time Series for sales and ...	A line chart that shows the relationship b...	10/22/2019 5:01:21 PM	31 KB	✓
Line - Product Revenue By Month - ...		10/22/2019 5:01:21 PM	21 KB	✓
Pie Matrix - Quantity By Region		10/22/2019 5:01:21 PM	21 KB	✓
Proportional Symbol Map - Sales b...		10/22/2019 5:01:21 PM	31 KB	✓

You can sort the list by clicking any column title. Each column features a three-step toggle sorting: ascending, descending, and return to the default. You can also specify which columns display in the content area by clicking the *Choose Column* button, and selecting the columns you want to view from the list. An example of an expanded columns menu is shown in the following image.



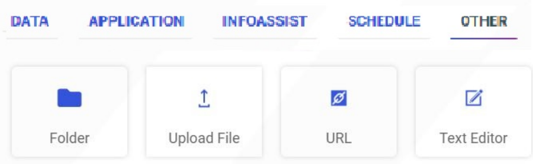
✓ Title
Name
Summary
Tags
✓ Last modified
✓ Created on
✓ Size
Owner
✓ Published
✓ Shown

In both the Grid and List views, you can see the additional information about your items, represented with special icons.

To refresh all content in a selected workspace or folder, click Refresh ⟳.

Action Bar

You can use the Action Bar to create analytical content with InfoAssist, upload files and spreadsheets, connect to data, build pages, schedule procedures, and generate sample content using the options available to you for a workspace or folder. The following image shows an example of the Action Bar an advanced or basic user has access to for a workspace or folder.



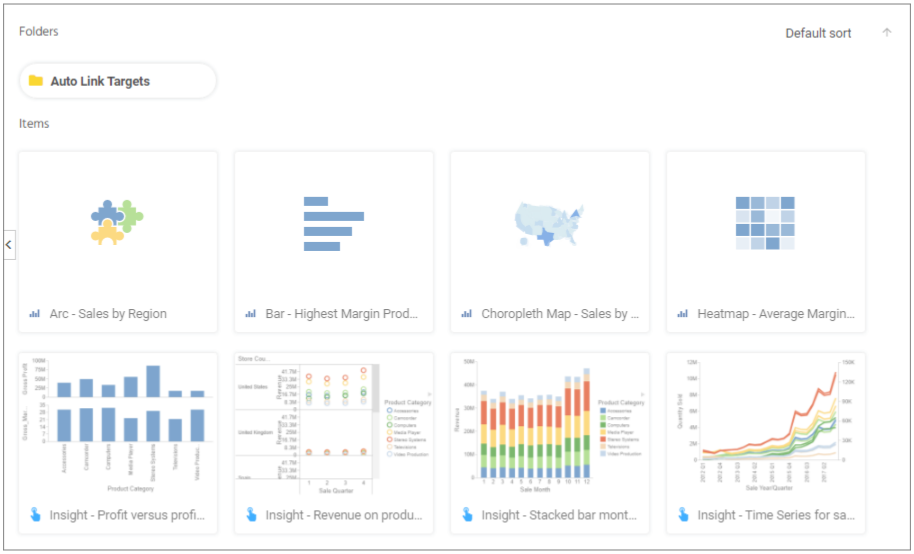
The default view of the Action Bar displays the most common options available to you.

The Action Bar only displays the items that you can create for the workspace or folder that is currently selected. If you click the Workspaces node, the Action Bar only displays the Workspace and Folder options. You can only see the items that are available to you based on your role.

You can collapse the Action Bar to make more room for the content area. To collapse the Action Bar, click the arrow in the upper-right corner of the Action Bar.

Content Area

The content area displays the items contained within a workspace or folder that is currently selected. An example of a content area is shown in the following image.


Folders
Default sort
Auto Link Targets
Items
Arc - Sales by Region
Bar - Highest Margin Prod...
Choropleth Map - Sales by ...
Heatmap - Average Margin...
Insight - Profit versus profi...
Insight - Revenue on produ...
Insight - Stacked bar mont...
Insight - Time Series for sa...

If a resource contains folders and items, the folders are always displayed first, followed by the items. Items shown in the content area include reports, charts, pages, and more.

You can sort your content by using the two sorting menus in the upper-left corner of the content area. One allows you to sort by various criteria, such as title, summary, size, last modified, and published dates. The other menu toggles between the descending and ascending orders. You can also create a customized sorting order by configuring the Sort Order property for the item. If the sort order is configured, the Default Sort option is added to the menu.

You can right-click any folder or item to access a shortcut menu of options. The following options are available when you right-click a folder or item in the content area.

Expand/Collapse

Expands or collapses a folder. This option is only available for folders.

Run

Runs the report procedure or schedule.

Run in new window

Runs the selected item in a new browser window or tab.

Run with SQL trace

Runs a report and includes an SQL trace of the report results.

Note: The Run with SQL Trace option is not available for visualizations.

Schedule

Launches the scheduling tool, where you can schedules a report procedure to be run. Scheduling options include Email, FTP, Printer, and Repository.

View log

Launches the Schedule Log Options dialog box, where you can select whether to view the most recent log report for the schedule, or specify a time range for which to obtain log reports. This option is only available for schedules.

View

Enables you to launch the URL or download a copy of the file to your browser. This option is available for URLs, Excel Workbooks, PDF, HTML, and Image Files.

Edit

Edits an item with the tool that was used to create it.

Edit with Designer

Edits an item in Db2 Web Query Designer. This option is only available for charts that were initially created in InfoAssist.

Edit with text editor

Edits an item in the Text Editor.

Duplicate

Makes a copy of the folder or item with an underscore and number appended to the title, for example, duplicating Sales_Report creates Sales_Report_1 in the same location. The number is incremented each time a duplicate is created.

Cut, Copy, and Paste

Cuts, copies, or pastes the folder and its contents or item.

Delete

Deletes the folder and its contents or item.

Publish/Unpublish

Allows or suppresses public access to the content.

Show/Hide

Shows or hides the folder or item.

Security

Provides access to the following option:

Owner

Allows you to set the resource owner, either Published (no specific owner), or a specific group or user.

Add to Favorites

Adds the item to the Favorites folder.

Remove from Favorites

Removes the Favorite from the list. This option is only available for Favorites.

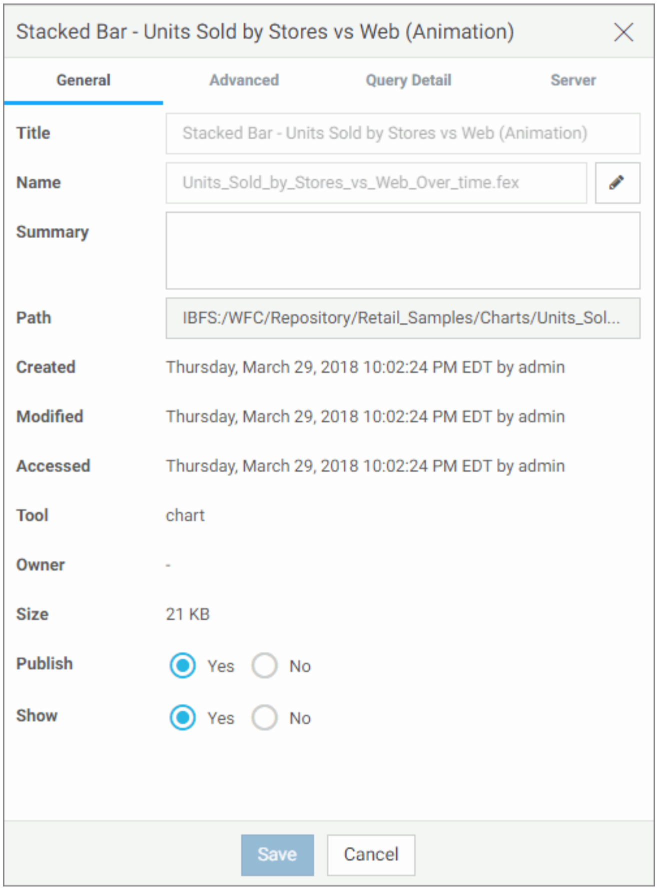
Properties

Opens the Properties panel to display the properties of the folder or item.

Folder and Item Properties

Authorized users can select the Properties option to review information about a folder or item, including when it was created, the date it was last modified, title, name, and settings that control the Db2 Web Query Server and applications it will access. There are also settings that control the functionality available to users when they access the folder, run, or schedule an item.

The following image shows the Properties panel for a stacked bar chart.



Stacked Bar - Units Sold by Stores vs Web (Animation)

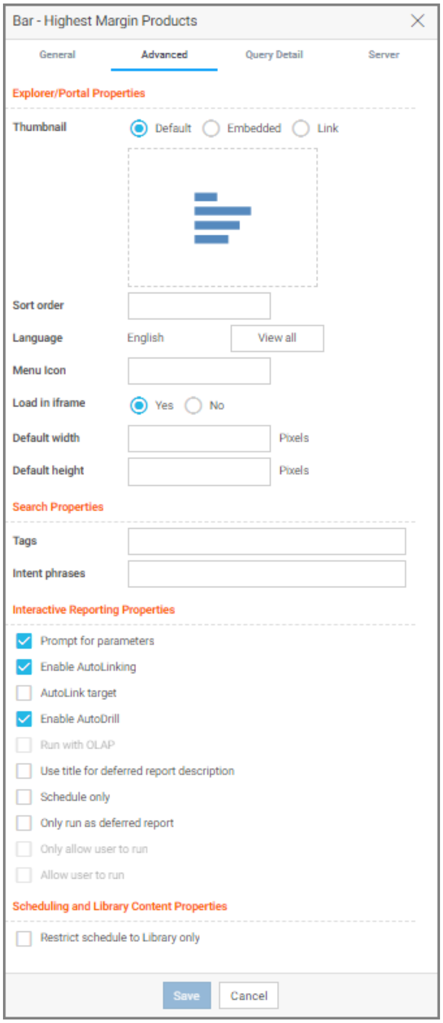
Title	Stacked Bar - Units Sold by Stores vs Web (Animation)
Name	Units_Sold_by_Stores_vs_Web_Over_time.fex
Summary	
Path	IBFS:/WFC/Repository/Retail_Samples/Charts/Units_Sol...
Created	Thursday, March 29, 2018 10:02:24 PM EDT by admin
Modified	Thursday, March 29, 2018 10:02:24 PM EDT by admin
Accessed	Thursday, March 29, 2018 10:02:24 PM EDT by admin
Tool	chart
Owner	-
Size	21 KB
Publish	● Yes ○ No
Show	● Yes ○ No

The General Properties tab includes the following settings:

- **Title.** Displays the value that you can use to identify the content in the Resources tree.
- **Name.** Refers to the unique name of the item or folder. Click the *Edit* button ✎ to change the name.
- **Summary.** Displays a detailed explanation that provides additional information about the item.
- **Path.** Shows the full path of the folder or item in the Db2 Web Query repository.
- **Created.** Shows the date the folder or item was created.

- **Modified.** Shows the date the folder or item properties were last modified, and identifies the user that modified them.
- **Accessed.** Shows the date the folder or item was last opened, and identifies the user that opened it.
- **Tool.** Specifies the type of an item.
- **Owner.** Identifies the user who created the folder or item.
- **Size.** Displays N/A bytes for folders. For items, displays the size in bytes.
- **Publish.** Provides an option to publish or unpublish the folder or item.
- **Show.** Provides an option to show or hide the folder or item.

The Advanced tab, which is shown in the following image, allows you to customize the functionality and behavior of the folder or item.



Bar - Highest Margin Products

Explorer/Portal Properties

Thumbnail ● Default ○ Embedded ○ Link

Sort order

Language English

Menu Icon

Load in iframe ● Yes ○ No

Default width Pixels

Default height Pixels

Search Properties

Tags

Intent phrases

Interactive Reporting Properties

☑ Prompt for parameters

☑ Enable AutoLinking

□ AutoLink target

☑ Enable AutoDrill

□ Run with OLAP

□ Use title for deferred report description

□ Schedule only

□ Only run as deferred report

□ Only allow user to run

□ Allow user to run

Scheduling and Library Content Properties

□ Restrict schedule to Library only

The Advanced Properties tab includes the following properties.

Explorer/Portal Properties:

- **Thumbnail.** Allows you to preview and change the thumbnail for an item.
- **Sort order.** Specifies the order to list the folder in the tree or item within other items in a folder.

- **Language.** Indicates the current language of your environment.
- **View All.** Opens the Language Properties dialog box, where you can select a language.
- **Load in iframe.** Grants or suppresses a permission to load an item in an iFrame container.
- **Default width.** Sets a default width value for the item.
- **Default height.** Sets a default height value for the item.

Search Properties:

- **Tags.** Adds tag values to the item that display in the search results.

Interactive Reporting Properties:

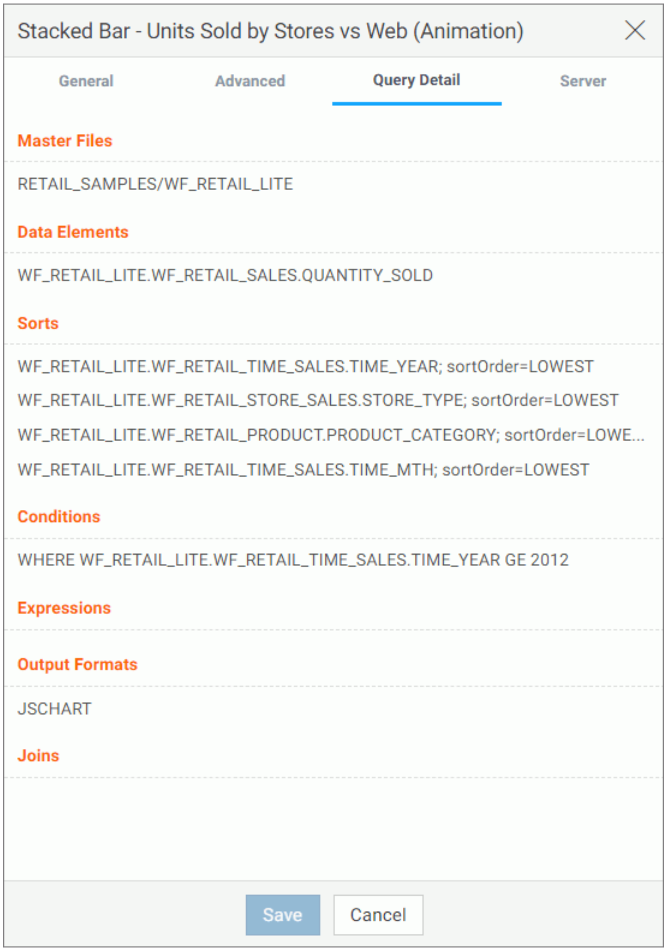
- **Prompt for parameters.** Indicates whether to display the Autoprompt HTML page to allow the user to view and select parameter values. When this check box is not selected, users will still be prompted if there are any parameters that do not have a value supplied.
- **Enable AutoLinking.** Establishes drill-down links from column data to the target reports that you link together.
- **AutoLink target.** Identifies a report as the target report that is run when the drill-down link is triggered.

  Note: The Auto Linking between reports only works if it the target report contains a parameterized field that corresponds to one of the fields in the primary Auto Link report.
- **Enable AutoDrill.** Creates drill-down links in reports and charts that contain hierarchical data.
- **Run with OLAP.** OLAP-enables the report.
- **Schedule only.** Indicates that the report can only be scheduled using Report Broker when the user does not have write access.
- **Only allow user to run.** Reserved for future use.
- **Allow user to run.** Reserved for future use.

Scheduling and Library Content Properties:

- **Restrict schedule to Library only.** Prevents emails distribution for security-sensitive scheduled reports, and restricts them to the library distribution only.

The Query Detail Properties tab, which is shown is the following image, is only available for reports, charts, and visualizations. It shows which data elements, parameters, and conditions were used in an item.



Master Files

RETAIL_SAMPLES/WF_RETAIL_LITE

Data Elements

WF_RETAIL_LITE.WF_RETAIL_SALES.QUANTITY_SOLD

Sorts

WF_RETAIL_LITE.WF_RETAIL_TIME_SALES.TIME_YEAR; sortOrder=LOWEST
WF_RETAIL_LITE.WF_RETAIL_STORE_SALES.STORE_TYPE; sortOrder=LOWEST
WF_RETAIL_LITE.WF_RETAIL_PRODUCT.PRODUCT_CATEGORY; sortOrder=LOWE...
WF_RETAIL_LITE.WF_RETAIL_TIME_SALES.TIME_MTH; sortOrder=LOWEST

Conditions

WHERE WF_RETAIL_LITE.WF_RETAIL_TIME_SALES.TIME_YEAR GE 2012

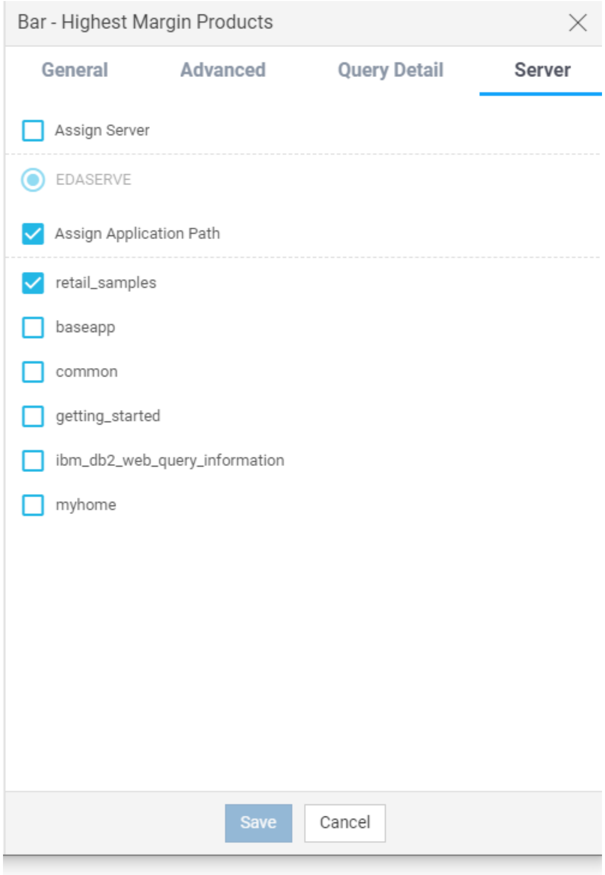
Expressions

Output Formats

JSCHART

Joins

The Server tab, which is shown in the following image, allows you to manage the server and application path settings.



□ Assign Server

● EDASERVE

☑ Assign Application Path

☑ retail_samples
□ baseapp
□ common
□ getting_started
□ ibm_db2_web_query_information
□ myhome

To change the server, select the *Assign Server* check box, and then select a server from the list. You can also change the application path in the same way.

By default, the Properties panel remains open while you browse between folders and items, allowing you to configure several resources quickly without closing the panel. To close the Properties panel, click the *Close* button in the upper-right corner of the panel.

Configuring and Using the Db2 Web Query Home Page

From the Db2 Web Query Home Page, you can create new workspaces or folders to organize content in your environment. You can also add thumbnails to view content easily, and add content to the Favorites view. Additionally, you can create URLs and other essential items that can enhance your content, if these items are enabled by your administrator.

This topic provides information about how you can create workspaces, configure folders, and further customize and use your Db2 Web Query Home Page.

Creating a New Workspace Folder

Your Home Page is the central location for all Db2 Web Query tasks. Setting it up in a way that is convenient, secure, and efficient helps you get the most out of your work. The Workspaces view of the Home Page opens the content area, and provides access to the available workspaces. Workspaces and folders that are located in the Resources tree organize all of your content items. Authorized users can create additional workspaces or folders, as well as customize the default structure of their view.

Note: The full view of the resources, in which the servers are listed as folders, is only available from the Legacy Home Page. To access the full view, on the Legacy Home Page, right-click *Workspaces*, point to *View*, and then click *Full View*.

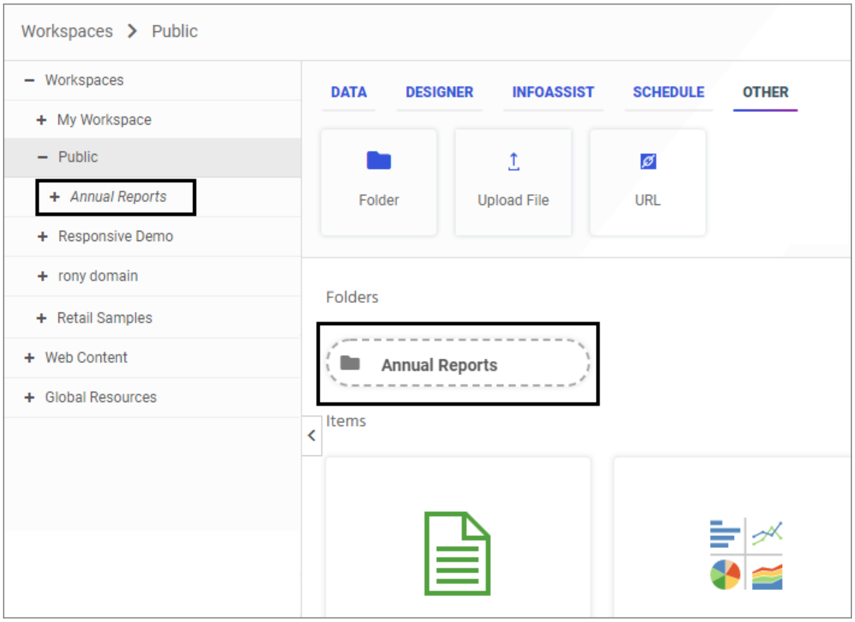
Procedure: How to Create a Folder

1. In the Resources tree, select the workspace where you want to create a folder.
2. On the Action Bar, click *Other*, and then click *Folder*.

   The New Folder dialog box opens.
3. Type a Title for your folder.

   The Name field automatically populates with the same text. You can edit the Name at any time.
4. Click *Create*.

The new folder is created and is now available in the Resources tree. Unlike a new workspace, a new folder is not published by default, so when it first appears in the Resources tree, it appears with a grayed-out icon and dotted border. Additionally, the title shown in the Resources tree is italicized. An example of a new folder is shown in the following image.



To publish a folder, right-click the folder, and then click *Publish*. By default, when you publish a folder, you automatically publish all its contents as well.

Dragging and Dropping Folders and Files

From the Db2 Web Query Home Page, you can drag and drop items between different areas of the Explorer.

You can drag and drop folders or files from the content area to other folders within the content area, or to other folders within the repository tree. You can also drag and drop folders from the repository tree to other folders within the repository tree, or to other folders within the content area.

You can drag and drop items, or press and hold the *Ctrl* key to drag and drop copies of items. Drag and drop functions the same as the cut-and-paste options from the context menu. Drag and drop while holding the *Ctrl* key functions the same as the copy and paste options from the shortcut menu.

Note: You can select multiple items by holding the *Ctrl* key while selecting each item.

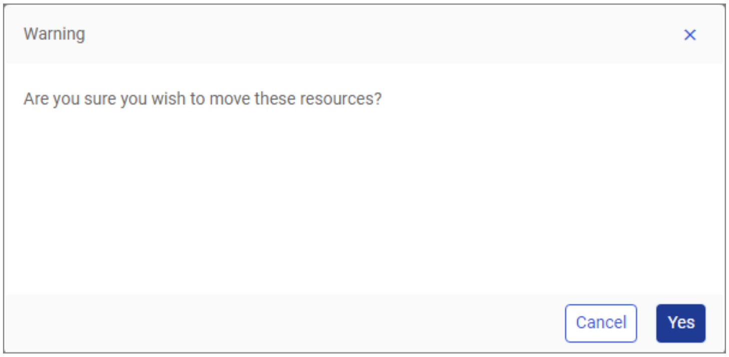
Dragging Indicators

When dragging an item, a thumbnail displays for that specific item type. If you select multiple items, a thumbnail for one of the dragged items displays.

If you drag an item over an area where it cannot be dropped, a No icon 🚫 displays.

Dropping Indicators

When dropping an item, a Warning message displays, as shown in the following image:



Warning

Are you sure you wish to move these resources?

Click *Yes* to move the item.

Click *Cancel* to cancel the operation.

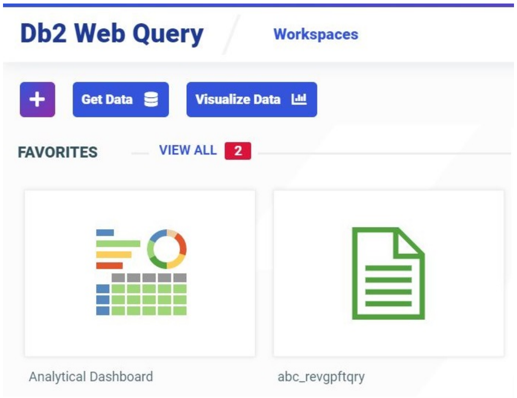
Adding Content to Favorites

You can create a list of favorites to easily access the items that are most important to you. When you save content as a favorite, it automatically appears in the Favorites area of the Home view. Here, you can interact with your content and view its properties.

Procedure: How to Add Content to Favorites

1. Right-click an item from any view of the Home Page, and then click *Add to Favorites*.
2. Navigate to the Home view by clicking the Db2 Web Query logo.

The item now appears in the Favorites area, as shown in the following image.



If you right-click the item, a shortcut menu of options opens. Here, you can do the following:

- ❑ Click *Run* or select a different way to run the item by pointing to *Run...*.
- ❑ Click *Edit* or *Edit with text editor* to edit the item in the appropriate tool or text editor.
- ❑ Click *Remove from Favorites* to remove the item from Favorites.
- ❑ Click *Properties* to open the Properties panel, and edit the properties of the item.

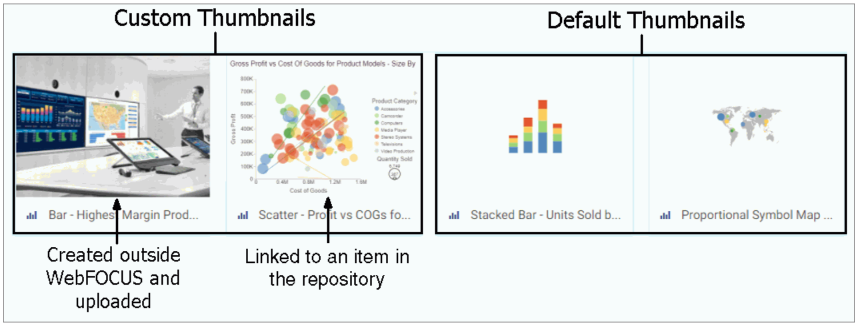
Adding Thumbnails to Content Items

To provide additional context to your items and simplify navigation, you can add custom thumbnails to your content items. These thumbnails might be created from any image files on your machine. You can also create thumbnails from the image files that can be found in your repository.

There are two types of thumbnails in the Home Page:

- ❑ **Default.** Assigned to an item by Db2 Web Query in the absence of a custom thumbnail.
- ❑ **Custom.** Created either from an uploaded image or linked to an item in your repository.

Examples of custom and default thumbnails are shown in the following image.



Custom Thumbnails
Default Thumbnails
Bar - Highest Margin Prod...
Scatter - Profit vs COGs fo...
Stacked Bar - Units Sold b...
Proportional Symbol Map ...

Created outside WebFOCUS and uploaded

Linked to an item in the repository

Thumbnails are only visible in the Grid view.

Procedure: How to Add Thumbnails to Content

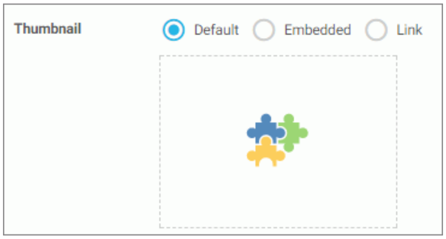
1. Right-click a content item to which you want to add a thumbnail, and then click *Properties*.

   The Properties panel opens.

2. Click the *Advanced* tab.

3. In the Thumbnail property, choose one of the following options:

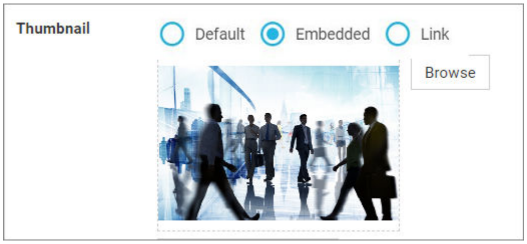
   - **Default.** Populates the thumbnail with the default graphic, as shown in the following image. This is a default option.

     Thumbnail ● Default ○ Embedded ○ Link

   - **Embedded.** Enables you to choose a custom image from your machine.

In the Open dialog box, select an image file and click *Open*. The custom image displays in the preview section of the Thumbnail property, as shown in the following image.

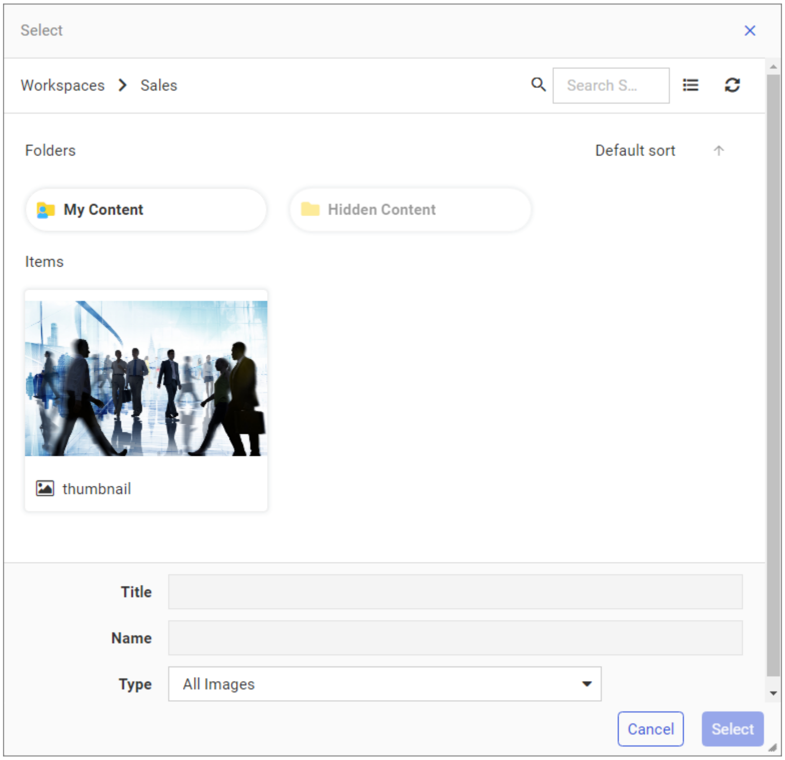


Thumbnail

○ Default ● Embedded ○ Link

To change an image, click *Browse* and select another image file.

- **Link.** Opens the Select dialog box, where you can navigate to any available image file in your repository, as shown in the following image.



Title

Name

Type: All Images

To change an image, click *Browse* and select another image file in your repository.

4. Once the thumbnail is selected, click *Save* in the Properties panel.

The thumbnail displays on the Db2 Web Query Home Page.

Creating URLs

You can create URLs to webpages and store them within the repository. These URLs can further enhance analytics, by providing an additional interactive context to your data. You can run a URL from the Home Page or add it to a personal page. If your URL is optimized for use in an iFrame, you can embed it within a container on a page. You can also publish a URL to make it available to other users and groups.

Note: Some web pages cannot be run inside an iFrame or inside the preview Run window on the Home Page. You can run these web pages in a new browser tab or window.

Procedure: How to Create a URL

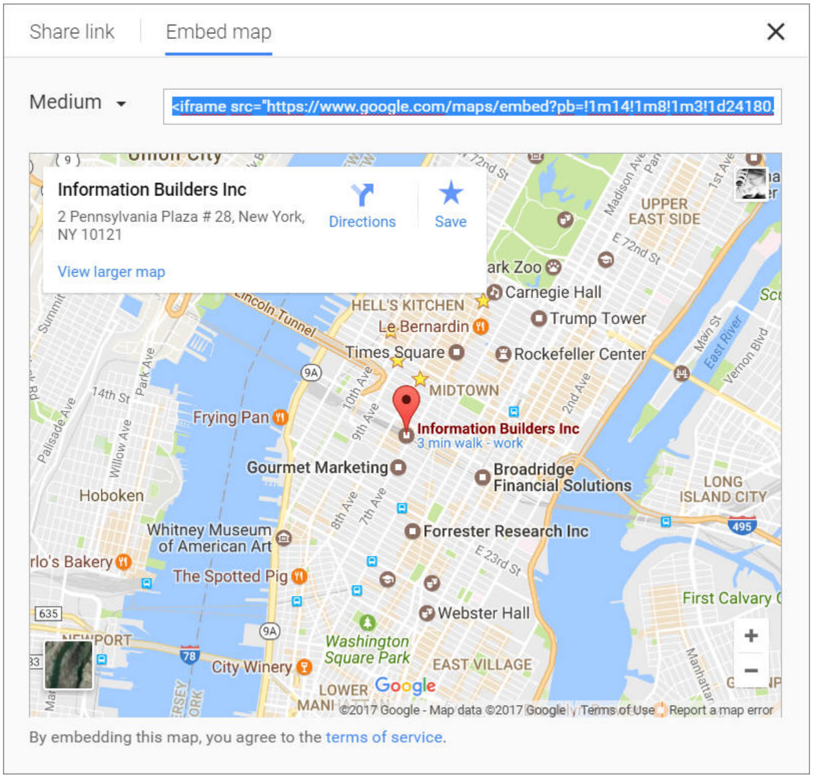
1. Select the workspace or folder where you want your new URL to reside.
2. On the Action Bar, under the *Other* tab, click *URL*.

   The New URL dialog box opens.
3. Type a Title for the URL, type the address of the URL, and then click *OK*.

   The URL now appears in your selected location.

Procedure: How to Embed an iFrame-Optimized URL in a Page

1. Copy an iFrame-optimized URL from the web, as shown in the following image.



<iframe src="https://www.google.com/maps/embed?pb=!1m14!1m8!1m3!1d24180

By embedding this map, you agree to the terms of service.

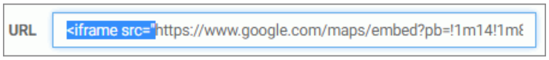
2. Select the workspace or folder where you want your new URL to reside.

3. On the Action Bar, click *URL*.

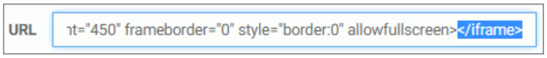
   The New URL dialog box opens.

4. Populate the Title field, and paste the copied URL in the URL field.

5. Remove any iFrame-related syntax that surrounds your URL. Examples are shown in the following images.



URL <iframe src="https://www.google.com/maps/embed?pb=!1m14!1m8



URL ıt="450" frameborder="0" style="border:0" allowfullscreen></iframe>

6. Click *OK* to close the New URL dialog box.

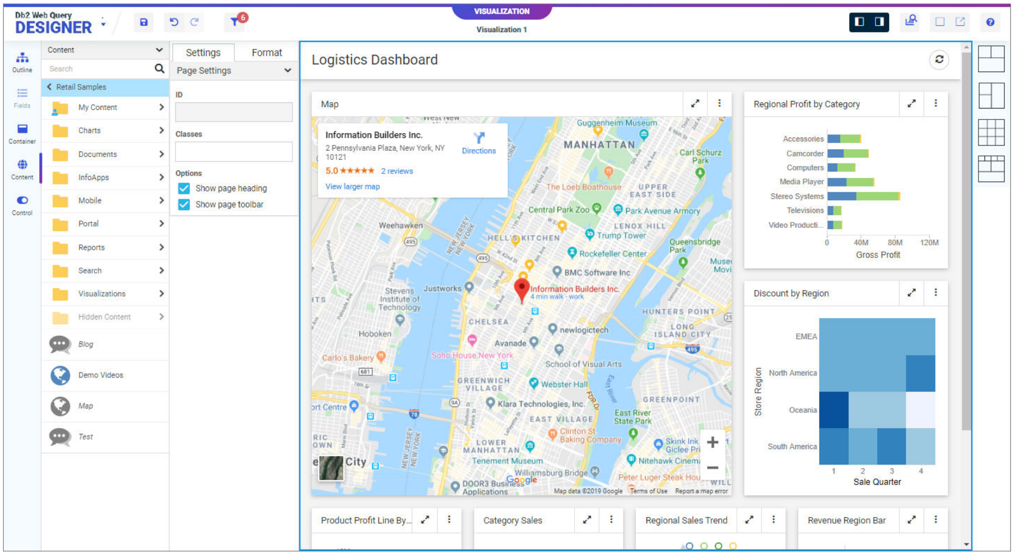
   Your URL item is complete and ready to be displayed in a page.

7. On the Home Page, click the plus (+) button and then click *Assemble Visualizations*.

   The Db2 Web Query Designer opens.

8. Add your new URL to the page.

   The URL content displays within the container, similar to other content items, as shown in the following image.



Options
☑ Show page heading
☑ Show page toolbar

Uploading Files

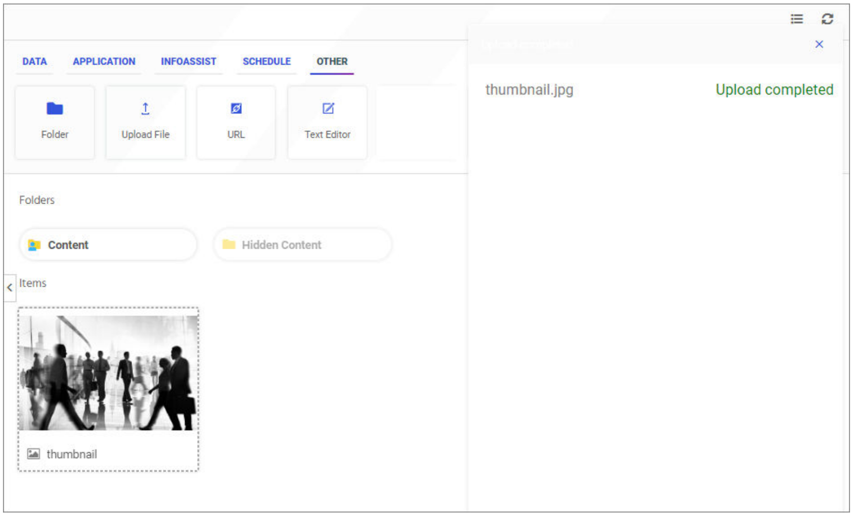
In addition to creating content using Db2 Web Query, you can upload other files, such as images and documents. These files can be used to enhance visualizations.

Procedure: How to Upload Files From The Action Bar

1. Select the workspace or folder where you want your uploaded file or files to reside.
2. On the Action Bar, under the *Other* tab, click *Upload File*.

   The Open dialog box opens.
3. Select one or multiples files from your machine and click *Open*.

The file uploads to your selected directory, and the *Upload completed* message appears, as shown in the following image.



thumbnail.jpg Upload completed

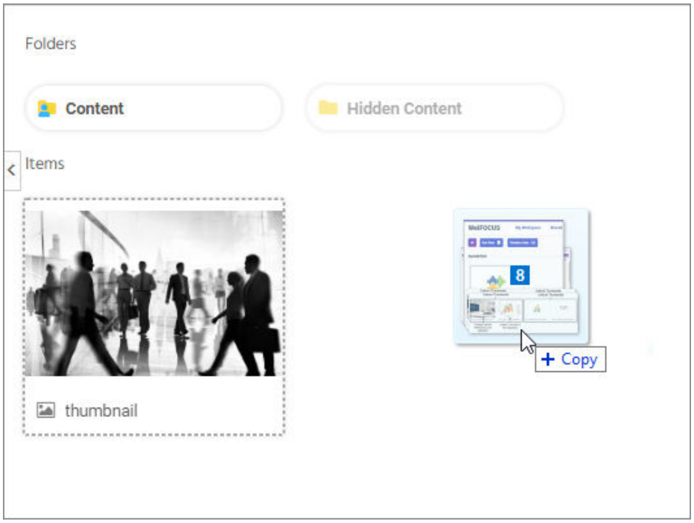
4. Close the *Upload completed* message and proceed with using the uploaded file.

   Note: When you upload an image, such as a .bmp, .jpg, .jpeg, .gif, or .png file, to the repository, an embedded thumbnail of the image automatically generates in the Db2 Web Query Explorer. To change the thumbnail that appears in the Db2 Web Query Explorer, right-click the thumbnail, and open the Properties panel. Click the *Advanced* tab, and modify the thumbnail.

Procedure: How to Upload Files by Dragging to the Explorer

1. On the Home Page, in the Workspaces view, select the workspace or folder where you want your uploaded file or files to reside.

2. Drag one or multiple files from your machine directly to the Explorer, as shown in the following image.



The files upload to your selected directory and the *Upload completed* message shows the status of each uploaded file. If an upload fails, the error appears in red in the *Upload completed* message.

3. Close the *Upload completed* message and proceed with using the uploaded files.

   Note: When you upload an image, such as a .bmp, .jpg, .jpeg, .gif, or .png file, to the repository, an embedded thumbnail of the image automatically generates in the Db2 Web Query Explorer. To change the thumbnail that appears in the Db2 Web Query Explorer, right-click the thumbnail, and open the Properties panel. Click the *Advanced* tab, and modify the thumbnail.

Chapter 2

Accessing Data

Learn how to access all your data from a single streamlined environment using Db2 Web Query. You can connect to various data sources, configure adapters, add or change connections, upload and profile data files before loading them, pivot columns to rows, and Join multiple tables. Then, visualize your data and use it to create charts, reports, and other content.

In this chapter:

Preparing Data for Upload

Uploading data to Db2 Web Query can be made easier if you first familiarize yourself with the data file, and then ensure that it is properly formatted for upload, so that geographic data, hierarchies, and other important aspects of your data are recognized. This is important because the synonym created for your uploaded data provides the basis for quality analytical content.

You can use the following techniques to prepare your data for the uploading process.

Naming Conventions and Microsoft Excel Sheet Names

For Excel spreadsheets, the name of the file is not important, but the name of the worksheet that contains your data is used to generate the synonym name. For the best results, note the following guidelines:

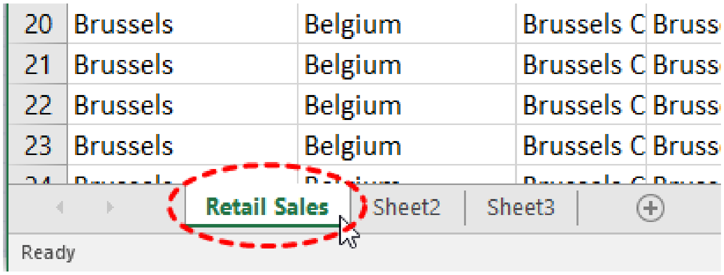
- Ensure that the worksheet name is meaningful. For example, Store Sales is better than Sheet1.

❑ Remove special characters from the worksheet name. Spaces will be converted to underscores, but all non-alphanumeric characters should be removed from the name, such as the following:

/`~!@#$%^()-+={}|;,

❑ Remove or replace NLS characters with standard alphanumeric characters.

The following image shows a worksheet with a meaningful name, Retail Sales.



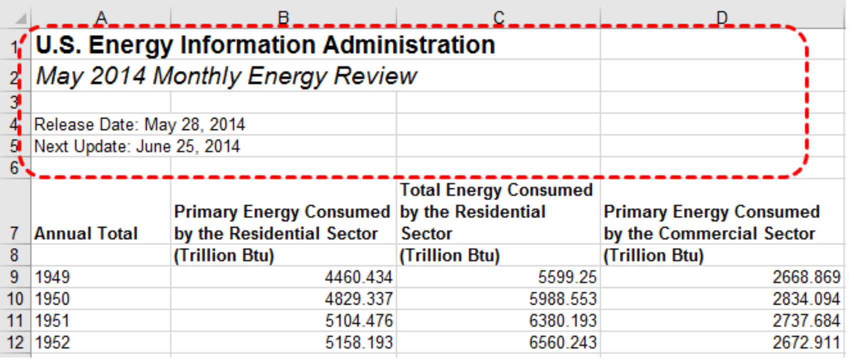
20	Brussels	Belgium	Brussels C	Bruss
21	Brussels	Belgium	Brussels C	Bruss
22	Brussels	Belgium	Brussels C	Bruss
23	Brussels	Belgium	Brussels C	Bruss

While you have an opportunity to edit the worksheet name within the Upload workflow, editing in Excel may be preferable. Note that your spreadsheet data and column titles may contain National Language Support data and special characters.

For CSV files, there is no worksheet name, so the CSV file name is used to generate the synonym name. For this reason, all of the limitations identified for Excel worksheet names apply to the CSV file name. Be sure to check and adjust the file name prior to the upload.

Removing Introductory Information

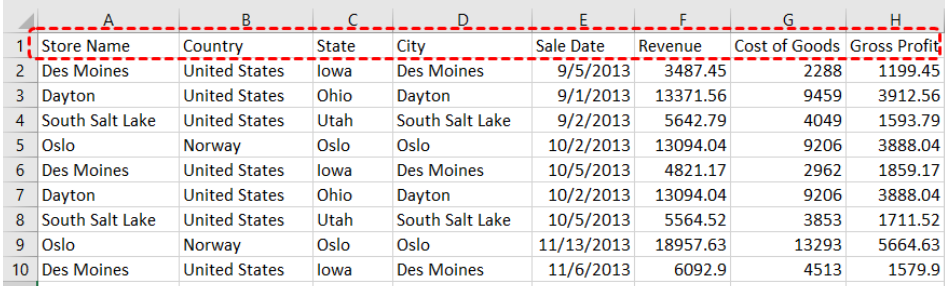
Sometimes, an Excel spreadsheet contains formatted headings in the first few rows. This information cannot be imported into Db2 Web Query and should be removed. Delete the introductory rows and save the file before uploading. Alternatively, you can define a data range within your worksheet and leave the introductory information in place. The following image shows an example spreadsheet with a heading and subheadings highlighted.



	A	B	C	D
1	U.S. Energy Information Administration			
2	*May 2014 Monthly Energy Review*			
3				
4	Release Date: May 28, 2014			
5	Next Update: June 25, 2014			
6				
7	Annual Total	Primary Energy Consumed by the Residential Sector	Total Energy Consumed by the Residential Sector	Primary Energy Consumed by the Commercial Sector
8		(Trillion Btu)	(Trillion Btu)	(Trillion Btu)
9	1949	4460.434	5599.25	2668.869
10	1950	4829.337	5988.553	2834.094
11	1951	5104.476	6380.193	2737.684
12	1952	5158.193	6560.243	2672.911

Placing Column Titles in the First Row

For data to be useful in Db2 Web Query, your data columns must be identified and properly described in the synonym that is generated during the upload process. You can make this easier by ensuring that the first few rows of your Excel spreadsheet contain column titles that are meaningful to you and to other users who will be using it. An example of meaningful column titles is shown in the following image.



	A	B	C	D	E	F	G	H
1	Store Name	Country	State	City	Sale Date	Revenue	Cost of Goods	Gross Profit
2	Des Moines	United States	Iowa	Des Moines	9/5/2013	3487.45	2288	1199.45
3	Dayton	United States	Ohio	Dayton	9/1/2013	13371.56	9459	3912.56
4	South Salt Lake	United States	Utah	South Salt Lake	9/2/2013	5642.79	4049	1593.79
5	Oslo	Norway	Oslo	Oslo	10/2/2013	13094.04	9206	3888.04
6	Des Moines	United States	Iowa	Des Moines	10/5/2013	4821.17	2962	1859.17
7	Dayton	United States	Ohio	Dayton	10/2/2013	13094.04	9206	3888.04
8	South Salt Lake	United States	Utah	South Salt Lake	10/5/2013	5564.52	3853	1711.52
9	Oslo	Norway	Oslo	Oslo	11/13/2013	18957.63	13293	5664.63
10	Des Moines	United States	Iowa	Des Moines	11/6/2013	6092.9	4513	1579.9

If your spreadsheet has more than one row of column titles, Db2 Web Query can merge the information when creating the synonym. You will be given an option to specify how many first rows of the Excel file contain title information in the Upload workflow.

Removing Aggregated Information

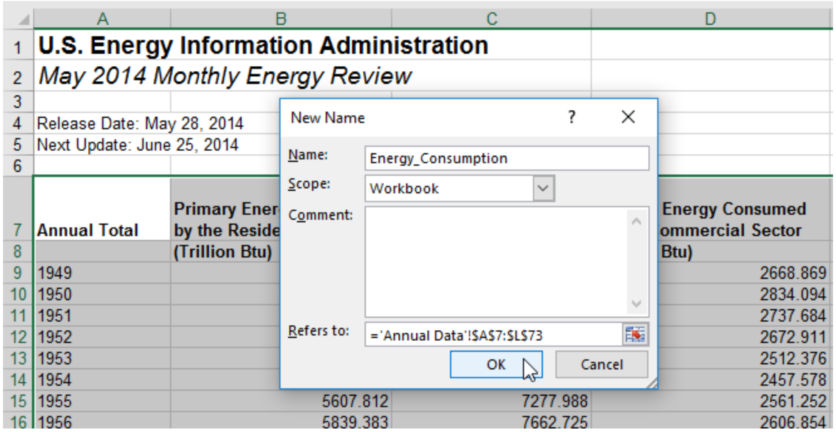
Excel spreadsheets may contain subtotals, grand totals, and other non-data row information. Data aggregation is performed by Db2 Web Query, so you should remove these kinds of rows from your spreadsheet and save the file before uploading it.

Using Excel Name Ranges

Data ranges defined within your Excel worksheet can be helpful for the following reasons:

- ❑ Your worksheet may have introductory information, such as formatted headings or non-data information, in the first few columns.
- ❑ You may not want to import all of the data columns found on your worksheet.

You can define a data range in your worksheet to remove the data that you want Db2 Web Query to process during upload, and leave your spreadsheet in its original format. An example of this is shown in the following image.



	A	B	C	D
1	U.S. Energy Information Administration			
2	May 2014 Monthly Energy Review			
3				
4	Release Date: May 28, 2014			
5	Next Update: June 25, 2014			
6				
7	Annual Total	Primary Ener by the Reside		Energy Consumed ommercial Sector
8		(Trillion Btu)		Btu)
9	1949			2668.869
10	1950			2834.094
11	1951			2737.684
12	1952			2672.911
13	1953			2512.376
14	1954			2457.578
15	1955	5607.812	7277.988	2561.252
16	1956	5839.383	7662.725	2606.854

New Name

Name: Energy_Consumption

Scope: Workbook

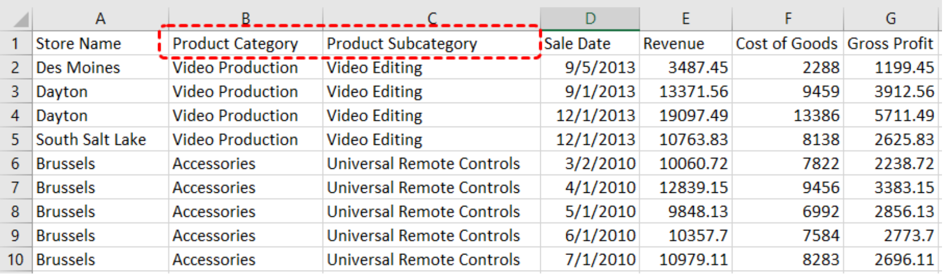
Comment:

Refers to: ='Annual Data'!$A$7:$L$73

Preparing Hierarchical Data Columns

Db2 Web Query recognizes columns of data that have hierarchical relationships. This is useful because the field names are arranged more logically in InfoAssist and because it facilitates Auto-drill capabilities in the content. Auto-drill lets you drill up and down a field hierarchy automatically, making the content engaging and useful.

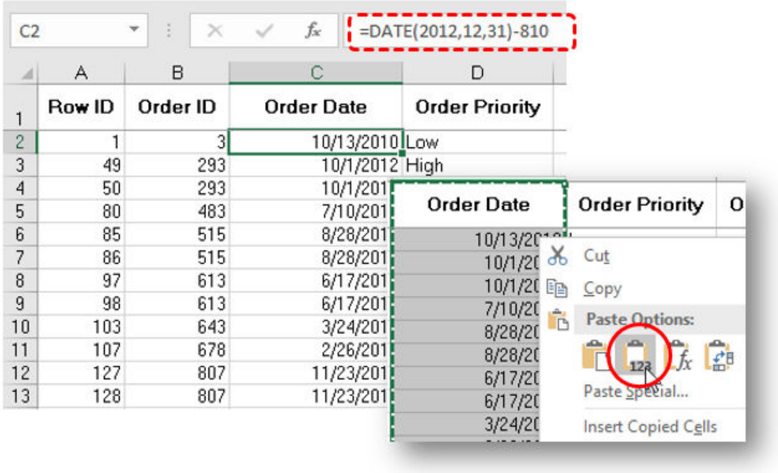
To help Db2 Web Query recognize hierarchical columns correctly, ensure that the column titles begin with a common word or words, and are arranged left-to-right in the correct top-to-bottom direction, as shown in the following image. In the Upload workflow, you can define and edit dimension hierarchies prior to creating the synonym. You can also do this prior to the upload in Excel.



	A	B	C	D	E	F	G
1	Store Name	Product Category	Product Subcategory	Sale Date	Revenue	Cost of Goods	Gross Profit
2	Des Moines	Video Production	Video Editing	9/5/2013	3487.45	2288	1199.45
3	Dayton	Video Production	Video Editing	9/1/2013	13371.56	9459	3912.56
4	Dayton	Video Production	Video Editing	12/1/2013	19097.49	13386	5711.49
5	South Salt Lake	Video Production	Video Editing	12/1/2013	10763.83	8138	2625.83
6	Brussels	Accessories	Universal Remote Controls	3/2/2010	10060.72	7822	2238.72
7	Brussels	Accessories	Universal Remote Controls	4/1/2010	12839.15	9456	3383.15
8	Brussels	Accessories	Universal Remote Controls	5/1/2010	9848.13	6992	2856.13
9	Brussels	Accessories	Universal Remote Controls	6/1/2010	10357.7	7584	2773.7
10	Brussels	Accessories	Universal Remote Controls	7/1/2010	10979.11	8283	2696.11

Removing Date Formulas

Spreadsheets may contain a date column where the values are computed by Excel using a formula. You need to convert these computed values into simple values before uploading a spreadsheet. To remove date formulas, select a column, right-click, and then click *Copy*. Then, right-click the selected column again and select *Values*. This can be found under the Paste Options menu, which is highlighted in the following image. Menu options may vary by Excel release. Now you can see that each cell contains a date value, and Db2 Web Query can decompose your dates into useful components for use in InfoAssist.



C2 =DATE(2012,12,31)-810

	A	B	C	D
1	Row ID	Order ID	Order Date	Order Priority
2	1	3	10/13/2010	Low
3	49	293	10/1/2012	High
4	50	293	10/1/201	
5	80	483	7/10/201	
6	85	515	8/28/201	
7	86	515	8/28/201	
8	97	613	6/17/201	
9	98	613	6/17/201	
10	103	643	3/24/201	
11	107	678	2/26/201	
12	127	807	11/23/201	
13	128	807	11/23/201	

Order Date	Order Priority	O
10/13/20		
10/1/20		
10/1/20		
7/10/20		
8/28/20		
8/28/20		
6/17/20		
6/17/20		
3/24/20		

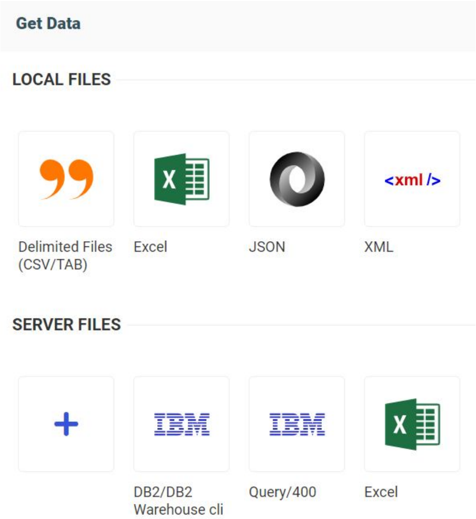
Uploading Data Files

You can upload delimited, Excel, JSON, or XML files into Db2 Web Query, where you can preview and modify the data before loading it to a database, then create visualizations or prepare it for future analysis.

Procedure: How to Upload Data Files

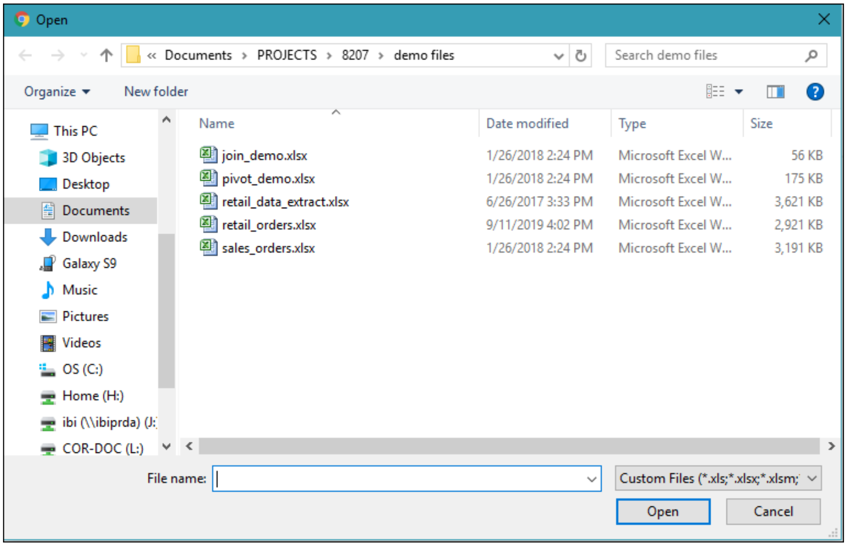
1. On the Db2 Web Query Home Page, click the *Get Data* button Get Data.

   The Get Data dialog box opens, as shown in the following image.



2. Under Local Files, click your upload file type. You can select from the following file types: Delimited Files, Excel, JSON, or XML. An Excel file can contain a single worksheet or multiple worksheets. Delimited, JSON, and XML files are individual data sources.

Your Open dialog box displays, as shown in the following image.



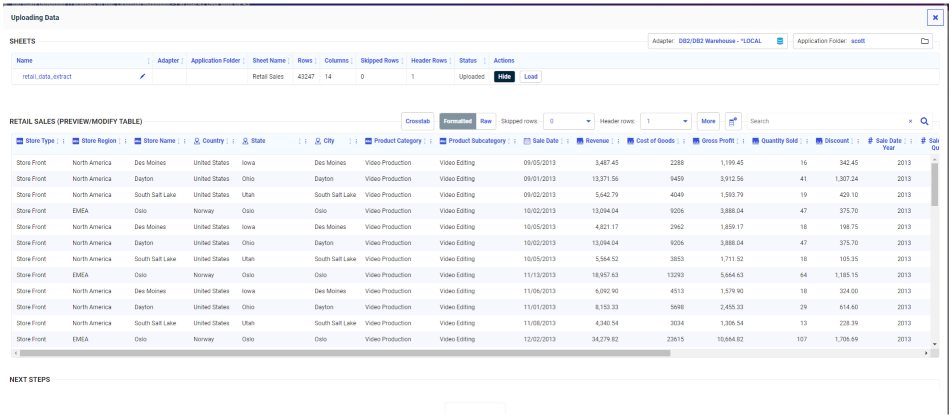
Name	Date modified	Type	Size
join_demo.xlsx	1/26/2018 2:24 PM	Microsoft Excel W...	56 KB
pivot_demo.xlsx	1/26/2018 2:24 PM	Microsoft Excel W...	175 KB
retail_data_extract.xlsx	6/26/2017 3:33 PM	Microsoft Excel W...	3,621 KB
retail_orders.xlsx	9/11/2019 4:02 PM	Microsoft Excel W...	2,921 KB
sales_orders.xlsx	1/26/2018 2:24 PM	Microsoft Excel W...	3,191 KB

File name:

Custom Files (*.xls;*.xlsx;*.xlsm;

3. In the Open dialog box for your operating system, select the file that you want to upload, and then click *Open*.

   This opens a preview, as shown in the following image. In this example, we are showing a multi-worksheet Excel file.



SHEETS

Adapter: DB2/DB2 Warehouse - *LOCAL

Application Folder: scott

Name	Adapter	Application Folder	Sheet Name	Rows	Columns	Skipped Rows	Header Rows	Status	Actions
retail_data_extract			Retail Sales	43247	14	0	1	Uploaded	Hide Load

RETAIL SALES (PREVIEW/MODIFY TABLE)

Skipped rows: 0

Header rows: 1

Search

Store Type	Store Region	Store Name	Country	State	City	Product Category	Product Subcategory	Sale Date	Revenue	Cost of Goods	Gross Profit	Quantity Sold	Discount	# Sale Date Year	# Sale Qu
Store Front	North America	Des Moines	United States	Iowa	Des Moines	Video Production	Video Editing	09/05/2013	3,487.45	2288	1,199.45	16	342.45	2013	
Store Front	North America	Dayton	United States	Ohio	Dayton	Video Production	Video Editing	09/01/2013	13,371.56	9459	3,912.56	41	1,307.24	2013	
Store Front	North America	South Salt Lake	United States	Utah	South Salt Lake	Video Production	Video Editing	09/02/2013	5,642.79	4049	1,593.79	19	429.10	2013	
Store Front	EMEA	Oslo	Norway	Oslo	Oslo	Video Production	Video Editing	10/02/2013	13,094.04	9206	3,888.04	47	375.70	2013	
Store Front	North America	Des Moines	United States	Iowa	Des Moines	Video Production	Video Editing	10/05/2013	4,821.17	2962	1,859.17	18	198.75	2013	
Store Front	North America	Dayton	United States	Ohio	Dayton	Video Production	Video Editing	10/02/2013	13,094.04	9206	3,888.04	47	375.70	2013	
Store Front	North America	South Salt Lake	United States	Utah	South Salt Lake	Video Production	Video Editing	10/05/2013	5,564.52	3853	1,711.52	18	105.35	2013	
Store Front	EMEA	Oslo	Norway	Oslo	Oslo	Video Production	Video Editing	11/13/2013	18,957.63	13293	5,664.63	64	1,185.15	2013	
Store Front	North America	Des Moines	United States	Iowa	Des Moines	Video Production	Video Editing	11/06/2013	6,092.90	4513	1,579.90	18	324.00	2013	
Store Front	North America	Dayton	United States	Ohio	Dayton	Video Production	Video Editing	11/01/2013	8,153.33	5698	2,455.33	29	614.60	2013	
Store Front	North America	South Salt Lake	United States	Utah	South Salt Lake	Video Production	Video Editing	11/08/2013	4,340.54	3034	1,306.54	13	228.39	2013	
Store Front	EMEA	Oslo	Norway	Oslo	Oslo	Video Production	Video Editing	12/02/2013	34,279.82	23615	10,664.82	107	1,706.69	2013	

   Your file will be loaded to the default data store of the displayed Adapter. You can click the Adapter name to select an adapter folder and connection where your data is loaded.

   You can click the Application Folder name to select an application folder and specify where your synonym is created.

You can change the name of your worksheet by clicking inside the Name field and typing a new name. From the Actions column, click the *Preview* or *Hide* buttons to control the preview of sample data.

You can select individual worksheets to preview or load. A single-sheet Excel file, or the first worksheet of a multi-sheet Excel file, is automatically previewed.

You can modify your data, such as changing the number of skipped rows or header rows, display your data as formatted or raw, or choose from more advanced and customized file options.

Uploaded Excel worksheets are examined for leading rows that contain titles or other extraneous information before the header rows and data values. These leading rows are automatically skipped. The number of rows to be skipped is displayed in the Skipped Rows field, and can be changed.

You can click the *Choose Columns* icon to select which columns you want to load to the target database or data store.

Column drop-down menu options let you change geographic roles and data types, or rename columns.

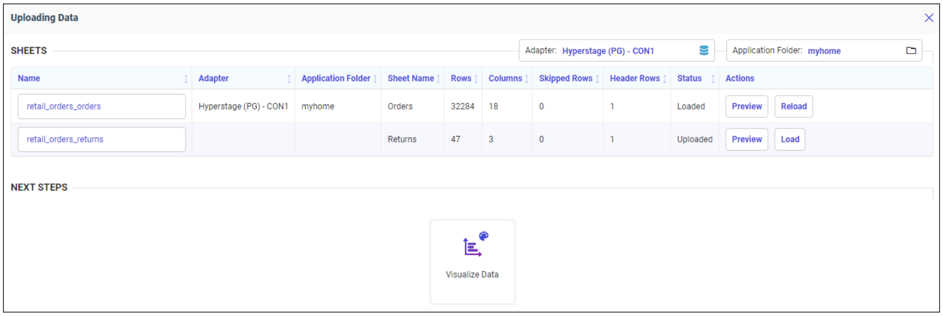
Numeric fields are generally classified as measures. If the name of a numeric field implies that it is more likely to be a dimension than a measure, it is classified as a dimension, by default. For example, fields ending with _ID such as ORDER_ID are classified as dimensions. This allows you to create content that more accurately represents your data.

To change field classifications from dimensions to measures, select *Convert to measure*.

To change field classifications from measures to dimensions, select *Convert to dimension*.

4. Click the *Load* button next to the worksheet you want to load.

   Your worksheet loads, and its load status is updated, as shown in the following image.



Uploading Data

SHEETS

Adapter: Hyperstage (PG) - CON1

Application Folder: myhome

Name	Adapter	Application Folder	Sheet Name	Rows	Columns	Skipped Rows	Header Rows	Status	Actions
retail_orders_orders	Hyperstage (PG) - CON1	myhome	Orders	32284	18	0	1	Loaded	Preview Reload
retail_orders_returns			Returns	47	3	0	1	Uploaded	Preview Load

NEXT STEPS

You can still make changes to your worksheet after it is initially loaded, and then click the *Reload* button to load your changes. Once your file is uploaded, you can create your own visualizations by clicking *Visualize Data*.

Changing a Target Connection When Uploading Files

When using Get Data to upload a file, with appropriate permission, you can override the default Adapter and Connection settings by selecting an alternative configured adapter from your current profile.

Procedure: How to Change a Target Connection

When uploading an Excel or delimited file, you can select an adapter folder and specify the connection where your data is loaded.

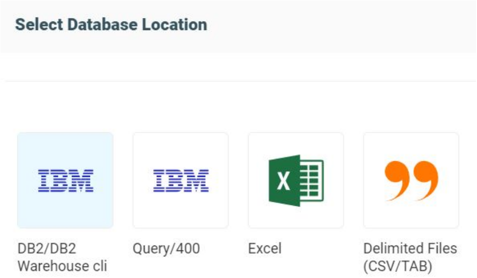
1. Upload an Excel or delimited file using Get Data.

   The Preview page opens.

   The Adapter column remains blank until your data is loaded. Once you load a file, the Adapter column displays the adapter and connection where your data was loaded.

2. Click the Adapter field.

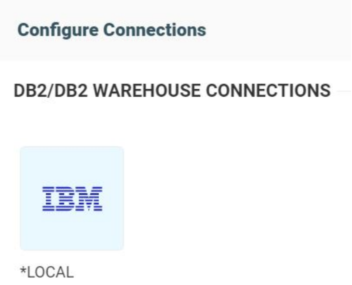
   The Select Database Location dialog box opens, and displays the adapters configured on your server, as shown in the following image.



Select Database Location

3. Click an available adapter, and then click *Select*.

The Configure Connections dialog box opens, and displays the connections configured for the adapter you selected, as shown in the following image.



Configure Connections

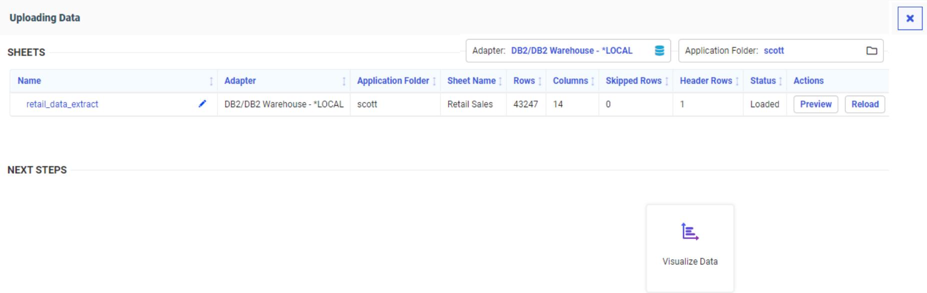
DB2/DB2 WAREHOUSE CONNECTIONS

4. Click an available connection and then click *Select*.

   The Preview page displays.

5. Click *Load*.

   Your selected connection displays in the Adapter column, as shown in the following image.



Uploading Data

SHEETS

Adapter: DB2/DB2 Warehouse - *LOCAL

Application Folder: scott

Name	Adapter	Application Folder	Sheet Name	Rows	Columns	Skipped Rows	Header Rows	Status	Actions
retail_data_extract	DB2/DB2 Warehouse - *LOCAL	scott	Retail Sales	43247	14	0	1	Loaded	Preview Reload

NEXT STEPS

Your data is saved by default to this connection for subsequent file loads.

Changing a Target Application When Uploading Files

When using Get Data to upload a file, with appropriate permission, you can override the default Application setting where the synonym is created by selecting an application folder from your current application path.

Procedure: How to Change a Target Application

When uploading an Excel or delimited file to create a synonym, you can select an application folder and specify where your synonym is created.

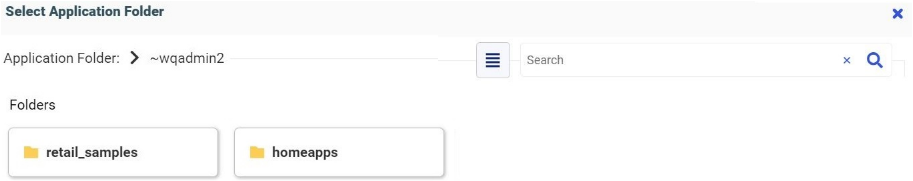
1. Upload an Excel or delimited file using Get Data.

The Preview page opens.

The Application Folder column remains blank until your data is loaded. If you have previously loaded a file, the Application Folder column displays the application where you created your synonym.

2. Click the Application Folder name.

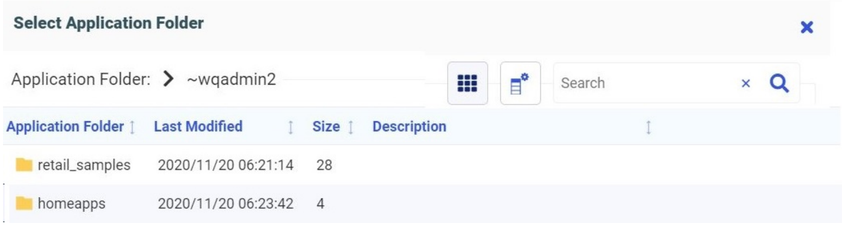
   The Select Application Folder dialog box opens and displays your application folders in your current application path, as shown in the following image.



Select Application Folder

Application Folder: > ~wqadmin2

   You can click the View icon to switch between the Grid view, as shown above, and the List view, as shown in the following image.



Select Application Folder

Application Folder: > ~wqadmin2

Application Folder	Last Modified	Size	Description
retail_samples	2020/11/20 06:21:14	28	
homeapps	2020/11/20 06:23:42	4	

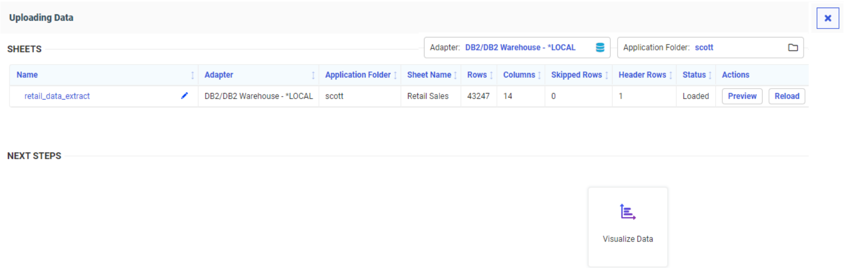
   You can click the Columns icon to select which columns to display in the List view.

3. Click your target application folder, for example, Scott.
4. Click *Select*.

   The Preview page displays.

5. Click *Load*.

Your selected application displays in the Application Folder column, as shown in the following image.



Adapter: DB2/DB2 Warehouse - *LOCAL

Application Folder: scott

Name	Adapter	Application Folder	Sheet Name	Rows	Columns	Skipped Rows	Header Rows	Status	Actions
retail_data_extract	DB2/DB2 Warehouse - *LOCAL	scott	Retail Sales	43247	14	0	1	Loaded	Preview Reload

Your data is saved by default to this application folder for subsequent file loads.

Choosing Columns to Load When Uploading Files

When using Get Data to upload a file, you can choose which columns will be loaded to the target database or data store, so you only load the columns you need.

Procedure: How to Choose Columns to Load

When uploading an Excel or delimited file, you can choose the columns that you want to be loaded.

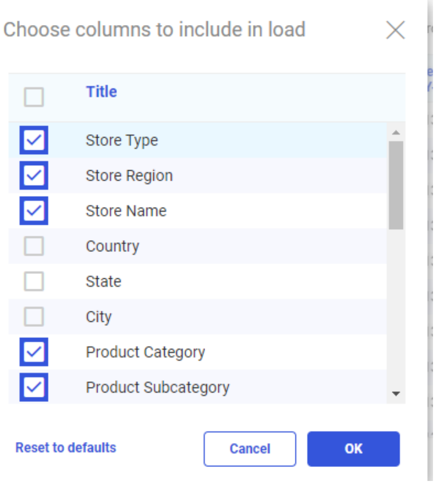
1. Upload an Excel or delimited file using Get Data.

   The Preview page displays.

2. Click the *Choose Columns* icon .

   The Choose Columns dialog box displays. All columns are selected, by default.

3. Clear the check boxes for the columns you do not want to be loaded, as shown in the following image.



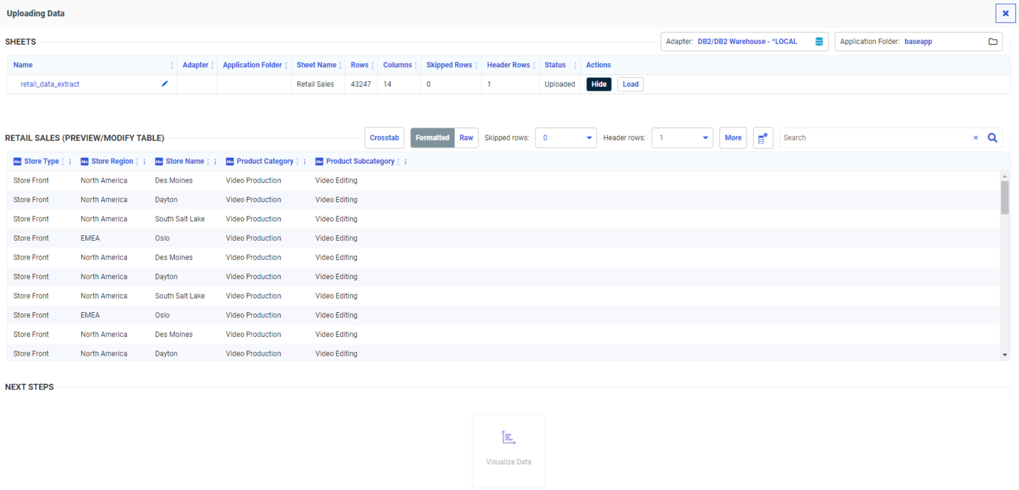
Choose columns to include in load

□	Title
☑	Store Type
☑	Store Region
☑	Store Name
□	Country
□	State
□	City
☑	Product Category
☑	Product Subcategory

You can return the columns to their original selected settings by clicking *Reset to defaults*.

4. Click *OK*.

The Preview page displays, and shows only the columns you want to be loaded, as shown in the following image.



Uploading Data

SHEETS

Adapter: DB2/DB2 Warehouse - *LOCAL

Application Folder: baseapp

Name	Adapter	Application Folder	Sheet Name	Rows	Columns	Skipped Rows	Header Rows	Status	Actions
retail_data_extract			Retail Sales	43247	14	0	1	Uploaded	Hide Load

RETAIL SALES (PREVIEW/MODIFY TABLE)

Skipped rows: 0

Header rows: 1

Store Type	Store Region	Store Name	Product Category	Product Subcategory
Store Front	North America	Des Moines	Video Production	Video Editing
Store Front	North America	Dayton	Video Production	Video Editing
Store Front	North America	South Salt Lake	Video Production	Video Editing
Store Front	EMEA	Oslo	Video Production	Video Editing
Store Front	North America	Des Moines	Video Production	Video Editing
Store Front	North America	Dayton	Video Production	Video Editing
Store Front	North America	South Salt Lake	Video Production	Video Editing
Store Front	EMEA	Oslo	Video Production	Video Editing
Store Front	North America	Des Moines	Video Production	Video Editing
Store Front	North America	Dayton	Video Production	Video Editing

NEXT STEPS

Note: You can click the *Choose Columns* icon again to change which columns you want to be loaded.

5. Click *Load* to load your data.

Pivoting Data

Some data files may contain repeating values, such as sales figures for a series of years. When uploading a spreadsheet or delimited file with columns of repeating values, you can preview these columns and transform them into rows.

Procedure: How to Pivot Columns Into Rows

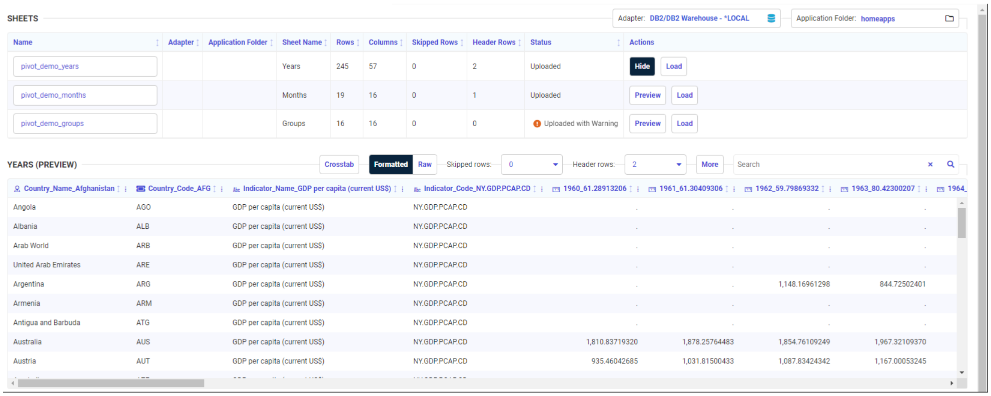
1. On the Db2 Web Query Home Page, click the *Get Data* button Get Data.

   The Get Data dialog box opens.

2. Under Local Files, click your Delimited or Excel file type for upload. An Excel file can contain a single worksheet or multiple worksheets. A delimited file is an individual data source.

3. In the Open dialog box for your operating system, select the file that you want to upload, and then click *Open*.

   This opens a preview, as shown in the following image. In this example, we are showing a multi-worksheet Excel file. The previewed worksheet contains columns for each year from 1960 to 2012.



SHEETS

Adapter: DB2/DB2 Warehouse - *LOCAL

Application Folder: homeapps

Name	Adapter	Application Folder	Sheet Name	Rows	Columns	Skipped Rows	Header Rows	Status	Actions
pivot_demo_years			Years	245	57	0	2	Uploaded	Hide Load
pivot_demo_months			Months	19	16	0	1	Uploaded	Preview Load
pivot_demo_groups			Groups	16	16	0	0	Uploaded with Warning	Preview Load

YEARS (PREVIEW)

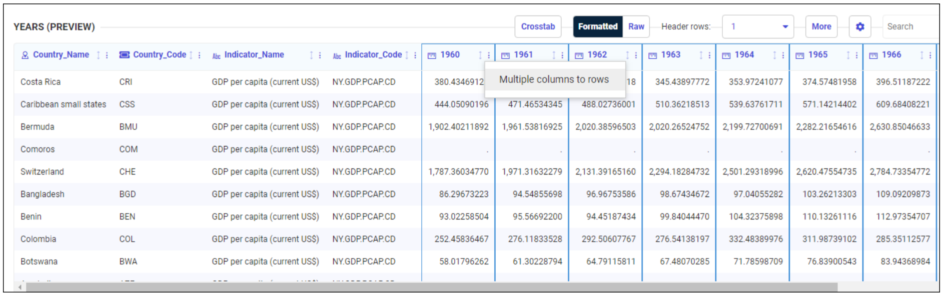
Skipped rows: 0

Header rows: 2

Country_Name_Afghanistan	Country_Code_AFG	Indicator_Name_GDP per capita (current US$)	Indicator_Code_NY.GDP.PCAP.CD	1960_61.28913206	1961_61.30409306	1962_59.79869332	1963_80.42300207	1964_
Angola	AGO	GDP per capita (current US$)	NY.GDP.PCAP.CD	.	.	.	.	
Albania	ALB	GDP per capita (current US$)	NY.GDP.PCAP.CD	.	.	.	.	
Arab World	ARB	GDP per capita (current US$)	NY.GDP.PCAP.CD	.	.	.	.	
United Arab Emirates	ARE	GDP per capita (current US$)	NY.GDP.PCAP.CD	.	.	.	.	
Argentina	ARG	GDP per capita (current US$)	NY.GDP.PCAP.CD	.	.	1,148.16961298	844.72502401	
Armenia	ARM	GDP per capita (current US$)	NY.GDP.PCAP.CD	.	.	.	.	
Antigua and Barbuda	ATG	GDP per capita (current US$)	NY.GDP.PCAP.CD	.	.	.	.	
Australia	AUS	GDP per capita (current US$)	NY.GDP.PCAP.CD	1,810.83719320	1,878.25764483	1,854.76109249	1,967.32109370	
Austria	AUT	GDP per capita (current US$)	NY.GDP.PCAP.CD	935.46042685	1,031.81500433	1,087.83424342	1,167.00053245	

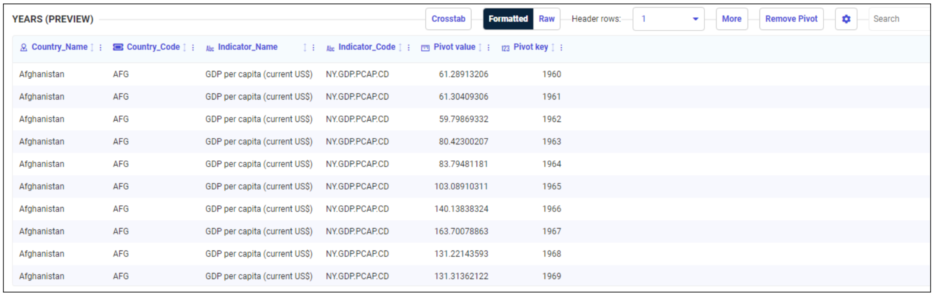
4. To multi-select all the year columns, click inside the first year column, hold down the Shift key, and click inside the last year column.

5. Click the menu button in any of the selected year columns, and click *Multiple columns to rows*, as shown in the following image.



Country_Name	Country_Code	Indicator_Name	Indicator_Code	1960	1961	1962	1963	1964	1965	1966
Costa Rica	CRI	GDP per capita (current US$)	NY.GDP.PCAP.CD	380.4346912		18	345.43897772	353.97241077	374.57481958	396.51187222
Caribbean small states	CSS	GDP per capita (current US$)	NY.GDP.PCAP.CD	444.05090196	471.46534345	488.02736001	510.36218513	539.63761711	571.14214402	609.68408221
Bermuda	BMU	GDP per capita (current US$)	NY.GDP.PCAP.CD	1,902.40211892	1,961.53816925	2,020.38596503	2,020.26524752	2,199.72700691	2,282.21654616	2,630.85046633
Comoros	COM	GDP per capita (current US$)	NY.GDP.PCAP.CD	.	.	.	.	.	.	.
Switzerland	CHE	GDP per capita (current US$)	NY.GDP.PCAP.CD	1,787.36034770	1,971.31632279	2,131.39165160	2,294.18284732	2,501.29318996	2,620.47554735	2,784.73354772
Bangladesh	BGD	GDP per capita (current US$)	NY.GDP.PCAP.CD	86.29673223	94.54855698	96.96753586	98.67434672	97.04055282	103.26213303	109.09209873
Benin	BEN	GDP per capita (current US$)	NY.GDP.PCAP.CD	93.02258504	95.56692200	94.45187434	99.84044470	104.32375898	110.13261116	112.97354707
Colombia	COL	GDP per capita (current US$)	NY.GDP.PCAP.CD	252.45836467	276.11833528	292.50607767	276.54138197	332.48389976	311.98739102	285.35112577
Botswana	BWA	GDP per capita (current US$)	NY.GDP.PCAP.CD	58.01796262	61.30228794	64.79115811	67.48070285	71.78598709	76.83900543	83.94368984

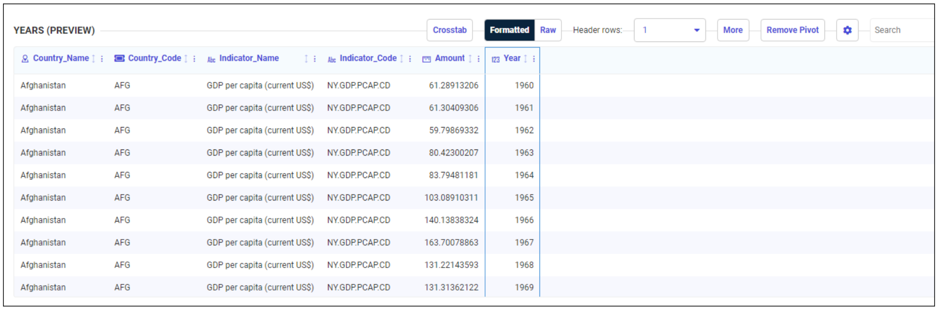
The repeating year columns now display as rows, as shown in the following image.



Country_Name	Country_Code	Indicator_Name	Indicator_Code	Pivot value	Pivot key
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	61.28913206	1960
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	61.30409306	1961
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	59.79869332	1962
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	80.42300207	1963
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	83.79481181	1964
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	103.08910311	1965
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	140.13838324	1966
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	163.70078863	1967
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	131.22143593	1968
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	131.31362122	1969

6. Click the menu button in the Pivot Value column, click *Rename*, and type a new column title in the dialog box that reflects the numeric cell you are describing. For example: Amount.
7. Click *OK*.
8. Click the menu button in the Pivot key column, click *Rename*, and type a new column title in the dialog box that reflects the numeric cell you are describing. For example: Year.
9. Click *OK*.

The column titles are renamed, as shown in the following image.



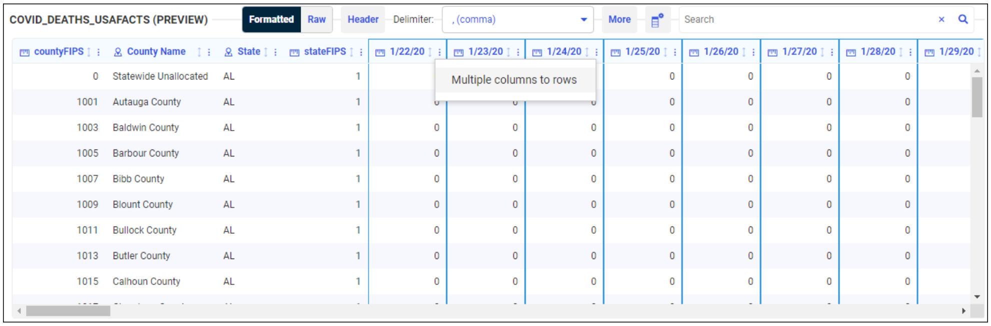
Country_Name	Country_Code	Indicator_Name	Indicator_Code	Amount	Year
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	61.28913206	1960
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	61.30409306	1961
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	59.79869332	1962
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	80.42300207	1963
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	83.79481181	1964
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	103.08910311	1965
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	140.13838324	1966
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	163.70078863	1967
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	131.22143593	1968
Afghanistan	AFG	GDP per capita (current US$)	NY.GDP.PCAP.CD	131.31362122	1969

To revert your pivoting changes, click *Remove Pivot*.

Pivot Columns with Date Headers

If the repeating columns have headers that are dates, the fields are recognized as date values when pivoted into rows, so you can report from the table using date functions.

An uploaded data file with selected repeating columns that have date headers is shown in the following image.



countyFIPS	County Name	State	stateFIPS	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20	1/27/20	1/28/20	1/29/20
0	Statewide Unallocated	AL	1				0	0	0	0	
1001	Autauga County	AL	1	0	0	0	0	0	0	0	
1003	Baldwin County	AL	1	0	0	0	0	0	0	0	
1005	Barbour County	AL	1	0	0	0	0	0	0	0	
1007	Bibb County	AL	1	0	0	0	0	0	0	0	
1009	Blount County	AL	1	0	0	0	0	0	0	0	
1011	Bullock County	AL	1	0	0	0	0	0	0	0	
1013	Butler County	AL	1	0	0	0	0	0	0	0	
1015	Calhoun County	AL	1	0	0	0	0	0	0	0	

Connecting to Data Files

You can connect to various data sources and create synonyms, configure new adapters, add connections to existing adapters, or change adapter connections. Then, use your connected data for content creation or analysis.

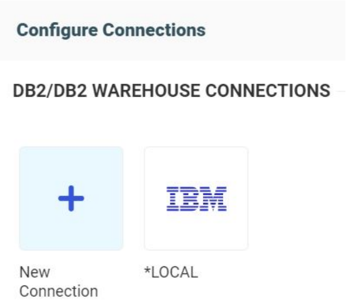
Procedure: How to Connect to Data Files

1. On the Db2 Web Query Home Page, click the *Get Data* button Get Data.

   The Get Data dialog box opens.

2. Under Server Files, click a configured adapter.

The Configure Connections dialog box opens, as shown in the following image.

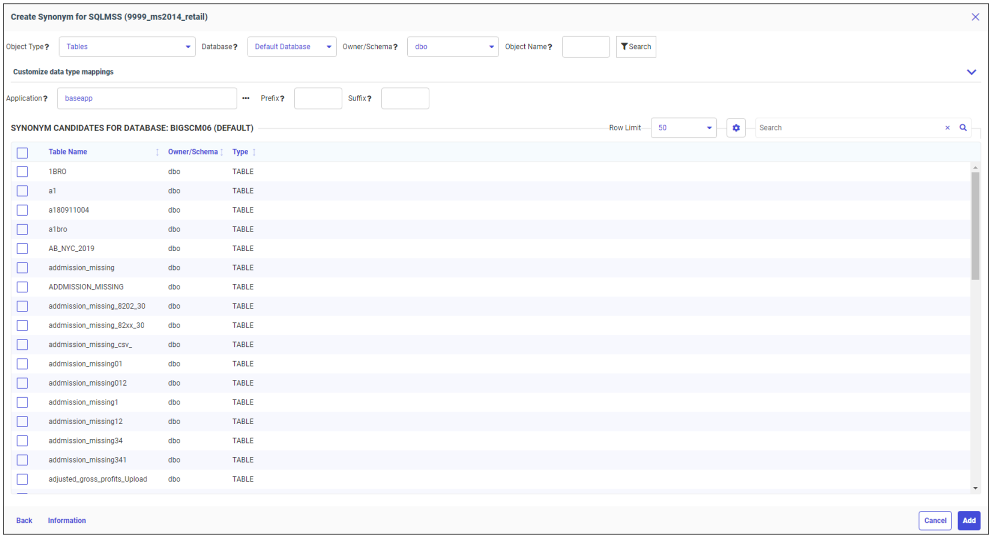


Configure Connections

DB2/DB2 WAREHOUSE CONNECTIONS

3. Double-click an available connection.

   The Create Synonym dialog box opens, as shown in the following image.



Create Synonym for SQLMSS (9999_ms2014_retail)

Object Type? Tables Database? Default Database Owner/Schema? dbo Object Name?

Customize data type mappings

Application? baseapp Prefix? Suffix?

SYNONYM CANDIDATES FOR DATABASE: BIGSCM06 (DEFAULT)

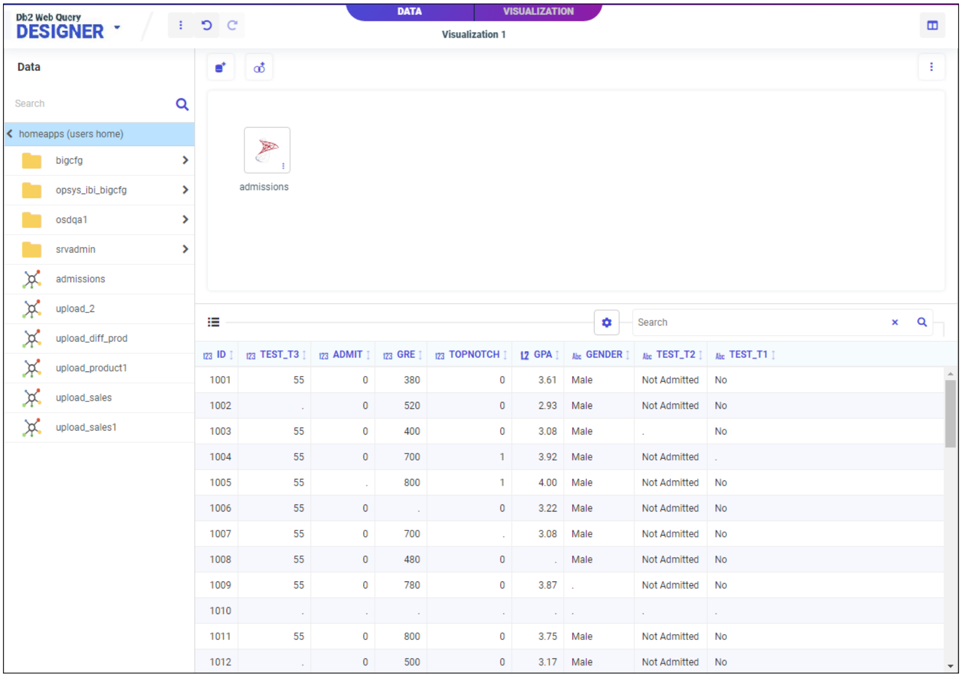
Row Limit 50

□	Table Name	Owner/Schema	Type
□	1BRO	dbo	TABLE
□	a1	dbo	TABLE
□	a180911004	dbo	TABLE
□	a1bro	dbo	TABLE
□	AB_NYC_2019	dbo	TABLE
□	addmission_missing	dbo	TABLE
□	ADDMISSION_MISSING	dbo	TABLE
□	addmission_missing_8202_30	dbo	TABLE
□	addmission_missing_82xx_30	dbo	TABLE
□	addmission_missing_csv_	dbo	TABLE
□	addmission_missing01	dbo	TABLE
□	addmission_missing012	dbo	TABLE
□	addmission_missing1	dbo	TABLE
□	addmission_missing12	dbo	TABLE
□	addmission_missing34	dbo	TABLE
□	addmission_missing341	dbo	TABLE
□	adjusted_gross_profits_Upload	dbo	TABLE

4. Select a data source for synonym creation.

5. Click the *Add* button.

Your synonym opens in Db2 Web Query Designer, where you can create content. Click the *DATA* tab to view your synonym, as shown in the following image.



ID	TEST_T3	ADMIT	GRE	TOPNOTCH	GPA	GENDER	TEST_T2	TEST_T1
1001	55	0	380	0	3.61	Male	Not Admitted	No
1002	.	0	520	0	2.93	Male	Not Admitted	No
1003	55	0	400	0	3.08	Male	.	No
1004	55	0	700	1	3.92	Male	Not Admitted	.
1005	55	.	800	1	4.00	Male	Not Admitted	No
1006	55	0	.	0	3.22	Male	Not Admitted	No
1007	55	0	700	.	3.08	Male	Not Admitted	No
1008	55	0	480	0	.	Male	Not Admitted	No
1009	55	0	780	0	3.87	.	Not Admitted	No
1010	.	.	.	.	.	.	.	.
1011	55	0	800	0	3.75	Male	Not Admitted	No
1012	.	0	500	0	3.17	Male	Not Admitted	No

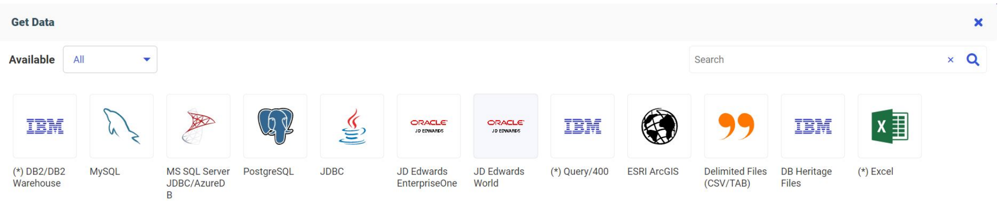
Procedure: How to Configure a New Adapter

1. Open the Get Data dialog box.

   Depending on your version of Db2 Web Query and your permissions, you may have the option to configure a new adapter.

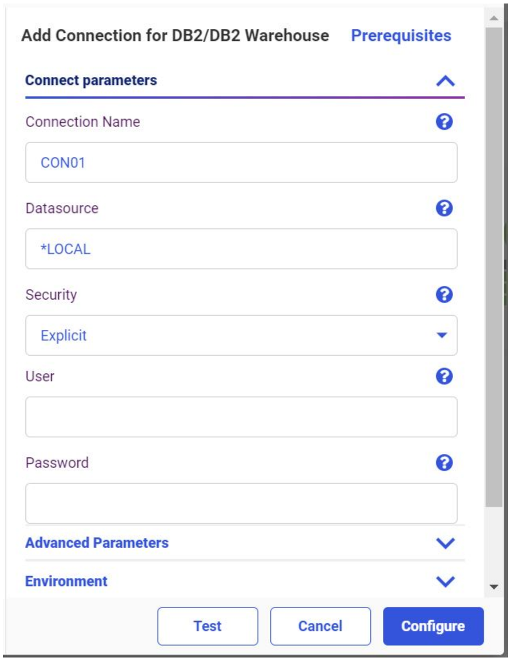
2. Under the Server Files area, click the New Connection icon ＋.

   The Available adapter dialog box opens, as shown in the following image.



3. Right-click an adapter and click *Configure*.

A parameters panel opens, as shown in the following image.



Add Connection for DB2/DB2 Warehouse

Connect parameters

Connection Name

CON01

Datasource

*LOCAL

Security

Explicit

User

Password

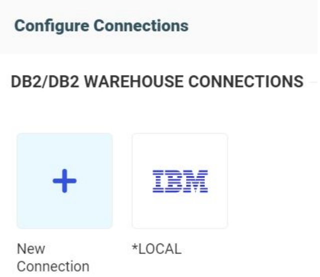
4. Enter your parameters in the panel.
5. Click *Test*, if available, to test your configuration.
6. Click *Configure*.

   Your configured adapter will display in the Get Data dialog box.

Procedure: How to Add a Connection

1. From the Get Data dialog box, under Server Files, click a configured adapter.

   The Connections dialog box opens, as shown in the following image.



Configure Connections

DB2/DB2 WAREHOUSE CONNECTIONS

2. Click the New Connection icon + .

   A parameters panel opens.

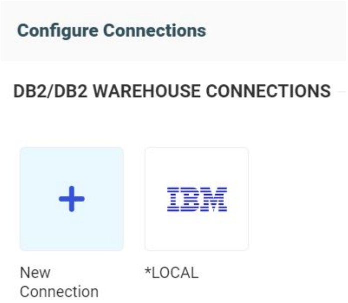
3. Enter your connection parameters.
4. Click *Test*, if available, to test your connection.
5. Click *Configure*.

   Your configured connection will display in the Connections dialog box.

Procedure: How to Change a Connection

1. From the Get Data dialog box, under Server Files, click a configured adapter.

   The Configure Connections dialog box opens.



Configure Connections

DB2/DB2 WAREHOUSE CONNECTIONS

2. Right-click a connection.
3. Select *Properties* from the shortcut menu.

   A parameters panel opens.

4. Change your connection parameters.
5. Click *Test*, if available, to test your connection.
6. Click *Configure*.

   Your changed connection will display in the Connections dialog box.

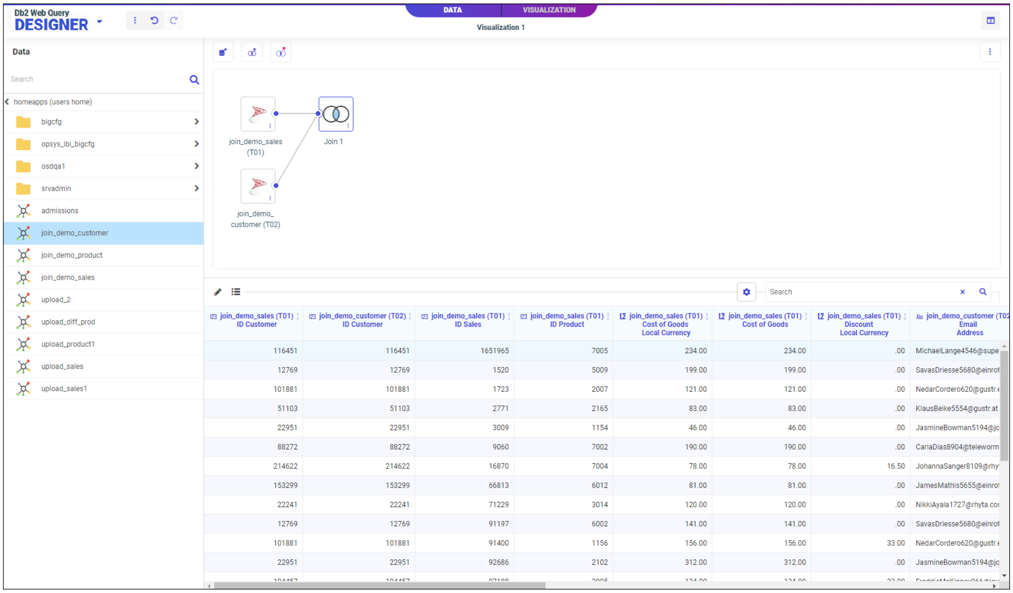
Joining Data

You can Join data sources from the Data tab in Db2 Web Query Designer, and then use your joined data to create new content from within the Visualization tab.

The joined data structure is not applied permanently to the metadata. This allows you to virtually join multiple data sources as if they were a single data source, from which you can report in a single request. This can increase the number of relevant fields available for use in your content, giving you an expanded selection of data specific to your charting purposes.

The Join is created when fields that match in your original data source and second data source are identified. For example, you can Join two data sources on a Join field of Product ID, using just one data source in your chart. You can then create content using your Joined data source.

The following image shows an example of joined data in Db2 Web Query Designer.



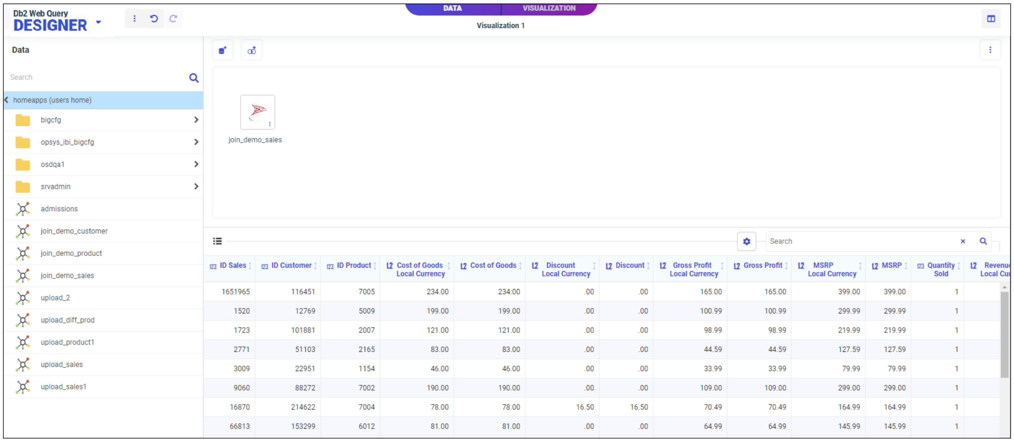
join_demo_sales (T01) ID Customer	join_demo_customer (T02) ID Customer	join_demo_sales (T01) ID Sales	join_demo_sales (T01) ID Product	join_demo_sales (T01) Cost of Goods Local Currency	join_demo_sales (T01) Cost of Goods	join_demo_sales (T01) Discount Local Currency	join_demo_customer (T02 Email Address
116451	116451	1651965	7005	234.00	234.00	.00	MichaelLange4546@supe
12769	12769	1520	5009	199.00	199.00	.00	SavasDriesse5680@einrot
101881	101881	1723	2007	121.00	121.00	.00	NedarCordero620@gustr.e
51103	51103	2771	2165	83.00	83.00	.00	KlausBeike5554@gustr.at
22951	22951	3009	1154	46.00	46.00	.00	JasmineBowman5194@jo
88272	88272	9060	7002	190.00	190.00	.00	CarlaDias8904@teleworm
214622	214622	16870	7004	78.00	78.00	16.50	JohannaSanger8109@rhy
153299	153299	66813	6012	81.00	81.00	.00	JamesMathis5655@einrot
22241	22241	71229	3014	120.00	120.00	.00	NikkiAyala1727@rhyta.cor
12769	12769	91197	6002	141.00	141.00	.00	SavasDriesse5680@einrot
101881	101881	91400	1156	156.00	156.00	33.00	NedarCordero620@gustr.e
22951	22951	92686	2102	312.00	312.00	.00	JasmineBowman5194@jo

Procedure: How to Create a Basic Join

You can Join data sources based on shared fields to enhance the data available to you, and then create charts based on the joined data source.

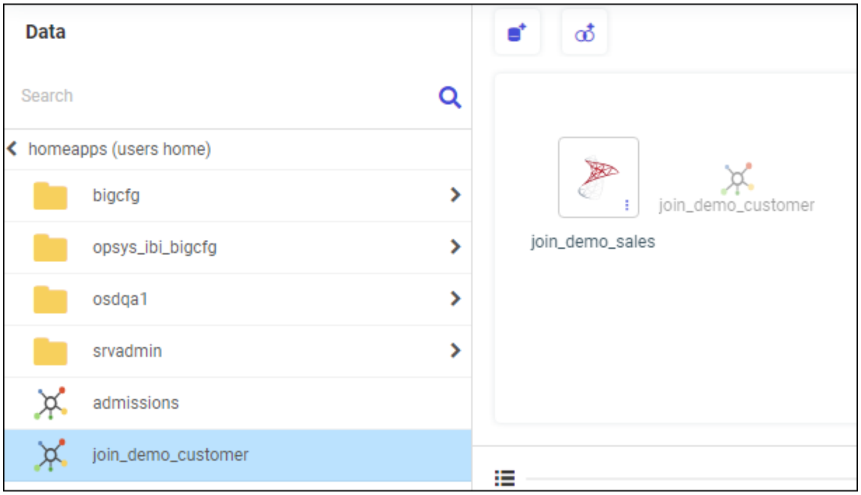
1. From the Db2 Web Query Designer Visualization tab, click the *Data* tab.

Your data source displays as a synonym on the Data tab canvas, and sample data displays in a tabular format similar to a spreadsheet, as shown in the following image.

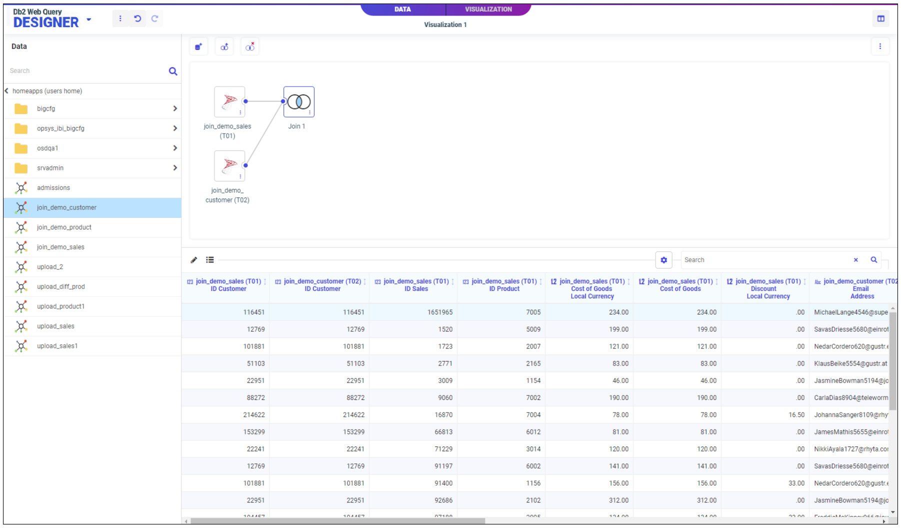


ID Sales	ID Customer	ID Product	Cost of Goods Local Currency	Cost of Goods	Discount Local Currency	Discount	Gross Profit Local Currency	Gross Profit	MSRP Local Currency	MSRP	Quantity Sold	Revenu Local Cu
1651965	116451	7005	234.00	234.00	.00	.00	165.00	165.00	399.00	399.00	1	
1520	12769	5009	199.00	199.00	.00	.00	100.99	100.99	299.99	299.99	1	
1723	101881	2007	121.00	121.00	.00	.00	98.99	98.99	219.99	219.99	1	
2771	51103	2165	83.00	83.00	.00	.00	44.59	44.59	127.59	127.59	1	
3009	22951	1154	46.00	46.00	.00	.00	33.99	33.99	79.99	79.99	1	
9060	88272	7002	190.00	190.00	.00	.00	109.00	109.00	299.00	299.00	1	
16870	214622	7004	78.00	78.00	16.50	16.50	70.49	70.49	164.99	164.99	1	
66813	153299	6012	81.00	81.00	.00	.00	64.99	64.99	145.99	145.99	1	

2. From the Data panel, drag a synonym to the canvas and drop it on your original data source to create a Join, as shown in the following image.



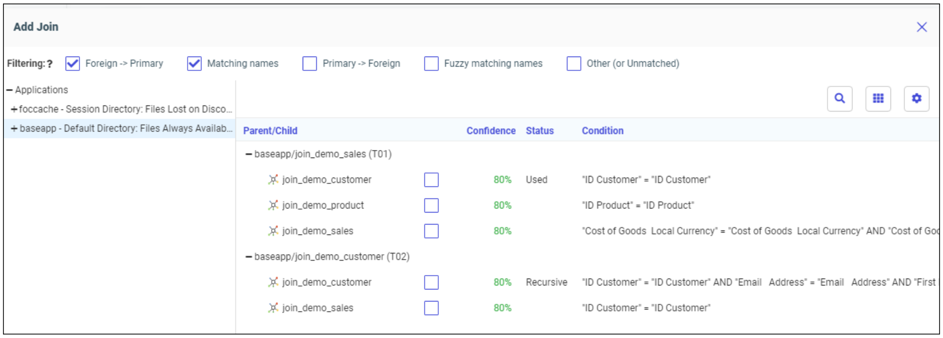
A Join object is added between the two synonyms on the canvas as shown in the following image.



join_demo_sales (T01) ID Customer	join_demo_customer (T02) ID Customer	join_demo_sales (T01) ID Sales	join_demo_sales (T01) ID Product	join_demo_sales (T01) Cost of Goods Local Currency	join_demo_sales (T01) Cost of Goods	join_demo_sales (T01) Discount Local Currency	join_demo_customer (T02 Email Address
116451	116451	1651965	7005	234.00	234.00	.00	MichaelLange4546@supe
12769	12769	1520	5009	199.00	199.00	.00	SavasDriesse5680@einrot
101881	101881	1723	2007	121.00	121.00	.00	NedarCordero620@gustr.e
51103	51103	2771	2165	83.00	83.00	.00	KlausBeike5554@gustr.at
22951	22951	3009	1154	46.00	46.00	.00	JasmineBowman5194@jo
88272	88272	9060	7002	190.00	190.00	.00	CarlaDias8904@teleworm
214622	214622	16870	7004	78.00	78.00	16.50	JohannaSanger8109@rhy
153299	153299	66813	6012	81.00	81.00	.00	JamesMathis5655@einrot
22241	22241	71229	3014	120.00	120.00	.00	NikkiAyala1727@rhyta.cor
12769	12769	91197	6002	141.00	141.00	.00	SavasDriesse5680@einrot
101881	101881	91400	1156	156.00	156.00	33.00	NedarCordero620@gustr.e
22951	22951	92686	2102	312.00	312.00	.00	JasmineBowman5194@jo

The canvas provides the following toolbar options:

- ❑ **Get Data.** Opens the Get Data dialog box, where you can select additional data sources to be joined with your original data source.
- ❑ **Add Join.** Displays potential Join candidates from a relational database, including confidence scores, as shown in the following image.


Add Join

Filtering: ? ☑ Foreign -> Primary ☑ Matching names □ Primary -> Foreign □ Fuzzy matching names □ Other (or Unmatched)

− Applications
+ foccache - Session Directory: Files Lost on Disco...
+ baseapp - Default Directory: Files Always Availab...

Parent/Child		Confidence	Status	Condition
− baseapp/join_demo_sales (T01)				
join_demo_customer	□	80%	Used	"ID Customer" = "ID Customer"
join_demo_product	□	80%		"ID Product" = "ID Product"
join_demo_sales	□	80%		"Cost of Goods Local Currency" = "Cost of Goods Local Currency" AND "Cost of Goo
− baseapp/join_demo_customer (T02)				
join_demo_customer	□	80%	Recursive	"ID Customer" = "ID Customer" AND "Email Address" = "Email Address" AND "First
join_demo_sales	□	80%		"ID Customer" = "ID Customer"

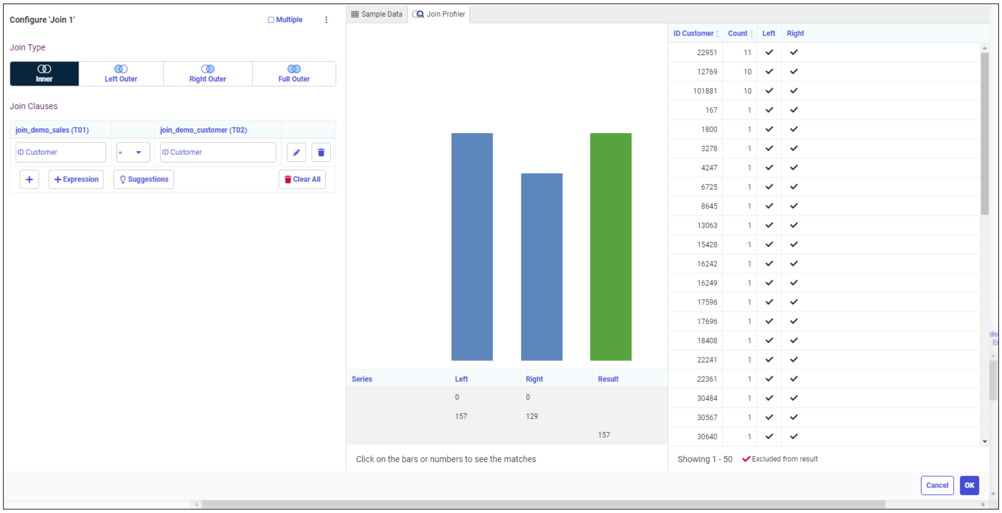
- ❑ **Clear Join.** Removes the Join from the data flow. You can also right-click the Join object or a synonym joined to the original data source, and click *Delete* to remove the Join.

Note: Before removing a recursive Join (when a table is joined to itself) you must remove any fields with active content from the first instance of the table.

The canvas provides the following menu options:

- **Flow view.** An icon represents each synonym, and an icon that displays the type of join represents each join.
- **Model view.** A list box represents each synonym, and a vector between the joined synonyms represents each join.
- **Arrangement.** Select a Layout option to define the spacing between objects on the canvas, or click *Auto Arrange* to reset the display of objects in the canvas to the default arrangement:
    - **Vertically spread out.**
    - **Vertically compressed.**
    - **Vertically compressed with long H-space.**
    - **Default.**
    - **Auto Arrange.**
- **More options.** Sets the following Join defaults:
    - **Insert child segment with snowflakes.** No is the default, which inserts only the segment selected. You can select *Yes* to insert the selected segment and all of its descendants.
    - **Display full component names.** Yes is the default. Full component names are field names qualified/prefixed with the synonym and segment name.
    - **Use Monotree.** Yes is the default. The monotree view displays the current folder in the Data panel and provides navigation to other folders. Select *No* to display all folders in a traditional tree view.
    - **Show undo/redo button with history.** No is the default. Select *Yes* to display the history button.

Click the pencil icon to display a Join configurator and data profiler, as shown in the following image.



Configure 'Join 1'

□ Multiple

Join Type: Inner

Join Clauses

join_demo_sales (T01)		join_demo_customer (T02)
ID Customer	=	ID Customer

Series	Left	Right	Result
	0	0	
	157	129	
			157

Click on the bars or numbers to see the matches

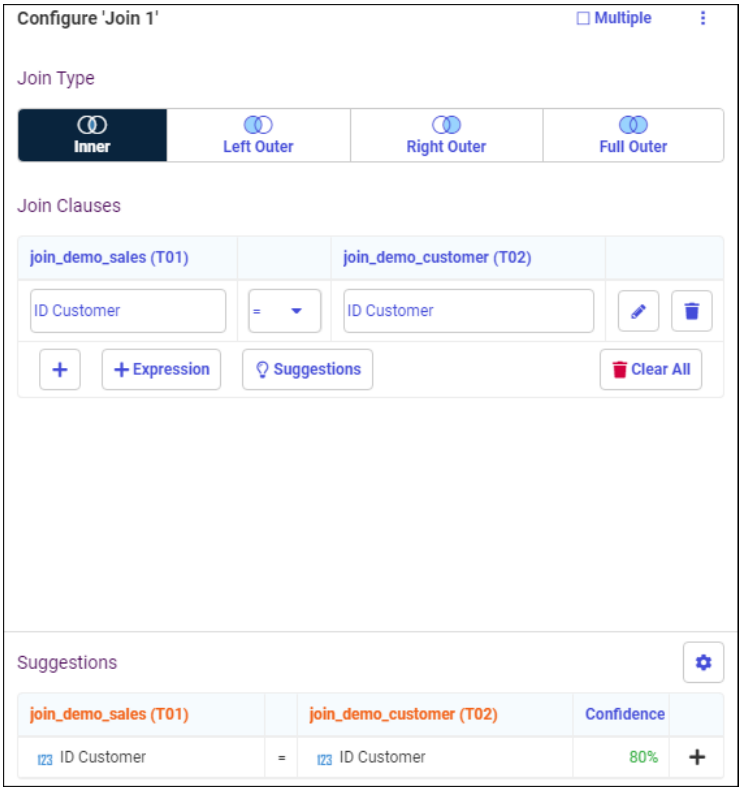
ID Customer	Count	Left	Right
22951	11	✓	✓
12769	10	✓	✓
101881	10	✓	✓
167	1	✓	✓
1800	1	✓	✓
3278	1	✓	✓
4247	1	✓	✓
6725	1	✓	✓
8645	1	✓	✓
13063	1	✓	✓
15428	1	✓	✓
16242	1	✓	✓
16249	1	✓	✓
17596	1	✓	✓
17696	1	✓	✓
18408	1	✓	✓
22241	1	✓	✓
22361	1	✓	✓
30484	1	✓	✓
30567	1	✓	✓
30640	1	✓	✓

Showing 1 - 50 ✓ Excluded from result

Using the Join configurator, you can change the join type or change expressions and operators. Check the Multiple check box to indicate a one-to-many Join. The following Join types are available:

- **Inner.** Returns only records where the selected join fields match between the left-side table and the right-side table. If there are no matches, no records are returned.
- **Left Outer.** Returns all records from the left-side table, and only matched records from the right-side table.
- **Right Outer.** Returns all records from the right-side table, and only matched records from the left-side table.
- **Full Outer.** Returns all records, matched and unmatched, from the left-side table and the right-side table. This type of Join can potentially return very large result sets and can take longer to run.

Click *Suggestions* to display Join Suggestions, including a confidence score based on any database relationships and the number of matching names in the fields, as shown in the following image.

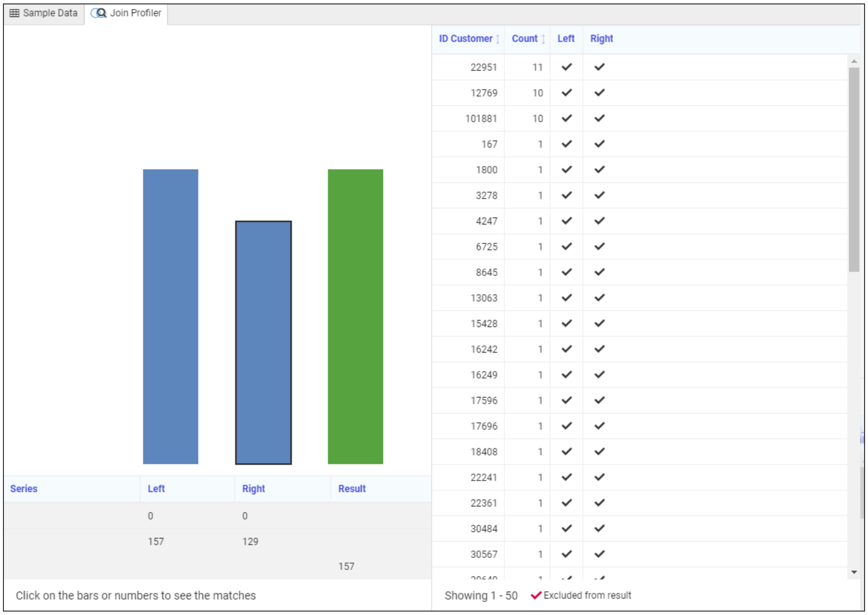


join_demo_sales (T01)		join_demo_customer (T02)	
ID Customer	=	ID Customer	

join_demo_sales (T01)		join_demo_customer (T02)	Confidence	
ID Customer	=	ID Customer	80%	+

Click the plus sign + to add the suggestion to the join pair list.

In the *Join Profiler* tab, a join analyzer chart displays the number of matching values from each part of the join and the total number of values included in the data source as a result of the join, and a key fields grid showing LHS and RHS match states, as shown in the following image.



Series	Left	Right	Result
	0	0	
	157	129	
			157

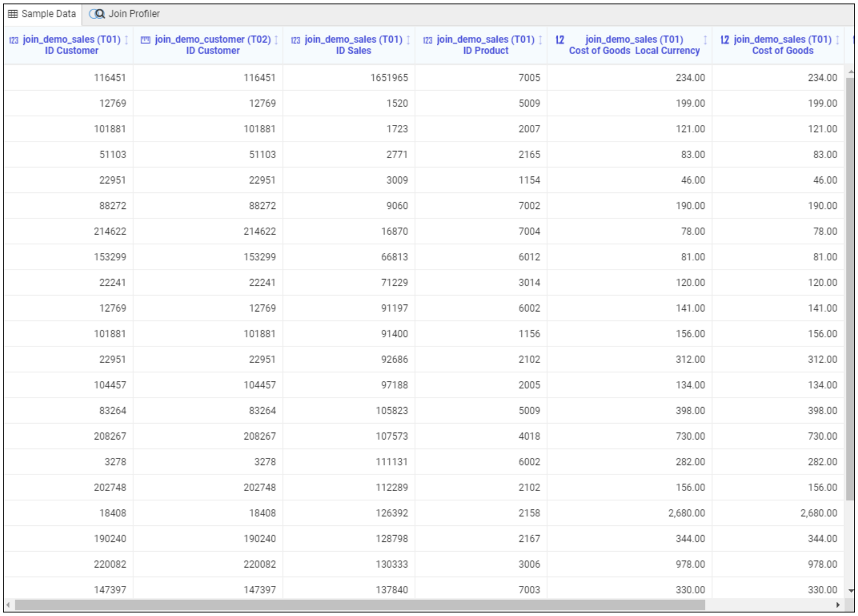
Click on the bars or numbers to see the matches

ID Customer	Count	Left	Right
22951	11	✓	✓
12769	10	✓	✓
101881	10	✓	✓
167	1	✓	✓
1800	1	✓	✓
3278	1	✓	✓
4247	1	✓	✓
6725	1	✓	✓
8645	1	✓	✓
13063	1	✓	✓
15428	1	✓	✓
16242	1	✓	✓
16249	1	✓	✓
17596	1	✓	✓
17696	1	✓	✓
18408	1	✓	✓
22241	1	✓	✓
22361	1	✓	✓
30484	1	✓	✓
30567	1	✓	✓

Showing 1 - 50 ✓ Excluded from result

You can click the table bars or numbers to display the matching fields in the report, and then click the *Clear selections* icon to clear the matches and return the report to show all values.

In the *Sample Data* tab, the values for the field being joined in each synonym are displayed, as shown in the following image.



join_demo_sales (T01) ID Customer	join_demo_customer (T02) ID Customer	join_demo_sales (T01) ID Sales	join_demo_sales (T01) ID Product	join_demo_sales (T01) Cost of Goods Local Currency	join_demo_sales (T01) Cost of Goods
116451	116451	1651965	7005	234.00	234.00
12769	12769	1520	5009	199.00	199.00
101881	101881	1723	2007	121.00	121.00
51103	51103	2771	2165	83.00	83.00
22951	22951	3009	1154	46.00	46.00
88272	88272	9060	7002	190.00	190.00
214622	214622	16870	7004	78.00	78.00
153299	153299	66813	6012	81.00	81.00
22241	22241	71229	3014	120.00	120.00
12769	12769	91197	6002	141.00	141.00
101881	101881	91400	1156	156.00	156.00
22951	22951	92686	2102	312.00	312.00
104457	104457	97188	2005	134.00	134.00
83264	83264	105823	5009	398.00	398.00
208267	208267	107573	4018	730.00	730.00
3278	3278	111131	6002	282.00	282.00
202748	202748	112289	2102	156.00	156.00
18408	18408	126392	2158	2,680.00	2,680.00
190240	190240	128798	2167	344.00	344.00
220082	220082	130333	3006	978.00	978.00
147397	147397	137840	7003	330.00	330.00

To create content from your Join, click the *Visualization* tab and then click *Save* to save your content.

Editing Metadata

Metadata is data about data, and describes the characteristics of your data sources, or synonyms. Metadata can be generated whenever data is created, acquired, added to, deleted from, or updated.

Depending on your version of Db2 Web Query and your permissions, you may have access to additional metadata options, so you can edit previously created synonyms.

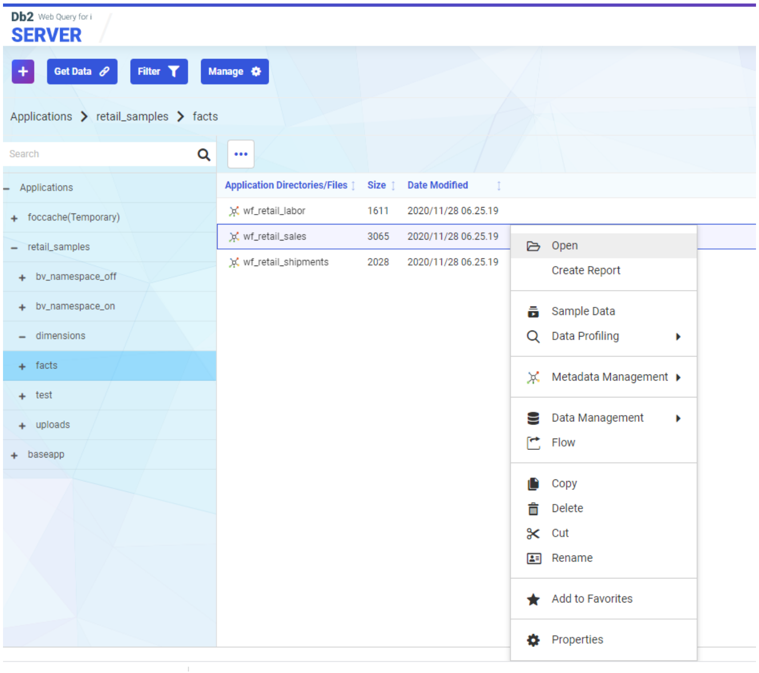
You can edit previously created synonyms in the following ways:

❑ From the Db2 Web Query Home page, click the plus sign +, and then click *Prepare and Manage Data*.

The Applications page opens, from which you can edit your synonym, or choose from other options.

Procedure: How to Edit a Metadata Object

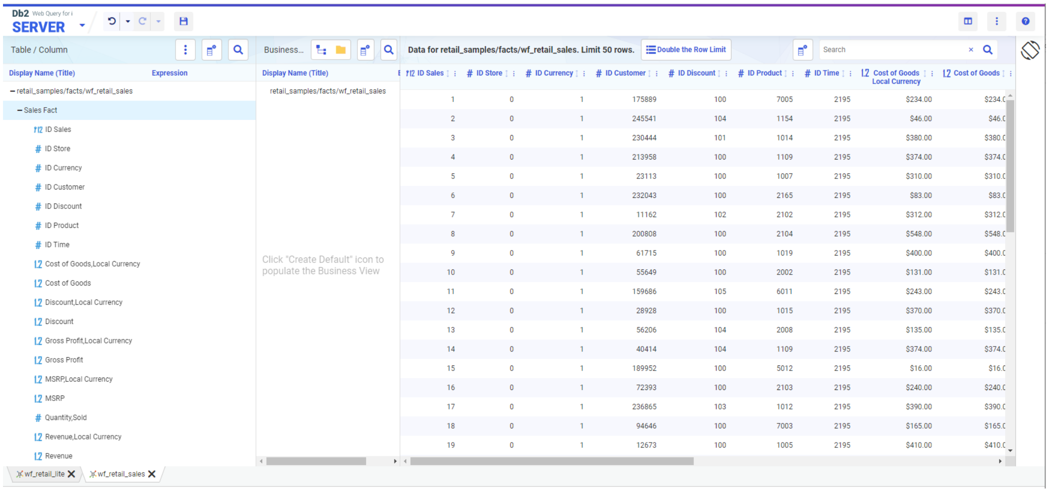
1. From the Applications page, in the Application Directories tree, click a folder to expand it, then right-click on a procedure or metadata file in the file panel and select *Open*, as shown in the following image.



Application Directories/Files	Size	Date Modified
wf_retail_labor	1611	2020/11/28 06.25.19
wf_retail_sales	3065	2020/11/28 06.25.19
wf_retail_shipments	2028	2020/11/28 06.25.19

Note: To edit as text, select *Metadata Management*, and then select *Edit as Text* or *Edit Access File as Text*.

The Synonym Editor opens, where you can modify your synonym, as shown in the following image.



Display Name (Title)	Expression
– retail_samples/facts/wf_retail_sales	
– Sales Fact	
ID Sales	
ID Store	
ID Currency	
ID Customer	
ID Discount	
ID Product	
ID Time	
Cost of Goods,Local Currency	
Cost of Goods	
Discount,Local Currency	
Discount	
Gross Profit,Local Currency	
Gross Profit	
MSRP,Local Currency	
MSRP	
Quantity,Sold	
Revenue,Local Currency	
Revenue	

Display Name (Title)
retail_samples/facts/wf_retail_sales

Click "Create Default" icon to populate the Business View

Data for retail_samples/facts/wf_retail_sales. Limit 50 rows.

ID Sales	ID Store	ID Currency	ID Customer	ID Discount	ID Product	ID Time	Cost of Goods Local Currency	Cost of Goods
1	0	1	175889	100	7005	2195	$234.00	$234.0
2	0	1	245541	104	1154	2195	$46.00	$46.0
3	0	1	230444	101	1014	2195	$380.00	$380.0
4	0	1	213958	100	1109	2195	$374.00	$374.0
5	0	1	23113	100	1007	2195	$310.00	$310.0
6	0	1	232043	100	2165	2195	$83.00	$83.0
7	0	1	11162	102	2102	2195	$312.00	$312.0
8	0	1	200808	100	2104	2195	$548.00	$548.0
9	0	1	61715	100	1019	2195	$400.00	$400.0
10	0	1	55649	100	2002	2195	$131.00	$131.0
11	0	1	159686	105	6011	2195	$243.00	$243.0
12	0	1	28928	100	1015	2195	$370.00	$370.0
13	0	1	56206	104	2008	2195	$135.00	$135.0
14	0	1	40414	104	1109	2195	$374.00	$374.0
15	0	1	189952	100	5012	2195	$16.00	$16.0
16	0	1	72393	100	2103	2195	$240.00	$240.0
17	0	1	236865	103	1012	2195	$390.00	$390.0
18	0	1	94646	100	7003	2195	$165.00	$165.0
19	0	1	12673	100	1005	2195	$410.00	$410.0

2. Make edits to the synonym, as needed.
3. To save your synonym, click the upper left Server icon and select the *Save* menu option.
4. To close the Synonym Editor, click the upper left Server icon and select the *Close* menu option.

Creating Folders

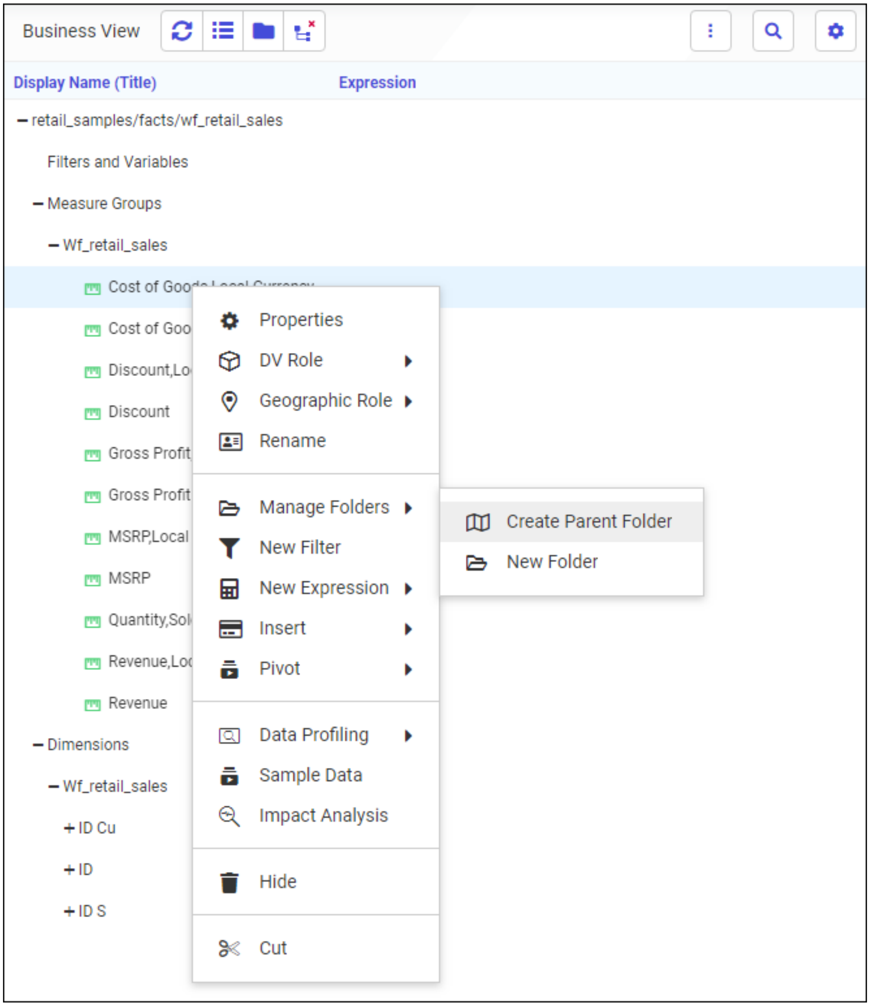
When you are working within a Db2 Web Query Business View panel, you can edit your data by selecting from folder and field shortcut menu options. These options are also available in the Table/Column and Output views.

You can create folders to organize your data. Folders function as segments to provide a view of the synonym, and define the accessible fields and their relationships. Folder relationships are the same as segment relationships, with parent folders, child folders, and sibling folders. Once you create a new folder, you can add data fields and assign roles. Synonym folder structures can be predefined or flexible.

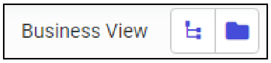
Predefined folder structures are categorized as Measure Groups, Dimensions, Hierarchies, and Attributes, and are created by default when using the Db2 Web Query Upload or Connect to Data workflows. In a predefined folder structure, a Measure Group contains measures or facts that are numeric values, such as Gross Profit or Cost of Goods Sold, that you may want to aggregate. All numeric values that can logically be summed can be categorized as Measures. All other fields are categorized as Dimensions, Hierarchies, or Levels.

Flexible folder structures allow you to insert any number of folders at any level to represent the Business need. A folder can contain any type of field categorized as a Measure or Dimension. However, it is good practice to keep different field types in separate folders.

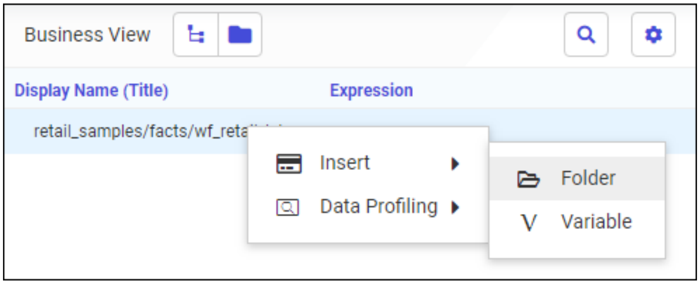
Within a Db2 Web Query Business View panel, you can create a new folder by right-clicking a folder or field, pointing to *Manage Folders*, and then clicking *New Folder*, as shown in the following image of a predefined folder structure.



When editing or creating a synonym, you can click the icons at the top of the Business View panel to create a populated, predefined folder structure, or an unpopulated, predefined folder structure, as shown in the following image.

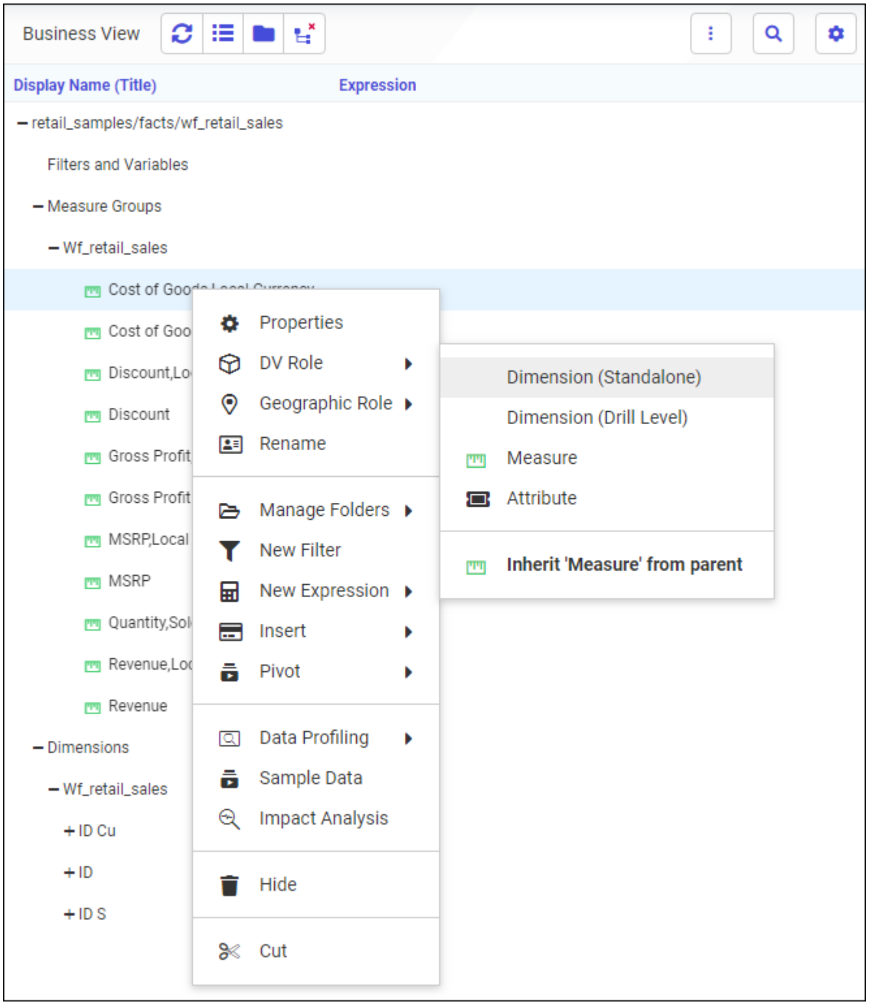


Note: You can start to create a new flexible folder structure by right-clicking the synonym name path, pointing to *Insert*, and then clicking *Folder*, as shown in the following image.



Assigning DV Roles

Categorization can be assigned using the DV Role option. When editing a synonym within a Db2 Web Query Business View panel, you can assign a DV role to a folder or field by right-clicking the folder or field, clicking *DV Role*, and then clicking a DV Role option, as shown in the following image of a predefined folder structure.



You can explicitly assign a DV Role to a folder or field, or have it automatically inherit a role from its parent. If you explicitly assign a DV Role, that role moves with the object when dragged to a different location within the Business View structure. If you do not explicitly assign a DV Role, the role changes when the object is dragged under a new parent. However, if the object is dragged under a parent with a field with a Drill Level role, it inherits that role.

A folder can only have one drill level field hierarchy. However, you can use the same fields in multiple hierarchies by placing each hierarchy in a separate folder. A folder with a drill level hierarchy is not limited to the hierarchy and can contain other fields with different DV Roles.

You can assign the following DV Roles.

- **Dimension.** A dimension is a way to categorize data or sort output. You can use a dimension to analyze and compare measures. Dimension fields are generally alphanumeric fields, such as Product.

  Dimension fields can be organized into hierarchies to define the relationships between them, and provide information for the AUTODRILL feature. For example, a Geography hierarchy can contain the Continent, Country, State, and City fields. When AUTODRILL is turned on, automatic drill-down links are created in the report or chart output.

  A dimension field, when double-clicked or dragged onto the report or chart canvas, will automatically be added to the request as a vertical (BY) sort field. A folder can be assigned the role Dimension. A field can be assigned the role Dimension (Standalone) or Dimension (Drill Level). When it is assigned the role Dimension (Drill Level), it becomes part of a hierarchy, where the levels depend on the order of the fields in the folder. You can also define dimension fields that are not part of a dimension hierarchy.

- **Measure.** You can use a measure to aggregate data. A measure field, when double-clicked or dragged onto the report or chart canvas, will automatically be added to the request as an aggregated value (SUM), if it is numeric. A folder or field can be assigned the role Measure.

- **Attribute.** An attribute is a field or collection of fields that add additional information about a dimension. An attribute field, when double-clicked or dragged onto the report or chart canvas will automatically be added to the request as an aggregated value (SUM), if it is numeric, or as a vertical sort field (BY), if it is alphanumeric.

- **Folder.** A folder is a virtual segment in a Business View. It can be assigned the roles Dimension, Measure, or Attribute.

  Note: When a folder is inserted as a child of a field, the PARENT_FIELD attribute describes this relationship. By default, such a folder and its fields will be assumed to have the Attribute role.

Chapter 3

Organizing Content Into Pages

Pages use containers to display multiple content items, and provide run-time options such as the ability to expand and collapse containers, navigate between items in a multi-content container, apply and clear filter selections, and even use custom functionality added using JavaScript. You can customize and style the entire page as well as sections, containers, and filter controls on the page using an intuitive set of options as well as CSS themes and custom CSS enhancements.

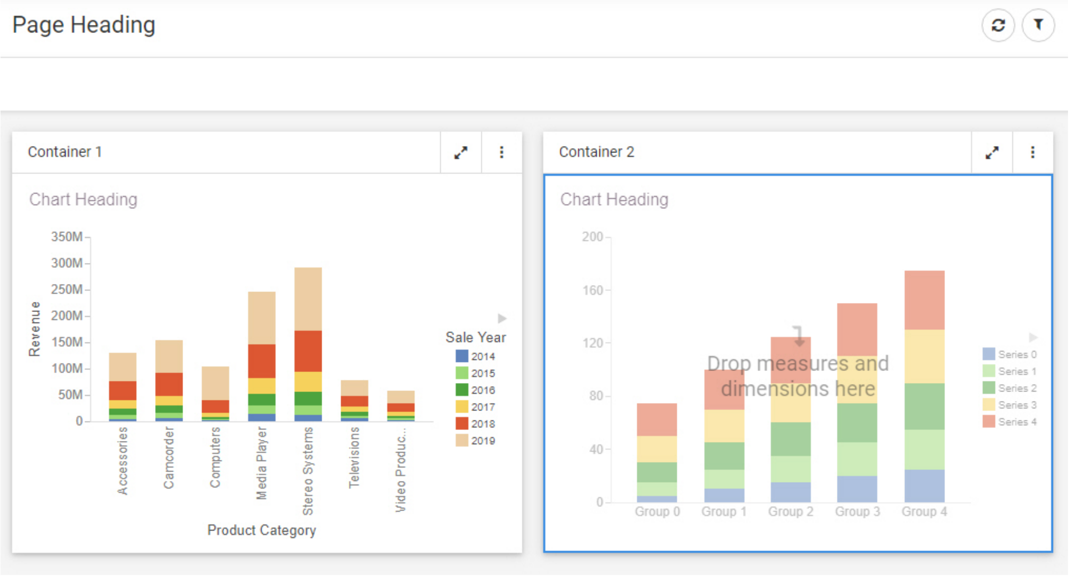
You can create a page in two ways. One the Db2 Web Query Home Page, click the plus button and then click *Assemble Visualizations* to create pages using existing content from your environment. Alternatively, click *Visualize Data* or click the plus button and click *Create New Visualization* to create new content. When you click *Add visualization* on the Visualization toolbar to add a second content item, or click *Convert to page*, the visualization transforms from a single chart or report into a page.

In this chapter:

Creating Pages

You can create a page in two ways. When you create a visualization with new content using the *Create New Visualization* option and then add a second item to it by clicking *Add visualization*, it transforms from a single item into a page. You can also transform a single item into a page by clicking *Convert to page*. Alternatively, you can create a page from existing content by clicking *Assemble Visualization* to open Db2 Web Query Designer. Pages can include multiple sections, each with multiple containers of content. Each of these containers can also be configured to contain multiple content items, providing a thoroughly comprehensive overview of your data. Pages also include run-time filtering, content navigation, and content customization, allowing users to analyze, explore, and enhance the information provided on each page.

To add a second item to a visualization, transforming it into a page, click *Add visualization* on the Visualization toolbar. The first content item is moved into a container, and a second container is added where you can create new content. A visualization to which a second content item has just been added is shown in the following image.



Select an item on the page from the canvas or the outline to edit it using a relevant set of tools and options, whether the item is a content item or a page component. The Settings and Format tabs on the Properties panel update accordingly. When assembling a page from existing content, you can also select multiple components, sections, or filters while holding down the Ctrl key in order to edit or style multiple items of the same type at once. Additionally, some options are applied hierarchically. For example, setting a theme for the page applies that theme to any sections, containers, and content items that use the default style option.

You can add more new content to the page by clicking *Add visualization*, or by clicking the *Container* tab on the sidebar and dragging a container onto an empty area of the page from the Resources panel. You can also drag a field onto an empty area of the page to create a new item.

If you are creating a page from existing content, you can add existing content from your repository by clicking *Content* on the sidebar, and then navigating to and dragging an item onto the page or into a container. These items can be external charts, reports, HTML pages, URLs, images, and more. You can navigate between folders in the Resources panel and locate an item by name using the search bar.

You can also duplicate an item by right-clicking the container or subsection that contains it and clicking *Duplicate*. This makes it easy to develop content by maintaining the styling, fields, and other properties from the existing content item and container. The title of the duplicated item is reused, with the text *(Copy)* added to the end. If you duplicate the content in a single-item panel container, a new container with the same content is added to the page. If you duplicate a content item in a multi-content container, such as a tab, carousel, or accordion container, a new subsection is added to the container with the selected content item duplicated.

You can move and resize containers on the page to organize them, or select a preset layout from the content picker to automatically create and arrange containers into a preset pattern.

You can save space on the page by placing multiple content items together in a tab, carousel, or accordion container.

If you are creating a page from existing content, to change a standard panel container into a tab, accordion, or carousel container, select *Content* on the sidebar and then drag a content item onto the container, already populated with content, on the canvas. The entire container is highlighted, indicating that the item will be dropped into the container. When you release the item into the container, you are prompted to replace the existing content, add the item in a new tab, accordion panel, or carousel slide, or cancel the operation. Alternatively, to instantly convert a container to another type, you can right-click it on the canvas and point to *Convert to*, then select a container type of your choice. If you convert a container with multiple items into a panel container, which only supports one content item, each content item from the original container is added to a separate panel container. For example, if you convert a tab container with three tabs into a panel container, three panel containers are created.

To delete a container from a page, right-click it and click *Delete container*. If the container includes multiple items, you can click *Delete Tab*, *Delete slide*, or *Delete Area* to delete a single sub-component, depending on the container type. You can also delete the selected container or sub-component by pressing the Delete key.

When creating new content in a page, you can filter the entire page by dragging a field from the Resource panel into the Filter toolbar, just like when you create a content item. You can select default values for the filter from the resulting filter control, and right-click it to change other filter properties. If you are creating a page with existing content, then when parameterized items are added to a page, a badge appears on the Filters tab of the sidebar, alerting you that you can create filter controls. Open the Filters tab and click *Add all filters to page* to create filter controls for all parameters, or right-click an individual parameter in the list and click *Add to page* to create a filter control for a single parameter. These controls will apply filters to any items that are parameterized for the same amper variable and field.

As you create your page, you can click *Run in new window* to see a run-time view of your page, allowing you to interact with filtering, built in container controls, In-Document Analytics, and any unlocked or interactive content on the page. You can check these variables by clicking *Info* on the Visualization toolbar.

Be sure to periodically save your page by clicking *Save* on the Designer toolbar.

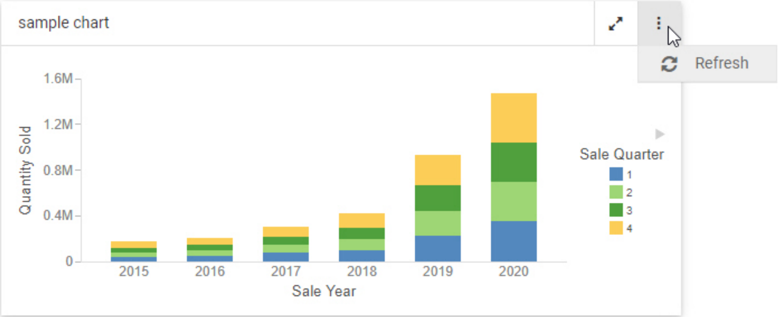
Styling Pages

A number of tools in Db2 Web Query Designer support your efforts to apply styling to pages and page components. You can resize content, hide content from smaller devices if it interferes with a responsive layout, edit page, section and container properties, apply filters, themes and styles, and create custom templates.

Working With Content Containers

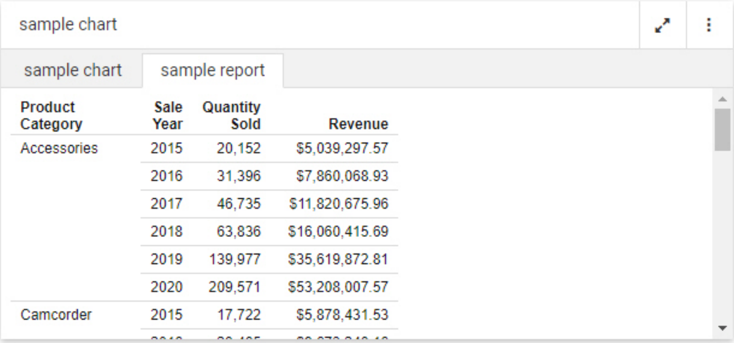
When you add content to a page, it is automatically placed in a container. To move and resize the content item, move and resize the container into which it was placed by dragging the container or its resizing handles. You can also configure and style a container by selecting it on the page and then using the options on the Settings and Format tabs of the Properties panel. Note that the options for a container are separate from those for the content item inside the container. To select the container, click the container toolbar, or select a container from the Outline tab of the sidebar. Empty containers can be added to the page by selecting the *Container* tab on the sidebar and dragging the selected container type into the canvas. A highlighted area shows where the container will be placed.

There are several types of containers. The default container for a single item is a panel container. The panel container is a basic container type that includes a title bar and toolbar with a run-time menu, as shown in the following image.



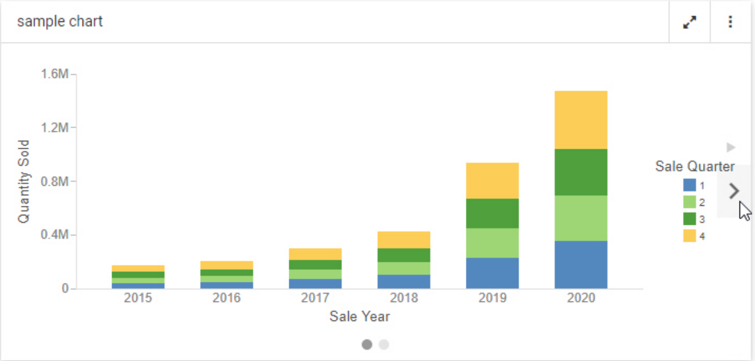
To add multiple items, such as charts, reports, and images, to a single container, use the tab, carousel, or accordion container types. These container types include the same features as the panel container, as well as controls to navigate between multiple content items. You can add another content item to a container by selecting the *Content* tab on the sidebar, navigating to an item, and dragging it onto the existing container. You are given options to add the new item to a new tab, panel, or slide, or replace the visible, existing item in the container.

In a tab container, navigate between items using the tabs below the title bar. You can double-click the text in a tab to change it, and right-click a tab to delete, duplicate, or reorder it. A tab container is shown in the following image.

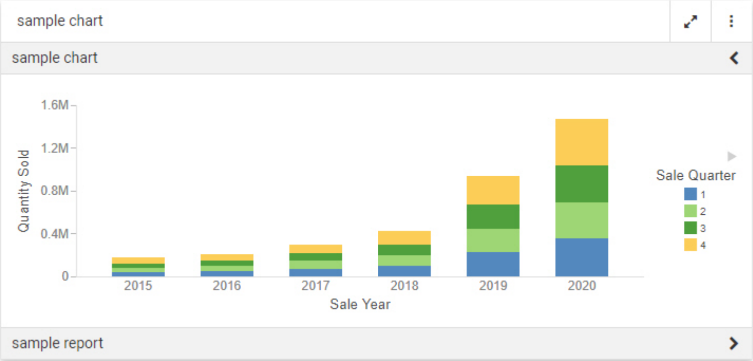


Product Category	Sale Year	Quantity Sold	Revenue
Accessories	2015	20,152	$5,039,297.57
	2016	31,396	$7,860,068.93
	2017	46,735	$11,820,675.96
	2018	63,836	$16,060,415.69
	2019	139,977	$35,619,872.81
	2020	209,571	$53,208,007.57
Camcorder	2015	17,722	$5,878,431.53

A carousel container allows you to switch between content items by clicking a dot at the bottom of the container, or by clicking arrows that appear when you point to the left or right side of the container. When creating a page, right-click a carousel container to duplicate, reorder, or delete a slide. You can set a carousel container to automatically cycle between content items using the options described in *Automatically Advancing Slides in a Carousel Container* on page 87. A carousel container, with navigation arrows, is shown in the following image.



Accordion containers place content in collapsible panels. You can expand and collapse items in an accordion container by clicking a panel label. When creating a page, right-click an individual panel in an accordion container to rename, reorder, duplicate, or delete the panel. An accordion container is shown in the following image.



Other types of containers serve more specific purposes. The grid container type, only available in pages assembled from existing content, can be used as a filter container, as an alternative or in addition to the filter toolbar. Once you add filter controls from you content to a page, you can drag them from the default location in the filter toolbar into the grid control. Select the filter bar within the panel from the outline, and then select the *Format* tab to change the number of columns in the grid container. You can also drag a text label or submit button to the grid from the Control tab on the sidebar.

You can use a panel group container to keep a set of vertically arranged panels together when responsive folding occurs as the size of the page changes. To use a panel grid container, add it to the page, then drag new containers, content items, or fields into it. For more information, see *Adding Panels to a Responsive Container* on page 134.

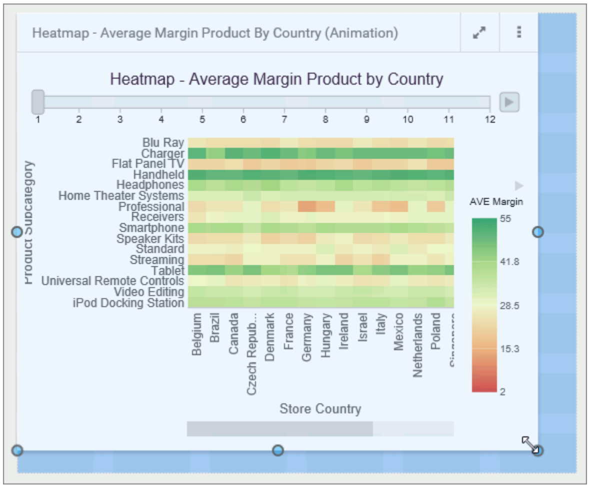
A link tile container allows you to display a content item or image on a page that can be clicked to access additional, outside content. Drag the link tile container onto the page, then, from the *Settings* tab, use the link tile Background, Content, Attach all parameters, and Target properties, to set the displayed and target content, and configure how the link tile behaves. For more information, see *Linking to External Content From a Page* on page 135.

You can use the Explorer widget to provide access to the Workspaces view of the Db2 Web Query Home Page. This allows anyone accessing the page to open and run content items and create new content, depending on their privileges. In the Explorer widget, you can navigate through workspaces, folders, and content items from the Resources tree and Content area, and create new content using options on the Action bar. For more information, see *Providing Access to Content Items and Db2 Web Query Tools* on page 141.

Right-click any type of container to access a shortcut menu of options. These include the ability to refresh the container content, change the title of the container, access the Settings or Format tabs, duplicate the visible content item, change the container type, or delete the container.

Resizing Content on a Page

You can resize content on a page, at any time, using the sizing handles and the shaded placement area that appear on the canvas. When you hover over a container on the canvas, a series of handles appear. Drag a handle in the direction that you want to resize the item, as shown in the following image.



The content container snaps to the nearest row and column when you resize it. To change the number of columns in a section on the page, allowing more granular container widths, select a section, open the *Format* tab, and select an option from the Number of Columns area. 12 is the default. You can also change the amount of space between containers by selecting the entire page and typing a size for the Margin property, on the Format tab.

Editing Page, Section, and Container Properties

The Properties panel displays the properties for the element that is selected. To access container properties, select a container from the canvas or outline. To access section properties, click the grid inside the canvas, or select a section from the outline. To access page properties, click the page header or the toolbar, or select a page from the outline. If the page header and the toolbar are hidden, you can also access page properties by clicking the canvas outside a grid section.

The following properties are available when you select an entire page.

Settings tab:

- **ID.** Contains a read-only unique CSS identifier.

- **Classes.** Allows you to add one or more custom CSS classes that you can reference in custom JavaScript and CSS code.
- **Show page heading.** Toggles between hiding and showing the title in the header.
- **Show page toolbar.** Toggles between hiding and showing the page toolbar.
  - **Show page refresh.** Toggles between showing and hiding the refresh button on the page toolbar.
  - **Show export.** Toggles between showing and hiding the Export menu on the page toolbar. You can export a page as a PDF or as a .png image. Select just the *PDF* or just the *Image* check box to make just one of these export options available from the page, in which case the Export menu turns into an Export button.
- **Include Page Filters.** Select this check box to add a filter toolbar to the page, even if no filters have been added yet. Clear this check box to remove the filter toolbar.

  If *Include Page Filters* is selected, you can select one of the following options to change the position of the filter toolbar:

  - **Below Header.** The filter toolbar is placed below the page toolbar and header. This is the default.
  - **Above Header.** The filter toolbar is placed above the page toolbar and header.
  - **Modal.** The filter toolbar is accessible in a modal window overlaid onto the page. Click the *Show filters* button on the page toolbar to open the modal window and make filter selections.

Format tab:

- **Theme.** Allows you to select a theme for the page. The default options are Designer 2018, Light, Midnight, and Vivid. Administrators can configure additional themes that will be available to users from this property. For more information, see *Applying Themes and Styles to Pages* on page 88.
- **Margin.** Controls the size of the margin between the border of the page and the content.

  Note: If the margin property is set to 0, you cannot select a section on the page from the canvas or access the section shortcut menu options. As a workaround, change the margin property value to 20px temporarily to gain access to the section and its options.
- **Maximum width.** Controls the maximum width of the page.

- **Page Heading Style.** Provides a selection of typeface styles for the page heading. Additional header text formatting can be done using CSS. For example, you can assign a class to the page and use the font-family property to change the header text font.

The following properties are available for sections.

Settings tab:

- **ID.** Contains a read-only unique CSS identifier.
- **Classes.** Allows you to add one or more custom CSS classes that you can reference in custom JavaScript and CSS code.
- **Collapsible.** Toggles between collapsible and noncollapsible modes of the selected section.

Format tab:

- **Style.** Allows you to choose a style for the selected section, or select *Default* to use the theme selected for the page.
- **Row height.** Sets the height of the section that is currently selected on the canvas. The default value is 60px.

  Note: Changing the row height does not change the margin, which can be configured on the page level.
- **Number of Columns.** Allows you to change the number of columns in a section, enabling more granular panel widths. Panels snap to the nearest column and row when you resize them. 12 columns are used by default, and 28, 45, and 80 column options are also available.

The following properties are available for containers.

Settings tab:

- **Container Settings.** Provides access to the following properties.
  - **ID.** Contains a read-only unique CSS identifier.
  - **Classes.** Allows you to add one or more custom CSS classes that you can reference in custom JavaScript and CSS code.
  - **Show title.** Toggles between hiding and showing the title.
  - **Show toolbar.** Toggles between hiding and showing the container toolbar.
  - **Show On.** Allows you to hide the selected container or item from displaying on specified devices. The options include desktop, tablet, and mobile.

  - **Autoplay interval.** Available for carousel containers. Allows carousel slides to automatically cycle based on the selected interval, in seconds. When set to 0, autoplay is disabled.

  - **Rerun content.** Available for carousel containers. When the Autoplay interval is set to a value higher than 0, the Rerun content option allows you to refresh the content on each slide of a carousel container when it appears. If Rerun content is not enabled, then the content on each slide only loads the first time it appears.

- **Container Customization.** Provides options that allow users to change the content in a page at run time. Includes the following properties.

  - **Lock container.** If disabled, allows users to change content in the panel at run time. This property is enabled by default.

  - **Select content from.** Allows you to set the initial directory that users will see when they click the *Add content* button.

    Note: To see the full value displayed in the field, widen the Properties panel area by clicking the separator and dragging it to the right.

  - **Lock path.** When enabled, limits the content selection area to the directory that is specified in the Path property. This property is enabled by default.

  - **Flatten list.** If enabled, hides the folder hierarchy and the breadcrumb trail within the directory that is specified in the Path property. This property is disabled by default.

    Note: This property is especially useful if you want to display different widgets to different users based on their roles. For example, you can create a series of subfolders with content and then apply security rules to show and hide these subfolders from users based on their group membership. Then you can configure the *Path* property to display the parent of these subfolders and flatten the list. Now each user can only see the widgets they are authorized to see in a simple list without any folders to navigate.

  - **Hide tags.** If enabled, hides tag buttons in a flattened list. This property is disabled by default. It remains inactive when the *Flatten list* property is disabled.

  - **Initial view.** Determines the initial view of the directory that users will see when they click the *Add Content* button. The options are Grid and List.

    Note: Users can change the view at run time inside the Select Item dialog box.

Format tab:

- **Style.** Allows you to choose a style for the selected container or item.

The following options are available for content items in a page assembled from existing content. New content items in a page provide standard content creation options, in addition to these.

Settings tab:

- ❑ **Content Properties.** Provides access to the following properties.
    - ❑ **ID.** Contains a read-only unique CSS identifier of the content item. You can reference this ID in custom JavaScript and CSS code as well as drill-down targeting procedures.
    - ❑ **Classes.** Allows you to add one or more custom CSS classes that you can reference in custom JavaScript and CSS code as well as drill-down targeting procedures.

Format tab:

- ❑ **Content Format.** Provides access to the following property.
    - ❑ **Theme.** Select a theme to apply to the content item. You can explicitly select a theme, or have the item inherit the theme used in the page. When the item is part of a page assembled from existing content, you can also select *Use content theme* to use the theme originally applied to the item, ensuring that it does not get overwritten by the page. For more information, see *Applying Themes and Styles to Pages* on page 88.

Hiding Content From Devices

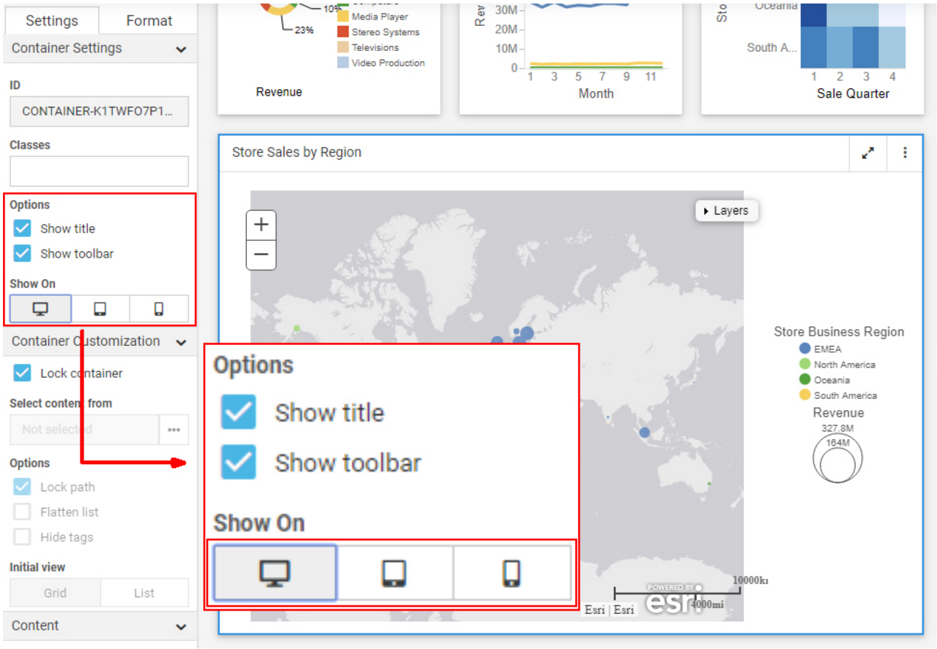
You can choose to hide certain content items from smaller devices, especially when dealing with large or multi-faceted items like maps or HTML pages. When an item is hidden from a device, the other items around it re-flow into the empty space and retain the responsive layout.

Procedure: How to Hide a Content Item From Devices

1. In Db2 Web Query Designer, select a large item that you want to hide from small-screen devices.
2. On the Properties panel, on the Settings tab, under the Show On property, click a device icon to hide the selected item from this type of device.

   Note: All three device buttons are selected, by default, making the item visible on all devices. Manually clear the buttons to hide the item from specific device types.

An example of hiding a large item from mobile phones and tablets is shown in the following image.



Settings | Format

Container Settings

ID: CONTAINER-K1TWFO7P1...

Classes:

Options

☑ Show title

☑ Show toolbar

Show On

Container Customization

☑ Lock container

Select content from: Not selected

Options

☑ Lock path

☐ Flatten list

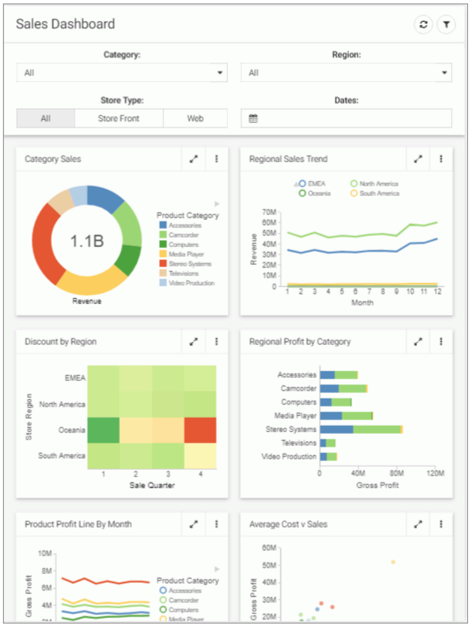
☐ Hide tags

Initial view: Grid | List

Content

3. Save the page.
4. Run the page on a device, from which the item was hidden.

The item does not display. All remaining content items rearrange to cover the empty space, as shown in the following image.



Sales Dashboard

Category: All

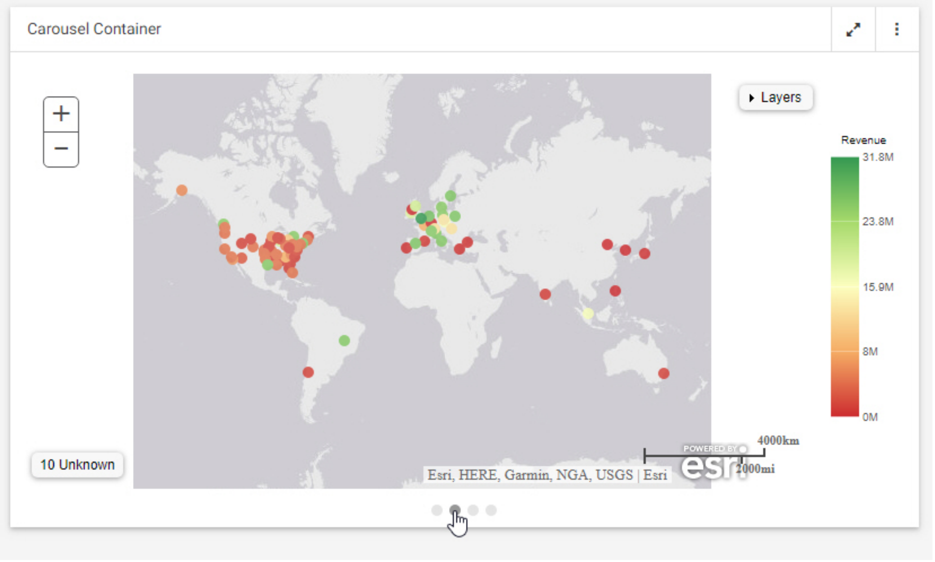
Region: All

Store Type: All | Store Front | Web

Dates:

Automatically Advancing Slides in a Carousel Container

Using a carousel container is one way to put multiple charts, reports, and other content items into a single container. Carousel containers include an unobtrusive control that allows users to cycle through the items added to it, as shown in the following image.



Unlike tabbed or accordion containers, carousel containers also include the option to autoplay the content within them, allowing you to show all of the content in the carousel container without requiring user interaction.

To enable autoplay for a carousel container, select the container and open the Settings tab on the Properties panel. The Autoplay interval and Rerun content properties are available in the Container Settings section. To enable autoplay, set the Autoplay interval to a value higher than zero (0) to enable autoplay. The Autoplay interval is the amount of time, in seconds, that each slide is shown before advancing to the next slide.

When the Autoplay interval is set, the Rerun content option becomes available. This option is turned off, by default. When enabled, the content on a slide of the carousel container refreshes each time it is displayed. When Rerun content is not enabled, the content on each slide only loads the first time it is displayed, and it is not refreshed unless manually prompted from the menu on the container toolbar.

Applying Themes and Styles to Pages

While customizing your page, you can apply themes and styles to various page elements. The general theme of the page is defined by the Theme setting, which you can configure in the Properties panel, on the page level. A theme affects the look of the entire set of elements on the page, including colors, opacity, and typeface styles. Themes also dictate the color schemes available as styles on the Format tab when editing sections, filters, or containers.

By default, the theme selected for a page is passed on to content in the page. You can select content items inside a container and change the theme for each one individually from the Format tab. If the content item was added to the page from the Content tab, you can additionally choose to use the original theme of the content item, instead of inheriting the page theme or selecting a specific theme to use.

There are four themes that Db2 Web Query Designer offers:

❑ Designer 2018

❑ Light

❑ Midnight

❑ Vivid

To add a theme to this list, create a folder in the Global Resources area, in the Custom folder within the Themes folder. In that folder, you can add a custom StyleSheet to use with charts and reports, and a custom CSS file to use for pages. The .sty file and .css file must both be named *themes*.

Note: To facilitate the creation of a custom theme, you can copy and modify properties from one of the themes included with Db2 Web Query. This makes it easy to identify the default classes used in a Designer page.

Once you select a theme, you can further modify it by configuring styles. You can also save your unique combination of a theme and styles as a custom theme.

Procedure: How to Apply a Theme to a Page

1. In Db2 Web Query Designer, click the page toolbar to select the entire page, and then click the *Format* tab on the Properties panel.

   The style properties appear.

2. In the Page Style section, from the Theme property drop-down list, select the theme that you want to use.

   The page refreshes with the new theme.

3. Save your changes.

Procedure: How to Apply Styles to Sections

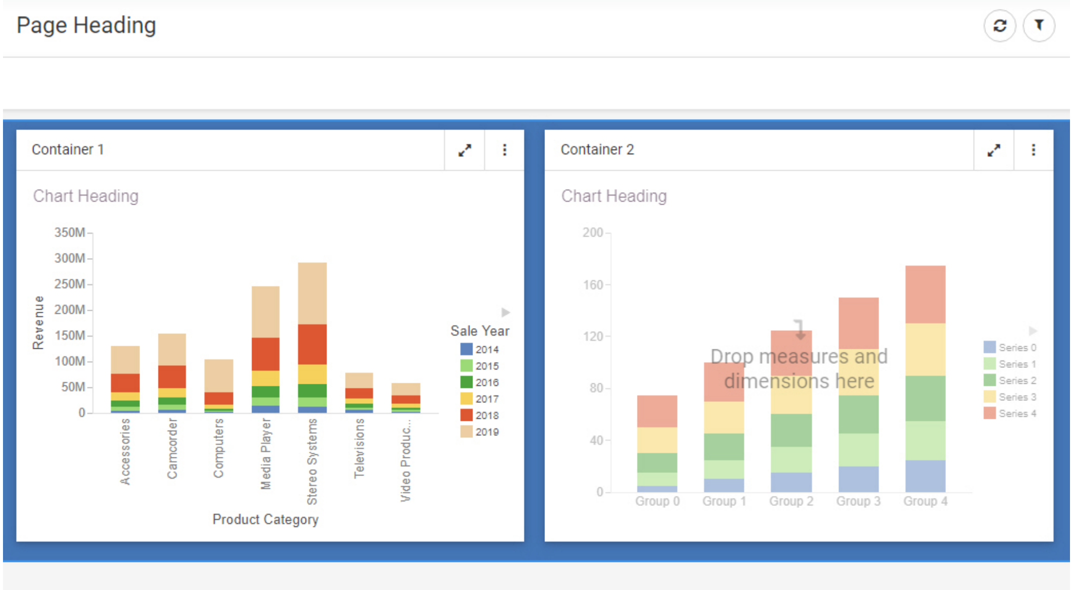
1. In Db2 Web Query Designer, select a page section by clicking the background area near a container, or by selecting a section from the outline.
2. On the Properties panel, click the *Format* tab.

   The Style properties appear.

3. Click a style that you want to use.

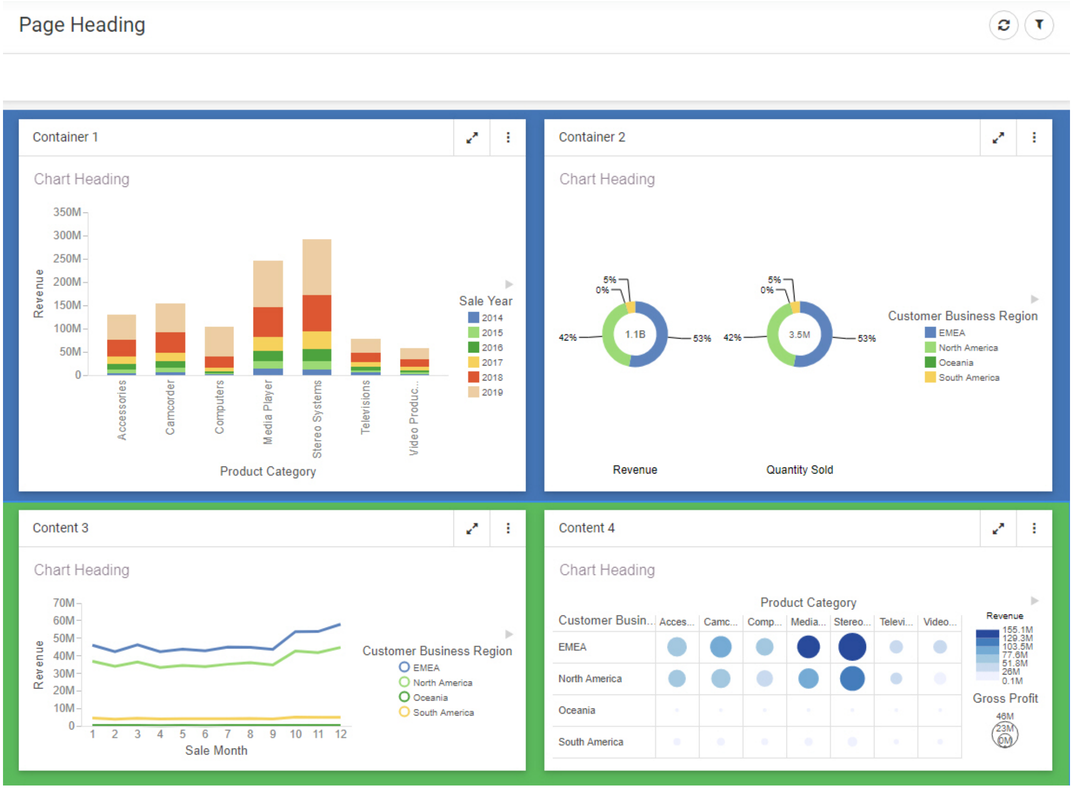
   Note: Styles are based on classes in the selected theme. For example, use the class .pd-style-default-color to configure the Default style of the theme, use the class .pd-style-color2 to configure Style 2 of the theme, and use the class .pd-style-color3 to configure Style 3 of the theme.

   The page refreshes and applies the style to the section, as shown in the following image, where the selection of Style 2 has made the first section blue.



4. Optionally, add more sections and apply styles to them.
5. Click the new section and apply a style to it, as described in step 3.

The page refreshes and applies the new style. You can apply different styles to different sections, as shown in the following image.



6. Save your changes.

Procedure: How to Apply Styles to Containers

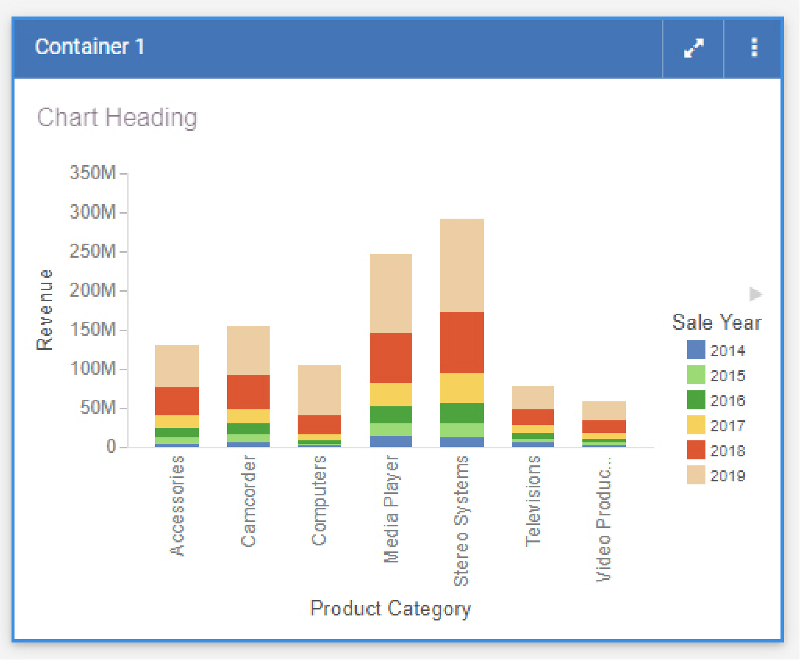
1. In Db2 Web Query Designer, create a visualization with multiple containers.
2. Click a container, and, on the Properties panel, click the *Format* tab.

   The Style properties appear.

3. Click a style that you want to use.

   Note: Styles are based on classes in the selected theme. For example, use the class .pd-style-default-color to configure the Default style of the theme, use the class .pd-style-color2 to configure Style 2 of the theme, and use the class .pd-style-color3 to configure Style 3 of the theme.

The page refreshes and applies the style to the container, as shown in the following image.



4. Optionally apply styles to other containers on the canvas.

   Note: When creating a visualization from existing content, to apply the same style to multiple containers, multi-select containers by holding the Ctrl key, and then click a style button.

5. Save your changes.

Procedure: How to Create a Custom Theme

1. Sign in to Db2 Web Query as an administrator.
2. On the Home Page, click *Workspaces* and, from the Resources tree, expand the *Global Resources* folder, and then expand the *Themes* folder.
3. Open the *Custom* folder, and then click *Folder* on the Action bar.

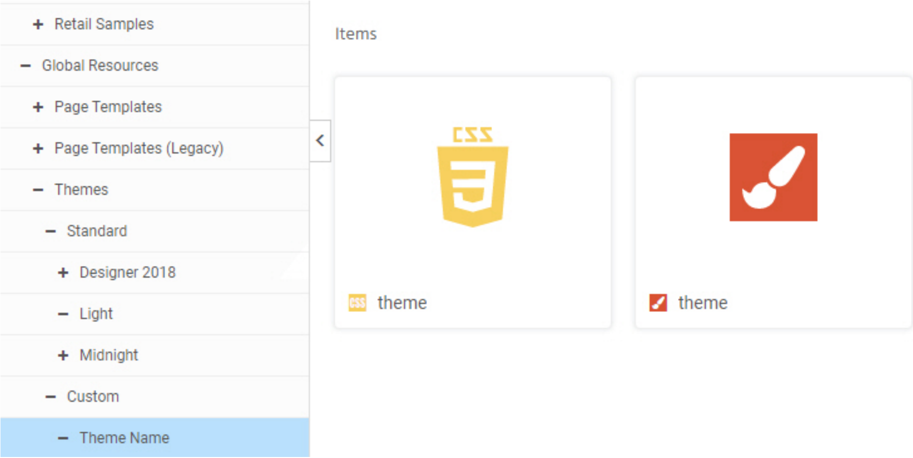
   The New Folder dialog box opens.

4. Populate the Title field with the name of your custom theme, and click *OK*.

The custom theme folder is created. Your theme CSS file will reside in this folder. If you know which CSS classes should be used for your theme, you can create a new CSS text file, add your code, and save it with the name *theme*. The file must be called *theme* to be available for selection in Db2 Web Query Designer. To facilitate the process, you can modify an existing theme CSS file. In this example, we copy and modify the theme CSS file for the Light theme.

5. Expand the *Standard* folder, and then expand the *Light* folder.
6. Copy the theme CSS file, and paste it inside your new custom theme folder.

   The following image shows the correct hierarchy of the custom theme file.



   Note: Do not modify the name of the theme CSS file. It is imperative that both the CSS file and StyleSheet in your custom theme folder are called theme. The name of the folder in which the file resides is the theme name that is visible in Db2 Web Query Designer.

7. Right-click the newly copied theme CSS file, and then click *Edit*.

   The Text Editor window opens.

8. Modify the code to achieve the desired look of the theme.

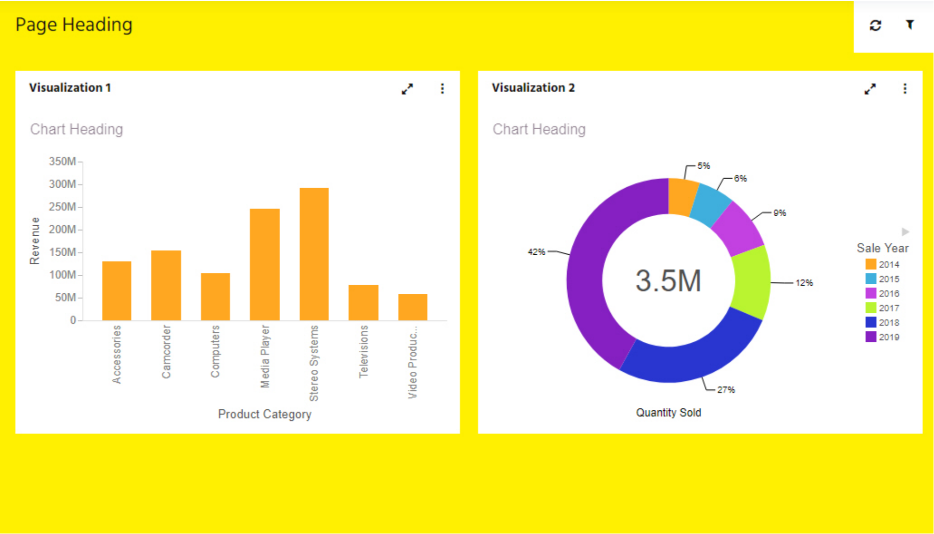
   In this example, you could change the background color from white #fff to yellow #ffff00.

9. Save and close the Text Editor.
10. Optionally, configure a related content theme by creating a StyleSheet (.sty) file called theme. You can copy this from an existing theme folder, such as the Light folder, as well.

    The styling properties in the StyleSheet will be applied automatically to content on the page when your custom theme is used, unless that content has a different theme explicitly assigned.

11. In Db2 Web Query Designer, apply the new custom theme to a page, as described in *How to Apply a Theme to a Page* on page 88.

    An example of a new theme applied to a page is shown in the following image.



    Notice that the content theme, or StyleSheet, associated with the page theme was automatically applied to the content on the page.

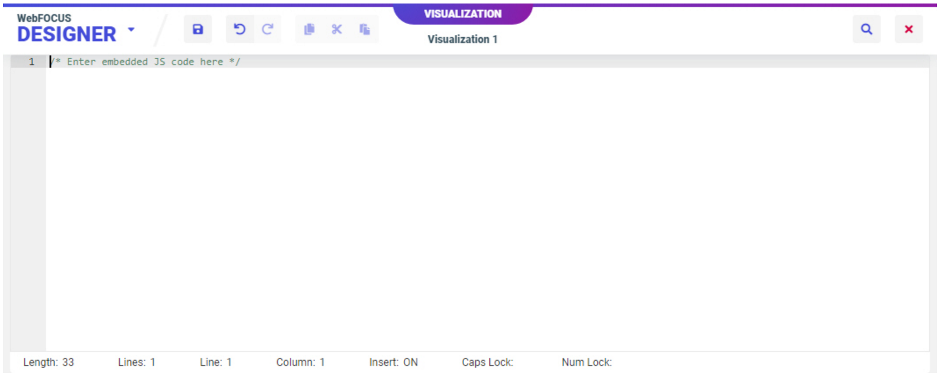
12. Save your changes.

Applying Custom CSS and JavaScript Code to a Page

You can apply custom cascading style sheet (CSS) properties and JavaScript code to a page, allowing you to significantly enhance the page with countless styling options and run-time behaviors.

This feature is intended for content developers with CSS and JavaScript coding experience. The CSS and JavaScript options are available to developers, administrators, and users with text editor access.

You can add custom CSS and JavaScript to a page by typing code into the CSS and Javascript text editors, respectively. To access the CSS or JavaScript text editor, click *Outline* on the sidebar, and then select *CSS* or *JavaScript*. The text editor opens over the canvas, and you can type your code into it, as shown in the following image.



```
/* Enter embedded JS code here */
```

Length: 33 Lines: 1 Line: 1 Column: 1 Insert: ON Caps Lock: Num Lock:

To return to the canvas, click the close icon in the corner of the page.

When you use the text editor, the Designer toolbar provides additional options to copy, cut, paste, and find and replace your text. The text editor also includes built-in features such as syntax highlighting, code folding, line numbering, autocomplete, alerts, indent guides, and a status bar. These features facilitate CSS and JavaScript code development and debugging by improving clarity and navigability in the text editor.

Note: Custom CSS and JavaScript is not displayed in the canvas. You must run the visualization to see the results of your custom code.

Applying Custom CSS to a Page

Custom CSS can be applied to specific objects on a page by specifying a CSS class for the object, and then using a class selector in your CSS code.

To specify a CSS class for an object on a page, select the object and, on the Properties panel, type a class name into the text box for the *Classes* property. You can specify multiple classes for an object by separating the class names with spaces, and you are encouraged to use the same class names for multiple items on the page. When you assign attributes to a class in the CSS tab, they will affect all elements on the page that are assigned to that class.

Procedure: How to Change the Color of a Panel Using CSS

You can change the background color of a panel by assigning a class to the panel and then adding the background-color attribute to the class using CSS.

1. Open Db2 Web Query Designer. On the default Db2 Web Query Home Page, click *Visualize Data*.

   Db2 Web Query Designer opens in a new browser tab.

2. Select a workspace and a data source available from that workspace.

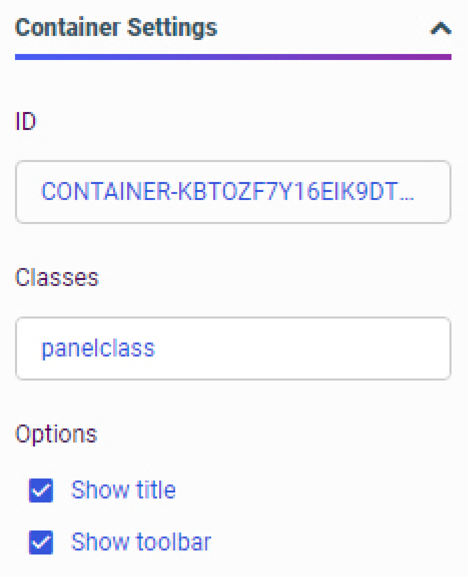
   Once you select a data source, Db2 Web Query Designer loads with options to create a single content item.

3. Click the *Add visualization* button to transform the visualization from a single content item to a page.

4. Optionally, create new content in the visualization and add new containers to the page for additional content.

5. Select a container on the page.

6. On the Settings tab of the Properties panel, in the text box for the *Classes* property, type a class name of your choice, as shown in the following image.

   Container Settings

   ID

   CONTAINER-KBTOZF7Y16EIK9DT...

   Classes

   panelclass

   Options

   ☑ Show title

   ☑ Show toolbar

7. Click *Outline* on the sidebar to display the visualization outline.

8. Click *CSS*.

   The CSS text editor opens.

9. Add a CSS declaration to change the background-color attribute for the class used by the panel.

    a. Reference the class name that you specified in the Classes property.

        Type a period (.) followed immediately by the class name, then an opening curly bracket ({), and press the Enter key.

        The closing bracket is added automatically.

    b. Within the brackets, type *background-color*, followed by a colon and then a color string.

        For example, to make the color of a panel, for which panelclass is the value of the Classes property, to a bright blue, you could use the following CSS declaration.

        ```
        .panelclass {
            background-color: rgb(0,180,240);
        }
        ```

10. Save the visualization and close Db2 Web Query Designer.

11. Run the visualization from the Db2 Web Query Home Page.

    The custom CSS is applied to the specified container.

Applying Custom JavaScript to a Page

You can use custom JavaScript to enhance the functionality of components on a page. Any component created on the page using the Designer canvas can be modified in the JavaScript text editor by referencing a CSS class name specified in the Classes field of the Properties panel for the object, or using the Designer JavaScript API, to standardized classes that correspond to different component types in a page. Each component class can utilize a set of predefined methods in addition to custom JavaScript. You can even use JavaScript to add objects to a page entirely through code, without using the canvas at all.

Procedure: How to Add a Button and a Menu Item to a Panel Container Using Custom JavaScript

You can use custom JavaScript code to add a button and a menu item to a panel that you have added to a page. You can configure the button and menu item to execute a command of your choice, such as a URL call.

For information about different JavaScript object classes, see *Designer JavaScript API Classes* on page 102.

1. Open Db2 Web Query Designer. On the default Db2 Web Query Home Page, click *Visualize Data*.

    Db2 Web Query Designer opens in a new browser tab.

2. Select a workspace and a data source available from that workspace.

   Once you select a data source, Db2 Web Query Designer loads with options to create a single content item.

3. Click the *Add visualization* button to transform the visualization from a single content item to a page.

4. Optionally, create new content in the visualization and add new containers to the page for additional content.

5. Select a container on the page.

6. On the Settings tab of the Properties panel, in the text box for the *Classes* property, type a class name of your choice.

7. Click *Outline* on the sidebar to display the visualization outline.

8. Click *JavaScript*.

   The JavaScript text editor opens.

9. To have the button and menu item appear when the page loads, use the iba_pageloading event listener by adding the following syntax:

   ```
   window.addEventListener("iba_pageloading", function (e){
   });
   ```

   The code to add a button and menu item will go within this event listener command.

10. Add a button to the panel using JavaScript.

    a. First, within the iba_pageloading event listener, create a variable to represent the panel, such as the following:

       ```
       var panel = document.querySelector(".class").ibaObject;
       ```

       where:

       panel

       Is the name of the variable that you define to represent the panel.

       class

       Is the name you previously typed in the Classes property for the panel on the page.

    b. After the panel variable, define the style and appearance of the button and add it to the panel, such as in the following syntax example. This example uses a home icon and places the button before the default resize button:

```
var button = panel.addButton(
    {"glyphClasses": "fa fa-home", "class": "buttonClass",
        "tooltip": "tooltip text"},
    ".pd-container-title-button-resize", true);
```

where:

button

Is the name of the variable that you define to represent the button.

panel

Is the variable representing the panel, defined in step 10a.

fa fa-home

Is a glyph class for a Font Awesome icon that looks like a house.

buttonClass

Is a CSS class assigned to the button itself. You can use this to apply CSS styling to the button, or reference the button elsewhere in your JavaScript code.

tooltip text

Is the text that you want to appear in the tooltip when you point to the button.

.pd-container-title-button-resize

Is the class of the resize button on the panel. This is the sibling of the button you are currently adding.

true

When the value of the before property is set to *true*, the button is placed before the sibling. Otherwise, it is placed after the sibling. If no sibling is specified, it is ordered last.

c. After defining the panel button, create an event listener that allows the button to execute a specified action. The following example causes the button to open the Information Builders website in a new window when it is clicked.

```
button.addEventListener("click", function(){
    window.open("http://www.informationbuilders.com");
    });
```

where:

button

Is the variable name that you assigned to the button in step 10b.

11. Add a menu item to the panel's run-time menu using JavaScript.

    a. Create a variable to define the menu item, such as in the following example.

    ```
    var menu = panel.addMenuItem({
        "text": "Menu text", "glyphClasses": "fa fa-globe","class":
    "menu-class"},
        ".class>.ibx_menu_item", true);
    ```

    where:

    menu

    Is a variable name that you want to use to represent the menu item.

    panel

    Is the name of the variable representing the panel to which the menu is added.

    Menu text

    Is the text of the menu item.

    `fa fa-globe`

    Is a glyph class for a Font Awesome icon that looks like the Earth.

    menu-class

    Is a CSS class assigned to the menu item itself. You can use this to apply CSS styling to the menu item, or reference the menu item elsewhere in your JavaScript code.

    `".`*class*`>.ibx_menu_item"`

    Is a selector for the Refresh button in the panel menu, where *class* is the CSS class assigned to the panel using the Classes property.

    `true`

    When the value of the before property is set to *true*, the menu item is placed before the sibling, which in this case is the Refresh option. Otherwise, it is placed after the sibling. If no sibling is specified, it is ordered last in the menu.

    b. After defining the menu item, create an event listener that allows the button to execute a specified action. The following example causes the menu item to run an HTML page in a new window through a URL call when it is clicked.

```
menu.addEventListener("click", function(){
    window.open(
       "http://localhost:12331/webquery/run.bip?
BIP_REQUEST_TYPE=BIP_LAUNCH&BIP_folder=IBFS%253A%252FWFC%252FRepository
%252FPublic%252F2019%252FHTML%252F&BIP_item=html_page.htm"
    );
});
```

where:

menu

Is the variable name that you assigned to the menu item in step 11a.

The complete custom JavaScript may resemble the following.

```
window.addEventListener("iba_pageloading", function (e){
    var panel = document.querySelector(".map-panel").ibaObject;
    var ibsite = panel.addButton({
            "glyphClasses": "fa fa-home", "class": "ibButton", "tooltip": "Click
to display help."},
        ".pd-container-title-button-resize", true);
    ibsite.addEventListener("click", function(){
        window.open("http://www.informationbuilders.com");});
    var runReport = panel.addMenuItem({
            "text": "Country Report", "glyphClasses": "fa fa-globe","class":
"globemenu"},
        ".map-panel>.ibx_menu_item",  true);
    runReport.addEventListener("click", function(){
        window.open(
            "http://localhost:12331/webquery/run.bip?
BIP_REQUEST_TYPE=BIP_LAUNCH&BIP_folder=IBFS%253A%252FWFC%252FRepository%252FPublic
%252F2019%252FHTML%252F&BIP_item=Basic_2_ctrl_page.htm");
    });
});
```

12. Save the page.

    Changes made using custom CSS and JavaScript do not appear on the canvas at design time. To see the impact of your custom code, you must run the visualization.

13. On the Db2 Web Query Home Page, right-click the visualization that you just created and click *Run in new window*.

    The visualization opens in a new browser tab or window, and the customized button and menu item are available. When clicked, they open the links you specified for them in a new tab or window.

Procedure: How to Use JavaScript to Automatically Resize an Image on a Page

Since the panels in a responsive page resize to fit the browser window, images added to the page may not always fit properly. If the image is too big, only part of the image may show in the panel, and scrollbars may be added. If the image is too small, the panel background may show around the image.

In order to autosize an image to fit a panel, you can use jQuery to set the height and width CSS properties for an image in a specified container. You can specify the container by referencing its class, which you can set using the Classes property on the Settings tab.

An example of the final JavaScript code is shown below:

```
$('.class').closest('.grid-stack-item').addClass('autoheight');
$('.class iframe').attr('scrolling','no').on('load', function(e){
    $(this).contents().find('img').height('100%');
    $(this).contents().find('img').width('100%');
})
```

where:

`class`

Is a class assigned to the panel that contains the image to be autosized.

`.attr('scrolling','no')`

Disables the generation of scrollbars in the container.

`.height('100%'), .width('100%')`

Sets the image to fill the height and width of the container. This may distort the image, depending on the dimensions of the container. As an alternative, replace *100%* with *auto* for either the height or width, to maintain the height-to-width ratio at all sizes. Using *auto* may reveal the panel background or cut off part of the image, depending on the dimensions of the container.

1. On the Db2 Web Query Home Page, click the plus button and then click *Assemble Visualizations*.

   Db2 Web Query Designer opens to assemble a page from existing content.

2. With *Content* selected on the sidebar, navigate to the location of an image that has been uploaded to the Repository and drag the image onto the canvas.

   The image is added to a container on the page. The size of the image in the container is the absolute size of the uploaded image file.

3. Optionally, resize and reposition the container that contains the image, and add other items to the page.

4. Select the container that holds the image. On the *Settings* tab of the Properties panel, type a class name of your choice into the Classes text box. This class will be used as an identifier for the container in the JavaScript code.

5. Click *Outline* on the sidebar, and then click *JavaScript*.

   The JavaScript text editor opens.

6. Paste the following code into the text editor.

   ```
   $('.class').closest('.grid-stack-item').addClass('autoheight');
   $('.class iframe').attr('scrolling','no').on('load', function(e){
       $(this).contents().find('img').height('100%');
       $(this).contents().find('img').width('100%');
   })
   ```

7. Replace *class* in the first two lines with the class name that you specified for the container in Step 4.
8. Click the red X in the top-right corner of the page to return to the page canvas.

   The image still displays at its original size. The height and width properties are only applied when the page is run.
9. Optionally, on the canvas, select the container that contains the image and, on the *Settings* tab on the Properties panel, clear the *Show title* and *Show toolbar* check boxes to hide the container title and toolbar, so that the image is the only item in the container.
10. On the Visualization toolbar, click *Run in new window* to run the page and execute the JavaScript code that you just added.

    The image stretches to fill the entire container at run time.

Designer JavaScript API Classes

As part of the Designer JavaScript API, you can use the following predefined classes.

❑ **ibaObject.** The base automation object.

❑ **ibaAccordionContainer.** Accordion container automation object.

❑ **ibaCarouselContainer.** Carousel container automation object.

❑ **ibaContent.** Content panel automation object.

❑ **ibaPage.** Page automation object. The event iba_pageloading will be triggered on the global window object. *event*.data will be the ibaPage object.

❑ **ibaPanelContainer.** Panel container automation object.

❑ **ibaPortal.** Portal automation object.

❑ **ibaSection.** Page section automation object.

❑ **ibaTabContainer.** Tab container automation object.

A help plug-in with information about each class is packaged with your Db2 Web Query installation. It is available through a URL at the following address:

```
http[s]://hostname:port/context/web_resource/doc/automation/index.html
```

where:

hostname

Is the name of the machine on which the Db2 Web Query is installed.

port

Is the port number on which the server is listening.

context

Is the context root of the Db2 Web Query application. For example, webquery.

Class: ibaObject

```
new ibaObject(element)
```

Is the base automation object. Automation objects are created internally by the system. You never directly instantiate an automation object.

where:

element

Is the node that is being automated.

Available methods include the following:

❏ **classId()**

Returns a string, which is the unique, automatically generated class ID of the object being automated. It can be used to filter enumerations.

❏ **customClasses(*classes*, *remove*)**

Returns a string. Sets or gets custom classes.

where:

classes

String.

Optional. A space-separated list of classes to be added or removed.

remove

Boolean.

Optional. If true, remove the classes specified in *classes*, otherwise add them.

❏ **element()**

Returns an element, the DOM node that is being automated by this object.

Class: ibaPage

```
new ibaPage(element)
```

Is the page automation object. Automation objects are created internally by the system. You never directly instantiate an automation object. The event iba_pageloading will be triggered on the global window object. event.data will be the ibaPage object.

where:

element

Is the node that is being automated.

The following triggers can be used with the page object:

- ❑ **event:iba_pageloaded.** The page is loaded in the running state.
- ❑ **event:iba_beforecontentdescribe.** Before the page content filtering information has been retrieved.
- ❑ **event:iba_contentdescribed.** After the page content filtering information has been retrieved, but before it has been processed.
- ❑ **event:iba_beforecontentload.** Before the content of a panel is about to load.

These are used in the following JavaScript syntax example:

```
window.addEventListener("iba_pageloading", function (e){
    var page = e.data;
    page.element().addEventListener("iba_pageloaded", function(e){
        var page = e.data;
    });
    page.element().addEventListener("iba_beforecontentdescribe", function (e){
        var describeInfo = e.data;
        var path = describeInfo.path; // The path to the content being described
        // e.preventDefault() will stop the describe from happening
    });
    page.element().addEventListener("iba_contentdescribed", function (e){
        var describeInfo = e.data;
        var path = describeInfo.path; // The path to the content being described
        var wfDescribe = describeInfo.wfDescribe; // The filtering information of the
content being described
    });
    page.element().addEventListener("iba_beforecontentload", function (e){
        var loadInfo = e.data;
        var path = loadInfo.path; // The path to the content being described
        var defaultValues = loadInfo.defaultValues; // If default values are used to
run the content
        // e.preventDefault() will stop the page from retrieving the content
    });
});
```

If you add multiple pages with custom JavaScript to the same Web Query portal, the iba_pageloading event listener, for the global window object, will execute whenever any page in the Web Query portal loads. To execute your code only when a specific page loads, remove the event listener after the code executes for the desired page using the window.removeEventListener method.

The iba_pageloading event listener can be used to access the page being loaded, since it receives a page object as a parameter. The page object can be stored or used to assign additional event handlers to the page as shown below:

```
function onPageLoading(e) {
    var page = e.data;
    page.element().addEventListener("iba_pageloaded", function(e){
        on load code
    });
   page.element().addEventListener("click", function(e)
   {
       on click code
   });
  window.removeEventListener("iba_pageloading", onPageLoading);
}
```

where:

`function onPageLoading`

Is an iba_pageloading event listener function.

`var page = e.data`

Accesses the page object via the event parameter.

`on load code`

Code to be executed when the page loads.

`on click code`

Code to be executed when the page is clicked.

`window.removeEventListener("iba_pageloading", onPageLoading);`

Removes the iba_pageloading listener.

You can use the *context* member object to supply the context information for the page.

Available methods include the following:

❑ **addButton(*options*, *sibling*, *before*)**

Returns an element, adding a button to the title bar of the page.

where:

options

Object.

One or more button options, which can include the following:

- **icon.** A URL to an image file in .jpeg, .png, or .gif format.
- **glyph.** A glyph or ligature, such as a Material icon.
- **glyphClasses.** Glyph classes. For example, 'material-icons'.
- **class.** Additional CSS classes.
- **tooltip.** A tooltip for the button.

sibling

Selector, element, or jQuery.

Optional. A sibling to add before or after. The default is Last.

before

Boolean.

Add button before or after the specified sibling.

- **addSection(*options*, *sibling*, *before*)**

  Returns an element, adding a section to the page.

  where:

  options

  Object.

  One or more section options, which can include the following:

  - **collapsible.** A Boolean value. When true, the section is collapsible.
  - **collapsed.** A Boolean value. When true, the section is created in a collapsed state.

  sibling

  Selector, element, or jQuery.

  Optional. A sibling to add before or after. The default is Last.

  before

  Boolean.

  Add the section before or after the specified sibling.

- **buttons(*selector*)**

  Returns the page title bar buttons as an array.

  where:

  selector

  Selector, element, or jQuery.

  Is the buttons selector. All is the default.

- **classId()**

  Returns a string, which is the unique, automatically generated class ID of the object being automated. It can be used to filter enumerations.

- **containers(*selector*)**

  Returns the page containers as an array.

  where:

  selector

  Selector, element, or jQuery.

  Is the containers selector. All is the default.

- **customClasses(*classes*, *remove*)**

  Returns a string. Sets or gets custom classes.

  where:

  classes

  String.

  Optional. A space-separated list of classes to be added or removed.

  remove

  Boolean.

  Optional. If true, remove the classes specified in *classes*, otherwise add them.

- **element()**

  Returns an element, the DOM node that is being automated by this object.

- **refreshFilters()**

  Use to redescribe the existing content and recreate filter panels.

- **removeButton(*button*)**

  Removes one or more buttons from the title bar of the page.

  where:

  `button`
  Selector, element, or jQuery.
  Optional. Is a button selector. All is the default.

- **removeSection(*selector*)**

  Removes one or more sections from the page.

  where:

  `selector`
  Selector, element, or jQuery.
  Optional. Is a section selector. All is the default.

- **sections(*selector*)**

  Returns the page sections as an array.

  where:

  `selector`
  Selector, element, or jQuery.

  Is the section selector. All is the default.

- **title(*title*)**

  Sets the title of the page. If *title* is not passed, the title of the container is returned as a string or jQuery.

  where:

  `title`
  String.
  Optional. Is the new title of the page.

Class: ibaSection

The ibaSection class extends ibaObject.

```
new ibaSection(element)
```

Is the page section automation object. Automation objects are created internally by the system. You never directly instantiate an automation object.

where:

`element`

Is the DOM node that is being automated.

Available methods include the following:

- **addContainer(*type*, *title*, *col*, *row*, *colSpan*, *rowSpan*)**

  Adds a new container to the section.

  where:

  `type`

  String.

  Is one of the following container types:

  - 'pane'
  - 'tab'
  - 'accordion'
  - 'carousel'

  `title`

  String.

  Is the container title.

  `col`

  Integer.

  Optional. The column of the section to insert to. By default, the container is added to the first available column.

  `row`

  Integer.

  Optional. The row of the section to insert to. By default, the container is added to the first available row.

  `colSpan`

  Integer.

  Optional. The width of the container in columns. 3 is the default.

  `rowSpan`

  Integer.

Optional. The height of the container in rows. 5 is the default.

- **classId()**

  Returns a string, which is the unique, automatically generated class ID of the object being automated. It can be used to filter enumerations.

- **collapsed(*collapsed*)**

  Makes the section collapsed, or returns the current collapsed state or a chainable automation object.

  where:

  `collapsed`

  Boolean.

  Optional. If true, collapse the section. Otherwise, expand.

- **collapsible(*collapsible*)**

  Makes the section collapsible, or returns the current collapsible state or a chainable automation object.

  where:

  `collapsible`

  Boolean.

  Optional. If true, make the section collapsible.

- **containers(*selector*)**

  Returns the containers in the section as an array.

  where:

  `selector`

  Selector, element, or jQuery.

  An optional containers selector. All is the default.

- **customClasses(*classes*, *remove*)**

  Returns a string. Sets or gets custom classes.

  where:

  `classes`

  String.

  Optional. A space-separated list of classes to be added or removed.

remove

Boolean.

Optional. If true, remove the classes specified in *classes*, otherwise add them.

❑ **element()**

Returns an element, the DOM node that is being automated by this object.

Class: ibaContainer

```
new ibaContainer(element)
```

Is the container automation object. Automation objects are created internally by the system. You never directly instantiate an automation object.

where:

element

Is the DOM node that is being automated.

Available methods include the following:

❑ **addButton(*options*, *sibling*, *before*)**

Returns an element, adding a button to the title bar of the container.

where:

options

Object.

One or more button options, which can include the following:

❑ **icon.** A URL to an image file in .jpeg, .png, or .gif format.

❑ **glyph.** A glyph or ligature, such as a Material icon.

❑ **glyphClasses.** Glyph classes. For example, 'material-icons'.

❑ **class.** Additional CSS classes.

❑ **tooltip.** A tooltip for the button.

sibling

Selector, element, or jQuery.

Optional. A sibling to add before or after. The default is Last.

before

Boolean.

Add button before or after the specified sibling.

- **addContent(*path*, *description*, *replaceExisting*)**

  Returns elements. Adds new content by replacing the content in the current content panel or adding a sub-panel.

  where:

  path

  String.

  Path of the content being added.

  description

  String.

  Description of the content being added.

  replaceExisting

  Boolean.

  Optional. When true, replace the existing content. When false, which is the default, add new content.

- **addMenuItem(*options*, *sibling*, *before*)**

  Returns an element, adding a menu item to the container menu.

  where:

  options

  Object.

  One or more menu item options, which can include the following:

  - **text.** The menu item text.
  - **icon.** A URL to an image file in .jpeg, .png, or .gif format.
  - **glyph.** A glyph or ligature, such as a Material icon.
  - **glyphClasses.** Glyph classes. For example, 'material-icons'.
  - **class.** Additional CSS classes.

  sibling

  Selector, element, or jQuery.

Optional. A sibling to add before or after. The default is Last.

before

Boolean.

Add the menu item before or after the specified sibling.

- ❑ **addMenuSeparator(*options*, *sibling*, *before*)**

  Returns an element, adding a separator to the container menu.

  where:

  options

  Object.

  Menu separator options. You can use the class property to specify additional CSS classes.

  sibling

  Selector, element, or jQuery.

  Optional. A sibling to add before or after. The default is Last.

  before

  Boolean.

  Add menu separator before or after the specified sibling.

- ❑ **buttons(*selector*)**

  Returns the container title bar buttons as an array.

  where:

  selector

  Selector, element, or jQuery.

  Is the buttons selector. All is the default.

- ❑ **classId()**

  Returns a string, which is the unique, automatically generated class ID of the object being automated. It can be used to filter enumerations.

- ❑ **contents(*selector*)**

  Returns the content panels in the container as an array.

where:

selector

Selector, element, or jQuery.

Is the content selector. All is the default.

- **customClasses(*classes*, *remove*)**

  Returns a string. Sets or gets custom classes.

  where:

  classes

  String.

  Optional. A space-separated list of classes to be added or removed.

  remove

  Boolean.

  Optional. If true, remove the classes specified in *classes*, otherwise add them.

- **element()**

  Returns an element, the DOM node that is being automated by this object.

- **removeButton(*button*)**

  Removes one or more buttons from the title bar of the container.

  where:

  button

  Selector, element, or jQuery.
  Optional. Is a button selector. All is the default.

- **removeContent(*selector*)**

  Removes content from the container.

  where:

  selector

  Selector, element, or jQuery.
  Optional. Is a content selector. All is the default.

- **removeMenuItem(*selector*)**

  Removes one or more menu items from the container menu.

where:

selector

Selector, element, or jQuery.
Optional. Is a menu item selector. All is the default.

- ❏ **removeMenuSeparator(*selector*)**

  Removes one or more separators from the container.

  where:

  selector

  Selector, element, or jQuery.
  Optional. Is a menu separator selector. All is the default.

- ❏ **title(*title*)**

  Sets the title of the container. If *title* is not passed, the title of the container is returned as a string or jQuery.

  where:

  title

  String.
  Optional. Is the new title of the container.

Class: ibaAccordionContainer

The ibaAccordionContainer class extends ibaContainer.

```
new ibaAccordionContainer(element)
```

Is the accordion container automation object. Automation objects are created internally by the system. You never directly instantiate an automation object.

where:

element

Is the DOM node that is being automated.

Available methods include the following:

- ❏ **addButton(*options*, *sibling*, *before*)**

  Returns an element, adding a button to the title bar of the accordion container.

where:

options

Object.

One or more button options, which can include the following:

- **icon.** A URL to an image file in .jpeg, .png, or .gif format.
- **glyph.** A glyph or ligature, such as a Material icon.
- **glyphClasses.** Glyph classes. For example, 'material-icons'.
- **class.** Additional CSS classes.
- **tooltip.** A tooltip for the button.

sibling

Selector, element, or jQuery.

Optional. A sibling to add before or after. The default is Last.

before

Boolean.

Add button before or after the specified sibling.

- **addContent(*path*, *description*, *replaceExisting*)**

  Returns elements. Adds new content by replacing the content in the current content panel or adding a new accordion tab.

  where:

  path

  String.

  Path of the content being added.

  description

  String.

  Description of the content being added.

  replaceExisting

  Boolean.

  Optional. When true, replace the existing content. When false, which is the default, add new content.

- **addMenuItem(*options*, *sibling*, *before*)**

  Returns an element, adding a menu item to the accordion container menu.

  where:

  options

  Object.

  One or more menu item options, which can include the following:

  - **text.** The menu item text.
  - **icon.** A URL to an image file in .jpeg, .png, or .gif format.
  - **glyph.** A glyph or ligature, such as a Material icon.
  - **glyphClasses.** Glyph classes. For example, 'material-icons'.
  - **class.** Additional CSS classes.

  sibling

  Selector, element, or jQuery.

  Optional. A sibling to add before or after. The default is Last.

  before

  Boolean.

  Add the menu item before or after the specified sibling.

- **addMenuSeparator(*options*, *sibling*, *before*)**

  Returns an element, adding a separator to the accordion container menu.

  where:

  options

  Object.

  Menu separator options. You can use the class property to specify additional CSS classes.

  sibling

  Selector, element, or jQuery.

  Optional. A sibling to add before or after. The default is Last.

  before

  Boolean.

Add menu separator before or after the specified sibling.

- ❑ **buttons(*selector*)**

  Returns the container title bar buttons as an array.

  where:

  selector

  Selector, element, or jQuery.

  Is the buttons selector. All is the default.

- ❑ **classId()**

  Returns a string, which is the unique, automatically generated class ID of the object being automated. It can be used to filter enumerations.

- ❑ **contents(*selector*)**

  Returns the content panels in the container as an array.

  where:

  selector

  Selector, element, or jQuery.

  Is the content selector. All is the default.

- ❑ **customClasses(*classes*, *remove*)**

  Returns a string. Sets or gets custom classes.

  where:

  classes

  String.

  Optional. A space-separated list of classes to be added or removed.

  remove

  Boolean.

  Optional. If true, remove the classes specified in *classes*, otherwise add them.

- ❑ **element()**

  Returns an element, the DOM node that is being automated by this object.

- ❑ **removeButton(*button*)**

  Removes one or more buttons from the title bar of the container.

where:

`button`

Selector, element, or jQuery.
Optional. Is a button selector. All is the default.

- ❑ **removeContent(*selector*)**

  Removes content from the container.

  where:

  `selector`

  Selector, element, or jQuery.
  Optional. Is a content selector. All is the default.

- ❑ **removeMenuItem(*selector*)**

  Removes one or more menu items from the container menu.

  where:

  `selector`

  Selector, element, or jQuery.
  Optional. Is a menu item selector. All is the default.

- ❑ **removeMenuSeparator(*selector*)**

  Removes one or more separators from the container.

  where:

  `selector`

  Selector, element, or jQuery.
  Optional. Is a menu separator selector. All is the default.

- ❑ **selectContent(*selector*)**

  Returns an element. Selects a content pane in the accordion container.

  where:

  `selector`

  Selector, element, or jQuery.
  Optional. Is a content selector. All is the default.

- ❑ **title(*title*)**

  Sets the title of the container. If *title* is not passed, the title of the container is returned as a string or jQuery.

where:

title

String.

Optional. Is the new title of the container.

Class: ibaCarouselContainer

The ibaCarouselContainer class extends ibaContainer.

`new ibaCarouselContainer(`*`element`*`)`

Is the carousel container automation object. Automation objects are created internally by the system. You never directly instantiate an automation object.

where:

element

Is the DOM node that is being automated.

Available methods include the following:

- ❑ **addButton(*options*, *sibling*, *before*)**

  Returns an element, adding a button to the title bar of the carousel container.

  where:

  options

  Object.

  One or more button options, which can include the following:

  - ❑ **icon.** A URL to an image file in .jpeg, .png, or .gif format.
  - ❑ **glyph.** A glyph or ligature, such as a Material icon.
  - ❑ **glyphClasses.** Glyph classes. For example, 'material-icons'.
  - ❑ **class.** Additional CSS classes.
  - ❑ **tooltip.** A tooltip for the button.

  sibling

  Selector, element, or jQuery.

  Optional. A sibling to add before or after. The default is Last.

  before

  Boolean.

Add button before or after the specified sibling.

- **addContent(*path*, *description*, *replaceExisting*)**

  Returns elements. Adds new content by replacing the content in the current content panel or adding a new carousel slide.

  where:

  path

  String.

  Path of the content being added.

  description

  String.

  Description of the content being added.

  replaceExisting

  Boolean.

  Optional. When true, replace the existing content. When false, which is the default, add new content.

- **addMenuItem(*options*, *sibling*, *before*)**

  Returns an element, adding a menu item to the carousel container menu.

  where:

  options

  Object.

  One or more menu item options, which can include the following:

  - **text.** The menu item text.
  - **icon.** A URL to an image file in .jpeg, .png, or .gif format.
  - **glyph.** A glyph or ligature, such as a Material icon.
  - **glyphClasses.** Glyph classes. For example, 'material-icons'.
  - **class.** Additional CSS classes.

  sibling

  Selector, element, or jQuery.

  Optional. A sibling to add before or after. The default is Last.

before

Boolean.

Add the menu item before or after the specified sibling.

- **addMenuSeparator(*options*, *sibling*, *before*)**

  Returns an element, adding a separator to the carousel container menu.

  where:

  options

  Object.

  Menu separator options. You can use the class property to specify additional CSS classes.

  sibling

  Selector, element, or jQuery.

  Optional. A sibling to add before or after. The default is Last.

  before

  Boolean.

  Add menu separator before or after the specified sibling.

- **buttons(*selector*)**

  Returns the container title bar buttons as an array.

  where:

  selector

  Selector, element, or jQuery.

  Is the buttons selector. All is the default.

- **classId()**

  Returns a string, which is the unique, automatically generated class ID of the object being automated. It can be used to filter enumerations.

- **contents(*selector*)**

  Returns the content panels in the container as an array.

  where:

  selector

  Selector, element, or jQuery.

Is the content selector. All is the default.

- ❑ **customClasses(*classes*, *remove*)**

  Returns a string. Sets or gets custom classes.

  where:

  classes

  String.

  Optional. A space-separated list of classes to be added or removed.

  remove

  Boolean.

  Optional. If true, remove the classes specified in *classes*, otherwise add them.

- ❑ **element()**

  Returns an element, the DOM node that is being automated by this object.

- ❑ **removeButton(*button*)**

  Removes one or more buttons from the title bar of the container.

  where:

  button

  Selector, element, or jQuery.
  Optional. Is a button selector. All is the default.

- ❑ **removeContent(*selector*)**

  Removes content from the container.

  where:

  selector

  Selector, element, or jQuery.
  Optional. Is a content selector. All is the default.

- ❑ **removeMenuItem(*selector*)**

  Removes one or more menu items from the container menu.

  where:

  selector

  Selector, element, or jQuery.
  Optional. Is a menu item selector. All is the default.

- **removeMenuSeparator(*selector*)**

  Removes one or more separators from the container.

  where:

  `selector`

  Selector, element, or jQuery.
  Optional. Is a menu separator selector. All is the default.

- **selectContent(*selector*)**

  Returns an element. Selects a content pane in the carousel container.

  where:

  `selector`

  Selector, element, or jQuery.
  Optional. Is a content selector. All is the default.

- **title(*title*)**

  Sets the title of the container. If *title* is not passed, the title of the container is returned as a string or jQuery.

  where:

  `title`

  String.
  Optional. Is the new title of the container.

Class: ibaPanelContainer

The ibaPanelContainer class extends ibaContainer.

```
new ibaPanelContainer(element)
```

Is the panel container automation object. Automation objects are created internally by the system. You never directly instantiate an automation object.

where:

`element`

Is the DOM node that is being automated.

Available methods include the following:

- **addButton(*options*, *sibling*, *before*)**

  Returns an element, adding a button to the title bar of the panel container.

where:

options

Object.

One or more button options, which can include the following:

- **icon.** A URL to an image file in .jpeg, .png, or .gif format.
- **glyph.** A glyph or ligature, such as a Material icon.
- **glyphClasses.** Glyph classes. For example, 'material-icons'.
- **class.** Additional CSS classes.
- **tooltip.** A tooltip for the button.

sibling

Selector, element, or jQuery.

Optional. A sibling to add before or after. The default is Last.

before

Boolean.

Add button before or after the specified sibling.

- **addContent(*path*, *description*, *replaceExisting*)**

  Returns elements. Adds new content by replacing the content in the current content panel or adding a new panel.

  where:

  path

  String.

  Path of the content being added.

  description

  String.

  Description of the content being added.

  replaceExisting

  Boolean.

  Optional. When true, replace the existing content. When false, which is the default, add new content.

- **addMenuItem(*options*, *sibling*, *before*)**

  Returns an element, adding a menu item to the container menu.

  where:

  options

  Object.

  One or more menu item options, which can include the following:

  - **text.** The menu item text.
  - **icon.** A URL to an image file in .jpeg, .png, or .gif format.
  - **glyph.** A glyph or ligature, such as a Material icon.
  - **glyphClasses.** Glyph classes. For example, 'material-icons'.
  - **class.** Additional CSS classes.

  sibling

  Selector, element, or jQuery.

  Optional. A sibling to add before or after. The default is Last.

  before

  Boolean.

  Add the menu item before or after the specified sibling.

- **addMenuSeparator(*options*, *sibling*, *before*)**

  Returns an element, adding a separator to the container menu.

  where:

  options

  Object.

  Menu separator options. You can use the class property to specify additional CSS classes.

  sibling

  Selector, element, or jQuery.

  Optional. A sibling to add before or after. The default is Last.

  before

  Boolean.

Add menu separator before or after the specified sibling.

❑ **buttons(*selector*)**

Returns the container title bar buttons as an array.

where:

selector

Selector, element, or jQuery.

Is the buttons selector. All is the default.

❑ **classId()**

Returns a string, which is the unique, automatically generated class ID of the object being automated. It can be used to filter enumerations.

❑ **contents(*selector*)**

Returns the content panels in the container as an array.

where:

selector

Selector, element, or jQuery.

Is the content selector. All is the default.

❑ **customClasses(*classes*, *remove*)**

Returns a string. Sets or gets custom classes.

where:

classes

String.

Optional. A space-separated list of classes to be added or removed.

remove

Boolean.

Optional. If true, remove the classes specified in *classes*, otherwise add them.

❑ **element()**

Returns an element, the DOM node that is being automated by this object.

❑ **removeButton(*button*)**

Removes one or more buttons from the title bar of the container.

where:

button

Selector, element, or jQuery.

Optional. Is a button selector. All is the default.

- ❑ **removeContent(*selector*)**

  Removes content from the container.

  where:

  selector

  Selector, element, or jQuery.

  Optional. Is a content selector. All is the default.

- ❑ **removeMenuItem(*selector*)**

  Removes one or more menu items from the container menu.

  where:

  selector

  Selector, element, or jQuery.

  Optional. Is a menu item selector. All is the default.

- ❑ **removeMenuSeparator(*selector*)**

  Removes one or more separators from the container.

  where:

  selector

  Selector, element, or jQuery.

  Optional. Is a menu separator selector. All is the default.

- ❑ **title(*title*)**

  Sets the title of the container. If *title* is not passed, the title of the container is returned as a string or jQuery.

  where:

  title

  String.

  Optional. Is the new title of the container.

Class: ibaTabContainer

The ibaTabContainer class extends ibaContainer.

`new ibaTabContainer(element)`

Is the carousel container automation object. Automation objects are created internally by the system. You never directly instantiate an automation object.

where:

`element`

Is the DOM node that is being automated.

Available methods include the following:

❑ **addButton(*options*, *sibling*, *before*)**

Returns an element, adding a button to the title bar of the tab container.

where:

`options`

Object.

One or more button options, which can include the following:

❑ **icon.** A URL to an image file in .jpeg, .png, or .gif format.

❑ **glyph.** A glyph or ligature, such as a Material icon.

❑ **glyphClasses.** Glyph classes. For example, 'material-icons'.

❑ **class.** Additional CSS classes.

❑ **tooltip.** A tooltip for the button.

`sibling`

Selector, element, or jQuery.

Optional. A sibling to add before or after. The default is Last.

`before`

Boolean.

Add button before or after the specified sibling.

❑ **addContent(*path*, *description*, *replaceExisting*)**

Returns elements. Adds new content by replacing the content in the current content panel or adding a new tab.

where:

path

String.

Path of the content being added.

description

String.

Description of the content being added.

replaceExisting

Boolean.

Optional. When true, replace the existing content. When false, which is the default, add new content.

- **addMenuItem(*options*, *sibling*, *before*)**

  Returns an element, adding a menu item to the tab container menu.

  where:

  options

  Object.

  One or more menu item options, which can include the following:

  - **text.** The menu item text.
  - **icon.** A URL to an image file in .jpeg, .png, or .gif format.
  - **glyph.** A glyph or ligature, such as a Material icon.
  - **glyphClasses.** Glyph classes. For example, 'material-icons'.
  - **class.** Additional CSS classes.

  sibling

  Selector, element, or jQuery.

  Optional. A sibling to add before or after. The default is Last.

  before

  Boolean.

  Add the menu item before or after the specified sibling.

- **addMenuSeparator(*options*, *sibling*, *before*)**

  Returns an element, adding a separator to the tab container menu.

where:

options

Object.

Menu separator options. You can use the class property to specify additional CSS classes.

sibling

Selector, element, or jQuery.

Optional. A sibling to add before or after. The default is Last.

before

Boolean.

Add menu separator before or after the specified sibling.

❑ **buttons(*selector*)**

Returns the container title bar buttons as an array.

where:

selector

Selector, element, or jQuery.

Is the buttons selector. All is the default.

❑ **classId()**

Returns a string, which is the unique, automatically generated class ID of the object being automated. It can be used to filter enumerations.

❑ **contents(*selector*)**

Returns the content panels in the container as an array.

where:

selector

Selector, element, or jQuery.

Is the content selector. All is the default.

❑ **customClasses(*classes*, *remove*)**

Returns a string. Sets or gets custom classes.

where:

classes

String.

Optional. A space-separated list of classes to be added or removed.

remove

Boolean.

Optional. If true, remove the classes specified in *classes*, otherwise add them.

- ❑ **element()**

  Returns an element, the DOM node that is being automated by this object.

- ❑ **removeButton(*button*)**

  Removes one or more buttons from the title bar of the container.

  where:

  button

  Selector, element, or jQuery.
  Optional. Is a button selector. All is the default.

- ❑ **removeContent(*selector*)**

  Removes content from the container.

  where:

  selector

  Selector, element, or jQuery.
  Optional. Is a content selector. All is the default.

- ❑ **removeMenuItem(*selector*)**

  Removes one or more menu items from the container menu.

  where:

  selector

  Selector, element, or jQuery.
  Optional. Is a menu item selector. All is the default.

- ❑ **removeMenuSeparator(*selector*)**

  Removes one or more separators from the container.

where:

`selector`

Selector, element, or jQuery.
Optional. Is a menu separator selector. All is the default.

- ❑ **selectContent(*selector*)**

  Returns an element. Selects a content pane in the tab container.

  where:

  `selector`

  Selector, element, or jQuery.
  Optional. Is a content selector. All is the default.

- ❑ **title(*title*)**

  Sets the title of the container. If *title* is not passed, the title of the container is returned as a string or jQuery.

  where:

  `title`

  String.
  Optional. Is the new title of the container.

Class: ibaContent

The ibaContent class extends ibaObject.

```
new ibaContent(element)
```

Is the content panel automation object. Automation objects are created internally by the system. You never directly instantiate an automation object.

where:

`element`

Is the DOM node that is being automated.

Available methods include the following:

- ❑ **classId()**

  Returns a string, which is the unique, automatically generated class ID of the object being automated. It can be used to filter enumerations.

- ❑ **customClasses(*classes*, *remove*)**

  Returns a string. Sets or gets custom classes.

  where:

  classes

  String.

  Optional. A space-separated list of classes to be added or removed.

  remove

  Boolean.

  Optional. If true, remove the classes specified in *classes*, otherwise add them.

- ❑ **element()**

  Returns an element, which is the DOM node that is being automated by this object.

- ❑ **path()**

  Returns the path of the content as a string.

Adding Panels to a Responsive Container

You can utilize responsive panel in panel behavior by adding items to a panel in panel container. When you run the page, top-level containers in the page, including the panel group container, fold responsively based on the size of the browser, and the panels within the panel group container fold based on its current size, independent of the rest of the page. Additionally, items in the panel group container are kept together when folding occurs, making the panel group container a good way to group together closely related content that you want to be visually separated from the other items on the page. This is particularly true when related items are arranged vertically, as items typically wrap based on rows.

To create a panel group container, click the *Container* tab on the sidebar, select the *Panel group* container, and drag it onto the canvas. You can then resize it like any other container.

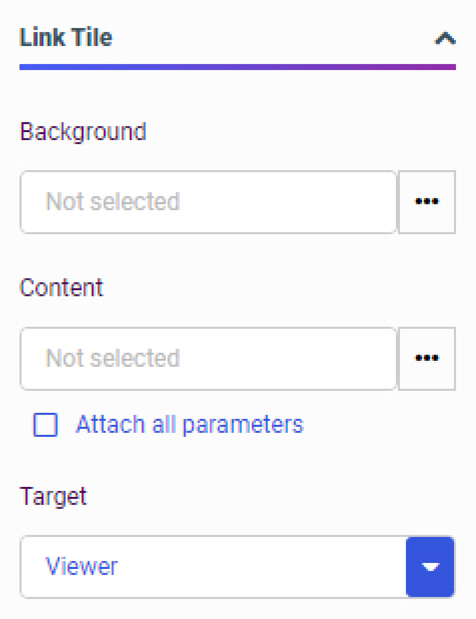
To add a panel to a panel group container, drag a content item or another container into it. You can then resize the new container inside the panel group container.

You can use the Properties panel to add a toolbar and title to the panel group container. Properties applied to the entire page or to the panel group container, such as margins or styling, affect all panels within it, unless specifically overridden.

Linking to External Content From a Page

The link tile widget provides additional innovative ways of using your content and incorporating it into your page. The link tile widget layers content in such a way that one content item becomes a click-through tile that opens another content item. This feature is especially useful when you need to incorporate a large or Insight-enabled item into a page and display it on all devices. The link tile widget can be used in pages assembled from existing content.

To add a link tile to a page, drag the link tile object from the Resources panel with *Container* selected on the sidebar. You configure a link tile using the Settings tab of the Properties panel, as shown in the following image.



Link Tile

Background

Not selected

Content

Not selected

□ Attach all parameters

Target

Viewer

When adding a link tile widget to a page, you configure the content that you want to display on the page using the Background property, the item that you want to link to using the Content property, and the location in which to open the linked content using the Target property.

If the link tile background content is a procedure, filters can be created for and applied to it just like any other content item.

Additionally, the filter selections in the page can be passed to the link tile target content if it is a procedure or a page by enabling the *Attach all parameters* option. When enabled, the link tile target is automatically filtered corresponding to the source page.

Procedure: How to Use the Link Tile Widget

1. Assemble a new page from existing content, or open an existing page in Db2 Web Query Designer.
2. On the sidebar, click the *Container* tab.
3. Drag a link tile container from the Resources panel onto the page

   The link tile container initially appears blank.
4. With the link tile selected on the page, expand the Link Tile area of the Settings tab on the Properties panel.

   Note: The link tile container does not include a title bar by default, so clicking the link tile on the canvas may only allow you to select the link tile background content. To select the link tile container more easily, use the outline.
5. In the Link Tile section of the *Settings* tab, click the ellipses button next to the Background option and navigate to the item from your repository that you would like to display in the link tile.

   You can select a FOCEXEC procedure such as a chart or report, a page, an image, a URL, an HTML page, and more. The item behaves as an image, so interactive features such as tooltips are not enabled.
6. Click the ellipses icon next to the Content option and select the target item. This content will run when users click on the link tile panel.
7. Select the *Attach all parameters* check box if parameter filter selections from the page should be passed to the target content, or leave it unselected to open the target content without passing filter selections.
8. Determine how the target item will display. Select *Viewer* to run the target content item in an overlay on the same browser tab as the link tile, or select *New window* to run the item in a new browser tab.
9. Run the visualization using the *Run in new window* button to test the link tile behavior.

   When you click the link tile, your target content opens as configured.

Enabling Container Customization

When creating a page with external content, you can enable users to select content to display at run time, which can make pages more engaging, useful, and interactive. With this feature you can also control which items users can access, by assigning a specific path to the Add content button in the Properties panel. This feature can be used with panel, tab, carousel, and accordion containers. If you enable this feature on a multi-content container, the Add content button automatically appears on every new tab, slide, or area.

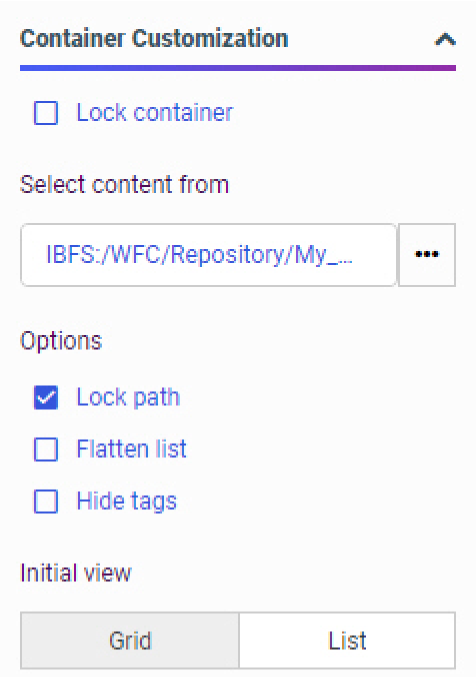
Procedure: How to Configure the Container Customization Properties

1. In a visualization using external content, drag an empty panel or a content item onto the canvas.
2. With the panel or content item selected, in the Properties panel, on the Settings tab, disable the *Lock container* option.

   If you used an empty panel, the *Add content* icon displays in the middle of the selected panel.
3. If you want to direct your selection to a specific area in the repository, click the ellipses next to the *Select content from* property to select a default workspace or folder that you want to display for content selection at run time.
4. Keep the *Lock path* option selected, if you want to limit your selection to the path specified in the previous step. Otherwise, users can navigate to other workspaces to access content there in addition to the default path.
5. Optionally, select the *Flatten list* option, if you want to hide the folder hierarchy inside your selection area and only display items.
6. Optionally, select the *Hide tags* option, if you want to hide all tags from your selection area.
7. Configure the *Initial View* option, choosing either Grid or List, to set the initial appearance of the window where users select the content to display.

   Note: Users can change the view at run time inside the Select Item dialog box.

The following image shows an example of the configured Container Customization properties.



Container Customization

□ Lock container

Select content from

IBFS:/WFC/Repository/My_...

Options

☑ Lock path

□ Flatten list

□ Hide tags

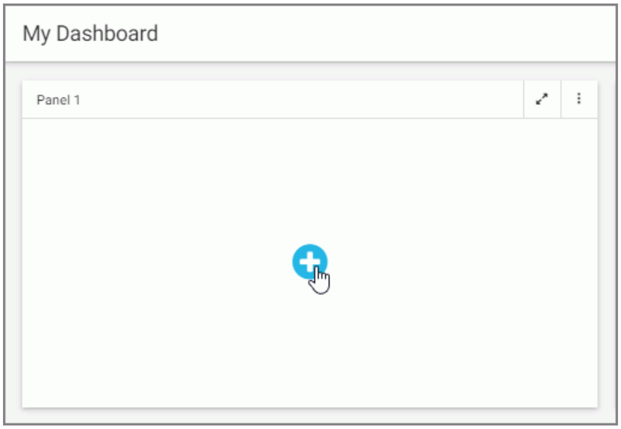
Initial view

8. Save the page.

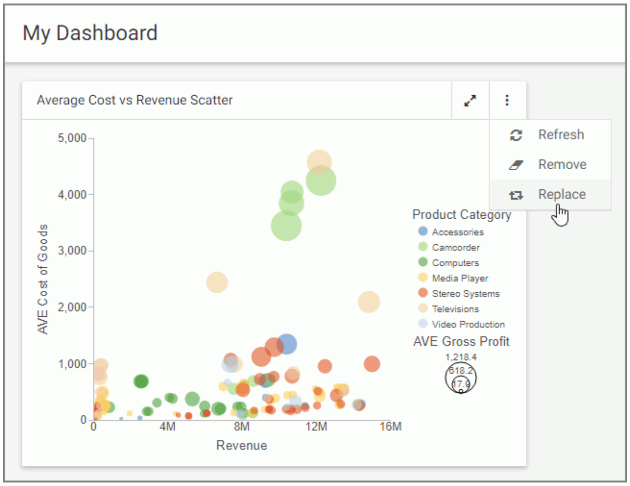
Procedure: How to Select Content at Run Time

1. Run the page that features interactive panels.

2. Click the *Add content* button, as shown in the following image.



If you enabled container customization for a container with content in it, click the Options menu and then click *Replace*, as shown in the following image.



The Select Item dialog box opens.

If the *Lock path* option is disabled, you can navigate between workspaces using the breadcrumb trail.

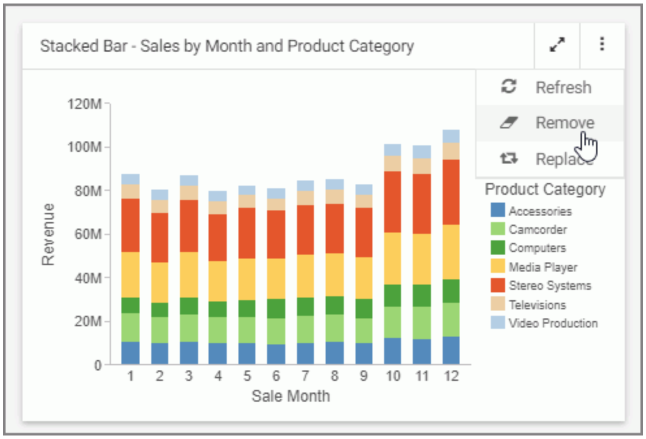
If the *Flatten list* option is disabled, you can navigate between the folders within the selection area hierarchy.

You can also use the search function, toggle between the Grid and List views, and refresh your selection area.

3. Make a selection and click *Add*.

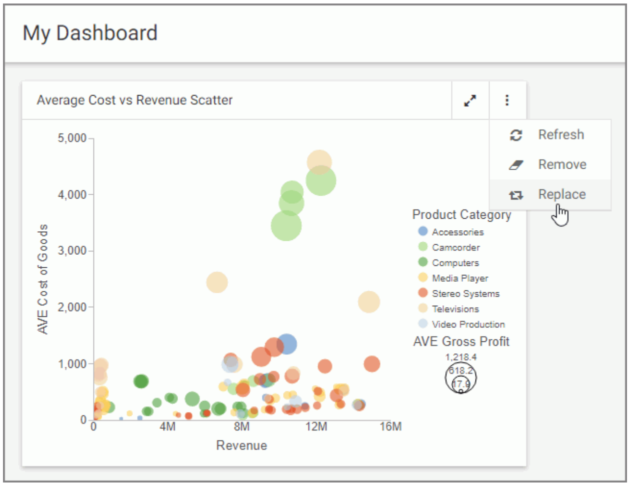
   The selected item displays inside the panel.

4. To go back to the initial view of the interactive panel, click *Options*, and then click *Remove*, as shown in the following image.



   The item is removed from the panel, and the *Add content* button displays.

5. To replace an item with another item, click *Options*, and then click *Replace*, as shown in the following image.



   The Select Item dialog box opens, and you can make a new selection.

Providing Access to Content Items and Db2 Web Query Tools

You can use the Explorer widget to provide access to the Workspaces view of the Db2 Web Query Home Page from a page assembled from existing content in Db2 Web Query Designer. This allows anyone accessing the page to open and run content items and create new content, depending on their privileges. In the Explorer widget, you can navigate through workspaces, folders, and content items from the Resources tree and Content area, and create new content using options on the Action bar.

When assembling pages from existing content in Db2 Web Query Designer, two widget containers are available that allow you to provide access to content saved to your Repository. These are the Workbench and Explorer widgets. The Explorer widget also allows users to create new content using the same options that are available on the Db2 Web Query Home Page.

The Workbench widget provides a resource tree that you can link to another container on the page using the *XXXXX* option on the Format tab. You can then run content items from your Repository, listed in the resource tree, in the selected target container by double-clicking them or by right-clicking them and clicking *Run*. When you run multiple content items, they are added to the container as new tabs, by default. You can change the container type to an accordion or carousel container by right-clicking it and pointing to *Convert to*, then selecting an option. In this case, new items that you run from the Workbench resource tree are added to new accordion panels or carousel slides.

By default, the Explorer widget fills the entire page in order to provide ample space to display any content items and content creation options.

Procedure: How to Use the Explorer Widget

You can add the Explorer widget to a page assembled from existing content.

1. On the Db2 Web Query Home Page, click the plus menu and then click *Assemble Visualizations*.

   Db2 Web Query Designer opens.

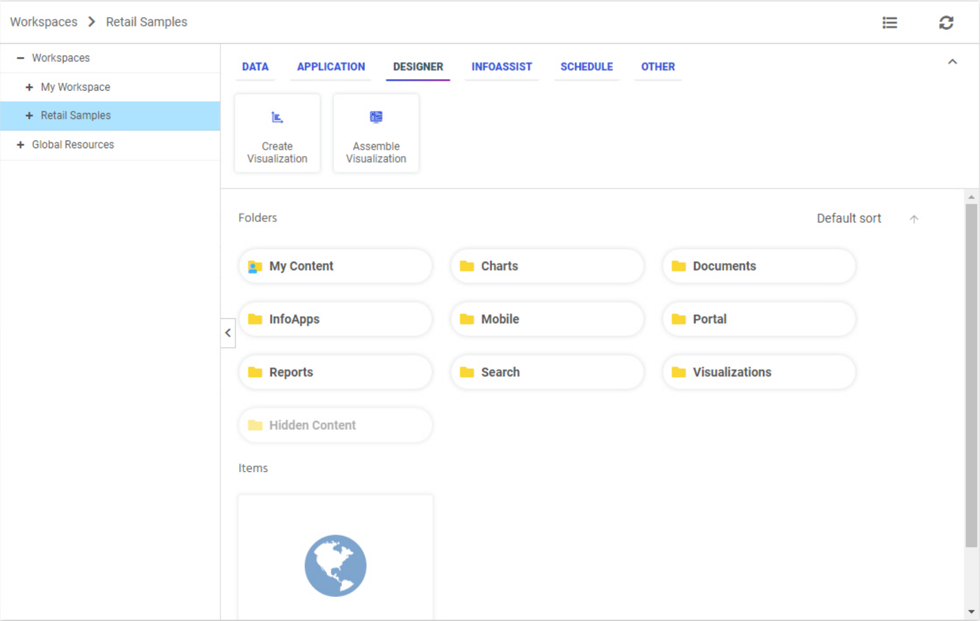
2. In the Choose Template dialog box, select the *Blank* template.
3. From the *Container* tab on the sidebar, drag the *Explorer* widget onto the canvas.

   The widget displays on the page and incorporates the full display of the Workspaces view of the Db2 Web Query Home Page.

4. Click the maximize button on the Explorer widget container to fill your page.
5. Optimize the look of the page by removing the page and container titles and toolbars. Select the Explorer widget container and, on the *Settings* tab, clear the *Show title* and *Show toolbar* check boxes.

Optionally, select the entire page and clear the *Show page heading* and *Show page toolbar* check boxes to show only the Explorer widget on the page.

An example of a completed page is shown in the following image.



6. Save your page.

Creating Custom Templates

Aside from the default templates that Db2 Web Query Designer offers, you can create custom templates that reflect your unique layout. They are a great way to add structural variation or highly customized layouts to your page. Custom templates are also useful for organizations that have a set style for all of their pages. You must have administrator access to Db2 Web Query to deploy custom templates.

Procedure: How to Create a Custom Template for Db2 Web Query Designer Pages

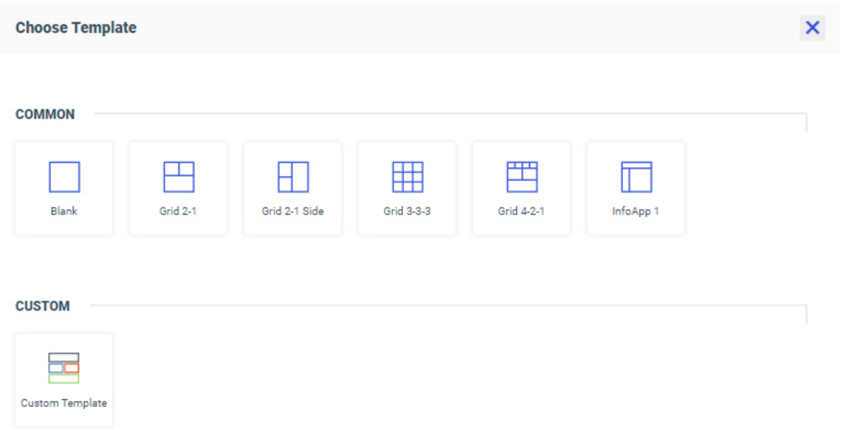
1. Create a new page. On the default Db2 Web Query Home Page, click the plus button.
2. Click *Assemble Visualizations*.

   Db2 Web Query Designer opens, and you are prompted to select a template.
3. Select one of the default templates that you want to use as the starting point for your custom template, to click *Blank* to build a new template from scratch.

4. Modify the page layout to reflect your preferences. For example, insert new sections, customize the sizing of each row in the template.
5. Save your new page to any folder and exit Db2 Web Query Designer.
6. Optionally, apply a custom thumbnail to your template page. If you do not assign a custom thumbnail, the default thumbnail Db2 Web Query associates with pages will display.
7. On the Workspace area of the Home Page, expand the *Global Resources* folder, and then expand the *Page Templates* folder.

   Two folders display inside the Page Templates folder: Standard and Custom.
8. Copy your new template page from its initial directory and paste it inside the *Custom* folder.

   The next time you launch Db2 Web Query Designer and select the Assemble Visualizations option, your custom template appears, as shown in the following image.



Passing Parameter Values to a Page

When a page includes content with dynamic parameter filters, you can pass parameter values to it from external content to set the filter values when the page initially loads. For example, you can create a chart or report with drill-down links that points to the page, and configure the drilldowns in such a way that the value the user clicks in the chart or report is used as a filter value in the page. This allows users to navigate between content items and determine the scope of the information that they want to see.

If the page is the target of a link tile in another page, parameter values in the source page are passed to the target page automatically if the Attach all parameters option is enabled and the same parameter filters are present. This allows users to see the same set of values in the target page without having to set filter values again.

Parameter values can also be passed to a page using parameter name-value pairs in the run-time URL. The run-time URL uses the following format:

```
http[s]://hostname:port/context_root/run/path_to_item[?
&variable_name1=variable_value1&variable_name2=variable_value2]
```

where:

hostname

Is the name of the system where Db2 Web Query is installed.

port

Is the port number used by Db2 Web Query.

context_root

Is the context root used for your Db2 Web Query application. For example, webquery.

path_to_item

Is the IBFS path of a page in your Db2 Web Query repository. You can find the path of a page by right-clicking it on the Db2 Web Query Home Page and clicking *Properties*. The file path is listed as the Path property.

Slashes (/) between folder names should be retained literally instead of encoded in the URL, and the colon (:) should be removed after *IBFS*. For example:

```
http://localhost:12331/webquery/run/IBFS/WFC/Repository/Retail_Sales/
example_page
```

&variable_name1=variable_value1&variable_name2=variable_value2

Are one or more parameter name-value pairs. To pass multiple values to a multiselect parameter in the page, separately list the same parameter name multiple times with each distinct value. For example, the URL,

```
http://localhost:12331/webquery/run/IBFS/WFC/Repository/Retail_Sales/
example_page?&TIME_YEAR=2018&TIME_YEAR=2019
```

passes 2018 and 2019 to the &TIME_YEAR parameter.

Note: If multiple filters in a page use the same amper variable name, for example, if they come from different data sources, the value or values defined in the URL are passed to each of these filters.

Procedure: How to Pass Parameter Values to a Page Using a URL Drilldown

You can drill down from a chart or report procedure to a page created in Db2 Web Query Designer. Drill-down parameters can be passed to the page to apply their values to any page filters.

1. Create and save a page that includes prompted filters. These filters can be created by adding parameterized external content to an assembled page and clicking *Add all filters to page* in the Filters tab of the sidebar, or can be created in a page with new content by dragging a field to the Filter toolbar.

   The filters in this page will be populated with values passed from a drill-down report.

   For information on creating a page, see *Creating Pages* on page 74.

2. Create a new report in InfoAssist.

   On the Db2 Web Query Home Page, in the Workspaces view, click the *INFOASSIST* tab on the Action bar and click *Report* or *Chart*.

   InfoAssist opens and you are prompted to select a data source.

3. Select the same data source that was used to filter the page, and click *Open*.

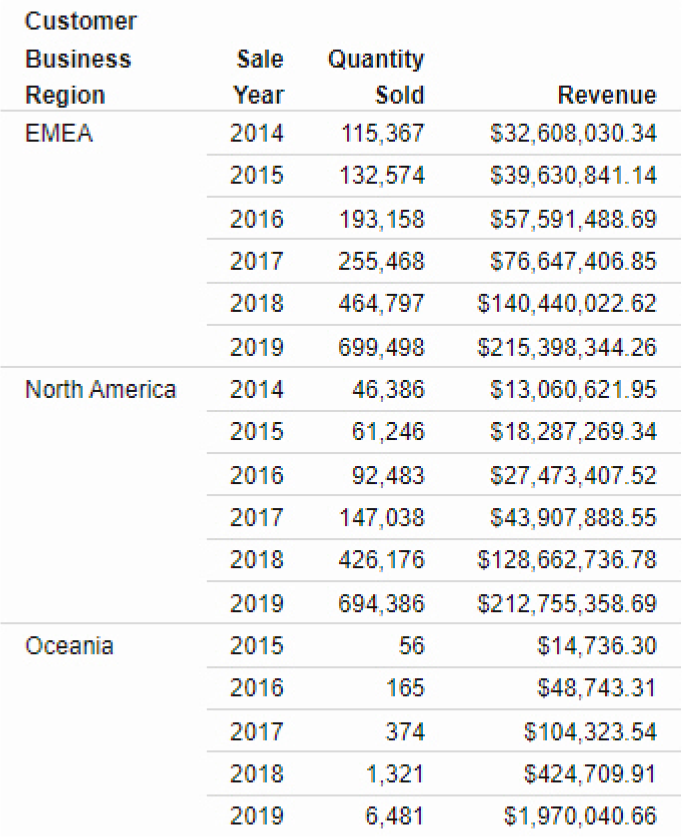
   The report or chart canvas appears.

4. Create content that is sorted by the same fields that are used as filters in the page. For example, if the page is filtered by Customer Business Region and Sale Year, create a report with those fields in the row bucket. Add other measures and dimensions to your content based on your needs and preferences.

5. Select an element of the chart or report to which to add drill-down links.

   If you are passing multiple parameter values through the drilldown, selecting a section of the chart or report that represents multiple sort values may help clarify to the user which values will be passed.

For example, in the report shown in the following image, the Sale Year field is the secondary sort field while Customer Business Region is the primary sort field, so each row in the Sale Year column represents a sale year and a business region.



Customer Business Region	Sale Year	Quantity Sold	Revenue
EMEA	2014	115,367	$32,608,030.34
	2015	132,574	$39,630,841.14
	2016	193,158	$57,591,488.69
	2017	255,468	$76,647,406.85
	2018	464,797	$140,440,022.62
	2019	699,498	$215,398,344.26
North America	2014	46,386	$13,060,621.95
	2015	61,246	$18,287,269.34
	2016	92,483	$27,473,407.52
	2017	147,038	$43,907,888.55
	2018	426,176	$128,662,736.78
	2019	694,386	$212,755,358.69
Oceania	2015	56	$14,736.30
	2016	165	$48,743.31
	2017	374	$104,323.54
	2018	1,321	$424,709.91
	2019	6,481	$1,970,040.66

In this case, adding the drill-down links to the Sale Year column would allow you to choose from all of the available Customer Business Region and Sale Year values, whereas adding the links to the Customer Business Region column would only allow you to pass parameter values for the first sale year in each business region group. As an alternative to creating drilldowns on the lower level sort field, you could ensure that all parameter values are passed from a report by using repeating sort values. On the *Format* tab, in the *Features* group, click *Repeat Sort Value*.

6. On the *Field* tab, in the *Links* group, click *Drill Down*.

   The Drill Down dialog box opens.

7. Change the drilldown type to a URL drilldown by selecting the *Web Page* radio button.

8. In the URL text box, type the run-time URL for the target page, without the parameters, using the format described above.

   The URL format should resemble the following:

   ```
   http[s]://hostname:port/context_root/run/path_to_page
   ```

   You can get the path to the page by right-clicking it on the Home Page and clicking *Properties*. It displays as the Path property. When you add it to the run-time URL for the drilldown, remember to remove the colon (:) after *IBFS*.

9. In the Description text box, type the name of the drilldown. This name displays in the tooltip from your content.

10. In the Target drop-down menu, select the location where the target page should run. It can be in a new tab, the same window where the report was run, or a user-specified location.

11. In the Parameters section, click *Add Parameter* to create a parameter to pass to each filter on the target page that you wish to populate.

    a. In the Name text box, type the name of the parameter. Typically, this matches the field name. You can check the parameter names used in an assembled page by clicking the *Info* button.

    b. From the Type menu, choose whether the value to pass should be the field value associated with the user drill-down link selection, or a constant value.

    c. If you selected *Field*, use the Value menu to select a field from your content. If you selected *Constant*, type a constant value to pass to the page.

12. Repeat step 11 for all parameters whose values you want to pass to the target page, then click *OK* to create the drilldown.

    In a report, drill-down links are added for each value of the field where the drilldown was created. In a chart, you can access drill-down links in a tooltip at run-time.

13. Run the chart or report.

14. At run-time, drill down to the target page.

    ❑ If you created a chart, point to a section of the chart and click the drill-down link in the tooltip.

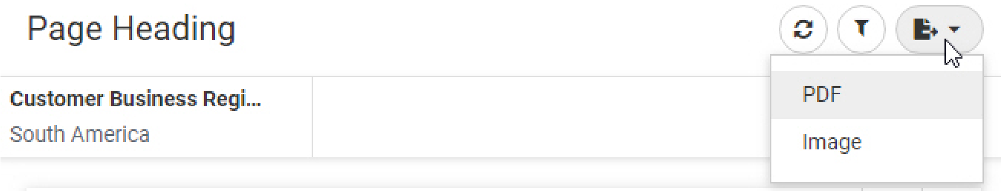
    ❑ If you created a report, click a drill-down link. If that field of the report has multiple drilldowns associated with it, select the drilldown that you just created.

15. The page opens in a new browser tab or window. Notice that the value or values represented by your selection in the parent content item are applied as filter values in the page.

Exporting a Page as a PDF or Image

You can instantly download a snapshot of your page as a PDF document or .png image using the Export to file menu on the page toolbar. This allows you and anyone running the page to quickly capture its current state, with filter selections applied, in a format that can be easily reused, distributed, added to PowerPoint presentations, and more. The ability to capture a snapshot at design time allows you to quickly capture multiple visual representations of your data as you develop your content, while the ability to capture a snapshot at run time allows you or a user running the page to save images of it with the currently selected filter values.

When designing or running the page, the Export to file menu appears on the page toolbar by default. Click it to display options to download a snapshot of the page as a PDF or image, as shown in the following image.



The .pdf or .png file is downloaded using your browser. The file can be opened immediately and saved to distribute and reuse later.

The snapshot that is produced is a run-time representation of your content with styling and filtering, including on-chart filtering, applied. Since the exported file is an image, it is not interactive, so tooltips, scrolling, and other behavior are unavailable.

You can change the export options that are visible on the page toolbar. Select the entire page, then open the Settings tab. The *Show export* option and *PDF* and *Image* options are selected by default. Clear the *PDF* or *Image* check box to change the format available when exporting the page. If only one format is enabled, the Export to file option on the page toolbar changes from a menu to a button. You can also clear the *Show export* check box to remove the export option from the page.

The following considerations may affect page export behavior:

- Content in containers on a page may not appear correctly when exported from Internet Explorer.
- When exporting a page from Microsoft Edge, ensure that the containers on the page are large enough to display their contents without scroll bars. Otherwise, the content may overlap.

- XML files and procedures with an XML output format added to a page produce an error when exporting.
- Certain chart extensions, for example, the Sparkline KPI, KPI with Sparkline Large, and KPI Table extensions, may not render correctly in an exported page.
- If a container on a page contains a URL that does not have the same protocol, host name, and port number as that of the environment from which the page is run, the export cannot be performed due to same-origin policy restrictions.
- If a page has too many rows to fit onto a single page of a PDF document, then containers and their contents may be split by page breaks in the exported file.
- PDF documents in containers on a page, including charts and reports with a PDF output format, do not display when the page is exported.