



Query SQL Data using DBXL Web Service

06/23/2021

The purpose of this document is to guide form designers in using DBXL's QueryDB Web Service to query data from a SQL database.

Contents

Prerequisites	1
Step 1: Build the query using Query Builder	1
Step 2: Add the QueryDBWithUserKeyService Data Connection.....	5
Step 3: Identify where the form should query data and add form rules.....	6
Step 4: Test your form	10

Prerequisites

- This document assumes that DBXL is already installed and running in your server and you have generated a key for submitting and querying documents. If you don't have the key, please ask your DBXL admin to provide or follow the steps in this [document](#).
- FormsViewer is already installed

Step 1: Build the query using Query Builder

1. In DBXL's Admin Tool, select your document type.
2. Click on the Query Builder tab and click **New Query**.

Home Catalog DocType Tools License

Qdabra DBXL
Administration Tool DocType: ExpenseReport

General Database Permissions Documents Statistics Email Reminder XML Import **Query Builder**

Use this tab to generate a query string for your Web service data connection or for a form rule.

Existing Queries **New Query** Refresh Queries

No queries found for this docType.

powered by Qdabra DBXL Copyright © 2006-2021 Qdabra Software. All rights reserved. Qdabra and the Qdabra logo are trademarks of Autonomy Systems LLC. www.qdabra.com English

- (Optional) Enter a query description.
- Click **Construct Now**.

Query

Query Description

Data Sources

Web Service Url	Web Service Name	Target
http://dbxl-ws2016.westus.cloudapp.azure.com	QdabraWebService	<input checked="" type="radio"/> SQL Database

Database Connection String **Construct Now**

Query Parameters Refresh Database Schema

Database Name	Table Name	Max Rows	Distinct
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

- Select **Specify a database connection string**:

Data Sources

Web Service Url	Web Service Name	Target
http://dbxl-ws2016.westus.cloudapp.azure.com	QdabraWebService	<input checked="" type="radio"/> SQL Database

Database Connection String **Clear** **Done**

Specify a database name or alias **Specify a database connection string**

Connection Parameters

Data Source (Server Name)	Initial Catalog (Database Name)	Authentication
<input type="text"/>	<input type="text"/>	<input checked="" type="radio"/> Windows <input type="radio"/> SQL Server

- Enter your **Data Source** (Server Name), **Initial Catalog** (Database Name) and select **Authentication** type.

Database Connection String Clear Done

Specify a database name or alias Specify a database connection string

Connection Parameters

Data Source (Server Name)	Initial Catalog (Database Name)	Authentication
DBXL-WSTEST	SampleDB	<input checked="" type="radio"/> Windows <input type="radio"/> SQL Server

Data Source=DBXL-WSTEST;Initial Catalog=SampleDB;Integrated Security=True

7. Click **Refresh Database Schema**:

Database Connection String Clear Done

Specify a database name or alias Specify a database connection string

Connection Parameters

Data Source (Server Name)	Initial Catalog (Database Name)	Authentication
DBXL-WSTEST	SampleDB	<input checked="" type="radio"/> Windows <input type="radio"/> SQL Server

Data Source=DBXL-WSTEST;Initial Catalog=SampleDB;Integrated Security=True

Query Parameters Refresh Database Schema

Database Name	Table Name	Max Rows Count	Distinct
SampleDB	ExpenseReports		

8. If connection to your database is successful, the Table Name dropdown will show the tables in the database you specified:

Query Parameters Refresh Database Schema

Database Name	Table Name	Max Rows Count	Distinct
SampleDB	ExpenseReports		

Query Columns Insert Column

Column Name	
ExpenseReports	<input checked="" type="checkbox"/>
GoodsRequested	<input type="checkbox"/>
PatientConsent	<input type="checkbox"/>

9. Select your table. Under **Query Columns**, click **Insert Column**.

Query Parameters Refresh Database Schema

Database Name	Table Name	Max Rows Count	Distinct
SampleDB	ExpenseReports		

Query Columns Insert Column

10. Select the columns that you want to include in your query:

Query Parameters		Refresh Database Schema	
Database Name	Table Name	Max Rows Count	Distinct
SampleDB	ExpenseReports	<input type="text"/>	<input type="checkbox"/>

Query Columns		Insert Column	
Column Name			
DocID	<input type="text"/>	<input type="text"/>	<input type="text"/>
ID	<input type="text"/>	<input type="text"/>	<input type="text"/>
Name	<input type="text"/>	<input type="text"/>	<input type="text"/>
Purpose	<input type="text"/>	<input type="text"/>	<input type="text"/>
Total	<input type="text"/>	<input type="text"/>	<input type="text"/>

11. If you want to add filters to your query, you can specify them under the **Query Filter** section – select the column that you want to filter and use the comparison operator to compare the column to a value or field name, here’s an example:

Query Filter			Insert Logical Op	
Column Name	Comparison Operator	Value or Column Name	Insert Comparison	
Name	Equal	{my:Name}	<input type="text"/>	<input type="text"/>

12. You can also add sorting by specifying the column to sort in the **Query Sort** section.

Query Sort		Insert	
Column Name	Sort Order		
ID	Ascending	<input type="text"/>	<input type="text"/>

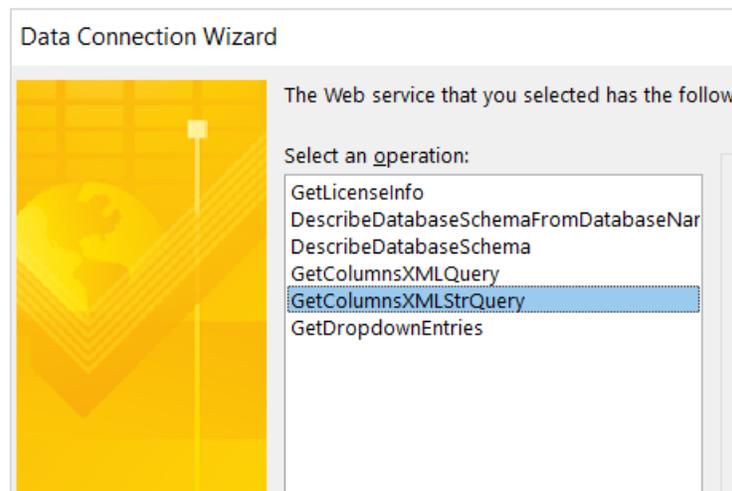
13. Click **Build Now**. This will generate the Query XML to be used in the form.

Query Strings	Build Now
Query XML to be used as a "sample value" in the Data Connection Wizard	<input type="text"/>
Query XML to be used as a "value" in the Data Connection Wizard	<input type="text"/>
Query XML to be used in the form Rules	<input type="text"/>
Query XML to be used in the form Code	<input type="text"/>

14. Take note of these queries since you're going to use them when creating your form data connection in the next section.
15. Save the query in Query Builder. This is optional but it would be handy when you need to copy it again next time.

Step 2: Add the QueryDBWithUserKeyService Data Connection

1. In your form template, create a receive data connection to <http://ServerName/QdabraWebService/QueryDBWithUserKey.asmx>
2. Select **GetColumnsXMLStrQuery**.



3. In the next window, supply the following values:
 - a. **Key** – this is your DBXL key
 - b. **queryXML** – this is the first Query XML generated in Query Builder.
4. Next, you will be asked to enter the values again. This time, for the queryXML, enter the second Query XML from Query Builder.
 - a. **Key** – this is your DBXL key
 - b. **queryXML** – this is the second Query XML from Query Builder.
5. Name your QueryDB connection.
6. **IMPORTANT:** Uncheck the box "Automatically retrieve data when form is opened"

Enter a name for this data connection:

Query Expense Reports

Automatically retrieve data when form is opened

Note: your form will open more quickly if you delay data queries until the data is needed. To do this, create a rule to query data when a button is clicked or when the value of a field changes.

7. Click Finish to close the wizard.

Step 3: Identify where the form should query data and add form rules

Now that you have added the QueryDB data connection, you can add a rule in your form to query this data connection on form load (qRules finishedLoading) or on field change / button click which depends on your form requirements.

1. After you've identified the place to add the query – in your form rule, add an action to query the data connection.

Rules

finishedLoading

Query data

New

Details for:
Query data

Condition:
None - Rule runs when field changes

Rule type:
Action

Run these actions:*
Query using a data connection: Query Expense Reports

Don't run remaining rules if the condition of this rule is met

2. To verify you're getting data correctly, drag the repeating row in the view:

QueryDB Sample

Doc ID	ID	Name	Purpose	Total

Repeating Table

Fields

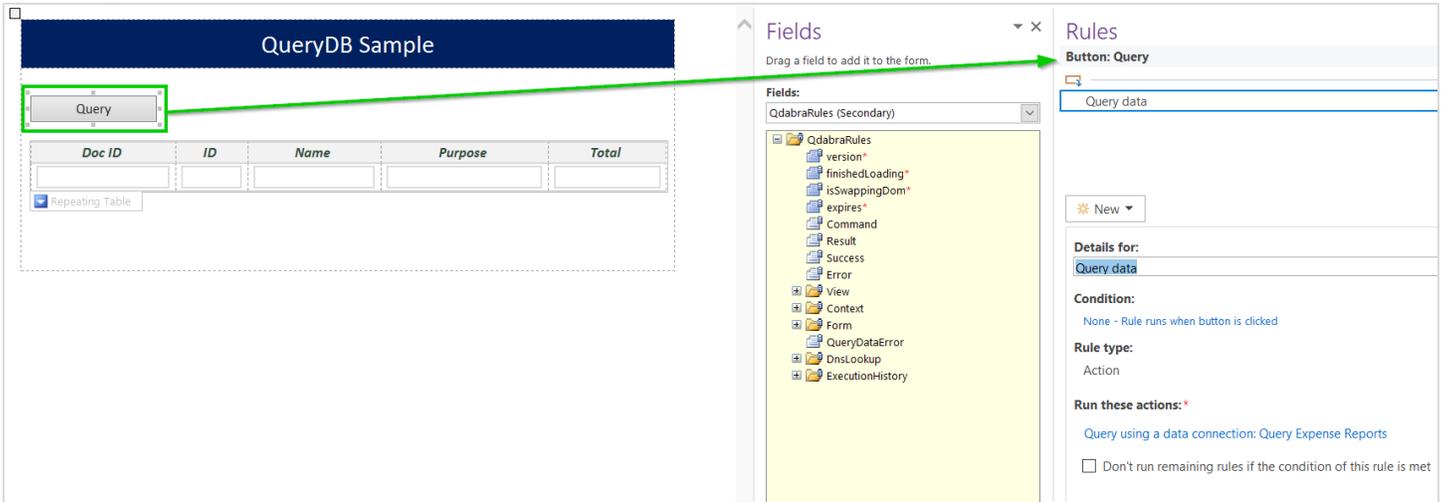
Drag a field to add it to the form.

Fields:
Query Expense Reports (Secondary)

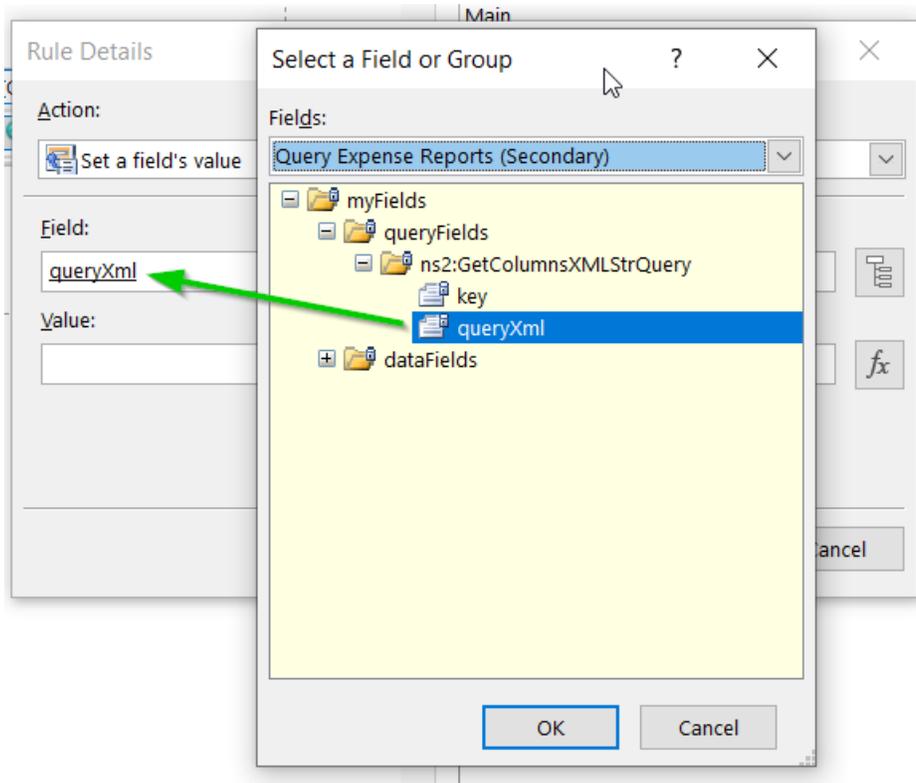
- myFields
 - queryFields
 - dataFields
 - ns2:GetColumnsXMLStrQueryResponse
 - GetColumnsXMLStrQueryResult
 - .Rows
 - table
 - count
 - row

- DocID
- ID
- Name
- Purpose
- Total

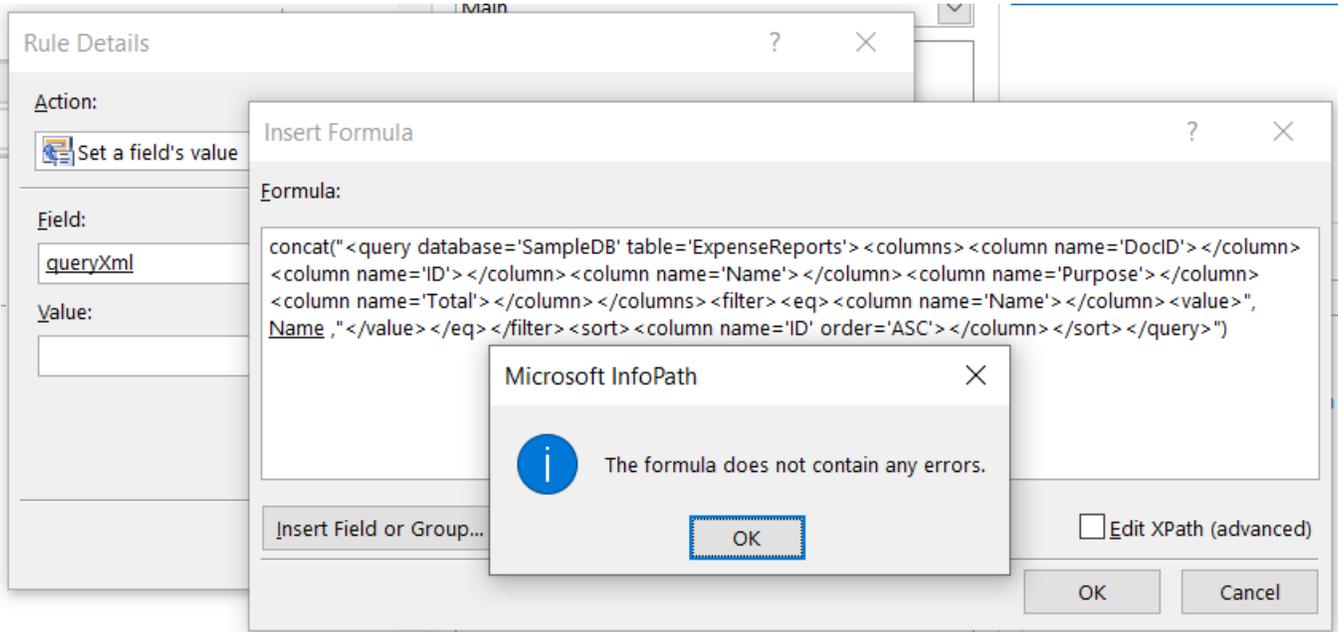
3. If you want to query data on field change or button click, add the rule / query for data action in the field or button, for example:



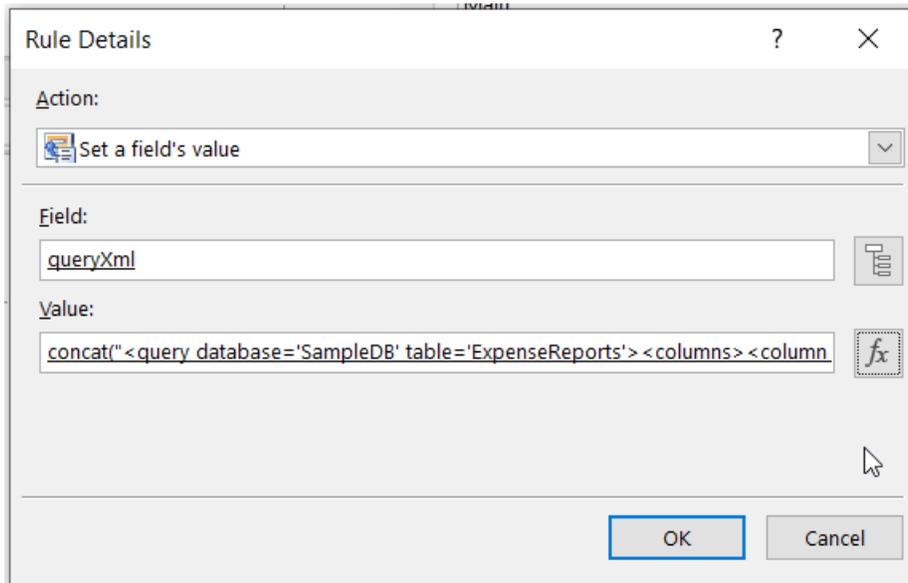
4. If the query XML in your data connection is not filtered, and you want to query with filters, go back to Query Builder and add a filter to your Query XML and build / generate the query again. You can add a filtered query on field change or button click by using the third Query XML from Query Builder. Here's how to do this:
 - a. In your form rule, add an action to set a field's value.
 - b. For the **Field**, select the queryXml field in the QueryDB data connection.



- c. For the **Value**, select the **fx** button and enter the third Query XML from Query Builder.
- d. Click **Verify Formula**. It should verify and display no errors. Click OK.



- e. Now you'll have something like this:



- f. Click OK.

g. Add a second action to query the data connection.

Details for:
Query data

Condition:
None - Rule runs when button is clicked

Rule type:
Action

Run these actions: * Add ▾

Set a field's value: queryXml = concat("<query database='SampleDB' table='Expen...")

Query using a data connection: Query Expense Reports

Don't run remaining rules if the condition of this rule is met

5. After making the changes above, save your form template.

Step 4: Test your form

1. Upload your form in FormsViewer.
2. Open the form.
3. If you've set the form to query data on finishedLoading, the form should query data on load.
4. If a field or button triggers the query, select the field or click the button.
5. Verify that your QueryDB is correctly retrieving or displaying data.

QueryDB Sample

Query

Doc ID	ID	Name	Purpose	Total
214	2	Jo-Anne Tandaan	Training	\$100.00
215	3	Jo-Anne Tandaan	Lodging	\$250.00



Contact Information

Qdabra Software
218 Main Street,
Suite 731,
Kirkland, WA
98033

Phone: 877.544.2389
Email: Support@Qdabra.com
Website: <http://www.qdabra.com>
Community: <http://www.infopathdev.com>