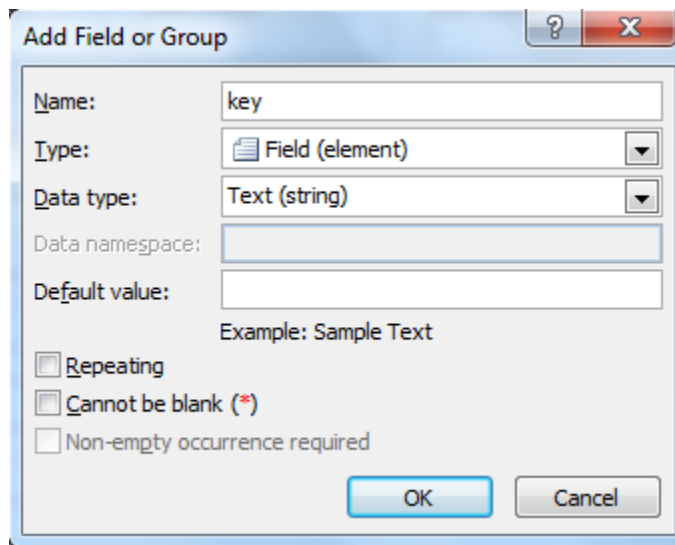


LINK TO YOUR DBXL FORMS

RETRIEVE THE DBXL DOCID AND CREATE A LINK FOR A SUBMITTED FORM

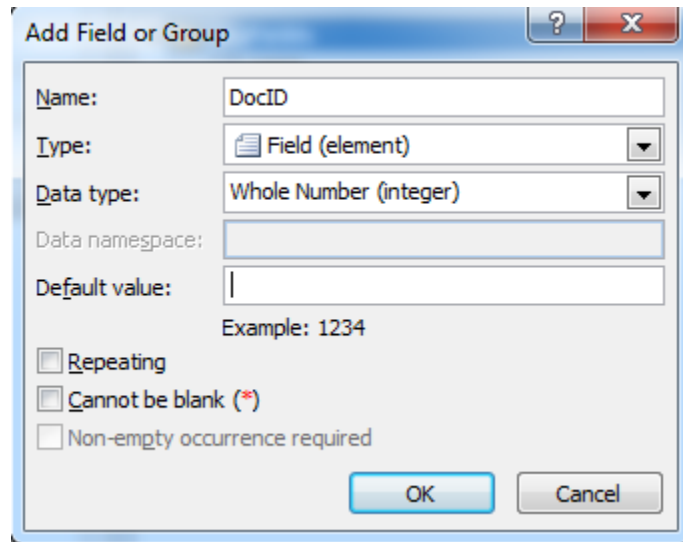
Sometimes it is useful to know the DocID of a document that was just submitted to DBXL. For example, you might want to send an email immediately after submission. So how can we retrieve the DocID of the document?

1. Open your Expense Report XSN in design mode.
2. Add a new field and call it **key**.



3. Add one more node to the schema, and call it **DocID**. Make it an integer data type.

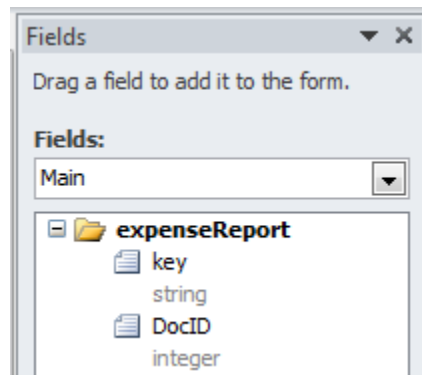
Qdabra DBXL Training: Module 7



The 'Add Field or Group' dialog box is shown. It has the following fields and options:

- Name:** DocID
- Type:** Field (element)
- Data type:** Whole Number (integer)
- Data namespace:** (empty)
- Default value:** (empty)
- Example:** 1234
- ☐ Repeating
- ☐ Cannot be blank (*)
- ☐ Non-empty occurrence required
- Buttons:** OK, Cancel

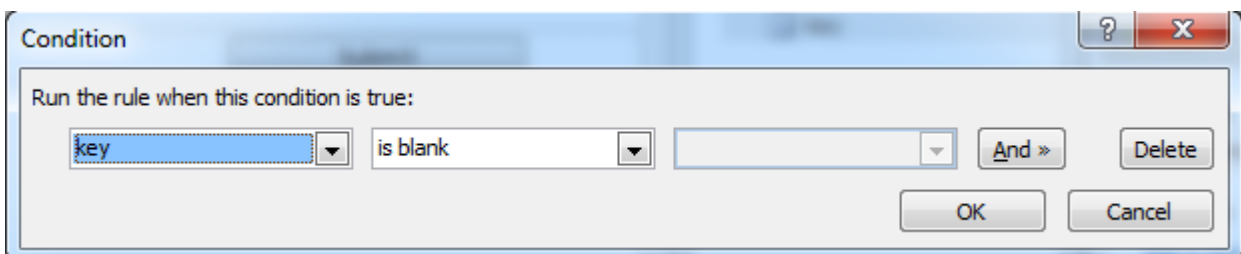
Your new schema will have these two new nodes:



The 'Fields' dialog box is shown. It has the following fields and options:

- Fields:** Main
- expenseReport** (folder icon)
 - key (document icon)
 - string
 - DocID (document icon)
 - integer

4. Click **Data > Form Load**.
5. Click **New > Action**. Name the rule *"Set the value of key"*.
6. Add a condition such that this rule will only execute when **key** is blank.



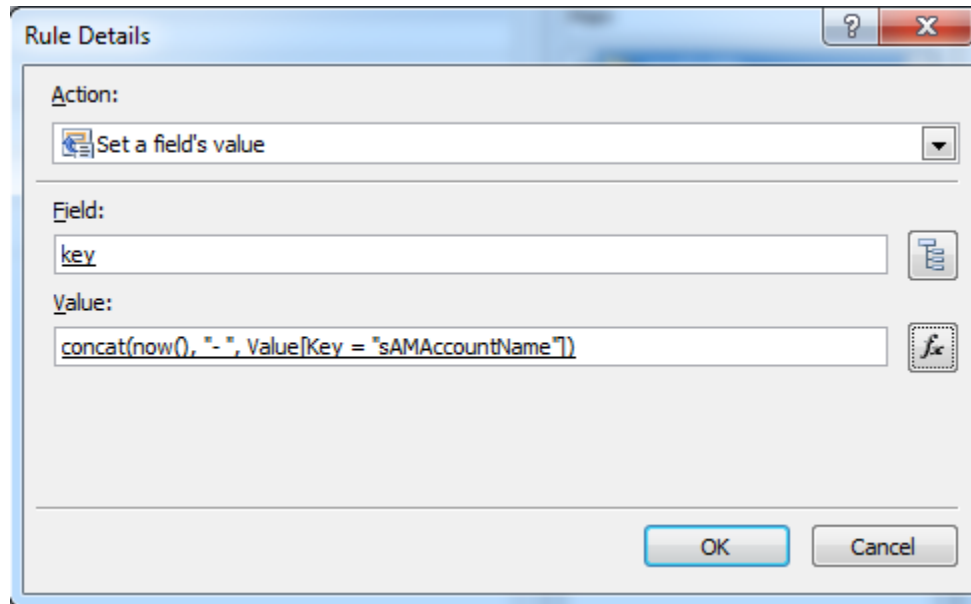
The 'Condition' dialog box is shown. It has the following fields and options:

- Run the rule when this condition is true:**
- key** (dropdown menu)
- is blank** (dropdown menu)
- Buttons:** And », Delete, OK, Cancel

7. Click **Add > Set a field's value**. This action will set the value of **key** to the following:

```
concat(xdDate:Now(), "- ",
xdXDocument:GetDOM("GetMyInfo")/dfs:myFields/dfs:dataFields/tns:GetMyInfoResponse/tns:GetMyInfoResult/tns:ADProp/tns:Value[../tns:Key = "sAMAccountName"]])
```

Note that you will need to click the **fx** button to enter the above formula. Your rule will look like this.



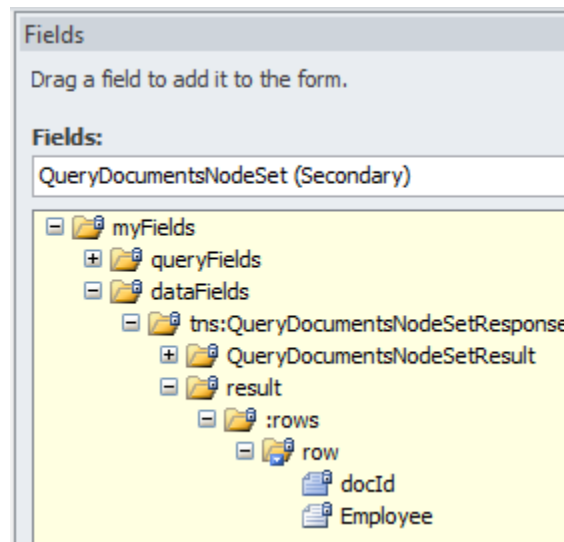
What is this rule doing? We are creating a unique key by concatenating the timestamp and the username. This creates a unique key because we can safely assume that one user won't be able to submit two documents at the same, exact moment.

8. Go to **Data > Data Connections > Add**. Follow the Data Connection Wizard to add a data connection that receives data from <http://<servername>/qdabrawebservice/dbxldocumentservice.asmx>. This data connection will call the method **QueryDocumentNodeSet** and will not be set to **"Automatically retrieve data when form is opened"**. In the parameter screen, when asked for the value of the tns:query parameter, enter the following:

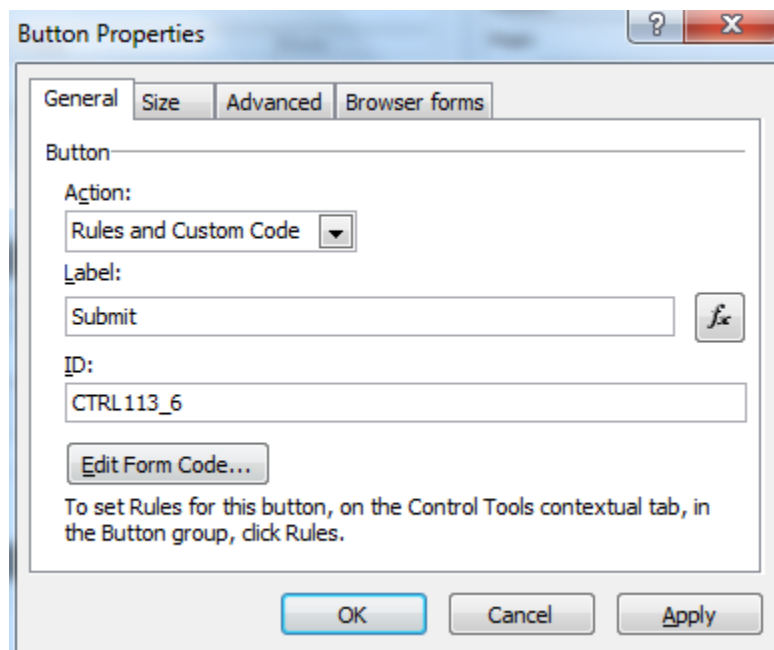
```
<query doctypeName="YOUR DOCUMENT TYPE"><columns repeatingPath="/my:expenseReport"><column name="my:employee/my:name" alias="Employee" /></columns></query>
```

You can confirm that this data connection successfully obtained a schema by checking the dataFields:

Qdabra DBXL Training: Module 7



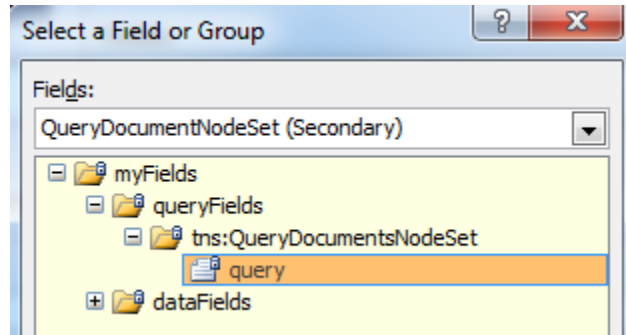
9. Right click on the Submit button and select **Properties**.
10. Change the button such that **Action** is set to "Rules and Custom Code" and change the **Label** to "Submit". Click **OK**.



11. Select the Submit button and go to **Home > Manage Rules**.
12. Add an action that submits using the data connection which submits to DBXL. This was created in the first lab of this training.

Qdabra DBXL Training: Module 7

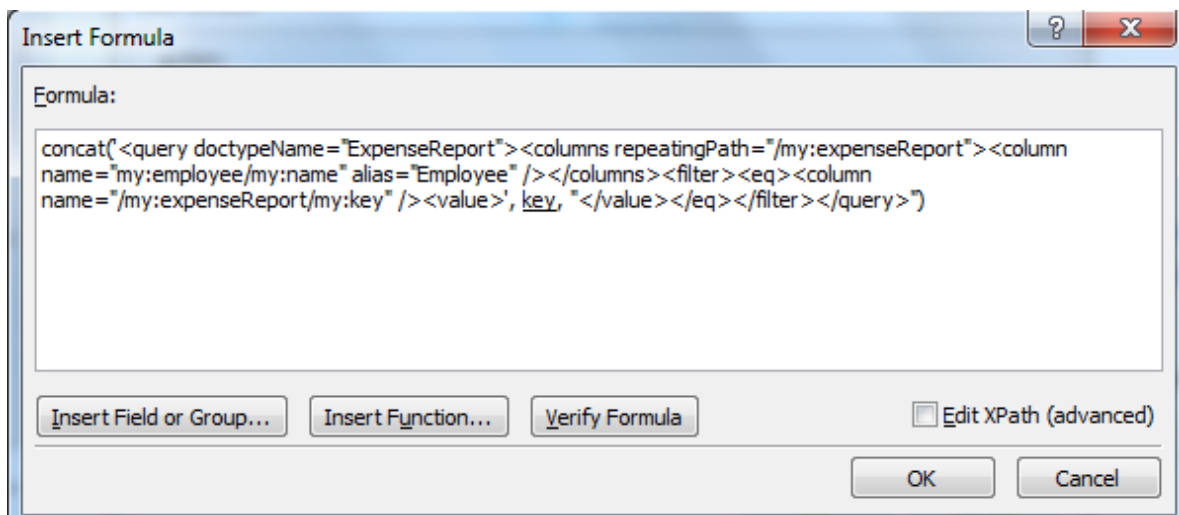
13. Add an action that sets the value of the query node:



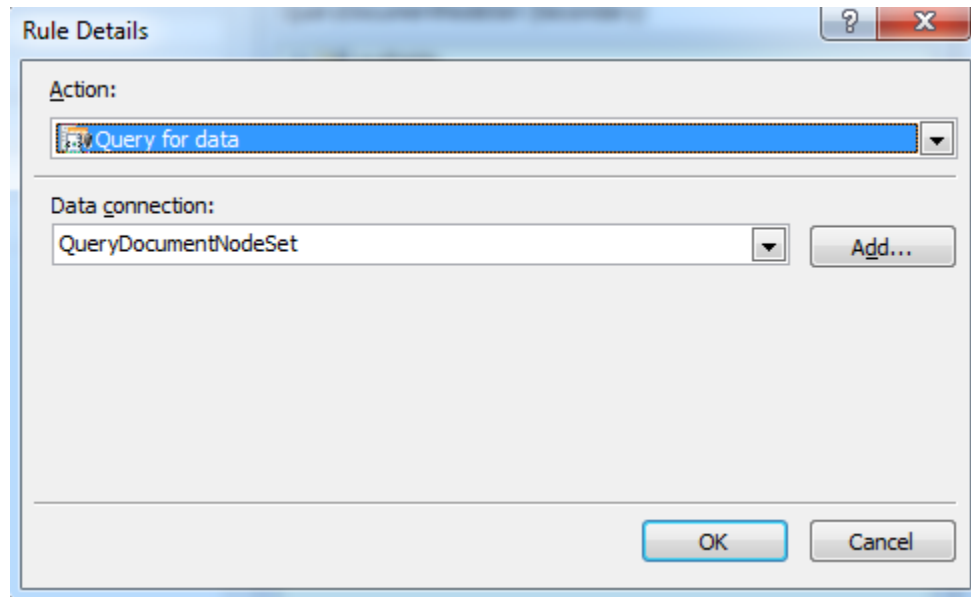
The value will be:

```
concat('<query doctypeName="YOURDOCTYPENAME"><columns
repeatingPath="/my:expenseReport"><column name="my:employee/my:name" alias="Employee"
/></columns><filter><eq> <column name="/my:expenseReport/my:key"
/><value>'<u>key</u>'</value></eq></filter></query>')
```

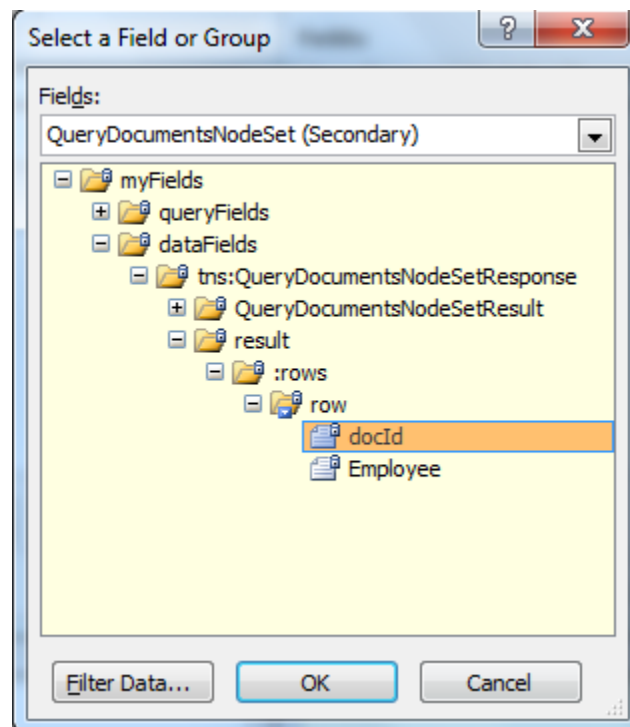
You will need to use the **fx** button when entering this value. In it, you'll use YOUR document type name and reference the field **key**, which is the field we added earlier in the steps.



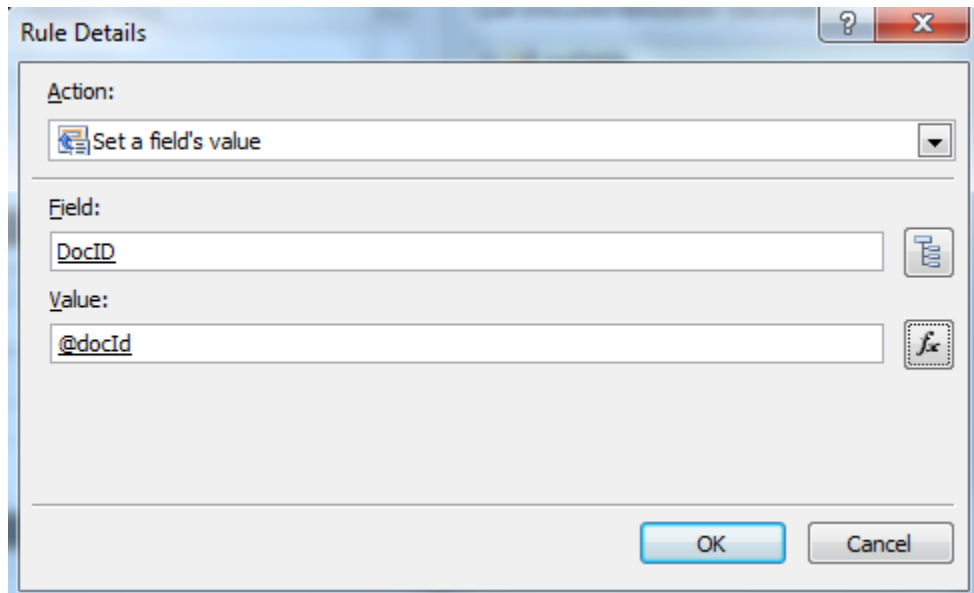
14. Add another action that queries data from the QueryDocumentNodeSet data connection.



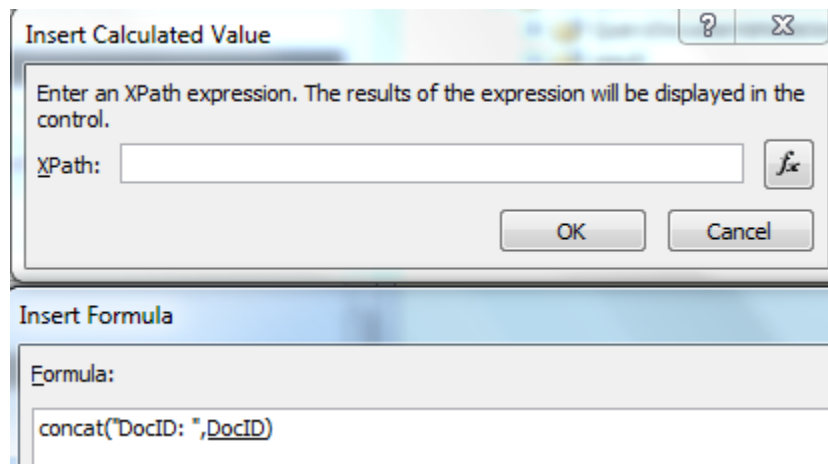
15. Add another action, this time to set the value of my:DocID to the value returned by the data connection.



Qdabra DBXL Training: Module 7



16. Finally, add a Calculated Value in the form in order to display the DocID.



17. Save the XSN, re-attach it to your document type in DAT, and **Save**.

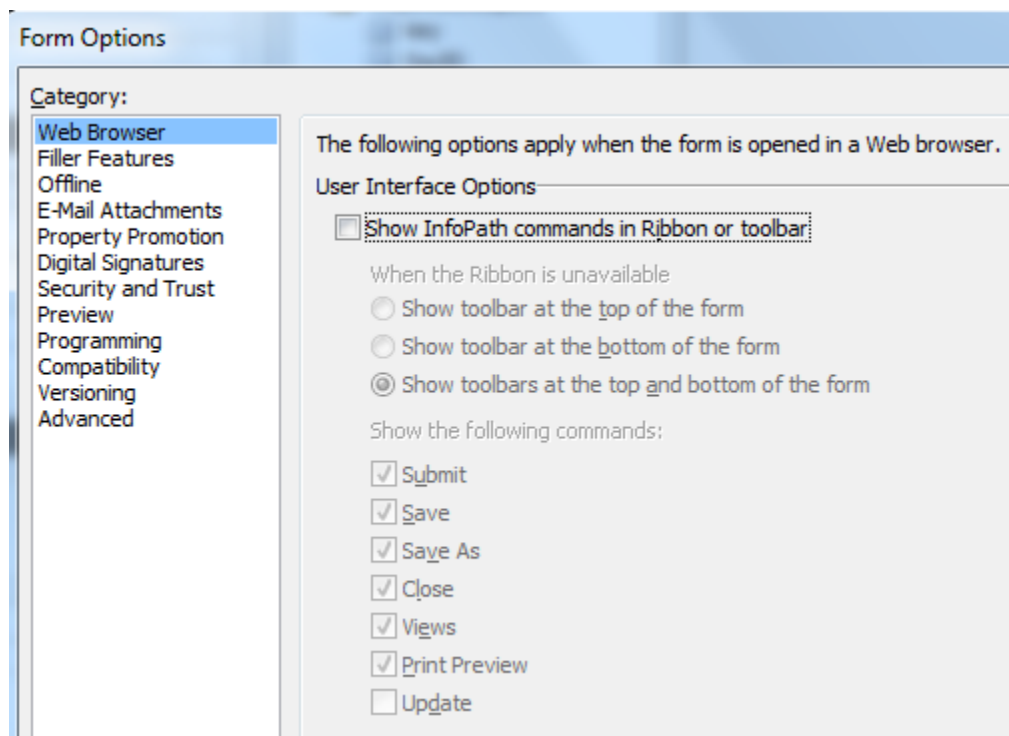
You can now test by submitting a new document and verifying that the DocID is returned to the form immediately after submitting.

Qdabra DBXL Training: Module 7



A screenshot of a web form. At the top, there is a blue button with the text "Submit". Below the button, the text "DocID: 6481" is displayed. The form is enclosed in a light gray border.

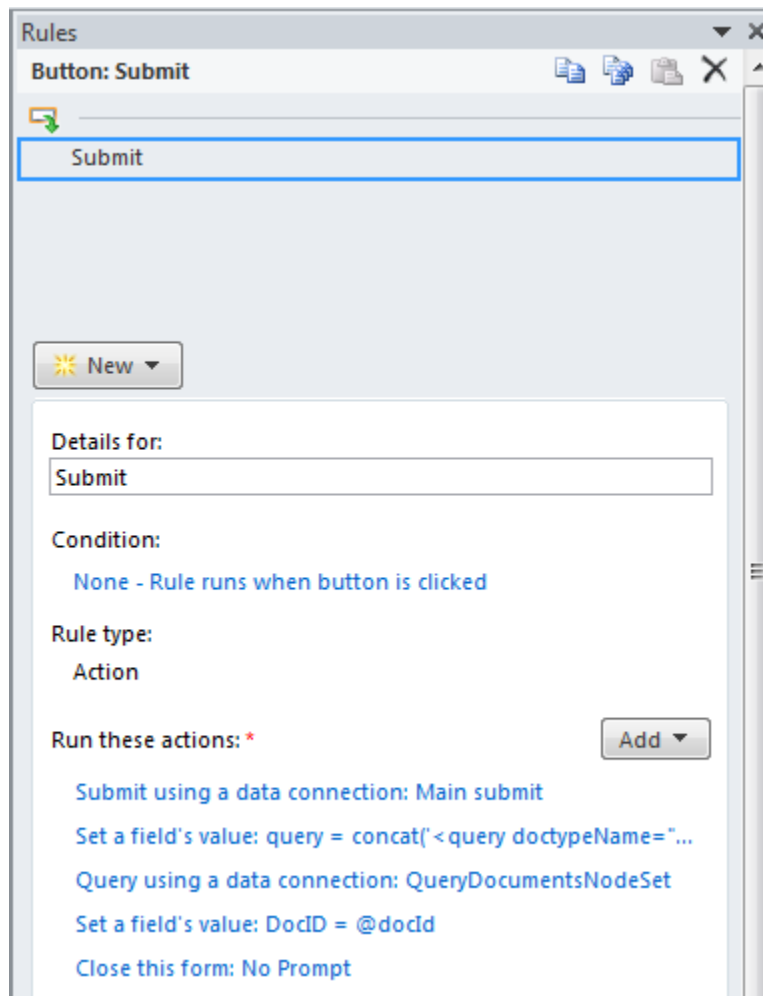
18. Return to the designer. Since we are now submitting via the button on the form, let's remove the ribbon in the browser form. Do this in **File > Form Options**, and uncheck "Show InfoPath commands in Ribbon or toolbar".



A screenshot of the "Form Options" dialog box. The "Category:" list on the left has "Web Browser" selected. The main area is titled "The following options apply when the form is opened in a Web browser." Under "User Interface Options", the checkbox "Show InfoPath commands in Ribbon or toolbar" is unchecked. Below this, there are three radio button options for toolbar placement: "Show toolbar at the top of the form", "Show toolbar at the bottom of the form", and "Show toolbars at the top and bottom of the form" (which is selected). A section titled "Show the following commands:" contains a list of checkboxes: "Submit", "Save", "Save As", "Close", "Views", "Print Preview", and "Update". The "Submit", "Save", "Save As", "Close", "Views", and "Print Preview" checkboxes are checked, while "Update" is unchecked.

19. Add one final action to the submit button, such that the form closes after all the other actions execute. Then update the XSN attached to DAT.

Qdabra DBXL Training: Module 7



This DocID can be used to generate a direct URL to your form, for example. The DocID, along with concat, can be used to set a field's value to the URL, like this:

```
concat("http://YOURSERVERNAME/qdabrawebservice/documents/", DocID, "/", DocID, ".xml")
```

Note that this method is only necessary to get the DocID for a newly submitted document. There is another way to obtain the DocID for a document that already exists in DBXL, explained below.

EXTRA: DISPLAYING INFORMATION FROM THE PROCESSING INSTRUCTIONS

You can also obtain the DocID from the **processing-instructions (PI)**, but the processing instructions hold much more information.

There are two downsides about PIs:

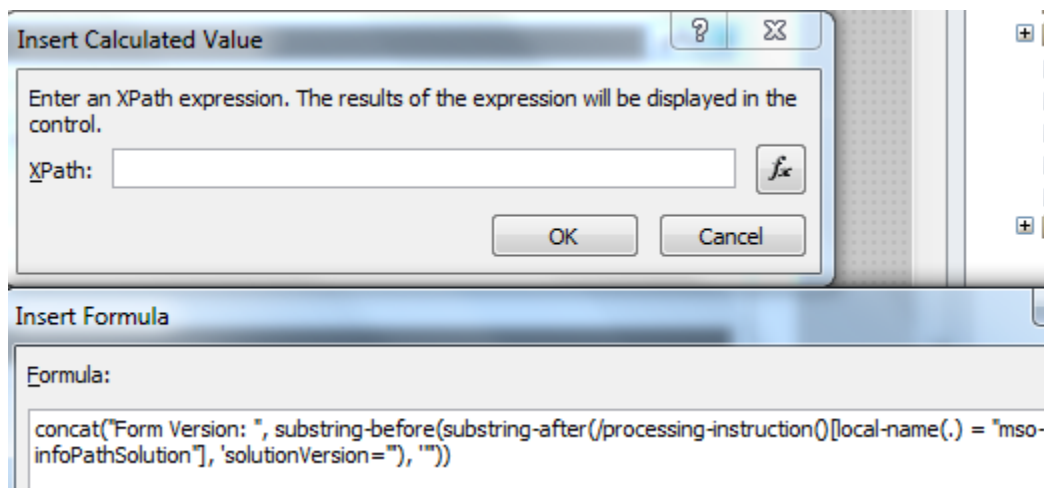
Qdabra DBXL Training: Module 7

- they do not allow XPath-style addressing of their attributes, so you are forced to use string-processing methods to retrieve values of the associated attributes.
- New forms do not have information such as DocID and version until submitted and reopened.

As an example of how to use the processing instructions:

20. Add a Calculated value to your form and use this formula, below. This formula will display the solution version of a form.

`concat("Form Version: ", substring-before(substring-after(/processing-instruction())[local-name(.) = "mso-infoPathSolution"], 'solutionVersion="'), '"'))`



It is often very helpful to display the version of a form in the view while a user is filling it out. This often helps when debugging because you can quickly and easily determine if the problem they are having is because of a versioning conflict.

21. Save the form and update it in DAT.

Now, when the form is opened, you'll see the version number:



Let's update the **Form Load** rules:

22. Go to **Data > Form Load**.

Qdabra DBXL Training: Module 7

23. Click **New > Action**.
24. Add a condition such that this rule only executes if the DocID is blank.

25. Click **Add > Set a field's value**.
26. For **Field**, select the **DocID** field.
27. For **Value**, click **fx** and enter the following:

```
concat("Document ID: ", substring-before(substring-after(/processing-instruction()[local-name(.) = "QdabraDBXL"], 'docid="'), ""))
```

This will grab the DocID from the processing instructions if, for some reason, that field is blank on load.

28. Save the form and update it in DAT.