



## Lab 3 – How to Query User Profiles

### Overview

The `GetUserProfileByName` command offers a way to retrieve SharePoint user profile data and display them in a form. A typical scenario would be to show the current user's manager name within a form that requires a manager's approval. This document will show you exactly how you can use the command in the Expense Report Form.

### Before you start

1. This lab assumes that you have finished the Lab 1 and 2 Exercises. Use the Start form provided in this lab.
2. **Important:** In this lab, we will get the user information from each respective site's Site Users List, using the **`GetUserProfileByName`** parameter **`/source=siteusers`**. The Site Users List only exposes a limited set of user values, and it does not include the Manager property by default.

You will need to carry out the following to enable querying the Manager properties.

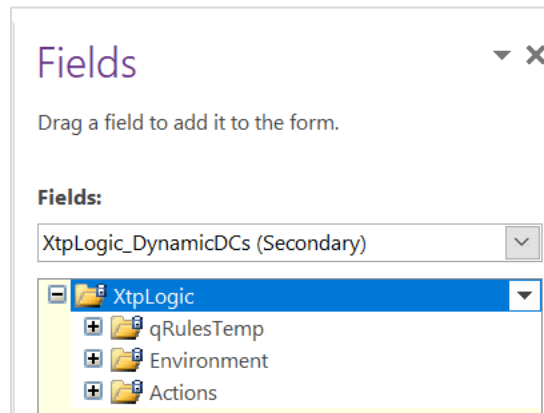
- In the SharePoint Admin Center, click Classic features → User Profiles Open → Manage User Properties.
- Find the Manager property, click that property name, then Edit.
- Under the Policy Settings section, check the Replicable checkbox. Note that this requires unchecking the User can override checkbox if it is checked. Click OK.

Note: It may take up to 24 hours for the properties to replicate to the Site Users List.

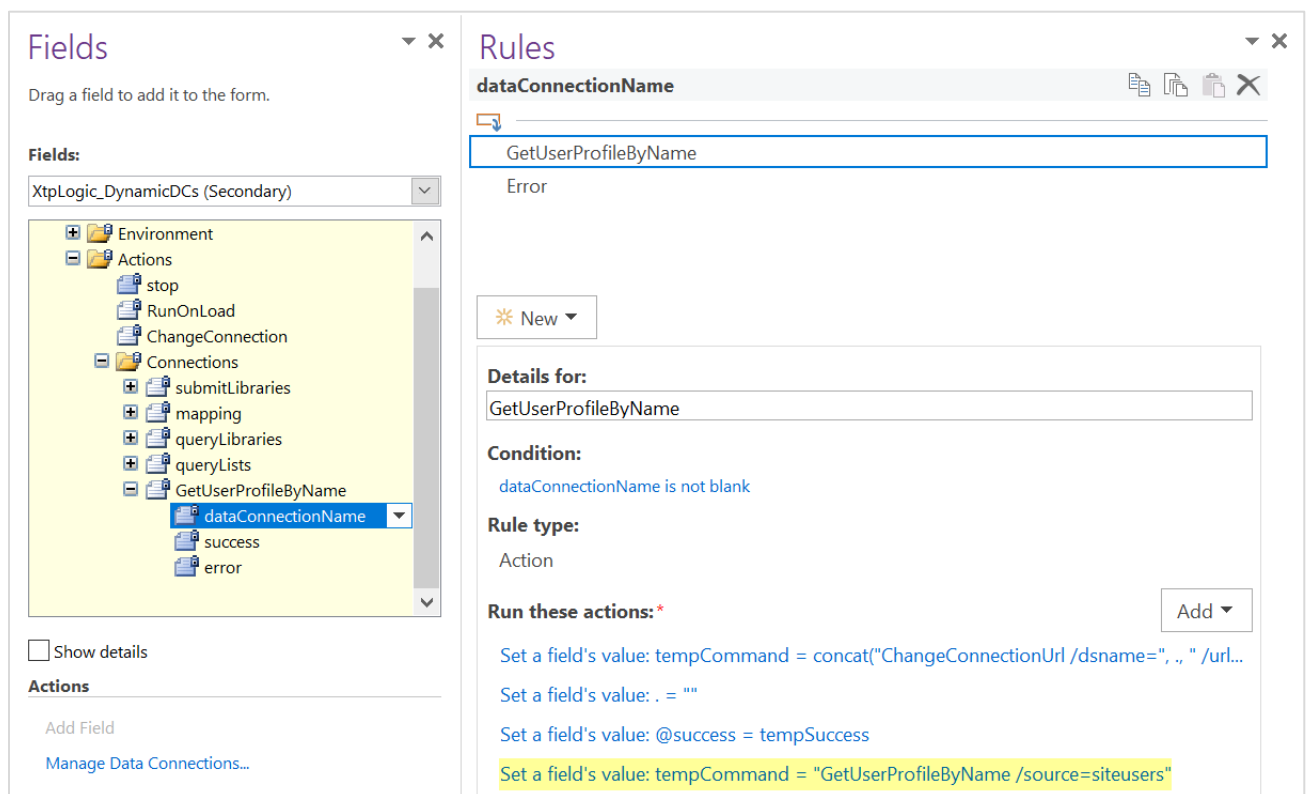
### Verify the `GetUserProfileByName` in Dynamic DCs XTP

In this section, we're going to see how the `GetUserProfileByName` command is applied and used in the form within the XTP implemented in the previous lab. The Site Users List only exposes a limited set of user values, and it does not include the Manager property by default.

1. Open the starter form for this lab in InfoPath Designer.
2. In the Fields pane, select the `XtpLogic_DynamicDCs` secondary data source.

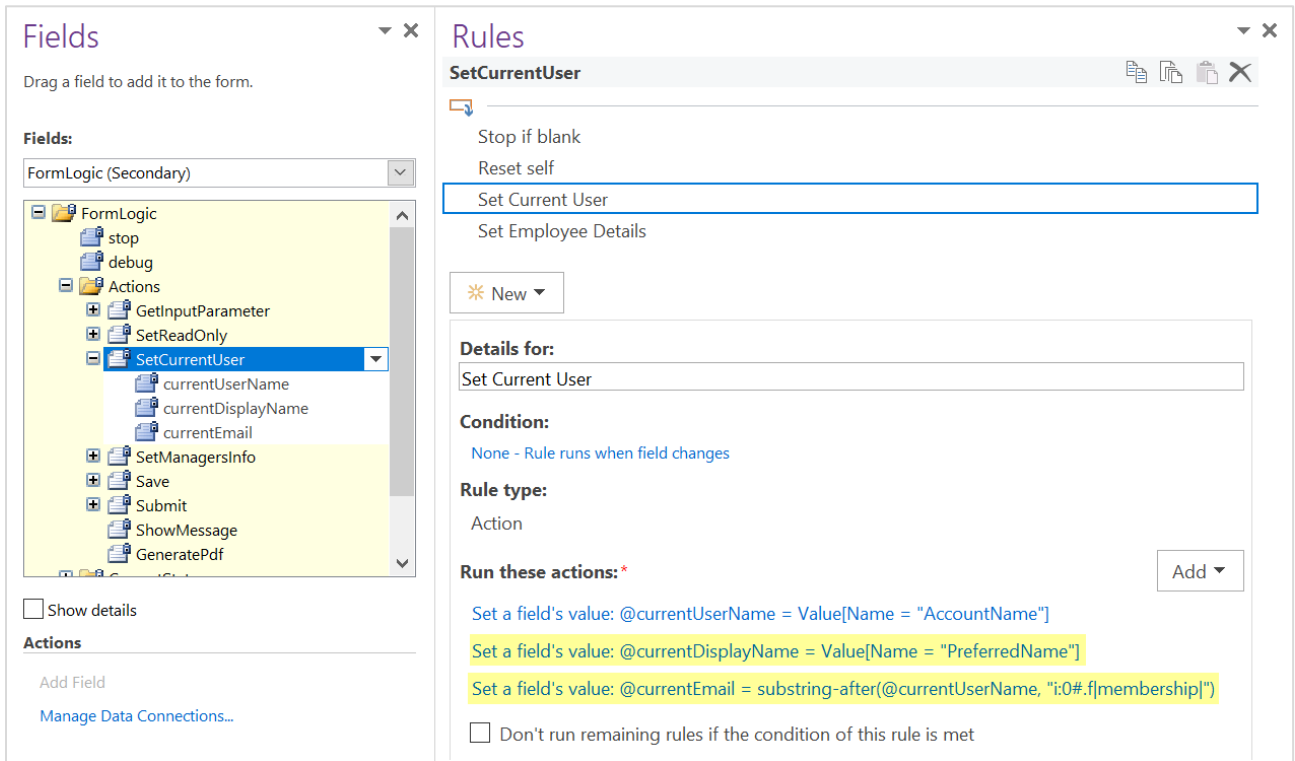


3. Navigate to /XtpLogic/Actions/Connections/GetUserProfileByName/@dataConnectionName. Here you can see that the GetUserProfileByName command is used.



The GetUserProfileByName, when used without any parameters, returns the current user's profile data.

The current user's name is set by getting the PreferredName property value in the user's profile information. To get the current user's email, the substring-after function is used for the AccountName property. These are all being set in /Util/Actions/SetCurrentUser node.



**Fields**

Drag a field to add it to the form.

**Fields:**

FormLogic (Secondary)

- FormLogic
  - stop
  - debug
  - Actions
    - GetInputParameter
    - SetReadOnly
    - SetCurrentUser**
      - currentUserName
      - currentDisplayName
      - currentEmail
    - SetManagersInfo
    - Save
    - Submit
    - ShowMessage
    - GeneratePdf

☐ Show details

**Actions**

Add Field

Manage Data Connections...

**Rules**

**SetCurrentUser**

- Stop if blank
- Reset self
- Set Current User**
- Set Employee Details

New

**Details for:**

Set Current User

**Condition:**

None - Rule runs when field changes

**Rule type:**

Action

**Run these actions:**

- Set a field's value: @currentUserName = Value[Name = "AccountName"]
- Set a field's value: @currentDisplayName = Value[Name = "PreferredName"]
- Set a field's value: @currentEmail = substring-after(@currentUserName, "i:0#.f|membership|")

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

These data in turn will be used to set the Employee details in the Main data source, which is done in the next rule, **Set Employee Details**:

### Fields

Drag a field to add it to the form.

**Fields:**

FormLogic (Secondary)

- FormLogic
  - stop
  - debug
  - Actions
    - GetInputParameter
    - SetReadOnly
    - SetCurrentUser**
      - currentUserName
      - currentDisplayName
      - currentEmail
    - SetManagersInfo
    - Save
    - Submit
    - ShowMessage
    - GeneratePdf

☐ Show details

**Actions**

Add Field

[Manage Data Connections...](#)

### Rules

**SetCurrentUser**

Stop if blank

Reset self

Set Current User

**Set Employee Details**

New

**Details for:**

Set Employee Details

**Condition:**

Name is blank

**Rule type:**

Action

**Run these actions:**

Set a field's value: Name = @currentDisplayName

Set a field's value: EmailAddress = @currentEmail

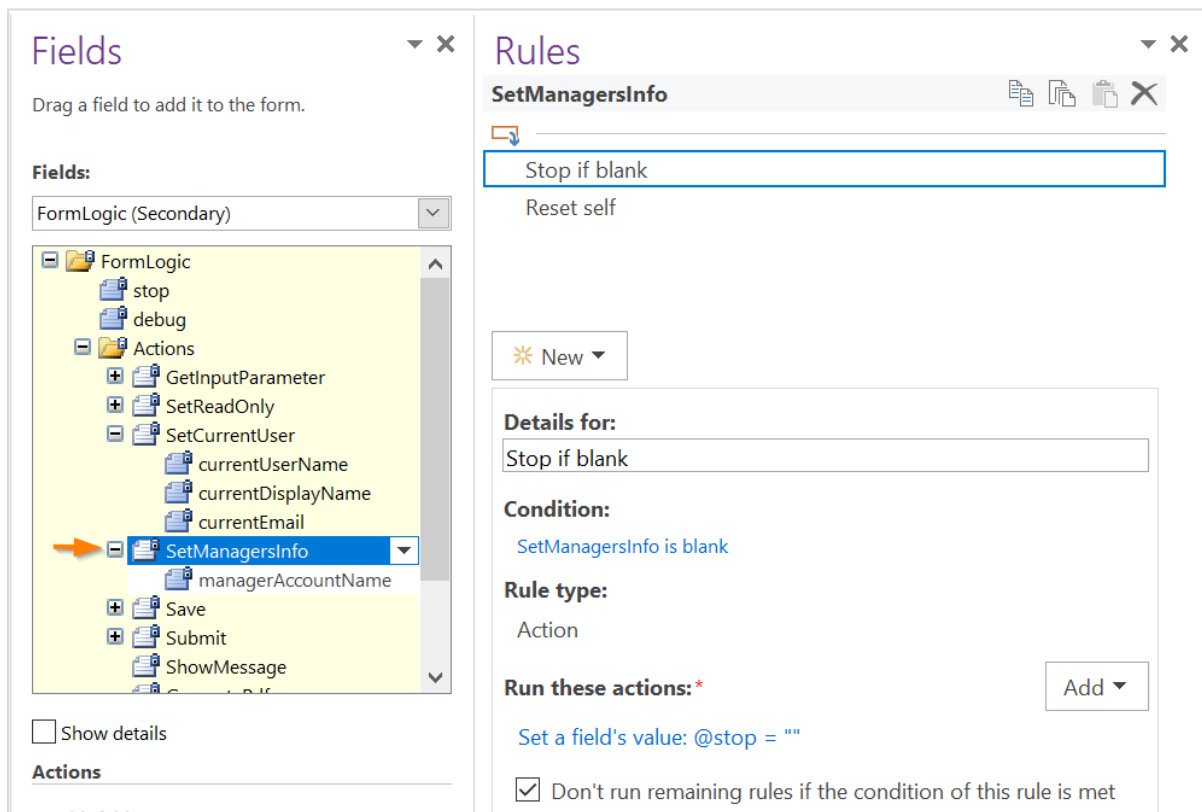
☐ Don't run remaining rules if the condition of this rule is met

Add

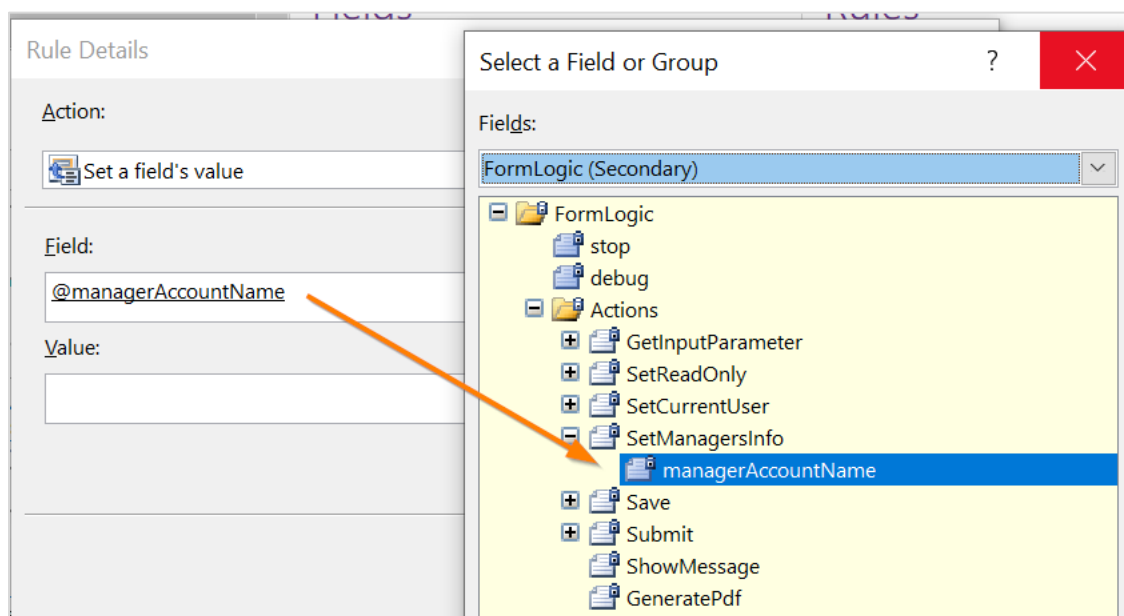
Now that we've seen how the Employee details are being set, we can do the same approach to set the Manager details.

## Get the Manager's Profile Data

1. In the Fields pane, select FormLogic secondary data source and navigate to /FormLogic/Actions/SetManagersInfo node.



2. Add a new rule and call it "Get Manager Profile".
3. Add a new action that sets the field **@managerAccountName** (attribute field under /FormLogic/Actions/SetManagersInfo/@managerAccountName).



- Click the *fx* button beside the **Value** field.

Rule Details

Action:  
Set a field's value

Field:  
@managerAccountName

Value:

fx

OK Cancel

- Click **Insert Field or Group** button.

Insert Formula

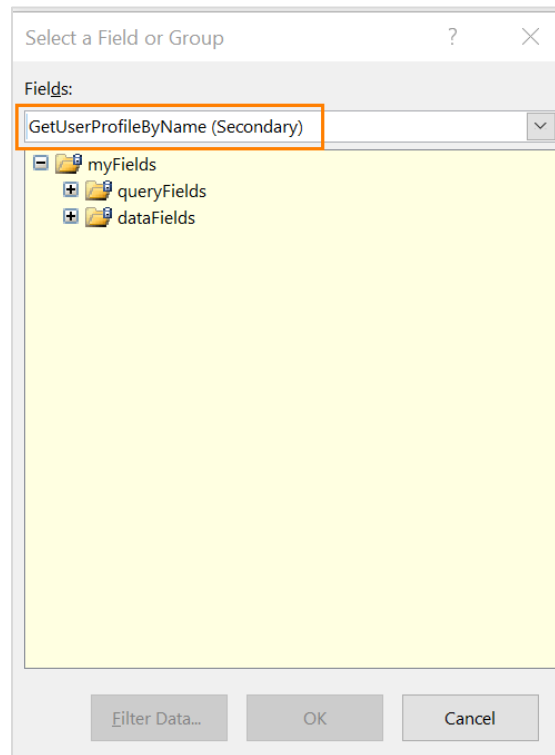
Formula:

Insert Field or Group... Insert Function... Verify Formula

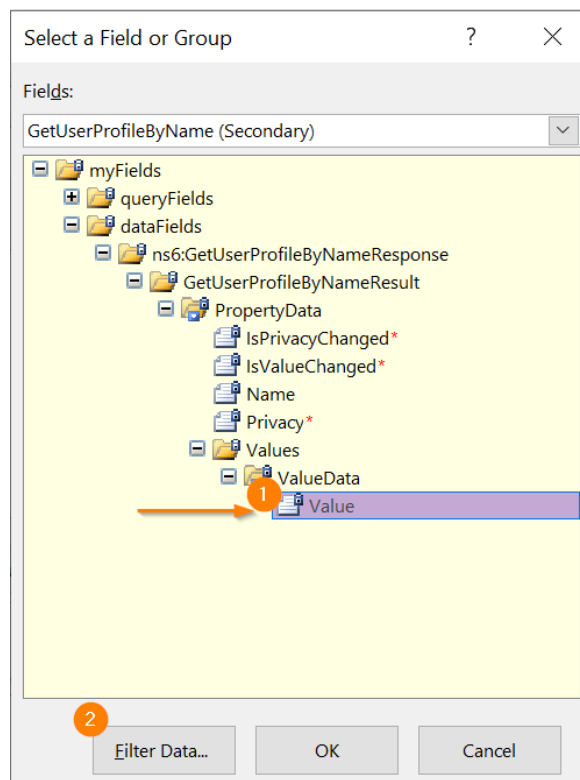
☐ Edit XPath (advanced)

OK Cancel

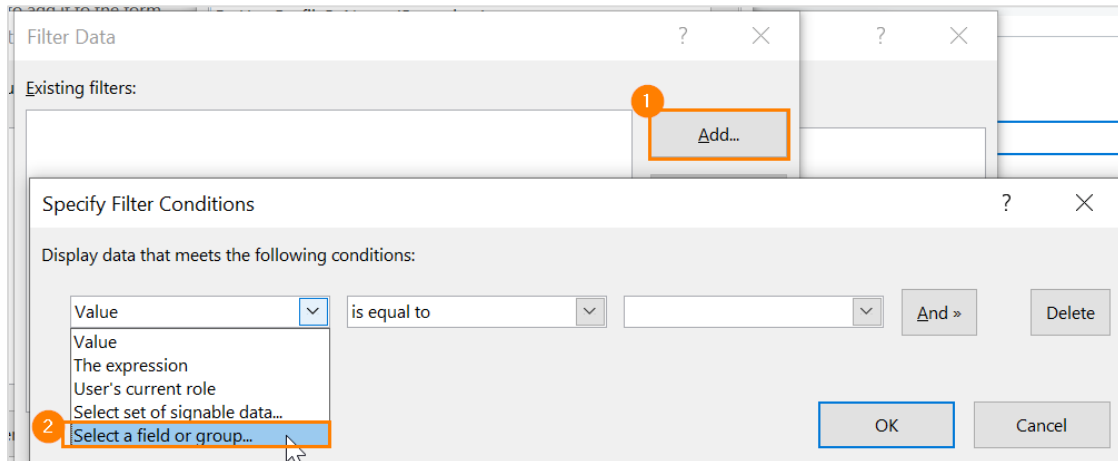
- In the dropdown, select the GetUserProfileByName secondary data source.



7. Expand **dataFields** and locate the **Value** node (under myFields/dataFields/GetUserProfileByNameResponse/GetUserProfileByNameResult/PropertyData/Values/ValueData). Then click **Filter Data**.

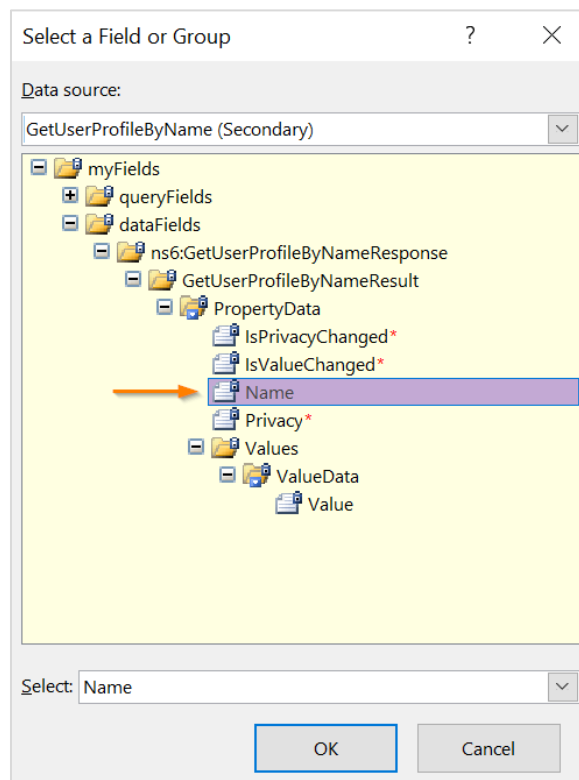


- In the Filter Data window, click **Add**. In the Filter condition, select from the first dropdown and choose **Select a field or group...**

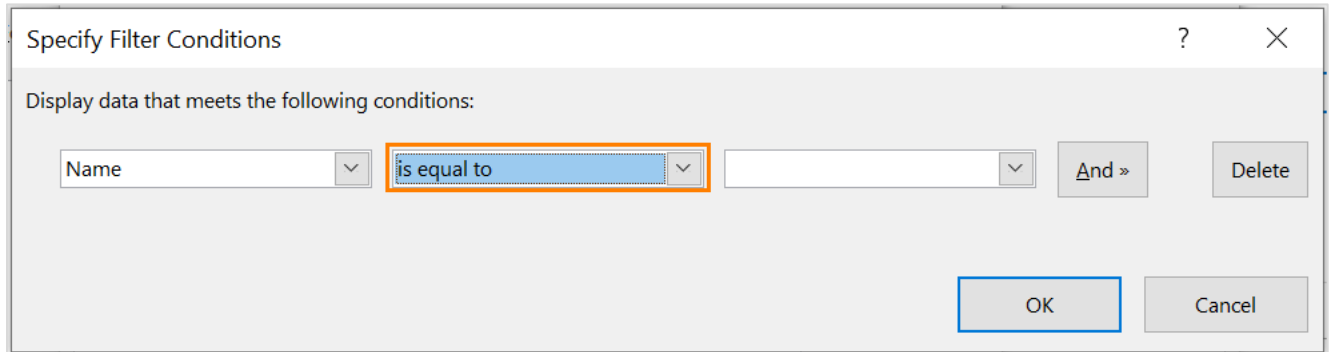


- Select the **Name** field in the GetUserProfileByName secondary data source. Then click **OK**.

The Name field here refers to the property name. Here are some of the user properties we can retrieve from SharePoint: AccountName, PreferredName, Department, Email, Manager. As long as these properties have value in SharePoint's user profile, we can retrieve the data by using the GetUserProfileByName command.

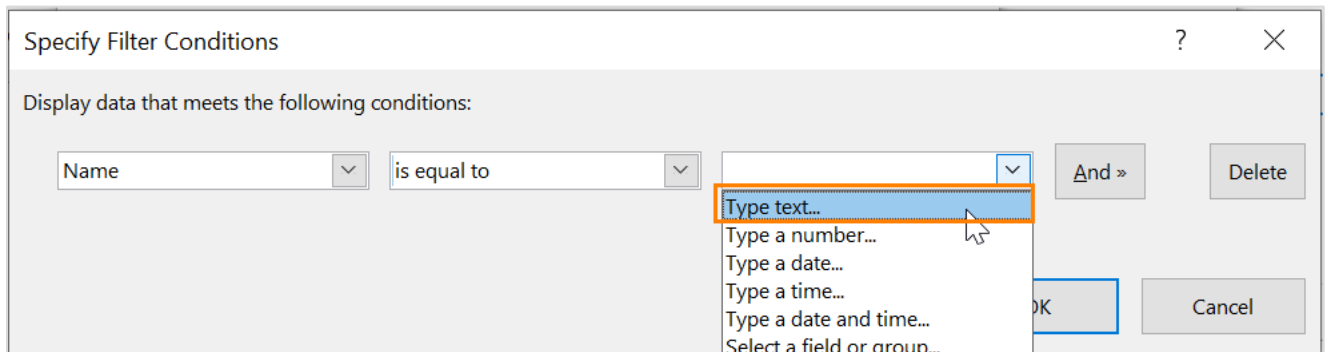


10. Leave the value **is equal to** in the second dropdown.

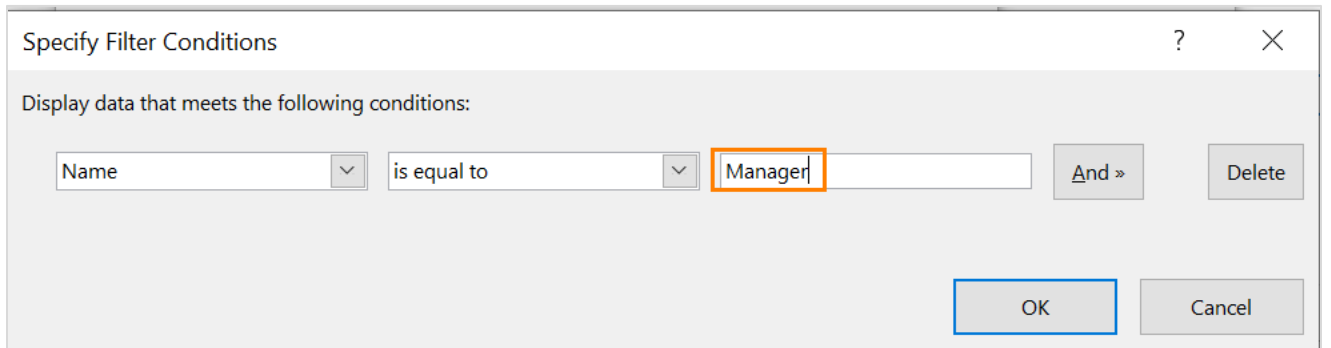


The dialog box titled "Specify Filter Conditions" shows the condition "Name" followed by a dropdown menu set to "is equal to". The "is equal to" option is highlighted with an orange box. Other options visible include "And »", "Delete", "OK", and "Cancel".

11. In the third dropdown, select **Type text...** and type **Manager**.



The dialog box shows the dropdown menu for the third field open. The "Type text..." option is highlighted with an orange box. Other options in the menu include "Type a number...", "Type a date...", "Type a time...", "Type a date and time...", and "Select a field or group...".



The dialog box shows the text "Manager" entered into the third field. The text is highlighted with an orange box. The "OK" button is visible at the bottom right.

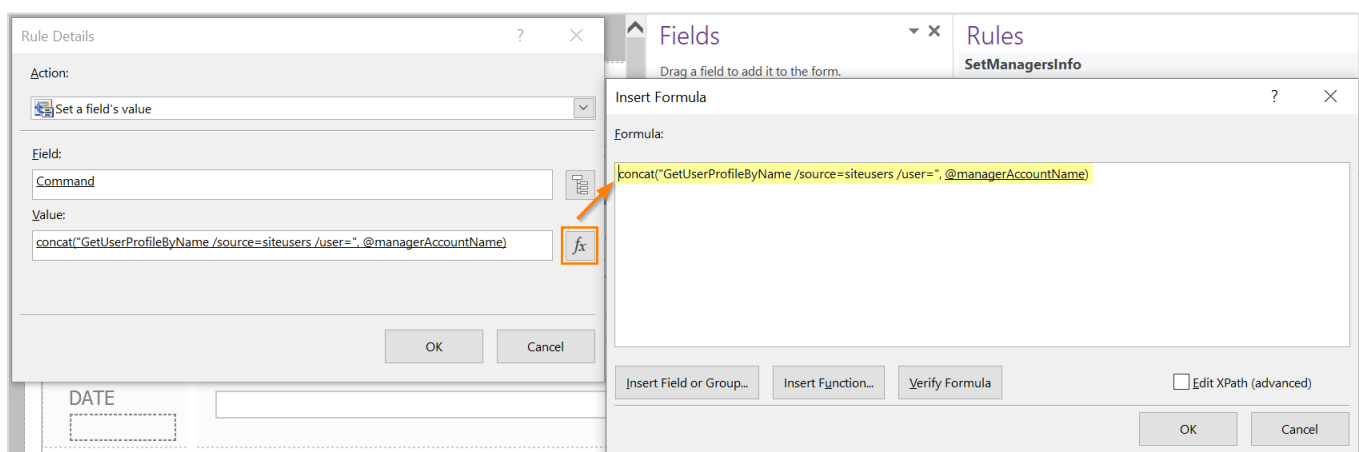
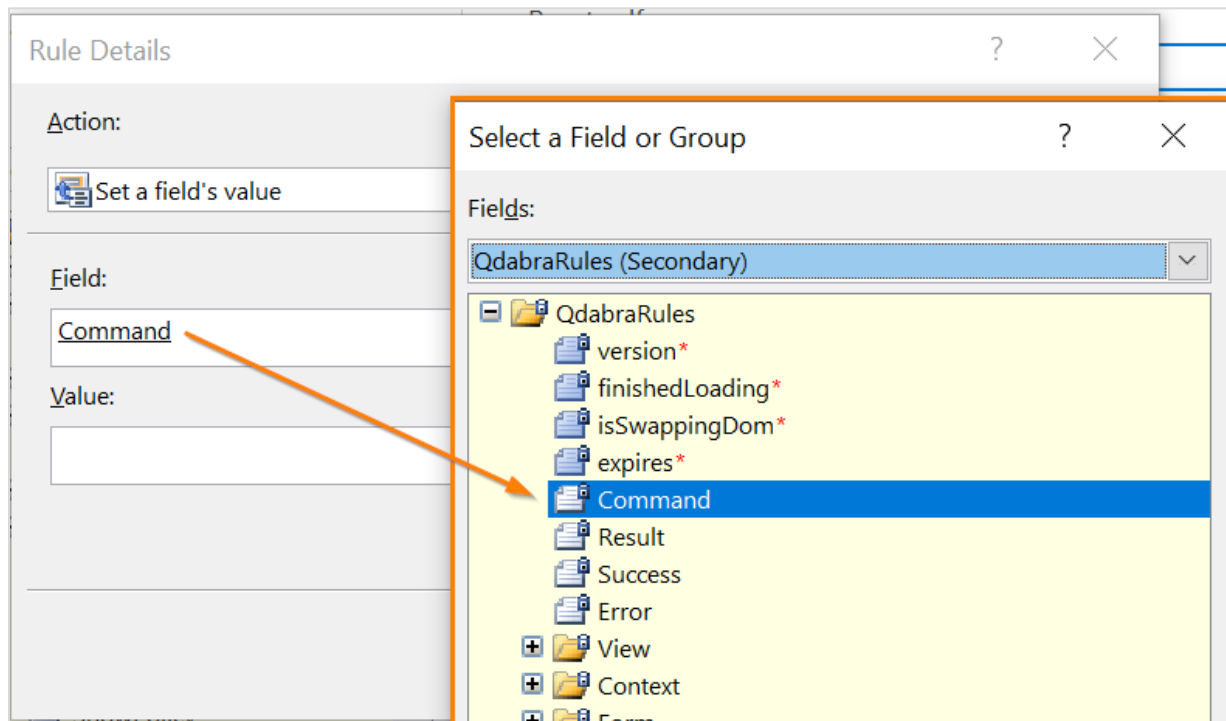
12. Click **OK** to close all the opened windows.

13. Add a new action to use the GetUserProfileByName to retrieve the manager's profile data. To do this, set the QdabraRules Command node with the following value:

**concat("GetUserProfileByName /source=siteusers /user=", @managerAccountName)**

where **@managerAccountName** is

/FormLogic/Actions/SetManagersInfo/@managerAccountName field



14. Click **OK** to close the windows. So far these are the actions added:

### Fields

Drag a field to add it to the form.

**Fields:**

FormLogic (Secondary)

- FormLogic
  - stop
  - debug
  - Actions
    - GetInputParameter
    - SetReadOnly
    - SetCurrentUser
      - currentUserName
      - currentDisplayName
      - currentEmail
    - SetManagersInfo**
    - managerAccountName
  - Save
  - Submit
  - ShowMessage

☐ Show details

**Actions**

Add Field

[Manage Data Connections...](#)

### Rules

**SetManagersInfo**

- Stop if blank
- Reset self
- Get Manager Profile**

New

**Details for:**

Get Manager Profile

**Condition:**

None - Rule runs when field changes

**Rule type:**

Action

**Run these actions:**

Add

- Set a field's value: @managerAccountName = Value[Name = "Manager"]
- Set a field's value: Command = concat("GetUserProfileByName /source=siteusers /user=", @managerAccountName)

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

## Set the Manager's Details

In this section, we're going to set the Manager's fields in the Main data source – the Manager's name and Manager's email address.

1. Back to the **SetManagersInfo** node under FormLogic secondary data source, add a new rule to set the manager's details. Let's call the rule **Set Managers Details**.
2. Add a condition if **ManagerName** is blank.

### Fields

Drag a field to add it to the form.

**Fields:**

FormLogic (Secondary)

FormLogic

stop
debug
Actions
GetInputParameter
SetReadOnly
SetCurrentUser
currentUserName
currentDisplayName
currentEmail
SetManagersInfo
managerAccountName
Save
Submit
ShowMessage

☐ Show details

**Actions**

### Rules

#### SetManagersInfo

Stop if blank
Reset self
Get Manager Profile
Set Managers Details

New

**Details for:**

Set Managers Details

**Condition:**

ManagerName is blank

**Rule type:**

Action

**Run these actions: \***

None

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

- In this rule, add a new action to set the ManagerName with the value when the GetUserProfileByName's **Name** field is equal to **PreferredName**. You can follow the same steps we did in numbers 5-11 but instead of typing in **Manager**, use **PreferredName** instead

Rule Details

Action:

Set a field's value

Field:

ManagerName

Value:

Value[Name = "PreferredName"]

OK

Cancel

4. Add another action to set the ManagerEmailAddress with the value when the GetUserProfileByName's Name field is equal to **Email**.

The 'Rule Details' dialog box is shown with the following configuration:

- Action:** Set a field's value
- Field:** ManagerEmailAddress
- Value:** Value[Name = "Email"]

Buttons: OK, Cancel

Here's how the rule should look after adding the two actions:

The screenshot shows the QDABRA interface with the 'SetManagersInfo' rule selected in the 'Rules' pane. The 'Fields' pane on the left shows the 'FormLogic (Secondary)' folder expanded, with 'SetManagersInfo' selected. The 'Rules' pane shows the 'SetManagersInfo' rule with the following details:

- SetManagersInfo**
- Details for:** Set Managers Details
- Condition:** ManagerName is blank
- Rule type:** Action
- Run these actions:**
  - Set a field's value: ManagerName = Value[Name = "PreferredName"]
  - Set a field's value: ManagerEmailAddress = Value[Name = "Email"]
- ☐ Don't run remaining rules if the condition of this rule is met

- For the third action, we'll set the values for the AP manager. Since there's no actual property in SharePoint's user profile where we can retrieve AP manager's info, let's just use the same data we used for the manager details. Let's copy the "Set Managers Details" rule and rename it to "Set AP Manager Details", as seen in the image below. You can find the **Copy Rule** and **Paste Rule** buttons in the Rules task pane.

### Fields

Drag a field to add it to the form.

**Fields:**

FormLogic (Secondary)

FormLogic

stop

debug

Actions

GetInputParameter

SetReadOnly

SetCurrentUser

SetManagersInfo

Save

Submit

ShowMessage

GeneratePdf

CurrentState

Validation

☐ Show details

**Actions**

Add Field

Manage Data Connections...

### Rules

SetManagersInfo

Stop if blank

Reset self

Get Manager Profile

Set Managers Details

Set AP Manager Details

New

**Details for:**

Set AP Manager Details

**Condition:**

ApName is blank

**Rule type:**

Action

**Run these actions: \***

Set a field's value: ApName = Value[Name = "PreferredName"]

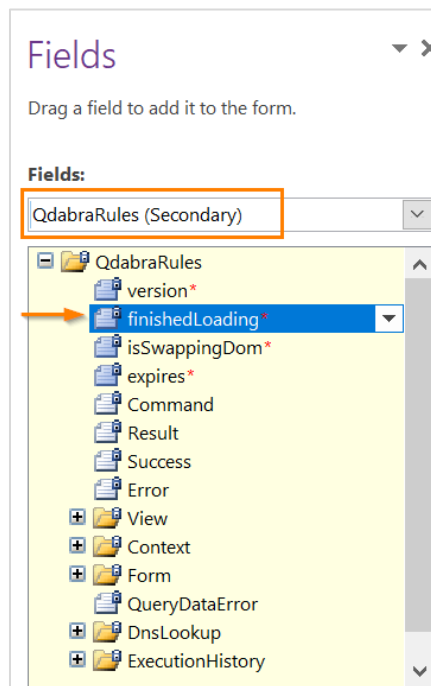
Set a field's value: ApEmail = Value[Name = "Email"]

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

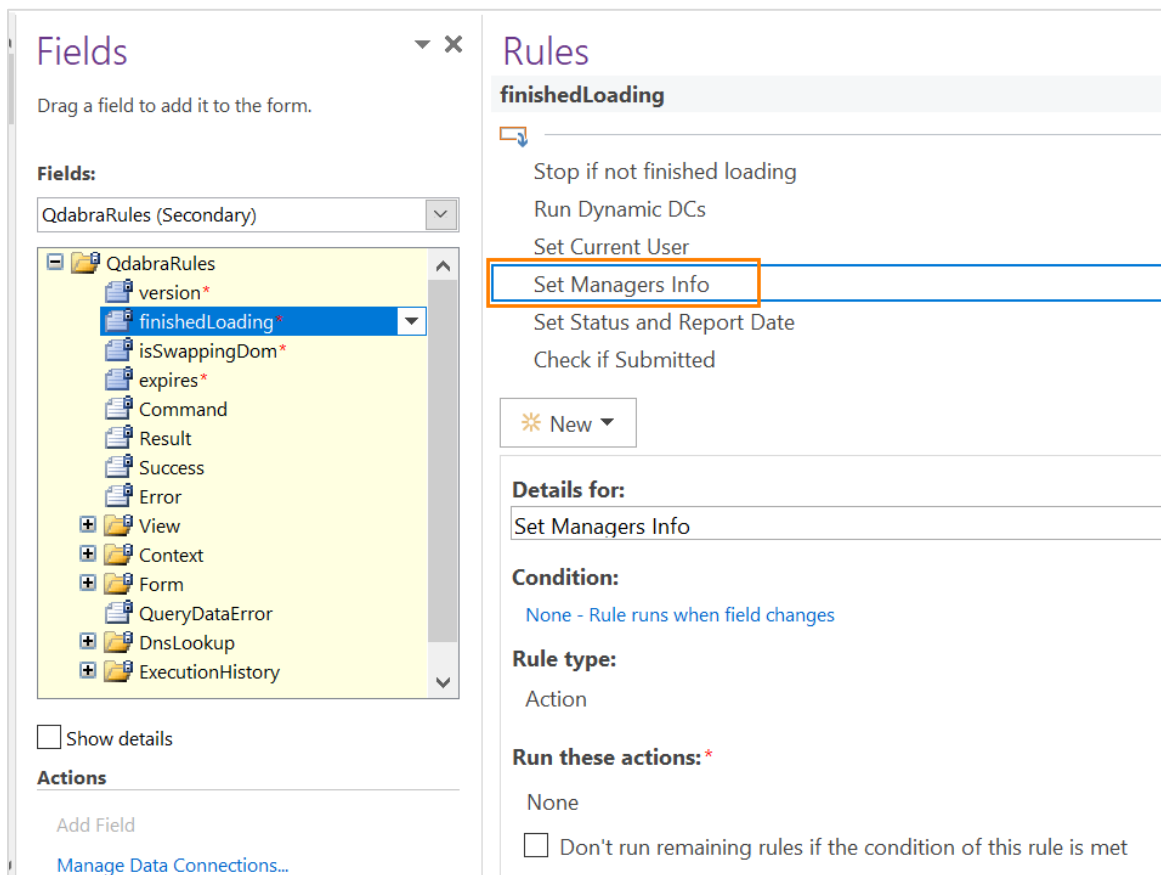
## Add Trigger on Form Load

Now, since we would like to display the employee and managers' details when the form opens, we need to make sure that the rules we created will be executed on form load. To do this:

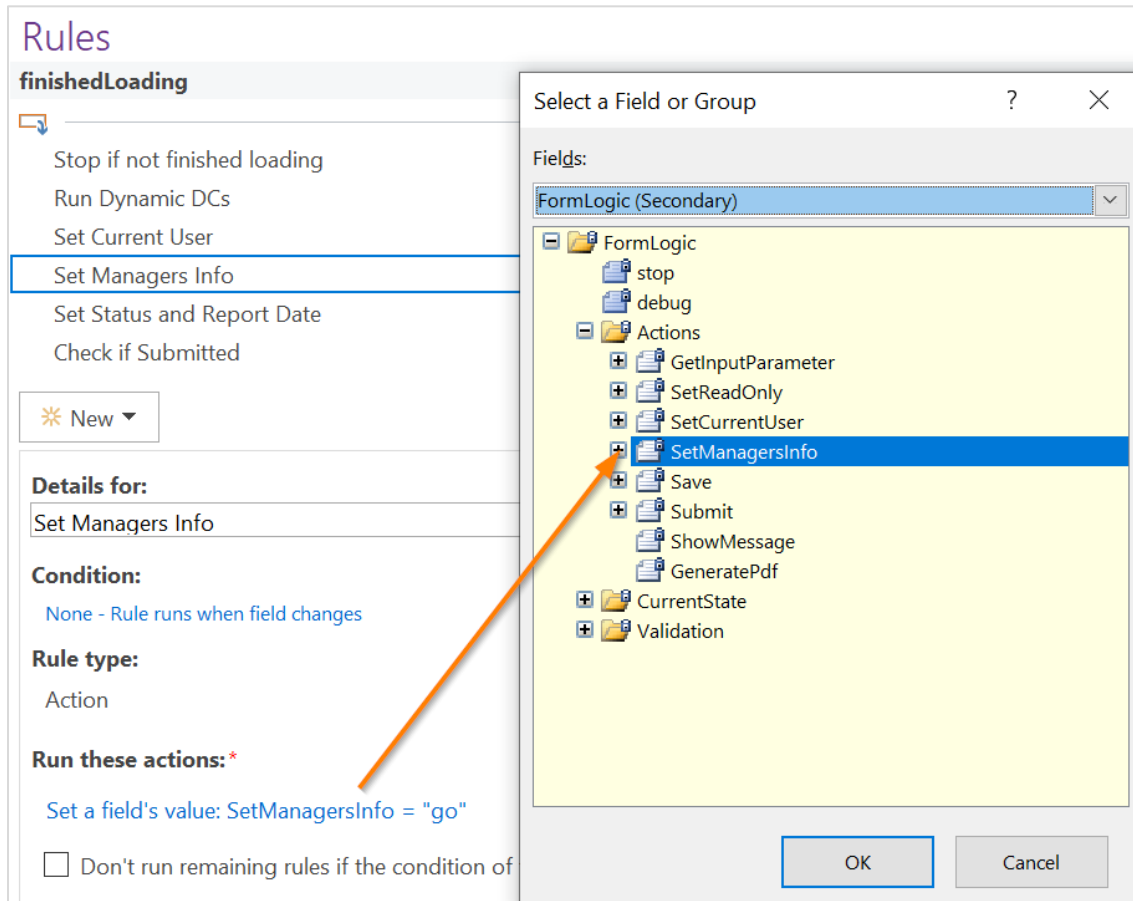
- In the Fields pane, select QdabraRules secondary data source and select the **finishedLoading** node. All rules that are expected to run when the form opens should be placed in this node.



2. Create a new rule under **Set Current User** and call it **Set Managers Info**.



3. Within the rule, add an action that sets the SetManagersInfo node to "go". You can find the node in FormLogic secondary data source, under /Util/Actions.



## Test the Form

1. Upload the updated Expense Report form to FormsViewer. You may also try to upload the end form provided in Lab 3 → *Lab 3 – GetUserProfileByName End.xsn*
2. Open a new Expense Report Form. The Employee, Manager, and Accounting Manager names should be displayed as seen in the image below:


[Close](#)

## EXPENSE REPORT

|                       |                             |                                        |
|-----------------------|-----------------------------|----------------------------------------|
| NAME<br>Kervi Laborte | MANAGER<br>Patrick Halstead | ACCOUNTING MANAGER<br>Patrick Halstead |
|-----------------------|-----------------------------|----------------------------------------|

REPORT DATE  
4/8/2020

BUSINESS PURPOSE\*

### EXPENSE ITEMS

FROM: TO:

| CATEGORY * | DATE * | DESCRIPTION | MERCHANT | RECEIPT     | AMOUNT * |
|------------|--------|-------------|----------|-------------|----------|
| Select...  |        |             |          | Attach file |          |

SUBTOTAL \$0.00

LESS CASH ADVANCE \$0.00

**TOTAL** \$0.00

Insert item

[Submit](#)

Solution Version: 1.0.1.1429

© 2020 Qdabra Software

You'll also see the email addresses being populated when you go to the debug section by clicking the Solution Version button at the bottom part of the form.

|                          |                        |                         |
|--------------------------|------------------------|-------------------------|
| Current User Name:       | Current Display Name:  | Current Email:          |
| i:0#.f membership ker... | Kervi Laborte          | kervi.laborte@qdabra... |
| Manager Name:            | Manager Email Address: |                         |
| Patrick Halstead         | Patrick.Halstead@Qd... |                         |

**TIP:** If you want to add more fields to the debug section, feel free to drag them from the Fields pane.

Congratulations! You have successfully added the GetUserProfileByName command in FormsViewer.

## Contact Information

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

**Phone:**

877.544.2389

**Email:**

[Support@Qdabra.com](mailto:Support@Qdabra.com)

**Website:**

<http://www.qdabra.com>

**Community:**

<http://www.infopathdev.com>