You

How do I, in power apps, get the name/id of a visual component in a formula box?

Copilot

In Power Apps, you can use named formulas to achieve this. Named formulas allow you to reuse app logic across different places. Here are a few use cases for named formulas:

1. **Pre-Filtering SharePoint Lists**: Suppose you have a SharePoint list of service vehicles with an "Active" column indicating whether a truck is in service or not. You want to show only active vehicles in your app. Create a named formula like this:

   ```
   // FILTER ACTIVE SERVICE VEHICLES
   fxServiceVehiclesActive = Filter('Service Vehicles', Active.Value = "Yes")
   ```

   Then use `fxServiceVehiclesActive` in the `Items` property of a gallery to display only active rows.

2. **Gathering Extended User Profile Data**: To retrieve user profile data (e.g., full account details) for the current app user, use the Office365Users connector. Create named formulas like this:

   ```
   // USER PROFILE INFO
   fxCurrentUser = Office365Users.MyProfile()
   fxCurrentUserPhoto = If(Office365Users.UserPhotoMetadata(fxCurrentUser.UserPrincipalName).HasPhoto, Office365Users.UserPhotoV2(fxCurrentUser.UserPrincipalNa
   ```

   These formulas prevent multiple calls to the Office365Users API and cache the result.

3. **Checking Security Roles For The Current User**: If you need to grant access to a hidden screen based on a custom security role (e.g., "Contoso Manager"), use named formulas to handle security roles.

Remember that named formulas recalculate each time they're referenced, ensuring the most up-to-date values. You can manage their definitions centrally, making your app logic more efficient and maintainable. 🚀👍

You

How can I use one to retrieve the name of a component in power apps?

Copilot

Certainly! To retrieve the name of a component in Power Apps, you can create a named formula that references the component's properties. Here's how you can do it:

1. **Create a Named Formula**:

   - Go to the "File" menu in Power Apps Studio.
   - Select "App settings."
   - Click on "Advanced settings."
   - Under "App formula," create a new named formula (e.g., `fxComponentName`).

2. **Reference the Component's Properties**:

   - In your named formula, reference the component whose name you want to retrieve. For example, if you want to get the name of a button, use the `Button1` control:

     ```
     // GET BUTTON NAME
     fxComponentName = Button1.Name
     ```

     Replace `Button1` with the actual name of your component.

3. **Use the Named Formula**:

   - Now you can use `fxComponentName` in other parts of your app. For instance, you can display the component's name in a label:

     ```
     TextLabel.Text = fxComponentName
     ```

Remember to adjust the formula based on the type of component you're working with (e.g., button, label, input, etc.). If you encounter any issues or need further assistance, feel free to ask! 😊👍

You

In power apps, you know how you can create a custom component that contains multiple buttons, text boxes, labels and more, how can I then retrieve the name of the custom component from a label within the custom component. Say the custom compenent's name is "Day_3" and I want to display "Day 3" in the text label within this custom com[ponent?>

Copilot

Certainly! To retrieve the name of a custom component in Power Apps, you can follow these steps:

1. **Create a Component Library**:

   - First, create a component library if you haven't already. A component library allows you to organize and reuse components across apps.
   - Go to **Power Apps** > **More** (left navigation) > **Discover all** > **Component libraries**.
   - Create a new component library (e.g., "Menu Components").

2. **Create Your Custom Component**:

   - Inside the component library, create your custom component (e.g., "Day_3").
   - Add the necessary buttons, text boxes, labels, etc., to your custom component.

3. **Retrieve the Component Name**:

   - In your custom component, add a label where you want to display the component name.
   - Set the label's **Text** property to the name of your custom component:

     ```
     TextLabel.Text = "Day 3"
     ```

     Replace `"Day 3"` with the actual name of your custom component.

4. **Use the Custom Component**:
    - Now, when you use your custom component in an app, the label within it will display the correct name (“Day 3”).

Remember that component libraries are a great way to manage reusable components, and they ensure consistency across your apps. If you have any more questions or need further assistance, feel free to ask! 😊👍.

You

How can I make it so that the text displayed changes if the custom component is renamed and alwauys shows the up to date name of the current component. The full scenario I’m aiming for is that new custom components are dynamically created with a new name e.g Day_3 tehn when another one is created the name is Day_4 and teh label should display this name accurately?

Copilot

To achieve this dynamic behavior where a label within a custom component always displays the up-to-date name of the component, you can follow these steps:

1. **Component Library**:
    - Create a component library (e.g., “Menu Components”) if you haven’t already. Component libraries allow you to organize and reuse components across apps.
    - Inside the component library, create your custom components (e.g., “Day_3,” “Day_4,” etc.).
2. **Custom Component Properties**:
    - In each custom component, add a label where you want to display the component name.
    - Set the label’s **Text** property to a formula that retrieves the component’s name dynamically. For example:

      ```
      TextLabel.Text = ThisItem.Name
      ```

      Here, `ThisItem` refers to the current component instance.
3. **Component Naming**:
    - When you create a new custom component (e.g., “Day_5”), the label within it will automatically display the correct name (“Day_5”).
    - If you rename the component, the label will update accordingly.

By following these steps, your custom components will always display their accurate names, even as new components are dynamically created. If you need further assistance or have any other questions, feel free to ask! 😊👍.

You

Yh but I tried that and the ThisItem variable that can be used to reference the component’s name doesn’t exist

Copilot

I apologize for any confusion. You’re right; my previous response was incorrect. In Power Apps, the `ThisItem` context variable is specific to galleries and data sources, not custom components. I appreciate your patience.

To dynamically display the name of a custom component within itself, you can use a combination of properties and formulas. Here’s an alternative approach:

1. **Component Properties**:
    - In your custom component, add a label where you want to display the component name.
    - Create a custom property (e.g., `ComponentName`) for your custom component. This property will hold the component’s name.
2. **Set the Custom Property**:
    - In the custom component, set the `ComponentName` property to the desired name (e.g., “Day_3” or “Day_4”).
3. **Label Text Property**:
    - Set the label’s **Text** property to reference the `ComponentName` property:

      ```
      TextLabel.Text = ThisComponent.ComponentName
      ```

      Replace `ThisComponent` with the actual name of your custom component.
4. **Dynamic Updates**:
    - Now, when you create a new instance of your custom component, set its `ComponentName` property to the appropriate name.
    - The label within each instance will display the correct name.

Remember to adjust the formula based on your specific component structure. If you have any further questions or need additional assistance, feel free to ask! 😊👍.