You

In power apps, I’m trying to use a named formula to count how many values are true from a list of variables, the true and false values of each of teh original varaibles is shown and the logic behind this named fomula as well as it’s output. As you can see, all the variables are false except day2Visible and the output of teh named formula is 9. How can I fix this?

Copilot

It looks like there’s an issue with the way the `CountIf` function is being used in your named formula. The formula you have is:

```
CountVisibleDays = CountIf(day1Visible, day2Visible, day3Visible, day4Visible, day5Visible, day6Visible, day7Visible, day8Visible, true);
```

This syntax is incorrect for counting the number of `true` values. The `CountIf` function in Power Apps is designed to count records in a table that meet a specified condition.

To fix this, you should use the `CountIf` function with a logical formula that evaluates each variable separately. Here’s how you can do it:

1. **Create a collection or table** that includes all your variables.
2. **Use the `CountIf` function** to count the `true` values in that collection or table.

Here’s an example of how you can do this:

1. **Create a collection** with your variables:

   ```
   ClearCollect(VisibleDays, {day: day1Visible}, {day: day2Visible}, {day: day3Visible}, {day: day4Visible}, {day: day5Visible}, {day: day6Visible}, {day: day7
   ```

2. **Use the `CountIf` function** to count the `true` values:

   ```
   CountVisibleDays = CountIf(VisibleDays, day = true);
   ```

This approach ensures that each variable is evaluated correctly, and the `CountIf` function will count only the `true` values.

Give this a try and let me know if it resolves the issue! 😊

You

How would it work if the variables can be dynamically changed based on user actions during runtime?

Copilot

To handle dynamically changing variables based on user actions during runtime, you can use collections and update them as needed. Here’s a step-by-step approach:

1. **Initialize the Collection**: Create a collection to store the visibility status of each day.

   ```
   ClearCollect(VisibleDays, {day: "day1", visible: day1Visible}, {day: "day2", visible: day2Visible}, {day: "day3", visible: day3Visible}, {day: "day4", visib
   ```

2. **Update the Collection**: Whenever a user action changes the visibility of a day, update the corresponding item in the collection.

   For example, if `day3Visible` changes:

   ```
   UpdateIf(VisibleDays, day = "day3", {visible: day3Visible});
   ```

3. **Count the True Values**: Use the `CountIf` function to count the number of `true` values in the collection.

   ```
   CountVisibleDays = CountIf(VisibleDays, visible = true);
   ```

4. **Dynamic Updates**: Ensure that any user action that changes a visibility variable also triggers an update to the collection.

Here’s a more detailed example:

1. **Initialize the Collection**:

   ```
   ClearCollect(VisibleDays,
       {day: "day1", visible: day1Visible},
       {day: "day2", visible: day2Visible},
       {day: "day3", visible: day3Visible},
       {day: "day4", visible: day4Visible},
       {day: "day5", visible: day5Visible},
       {day: "day6", visible: day6Visible},
       {day: "day7", visible: day7Visible},
       {day: "day8", visible: day8Visible}
   );
   ```

2. **Update the Collection on User Action**:

   ```
   // Example: When day3Visible changes
   UpdateIf(VisibleDays, day = "day3", {visible: day3Visible});
   ```

3. **Count the True Values**:

   ```
   CountVisibleDays = CountIf(VisibleDays, visible = true);
   ```

By using this approach, you can dynamically track and count the visibility status of each day based on user interactions. This ensures that your count is always accurate and reflects the current state of the variables.

Feel free to try this out and let me know if you need any further assistance! 😊

You

This is my logic for my power app. This is what is in teh OnVisible method of the screen: ClearCollect(VisibleDays, {day: “day9”, visible: day9Visible}, {day: “day2”, visible: day2Visible}, {day: “day3”, visible: day3Visible}, {day: “day4”, visible: day4Visible}, {day: “day5”, visible: day5Visible}, {day: “day6”, visible: day6Visible}, {day: “day7”, visible: day7Visible}, {day: “day8”, visible: day8Visible}, {day: “day10”, visible: day10Visible}); This is in the Calculate button: Set(hours, If(ComboBox1.Selected.‘Choose Hours’ = “Per Day”, TextBox2.Value, TextBox2.Value / 5));

Set(totalHours, 0); Set(totalMinutes, 0); Set(CountVisibleDays, CountIf(VisibleDays, visible = true)); Set(days, CountVisibleDays);

// Convert time values to date/time format Set(convertedST, TimeValue(startTime1.Value)); Set(convertedET, TimeValue(endTime1.Value)); Set(convertedLS, TimeValue(lunchStart1.Value)); Set(convertedLE, TimeValue(lunchEnd1.Value));

// Calculate time differences Set(mainDiffHours, DateDiff(convertedST, convertedET, TimeUnit.Hours)); Set(lunchDiffMinutes, DateDiff(convertedLS, convertedLE, TimeUnit.Minutes));

Set(adjustedWorkHours, mainDiffHours - (lunchDiffMinutes / 60)); Set(workHours, Int(adjustedWorkHours)); Set(workMinutes, (adjustedWorkHours - workHours) * 60); Set(totalHours, totalHours + workHours); Set(totalMinutes, totalMinutes + workMinutes);

// Convert time values to date/time format Set(convertedST, TimeValue(Text(Day_11.day2StartTime, “[$-en-US]HH:mm”))); Set(convertedET, TimeValue(Text(Day_11.day2EndTime, “[$-en-US]HH:mm”))); Set(convertedLS, TimeValue(Text(Day_11.day2LunchStart, “[$-en-US]HH:mm”))); Set(convertedLE, TimeValue(Text(Day_11.day2LunchEnd, “[$-en-US]HH:mm”)));

// Calculate time differences Set(mainDiffHours, DateDiff(convertedST, convertedET, TimeUnit.Hours)); Set(lunchDiffMinutes, DateDiff(convertedLS, convertedLE, TimeUnit.Minutes));

Set(adjustedWorkHours, mainDiffHours - (lunchDiffMinutes / 60)); Set(workHours, Int(adjustedWorkHours)); Set(workMinutes, (adjustedWorkHours - workHours) * 60); Set(totalHours, totalHours + workHours); Set(totalMinutes, totalMinutes + workMinutes);

Set(averageHours, Int(totalHours / days)); This is in the output box: Concatenate("Your daily work hours should be: “, hours, " hours”, Char(10), "Your total hours for all ", days, " days are ", totalHours, " hours and “, totalMinutes, " minutes.”, Char(10), "Your average hours per day are: “, averageHours, " hours”) Also in the app there are buttons that can be used to add/remove a day and this is the logic inside them: If (day2Visible = false, Set(day2Visible, true), Set(day2Visible, false)); Update(VisibleDays, day = “day2”, {visible: day2Visible}); I want the output to show the total number of days that are visible, however it just shows this at all times: Your daily work hours should be: 8 hours Your total hours for all 1 days are 8 hours and 30 minutes. Your average hours per day are: 8 hours; How can I fix this logic error so that where it says 1 days it should say e.g. 2 days if 2 days are visible?

Copilot

It looks like the logic for updating the `VisibleDays` collection and counting the visible days is mostly correct, but there might be a few areas to check and adjust to ensure the count updates dynamically.

Here are a few steps to troubleshoot and fix the issue:

1. **Ensure the Collection is Updated Correctly**:

   - Make sure the `Update` function is correctly updating the `VisibleDays` collection whenever a day visibility changes.

   ```
   If(day2Visible = false, Set(day2Visible, true), Set(day2Visible, false));
   UpdateIf(VisibleDays, day = "day2", {visible: day2Visible});
   ```

2. **Recalculate the Count After Updates**:

   - Ensure that the count of visible days is recalculated after any update to the visibility of a day.

   ```
   Set(CountVisibleDays, CountIf(VisibleDays, visible = true));
   Set(days, CountVisibleDays);
   ```

3. **Check the Output Logic**:

   - Ensure the output logic correctly references the updated `days` variable.

   ```
   Concatenate(
       "Your daily work hours should be: ", hours, " hours", Char(10),
       "Your total hours for all ", days, " days are ", totalHours, " hours and ", totalMinutes, " minutes.", Char(10),
       "Your average hours per day are: ", averageHours, " hours"
   )
   ```

Here is a revised version of your logic with these adjustments:

1. **OnVisible Method**:

   ```
   ClearCollect(VisibleDays,
       {day: "day9", visible: day9Visible},
       {day: "day2", visible: day2Visible},
       {day: "day3", visible: day3Visible},
       {day: "day4", visible: day4Visible},
       {day: "day5", visible: day5Visible},
       {day: "day6", visible: day6Visible},
       {day: "day7", visible: day7Visible},
       {day: "day8", visible: day8Visible},
       {day: "day10", visible: day10Visible}
   );
   ```

2. **Calculate Button**:

   ```
   Set(hours,
       If(ComboBox1.Selected.'Choose Hours' = "Per Day",
           TextBox2.Value,
           TextBox2.Value / 5
       )
   );

   Set(totalHours, 0);
   Set(totalMinutes, 0);
   Set(CountVisibleDays, CountIf(VisibleDays, visible = true));
   Set(days, CountVisibleDays);

   // Convert time values to date/time format
   Set(convertedST, TimeValue(startTime1.Value));
   Set(convertedET, TimeValue(endTime1.Value));
   Set(convertedLS, TimeValue(lunchStart1.Value));
   Set(convertedLE, TimeValue(lunchEnd1.Value));

   // Calculate time differences
   Set(mainDiffHours, DateDiff(convertedST, convertedET, TimeUnit.Hours));
   Set(lunchDiffMinutes, DateDiff(convertedLS, convertedLE, TimeUnit.Minutes));

   Set(
       adjustedWorkHours,
       mainDiffHours - (lunchDiffMinutes / 60)
   );
   Set(workHours, Int(adjustedWorkHours));
   Set(workMinutes, (adjustedWorkHours - workHours) * 60);
   Set(totalHours, totalHours + workHours);
   Set(totalMinutes, totalMinutes + workMinutes);

   // Convert time values to date/time format
   Set(convertedST, TimeValue(Text(Day_11.day2StartTime, "[$-en-US]HH:mm")));
   ```

```
Set(convertedET, TimeValue(Text(Day_11.day2EndTime, "[$-en-US]HH:mm")));
Set(convertedLS, TimeValue(Text(Day_11.day2LunchStart, "[$-en-US]HH:mm")));
Set(convertedLE, TimeValue(Text(Day_11.day2LunchEnd, "[$-en-US]HH:mm")));

// Calculate time differences
Set(mainDiffHours, DateDiff(convertedST, convertedET, TimeUnit.Hours));
Set(lunchDiffMinutes, DateDiff(convertedLS, convertedLE, TimeUnit.Minutes));

Set(
    adjustedWorkHours,
    mainDiffHours - (lunchDiffMinutes / 60)
);
Set(workHours, Int(adjustedWorkHours));
Set(workMinutes, (adjustedWorkHours - workHours) * 60);
Set(totalHours, totalHours + workHours);
Set(totalMinutes, totalMinutes + workMinutes);

Set(averageHours, Int(totalHours / days));
```

3. **Output Box**:

```
Concatenate(
    "Your daily work hours should be: ", hours, " hours", Char(10),
    "Your total hours for all ", days, " days are ", totalHours, " hours and ", totalMinutes, " minutes.", Char(10),
    "Your average hours per day are: ", averageHours, " hours"
)
```

4. **Button to Add/Remove a Day**:

```
If(day2Visible = false, Set(day2Visible, true), Set(day2Visible, false));
UpdateIf(VisibleDays, day = "day2", {visible: day2Visible});
Set(CountVisibleDays, CountIf(VisibleDays, visible = true));
Set(days, CountVisibleDays);
```

By ensuring that the `CountVisibleDays` and `days` variables are updated immediately after any change to the visibility of a day, the output should correctly reflect the number of visible days.

Give this a try and let me know if it resolves the issue! 😊