

Usar controles y funciones para crear una fórmula dinámica

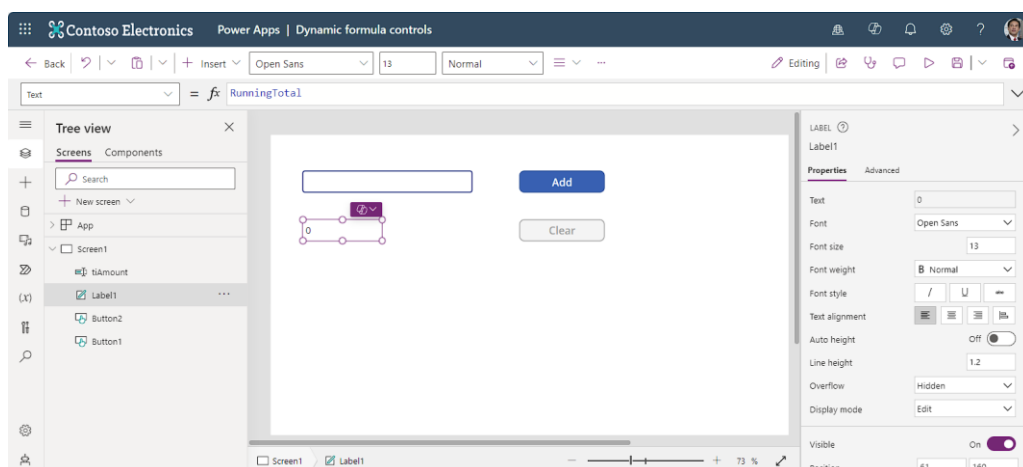
En el ejercicio siguiente se usa el control **Botón** y la función **UpdateContext** para crear una fórmula dinámica para cambiar la propiedad **DisplayMode** de un botón.

1. Vaya a [Power Apps](#).
2. Seleccione **+ Crear>Aplicación en blanco>Crear** (en Aplicación de lienzo en blanco). Asigne un nombre a su aplicación y seleccione **Crear**.
3. Seleccione la opción del menú **Insertar** y agregue un **Botón**.
4. Cambie la propiedad **Text** del botón a *Add*.

Sugerencia

Puede cambiar rápidamente la propiedad text de un control de botón si hace doble clic en el control mientras está en modo de edición.

5. Inserte otro control **Botón** y cambie la propiedad **Text** a *Clear*.
6. Agregue uno que sea **Etiqueta de texto** y otro **Entrada de texto** al lienzo.
7. Cambie el nombre del control de entrada de texto a *tiAmount* y, en la propiedad **Default**, elimine "Entrada de texto". Para Formato,** introduzca TextFormat.Number. (Consulte la imagen a continuación para ver la ubicación de los controles. La imagen muestra el ejercicio completado).



8. Seleccione el botón **Add** (Button1) y establezca la propiedad **OnSelect** en:

```
UpdateContext({RunningTotal: RunningTotal +  
Value(tiAmount.Text)})
```

9. Seleccione el botón **Clear** (Button2) y establezca la propiedad **OnSelect** en:

```
UpdateContext({RunningTotal: 0}); Reset (tiAmount)
```

10. Establezca la propiedad **DisplayMode** del botón de borrar en:

```
If(RunningTotal > 0, DisplayMode.Edit, DisplayMode.Disabled)
```

Tenga en cuenta que, después de haber actualizado la propiedad **DisplayMode**, esta aparece atenuada.

11. Seleccione la **Etiqueta** (Label1) y actualice la propiedad **Text** a nuestra variable de contexto: RunningTotal
12. Pruebe la aplicación en modo Versión preliminar. Escriba un número positivo en la entrada de texto y seleccione el botón **Add**.

En cuanto seleccione el botón **Add**, podrá seleccionar de nuevo el botón **Clear**.

13. Seleccione el botón **Clear** para borrar el campo de entrada y el RunningTotal.

En este ejemplo, hemos usado una función **If** para evaluar si el RunningTotal es mayor que 0. Si es mayor que 0, el botón **Clear** se vuelve editable (DisplayMode.Edit). Si la variable no tiene ningún valor o es 0, nuestro botón se deshabilita (DisplayMode.disabled).

Otra forma de lograr la misma funcionalidad es evitar la variable de contexto en las fórmulas. En este caso, podría poner de nombre a un control en la fórmula DisplayMode. Por ejemplo, podríamos usar el valor de nuestro control de etiqueta en la fórmula.

14. Vuelva a poner la aplicación en el modo de edición y seleccione el botón **Clear**. Seleccione la propiedad DisplayMode y cambie la fórmula a lo siguiente:

```
If(Value(Label1.Text) > 0, DisplayMode.Edit,  
DisplayMode.Disabled)
```

Dado que nuestro control de etiqueta almacena información como texto, no se olvide de utilizar la función **Value**. La función **Value** convierte una cadena numérica de texto en un número. Si intenta evaluar el texto con respecto a un valor, verá un error en la fórmula. Ajustar la propiedad text de la etiqueta en una función Value garantiza que Power Apps la interprete como un número.