

# Introducción a las funciones de Power Apps

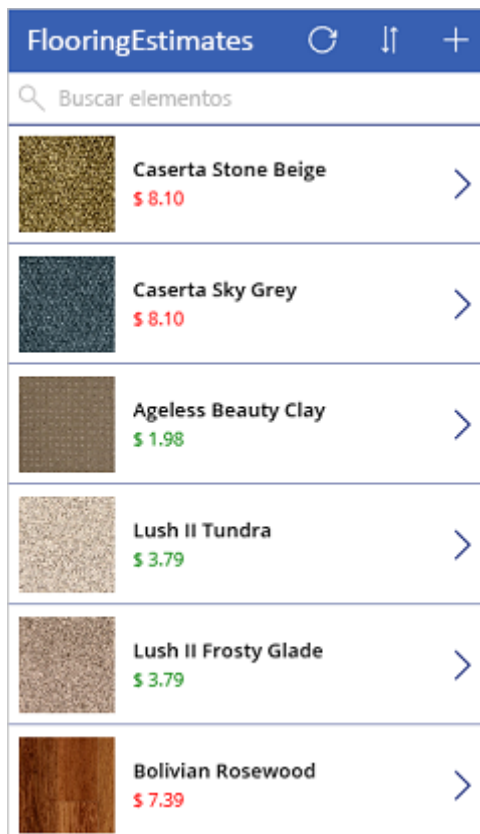
Cuando usa Microsoft Power Apps, no es necesario que escriba código de aplicación complicado como si fuese un desarrollador profesional. No obstante, debe expresar lógica en una aplicación y controlar su navegación, filtrado, ordenación y otras funcionalidades. Aquí es donde entran en juego las fórmulas.

Si ha usado funciones de Microsoft Excel, el enfoque que sigue Power Apps debería resultarle familiar. En esta unidad se muestran un par de fórmulas básicas para dar formato al texto y después se describen tres de las fórmulas que Power Apps incluye cuando genera una aplicación. Con esta información, comprenderá mejor lo que pueden hacer las fórmulas y podrá empezar a escribir las suyas por su cuenta.

## Introducción a las fórmulas y propiedades

En la unidad anterior ha descubierto los controles de las tres pantallas de una aplicación generada por Power Apps. En esta sección, modificará las propiedades del control de etiqueta para dar formato al precio del producto en la galería.

De forma predeterminada, el precio aparece como un número sin formato y sin símbolo de moneda. Suponga que desea agregar un símbolo de dólar y cambiar el color del texto según el coste del producto. Por ejemplo, rojo si es más de 5 \$, pero verde en caso contrario. En el gráfico siguiente se muestra el resultado esperado.

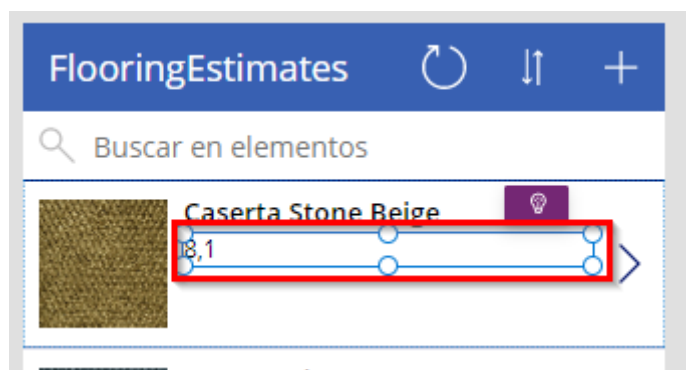


De forma predeterminada, Power Apps inserta un valor de precio para cada elemento. Este valor se establece como la propiedad **Text** de la etiqueta que muestra el precio.

### Nota

Asegúrese de completar los pasos en la unidad 1 de este módulo ya que el campo de descripción cambia a precio, tal y como se refleja en los siguientes pasos.

1. En **BrowseScreen1**, seleccione el precio del primer producto. En su **Vista de árbol**, es el control de etiqueta dentro de **BrowseGallery1** llamado **Subtitle1**.



2. En la lista desplegable de propiedades, seleccione **Text** (la propiedad predeterminada para un control de etiqueta).
3. Para agregar el símbolo de moneda de dólares estadounidenses, establezca la propiedad **Text** en esta fórmula:

```
Text(ThisItem.Price, "$ ##.00")
```

### Nota

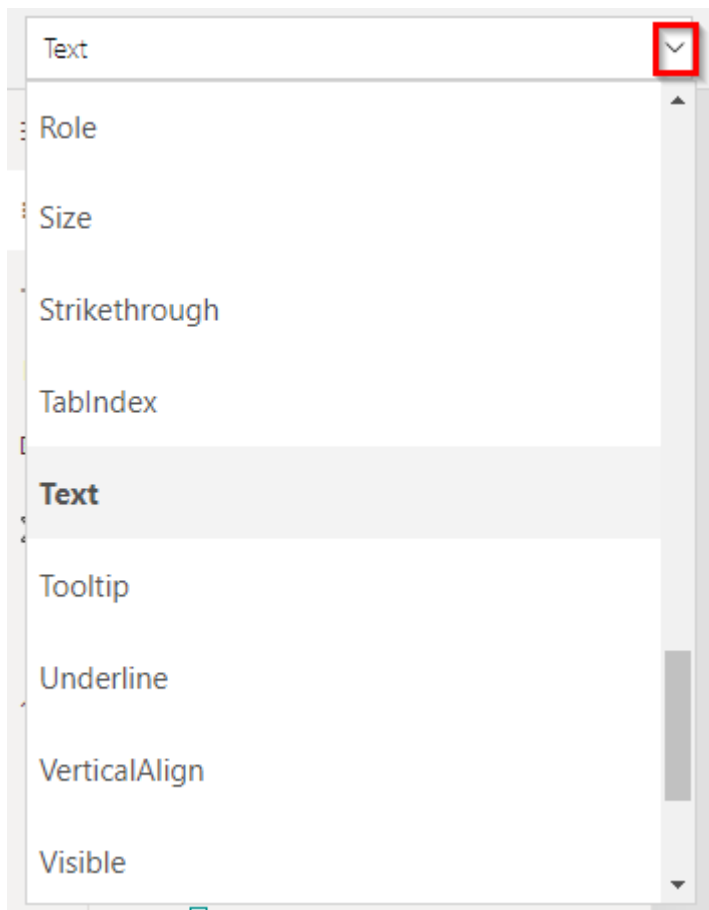
Si la fórmula devuelve un error, tenga en cuenta que la configuración de idioma del entorno de Power Apps puede afectar a algunos separadores y operadores. Por ejemplo, la fórmula anterior se expresa en un idioma y una región que utiliza un punto como separador decimal, como Japón o el Reino Unido.

Sin embargo, para esta misma fórmula en un idioma y una región donde se usa una coma como separador decimal, como Francia o España, la fórmula debería ser: `Text(ThisItem.Price; "$ ##,00")`

El operador de selección de propiedad (punto) en `ThisItem.Price` es siempre el mismo, sin importar cuál sea el separador decimal, pero observe que el separador decimal y el separador de la operación de encadenamiento han cambiado a coma y punto y coma respectivamente. Internamente, la fórmula no cambia, lo que cambia es cómo la muestra y edita el autor.

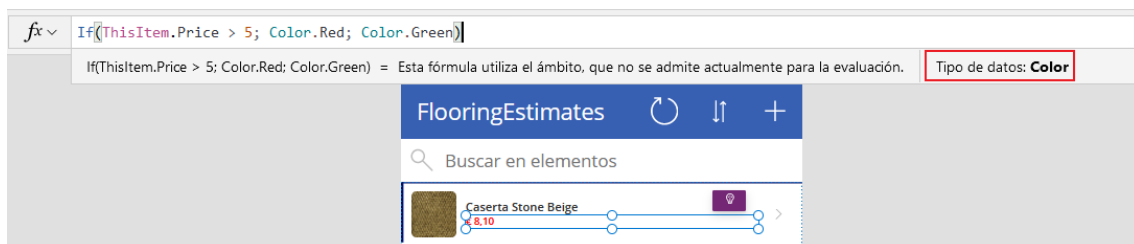
La función **Text** especifica cómo dar formato al número. La fórmula es como una función de Excel, pero las fórmulas de Power Apps hacen referencia a controles y otros elementos de una aplicación en lugar de hacerlo a las celdas de un libro.

Si selecciona un control y luego abre la lista desplegable de propiedades, aparecerá una lista de propiedades que son pertinentes para el control. Por ejemplo, esta es una lista parcial de las propiedades de un control **Etiqueta**. Algunas de ellas son pertinentes en un amplio conjunto de controles, pero otras solo lo son para un control específico.



La propiedad **Color** de un control de etiqueta afecta al color del texto. Los colores se pueden representar como nombres de color RGBA, HEX o HTML. En nuestra aplicación Flooring Estimates, formateemos condicionalmente el color del precio. Seleccione la etiqueta de precio (o **Subtitle1**), elija la propiedad **Color** y configúrela con esta fórmula:

`If(ThisItem.Price > 5, Color.Red, Color.Green)`



Verá que el color del texto era originalmente un valor RGBA equivalente al color negro; lo hemos cambiado a colores con nombre (Color.Red y Color.Green) según la condición de nuestra fórmula. Para que la fórmula funcione, Power Apps espera un valor que sea un color. Siempre que su fórmula proporcione datos de color como salida, puede agregar cualquier fórmula funcional como entrada.

## Fórmulas incluidas en la aplicación generada

Power Apps incluye fórmulas en cada aplicación que genera, así que exploremos dos de ellas. Ambos ejemplos proceden de la pantalla de navegación y funcionan con la propiedad **OnSelect**, la cual define lo que sucede cuando un usuario selecciona un control.

- La primera fórmula está asociada al control **IconNewItem1** icono Nuevo elemento (+). Cuando un usuario selecciona este control, la aplicación abre la pantalla editar/crear, donde puede crear un elemento. Para ver la fórmula, seleccione el icono de Nuevo elemento (+) y busque la propiedad **OnSelect** en la barra de fórmulas. Coloque el cursor en la barra de fórmulas, verá que la fórmula es la siguiente:

```
NewForm(EditForm1);Navigate(EditScreen1, ScreenTransition.None)
```

La fórmula crea una instancia de un formulario de edición en la pantalla editar/crear para que los usuarios puedan crear un elemento. El valor de `ScreenTransition.None` significa que no hay transición entre pantallas (por ejemplo, un fundido).

- La segunda fórmula está asociada con el control **IconSortUpDown1** (el botón con flechas hacia arriba y hacia abajo). El usuario selecciona este control para ordenar los elementos de la galería. Con la aplicación en modo de edición, continúe y selecciónela. La fórmula es la siguiente:

```
UpdateContext({SortDescending1: !SortDescending1})
```

La fórmula usa `UpdateContext` para actualizar una variable denominada `SortDescending1`. El signo de admiración "!" de la fórmula es un acceso directo a la función *Not*. El valor de la variable oscila cuando se selecciona el control. Esta variable le dice a la galería en esta pantalla cómo ordenar los elementos, ya sea en orden descendente o al revés.

Esta aplicación contiene muchas otras fórmulas, por lo que se recomienda invertir un tiempo en seleccionar diferentes controles y descubrir las fórmulas que están establecidas actualmente para las distintas propiedades.