

Microsoft



E-BOOK

POWER APPS

Expert




VISCARI
Inc.

SUMÁRIO

SEJA BEM-VINDO AO POWER APPS EXPERT.....	3
CRIAR UM APLICATIVO DE TELA COM DADOS DE UM ARQUIVO DO EXCEL	3
PERSONALIZAR UMA GALERIA NO POWER APPS	7
PERSONALIZAR FORMULÁRIOS EM APLICATIVOS DE TELA	15
CRIAR UM APLICATIVO DE TELA DO ZERO COM BASE NOS DADOS DO EXCEL	18
ADICIONAR E CONFIGURAR CONTROLES EM APLICATIVOS DE TELA	26
ADICIONAR UMA CAIXA DE LISTAGEM, UMA CAIXA DE OPÇÃO OU BOTÕES DE OPÇÃO A UM APLICATIVO DE TELA	35
MOSTRAR DADOS EM UM GRÁFICO DE LINHAS, DE PIZZA OU DE BARRAS EM APLICATIVOS DE TELA	
41	
COMO USAR ARQUIVOS MULTIMÍDIAS EM APLICATIVOS DE TELA	46
INTRODUÇÃO ÀS FÓRMULAS EM APLICATIVOS DE TELA	47
ENTENDER VARIÁVEIS EM APLICATIVOS DE TELA	54
SALVAR E PUBLICAR APLICATIVOS DE TELA	59



Microsoft POWER APPS



SEJA BEM-VINDO AO POWER APPS EXPERT

O E-book Power Apps Expert tem por objetivo servir como material de apoio para as videoaulas fazer você dominar o Microsoft Power Apps do básico ao avançado, independentemente da sua formação ou área de atuação, estamos falando do programa que mais vem ganhando espaço no mundo corporativo quando o assunto é criação de aplicativos para celular e tablets, sem nenhuma necessidade de conhecimento em linguagem de programação.

O QUE É O POWER APPS?

Ele é um software desenvolvido pela Microsoft que permite a criação de aplicativos de negócios personalizados que se conectam aos seus dados armazenados em uma plataforma à parte ou de forma on-line. Com ele, você irá precisar de poucos códigos para fazer sua criação, possibilitando uma modernização dos processos e soluções de desafios complexos nunca antes vista.

Você poderá criar apps avançados sem limitações, o que antes só era possível com desenvolvedores profissionais. Além disso, os aplicativos criados com o Power Apps contam com um design responsivo e podem ser executados continuamente em um navegador ou em dispositivos móveis (celulares ou tablets).

CRIAR UM APPLICATIVO DE TELA COM DADOS DE UM ARQUIVO DO EXCEL

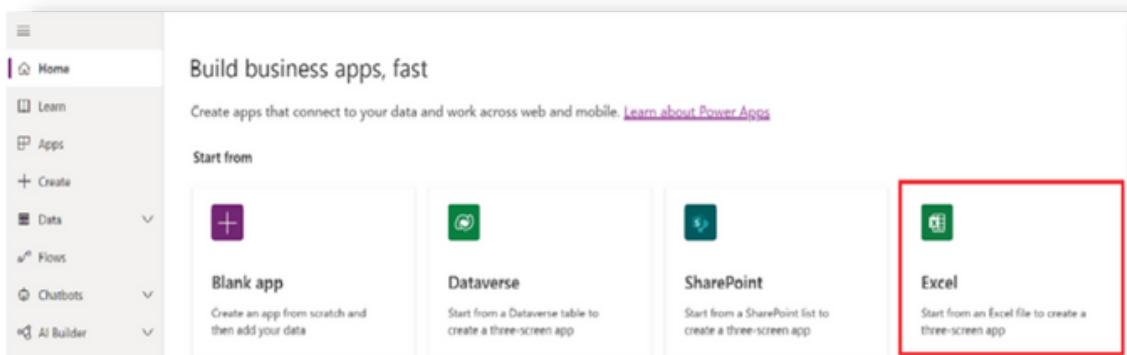
Neste tópico, você criará automaticamente seu primeiro aplicativo de tela no Power Apps usando os dados de uma tabela do Excel. Você selecionará um arquivo do Excel, criará um aplicativo e, em seguida, executará o aplicativo criado. Cada aplicativo criado inclui telas para procurar registros, mostrar detalhes do registro e criar ou atualizar registros. Gerando um aplicativo, você pode obter rapidamente um aplicativo funcional usando os dados do Excel e, em seguida, personalizar o aplicativo para atender melhor às suas necessidades.

O arquivo do Excel precisa estar em uma conta de armazenamento em nuvem, como o OneDrive, Google Drive ou Dropbox. Este tópico usa o OneDrive for Business.

Criando o aplicativo:

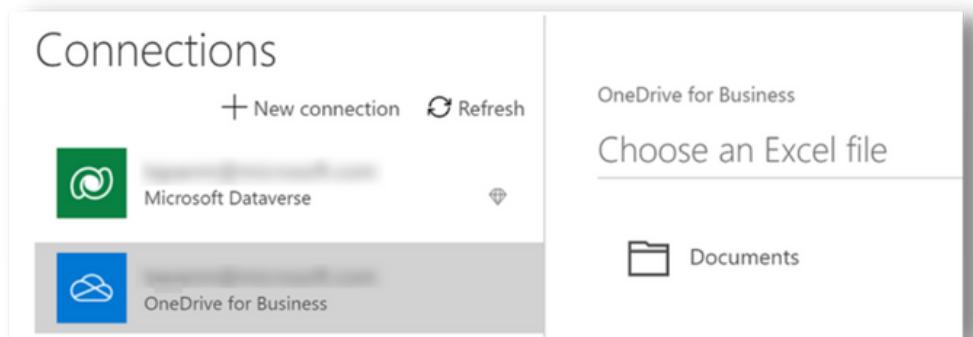
Entre no Power Apps.

Em **Iniciar Em**, selecione **Excel**.



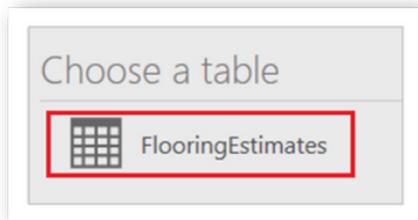
Se você ainda não tem uma conexão do **OneDrive for Business**, será solicitado que crie uma. Selecione **Criar** para criar a conexão.

Selecione a conexão do **OneDrive for Business**.



Navegue até o local onde você tem o arquivo do Excel —**FlooringEstimates.xlsx** neste exemplo.

Em **Escolher uma tabela**, selecione a tabela —**FlooringEstimates** neste exemplo e, depois, **Conectar**.



Executando o aplicativo:

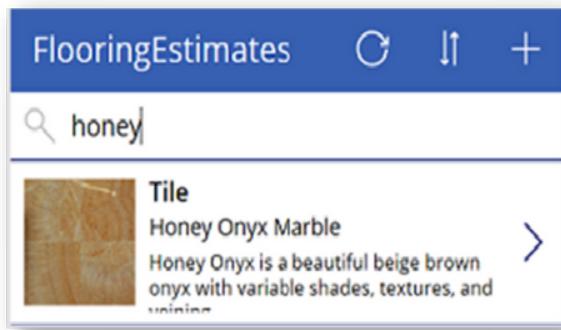
Abra a **Visualização** pressionando **F5** (ou clicando ou tocando no ícone de reprodução próximo ao canto superior direito).



Alterne a ordem de classificação clicando ou tocando no ícone de classificação próximo ao canto superior direito.

Carpet	Caserta Stone Beige	Extreme Series 18 in. x 18 in. carpet tiles are a durable and beautiful carpet >
Carpet	Caserta Sky Grey	Extreme Series 18 in. x 18 in. carpet tiles are a durable and beautiful carpet >
Carpet	Ageless Beauty Clay	Add some fashion to your floors with the Shaw Ageless Beauty Carpet collection. >
Carpet	Lush II Tundra	Made with 100% premium nylon fiber, this textured carpet creates a warm, >

Filtre a lista digitando ou colando um ou mais caracteres na barra de pesquisa. Por exemplo, digite ou cole **MeI** para mostrar o único registro para o qual essa sequência de caracteres aparece no nome, categoria ou visão geral do produto.



Adicione um registro:

Selecione o ícone de adição.



Adicione os dados desejados e selecione o ícone de marca de seleção para salvar suas alterações.



Edite um registro:

Selecione a seta para o registro que deseja editar.



Selecione o ícone de lápis.



Atualize um ou mais campos e selecione o **ícone de marca de seleção** para salvar suas alterações.



Exclua um registro:

Selecione a **seta** para avançar do registro que deseja excluir.



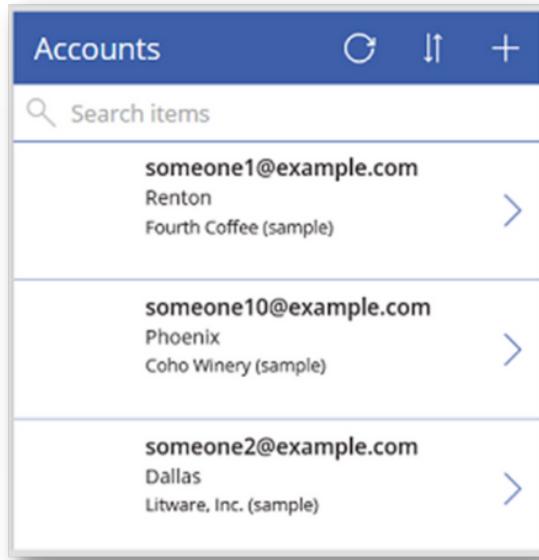
Selecione o ícone de lixeira.



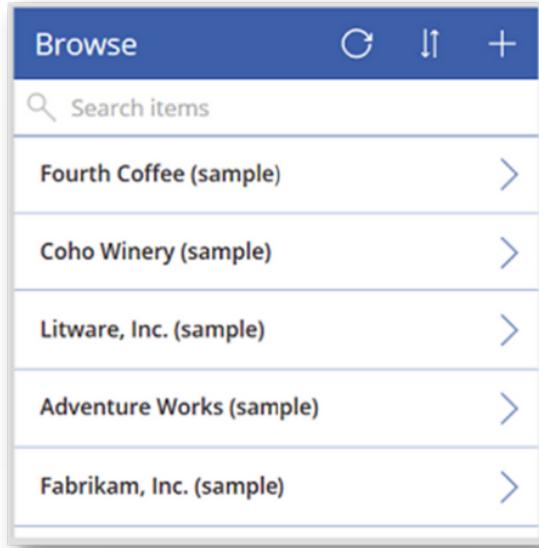
PERSONALIZAR UMA GALERIA NO POWER APPS

Neste tutorial, você personalizará uma lista de registros, chamada de galeria, e fará outras alterações em um aplicativo que foi gerado automaticamente no Microsoft Power Apps. Os usuários poderão gerenciar dados no aplicativo mesmo se você não fizer essas alterações, mas será mais fácil usar o aplicativo se você o personalizar de acordo com as necessidades da organização.

Por exemplo, a galeria deste tutorial corresponde a este gráfico por padrão. O endereço de email fica mais visível do que os outros tipos de dados e os usuários podem classificar e filtrar a galeria com base no texto neste endereço:



No entanto, talvez o nome da conta seja mais interessante para os seus usuários do que o endereço de email e, neste caso, você poderá reconfigurar a galeria para realçar, classificar e filtrar com base nos dados mais importantes para sua organização. Além disso, você poderá alterar o título da tela padrão para diferenciá-la das outras telas no aplicativo.



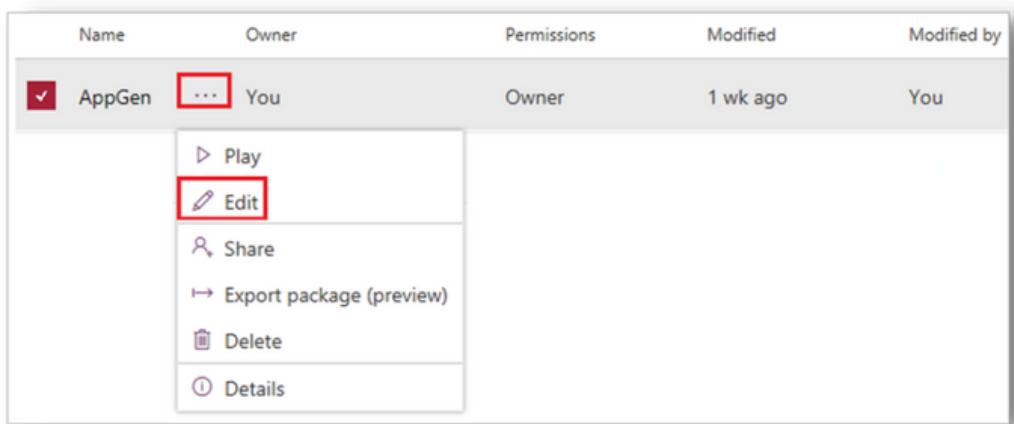
Você também adicionará uma barra de rolagem para que usuários que não tenham tela touch ou botão de rolagem do mouse possam procurar na galeria inteira.

- Alterar o layout da galeria
- Alterar o tipo de dados exibidos na galeria
- Alterar as colunas por meio das quais os usuários podem classificar e pesquisar os dados
- Alterar o título da tela
- Mostrar uma barra de rolagem

Este tutorial começa com um aplicativo que foi gerado de uma fonte de dados específica. No entanto, os mesmos conceitos aplicam-se a qualquer aplicativo que você gera no Power Apps, seja por uma lista criada com Listas da Microsoft, uma biblioteca do SharePoint, uma tabela do Excel ou alguma outra fonte de dados.

Abrir o aplicativo gerado:

Entre no Power Apps e então selecione **Aplicativos**, próximo à borda esquerda. Encontre o aplicativo que você gerou, selecione o ícone de reticências (...) dele e, em seguida, selecione **Editar**.

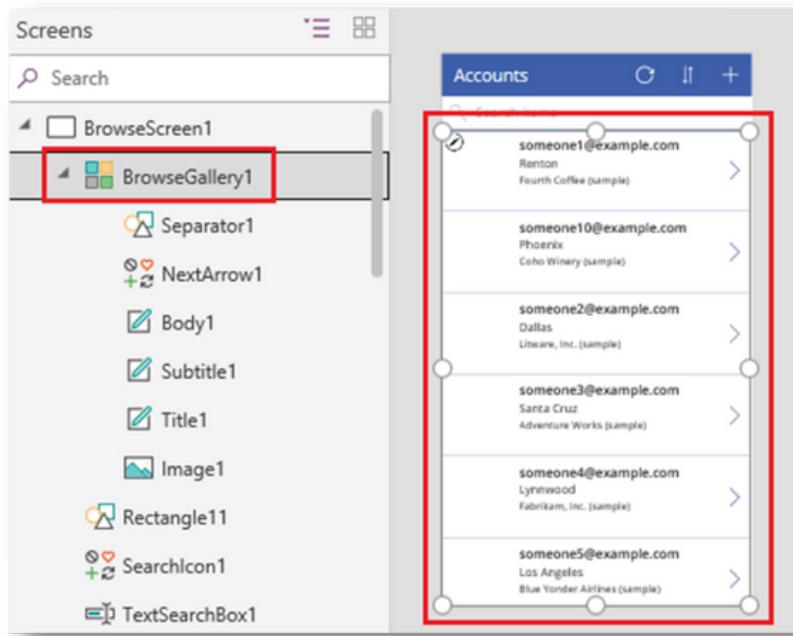


Se a caixa de diálogo **Bem-vindo ao Power Apps Studio** for exibida, selecione **Ignorar**.

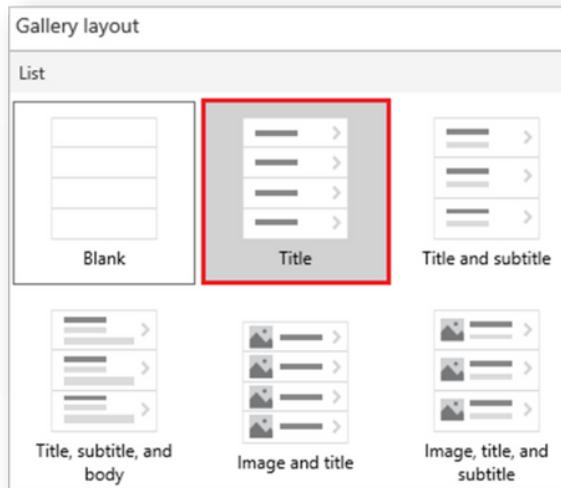
Altere o layout:

No painel de navegação esquerdo, selecione **BrowseGallery1**.

Quando a galeria for selecionada, uma caixa de seleção com alças a contornará.



Na guia **Propriedades** do painel direito, abra a lista de opções em **Layout** e selecione a opção que mostra apenas um título.



Ao lado de **Campos**, selecione **Editar** e selecione a seta para baixo para a caixa de título.

O nome deste controle terminará em um numeral, como **Title1**, mas o numeral poderá se diferenciar com base em outras ações que você possa ter realizado.

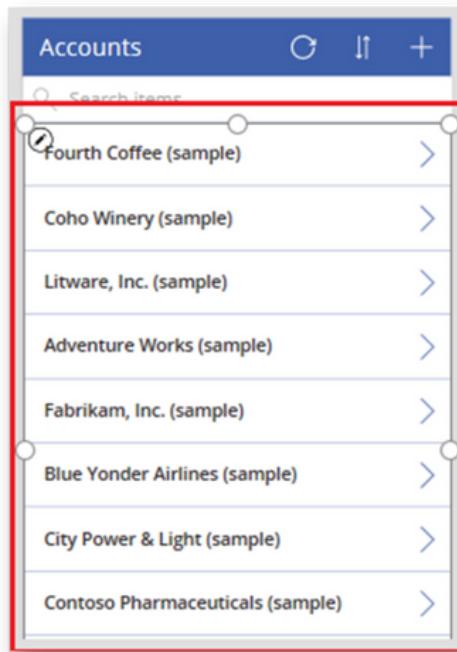
Na lista de opções, selecione **Nome da conta** e, em seguida, feche o painel **Dados**.

A galeria mostra o nome de cada conta.

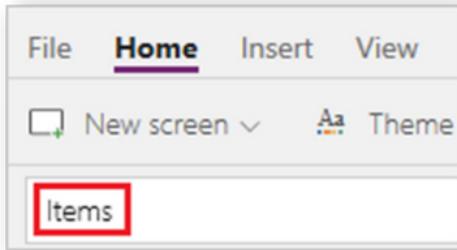


Alterar as colunas de classificação e pesquisa:

Selecione a galeria, conforme descrito na seção anterior.



Perto do canto superior esquerdo, confirme se a lista de propriedades mostra **Items**.



O valor dessa propriedade aparece na barra de fórmulas. Defina essa propriedade para especificar não somente a fonte de dados da galeria, mas também as colunas pelas quais os usuários podem classificar e pesquisar os dados.

Copie essa fórmula e cole-a na barra de fórmulas.

```
SortByColumns(Search(Accounts, TextBox1.Text, "name"), "name",  
If(SortDescending1, Descending, Ascending))
```

Ao usar esta fórmula, você garante que:

- Se um usuário digitar um ou mais caracteres na barra de pesquisa, a galeria mostrará apenas os nomes de conta que contêm o texto digitado pelo usuário.
- Se um usuário selecionar o ícone de classificação, a galeria será classificada em ordem alfabética pelo nome da conta, em ordem crescente ou decrescente, dependendo de quantas vezes o usuário seleciona o ícone.

Testar classificação e pesquisa:

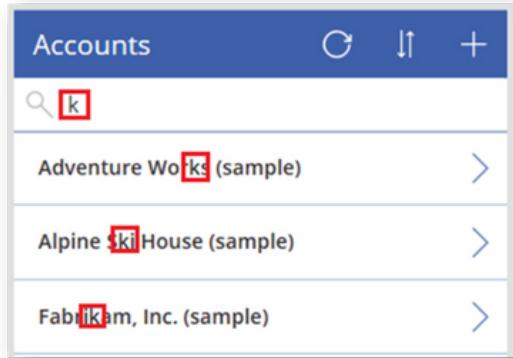
Abra o modo de **Visualização** pressionando **F5** (ou selecionando o botão de reprodução no canto superior direito).



Próximo ao canto superior direito da tela de navegação, selecione o ícone de classificação uma ou mais vezes para alterar a ordem alfabética de classificação entre crescente e decrescente.



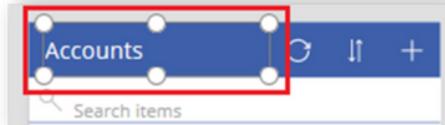
Na caixa de pesquisa, digite **k** para mostrar apenas os nomes de conta que contêm a letra digitada.



Remova todo o texto da barra de pesquisa e, em seguida, feche o modo Visualização pressionando Esc (ou selecionando o ícone de fechar perto do canto superior direito).

Alterar o título da tela:

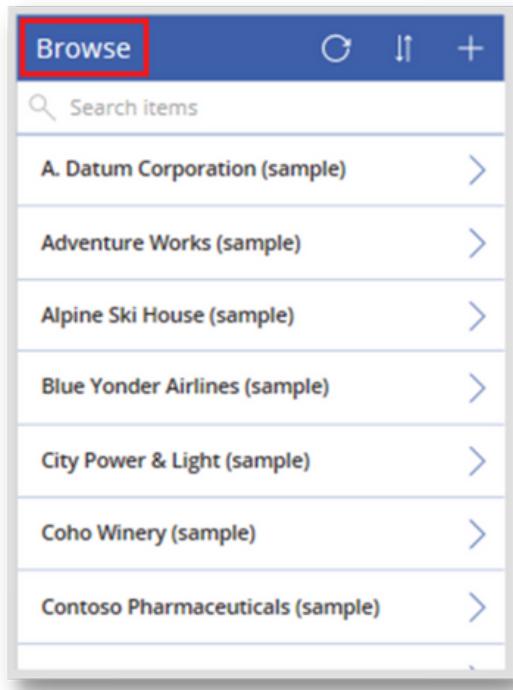
Selecione o título da tela clicando ou tocando nele.



Verifique se a lista de propriedades mostra **Text** e, em seguida, na barra de fórmulas, substitua **Accounts** por **Browse** (mantendo as aspas duplas).



A tela reflete a alteração.



Mostrar uma barra de rolagem:

Caso os usuários não tenham telas touch nem botões de rolagem do mouse, configure a galeria para mostrar uma barra de rolagem quando o usuário passar o mouse sobre ela. Dessa forma, os usuários podem mostrar todas as contas, mesmo se a tela não puder mostrá-las todas de uma vez.

Selecione a galeria conforme descrito no primeiro procedimento.



Defina a propriedade **Show scrollbar** da galeria como **true**.

PERSONALIZAR FORMULÁRIOS EM APLICATIVOS DE TELA

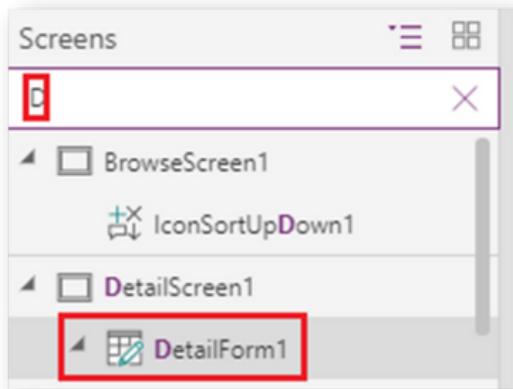
Em um aplicativo de tela, personalize um controle Display form e um controle Edit form para que eles mostrem os dados mais importante e na ordem mais intuitiva para ajudar os usuários a entender e atualizar os dados com facilidade.

Cada formulário consiste em um ou mais cartões, cada um mostrando os dados de uma determinada coluna na fonte de dados. Seguindo as etapas deste tópico, você poderá especificar quais cartões são exibidos em um formulário e movê-los para cima e para baixo em um formulário.

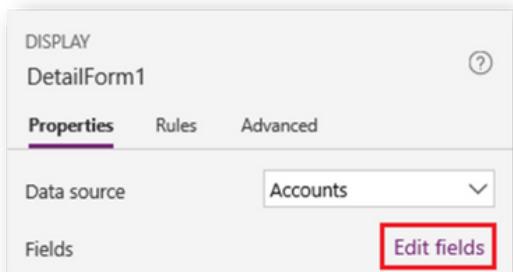
Mostrar e ocultar cartões:

Entre no Power Apps e abra o aplicativo que você gerou e personalizou.

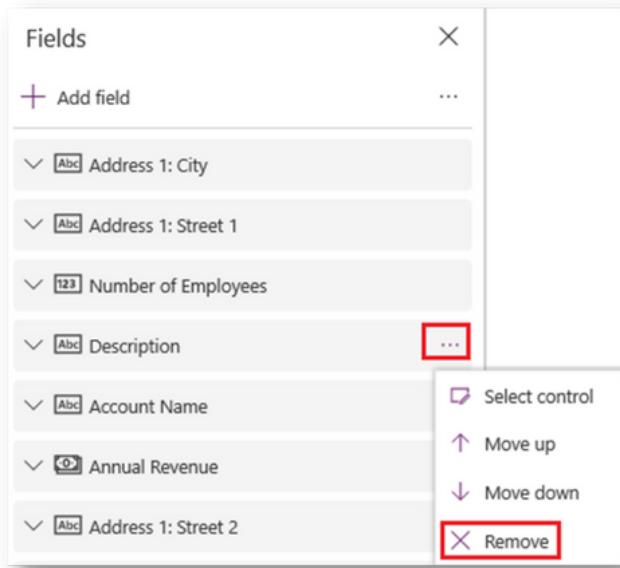
Na barra de navegação à esquerda, digite ou cole **D** na barra de pesquisa para filtrar a lista de elementos e, em seguida, selecione **DetailForm1**.



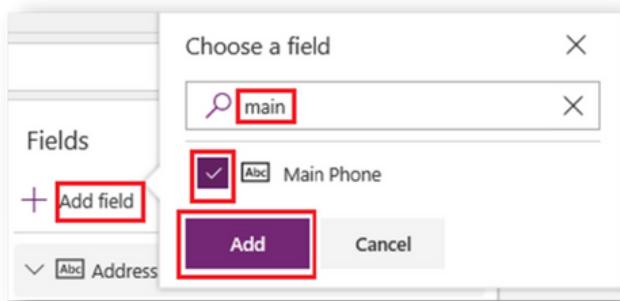
Na guia **Propriedades** do painel direito, selecione **Editar campos** para abrir o painel **Campos**.



Ocute um campo, como **Descrição**, passando o mouse sobre ele, selecionando as reticências (...) exibidas e, em seguida, selecionando **Remover**.



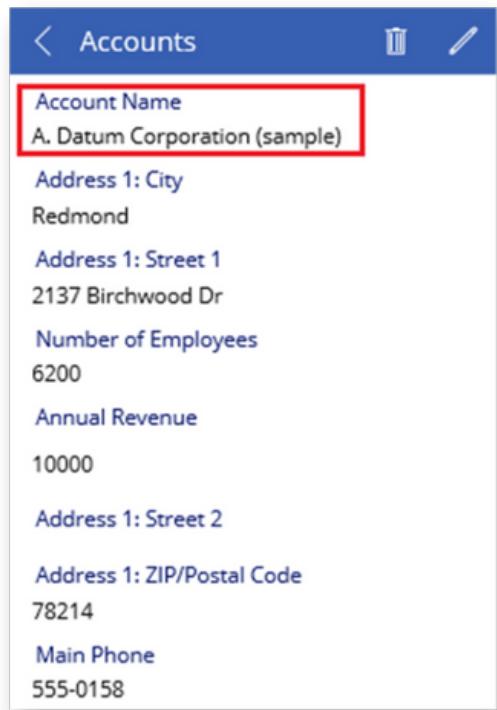
Mostre um campo selecionando **Adicionar campo**, digitando ou colando as primeiras letras do nome do campo na caixa de pesquisa, marcando a caixa de seleção do campo e selecionando **Adicionar**.



Reordenar cartões:

No painel **Campos**, arraste o campo **Nome da Conta** para a parte superior da lista de campos.

Os cartões no **DetailForm1** refletem a alteração.



(opcional) Reordene os outros cartões nesta sequência:

- Nome da Conta
- Número de Funcionários
- Receita Anual
- Telefone Principal
- Endereço 1: Rua 1
- Endereço 1: Rua 2
- Endereço 1: Cidade
- Endereço 1: CEP

Na barra de navegação à esquerda, digite ou cole **Ed** na barra de pesquisa e, em seguida, selecione **EditForm1**.

Repita as etapas do procedimento anterior, incluindo esta, de modo que os campos de **EditForm1** correspondam àqueles de **DetailForm1**.

CRIAR UM APLICATIVO DE TELA DO ZERO COM BASE NOS DADOS DO EXCEL

Crie seu próprio aplicativo de tela do zero com base em dados do Excel formatados como tabela e, em seguida, adicione dados de outras fontes, caso deseje. Seguindo este tutorial, você criará um aplicativo que contém duas telas.

Em uma tela, os usuários podem navegar por meio de um conjunto de registros. Na outra tela, os usuários podem criar um registro, atualizar um ou mais campos em um registro ou excluir um registro inteiro. Essa abordagem demora mais do que criar um aplicativo básico no Excel, mas os criadores de aplicativos mais experientes podem usá-la para criar um aplicativo que melhor atenda às necessidades deles.

Para executar as etapas deste tutorial com precisão, primeiro crie um arquivo do Excel usando estes dados de exemplo.

Copie esses dados e, em seguida, cole-os em um arquivo do Excel.

StartDay	StartTime	Voluntário	Backup
sábado	10h – meio-dia	Vasquez	Kumashiro
sábado	meio-dia – 14h	Ice	Singhal
sábado	14h – 16h	Myk	Mueller
domingo	10h – meio-dia	Li	Adams
domingo	meio-dia – 14h	Singh	Morgan
domingo	14h – 16h	Batye	Nguyen

Formate esses dados como uma tabela, chamada **Agenda**, de modo que o Power Apps possa analisar as informações.

Salve o arquivo com o nome **eventsSignup.xls**, feche-o e, em seguida, carregue-o em uma conta de armazenamento em nuvem, como o OneDrive.

Abrir um aplicativo em branco:

Crie um aplicativo de tela em branco com layout para **Smartphone**.

Conectar-se aos dados:

No meio da tela, selecione **Conectar aos dados**.

No painel **Dados**, selecione a conexão de sua conta de armazenamento em nuvem, se ela aparecer. Caso contrário, execute estas etapas para adicionar uma conexão:

Selecione **Nova conexão**, selecione o bloco da sua conta de armazenamento em nuvem e, em seguida, **Criar**.

Forneça as credenciais dessa conta, se for solicitado.

Em **Escolher um arquivo do Excel**, digite ou cole as primeiras letras de **eventsSignup** para filtrar a lista e, depois, selecione o arquivo que você carregou.

Em **Escolher uma tabela**, marque a caixa de seleção **Agenda** e, em seguida, selecione **Conectar**.

Feche o painel **Dados** no canto superior do painel, selecionando o ícone para fechar (X).

Criar a tela de exibição:

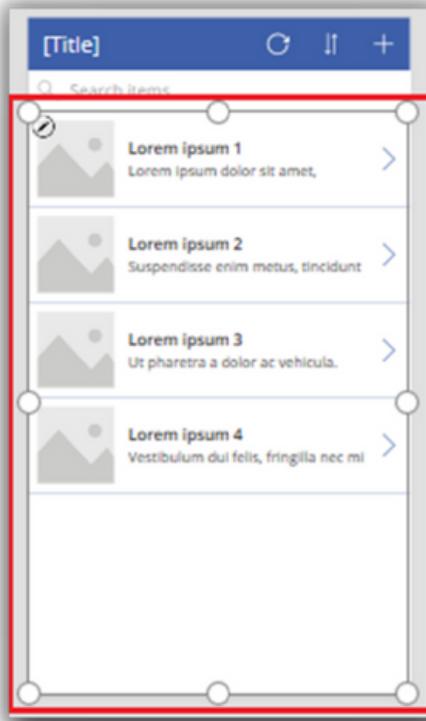
Na guia Página Inicial, selecione a seta para baixo ao lado de Nova tela para abrir uma lista de tipos de tela e, em seguida, selecione Lista.

- Uma tela é adicionada com diversos controles padrão, como uma caixa de pesquisa e um controle Gallery. A galeria abrange toda a tela abaixo da caixa de pesquisa.

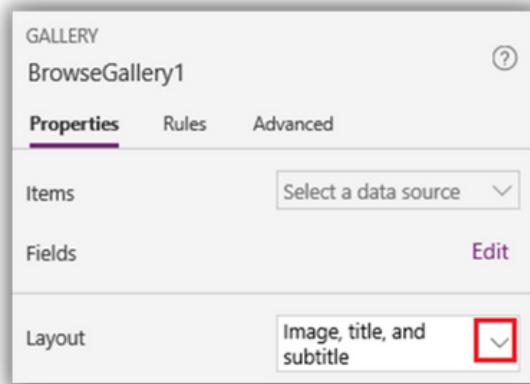
Na parte superior da nova tela, selecione o controle Label e, em seguida, substitua [Título] por Exibir registros.



Na barra de navegação à esquerda, selecione **BrowseGallery1**.
Uma caixa de seleção com alças contorna a galeria.



Na guia **Propriedades** do painel direito, selecione a seta para baixo do menu **Layout**.



Selecione **Título, subtítulo e corpo**.

Na barra de fórmulas, substitua **CustomGallerySample** por **Agenda**, e substitua as duas instâncias de **SampleText** por **Voluntário**.

Na borda direita da barra de fórmulas, selecione a seta para baixo e, em seguida, selecione **Formatar texto**.

A fórmula corresponde a este exemplo:

```

SortByColumns(
    Search(
        Schedule,
        TextSearchBox1.Text,
        "Volunteer"
    ),
    "Volunteer",
    If(
        SortDescending1,
        SortOrder.Descending,
        SortOrder.Ascending
    )
)

```

Na guia **Propriedades** do painel direito, selecione **Editar** ao lado do rótulo **Campos**.

Na caixa **Title2**, selecione **Voluntário**, na caixa **Subtitle2**, selecione **StartDay**, e na caixa **Body1**, selecione **StartTime**.

Feche o painel **Dados** no canto superior do painel, selecionando o ícone para fechar (X).

Os usuários podem classificar e filtrar a galeria por nome de voluntário com base nas funções **SortByColumns** e **Search** na fórmula.

- Se um usuário digitar pelo menos uma letra na caixa de pesquisa, a galeria mostrará apenas os registros para os quais o campo Voluntário contém o texto digitado pelo usuário.
- Se um usuário selecionar o botão de classificação (entre o botão de atualização e o botão de adição na barra de título), a galeria mostrará os registros em ordem crescente ou decrescente (dependendo de quantas vezes o usuário selecionar o botão) com base no campo Voluntário.

Criar a tela de alteração:

Na guia **Página Inicial**, selecione a seta para baixo ao lado de **Nova tela** e, em seguida, selecione **Formulário**.

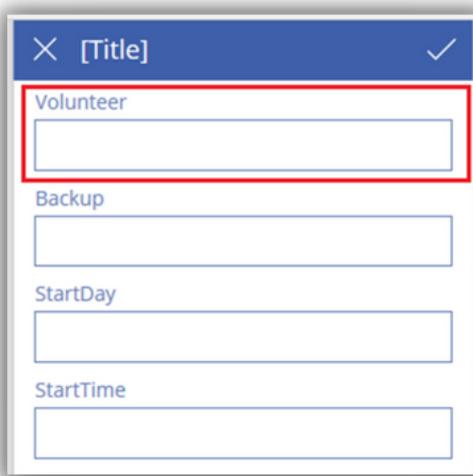
Na barra de navegação à esquerda, selecione **EditForm1**.

Na guia **Propriedades** do painel direito, selecione a seta para baixo ao lado de **Fonte de dados** e, em seguida, selecione **Agenda** na lista que é exibida.

Na fonte de dados que você especificou, selecione **Editar campos**.

No painel **Campos**, selecione **Adicionar campo**, marque a caixa de seleção de cada campo e, em seguida, selecione **Adicionar**.

Selecione a seta ao lado do nome de cada campo para recolhê-lo e, em seguida, arraste o campo **Voluntário** para cima, para que ele apareça na parte superior da lista de campos.

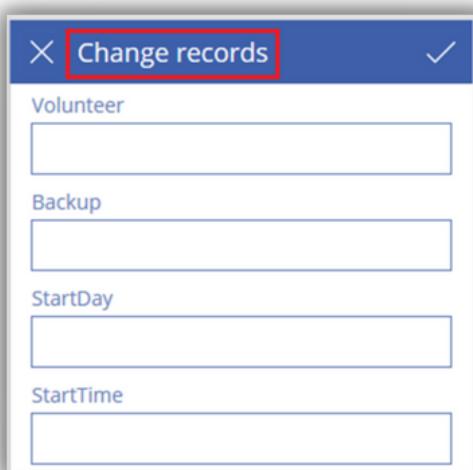


No canto superior direito do painel **Campos**, feche-o, selecionando o ícone para fechar (X).

Defina a propriedade **Item** do formulário como esta expressão, digitando ou colando-a na barra de fórmulas:

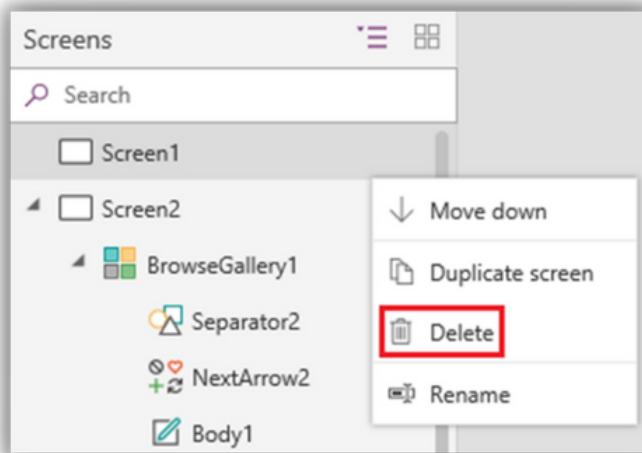
BrowseGallery1.Selected

Na parte superior da tela, selecione o controle **Label** e, em seguida, substitua **[Title]** por **Alterar registros**.



Excluir e renomear telas:

Na barra de navegação à esquerda, selecione as reticências (...) de **Screen1** e, depois, **Excluir**.



Selecione as reticências (...) de **Screen2**, selecione **Renomear** e, em seguida, digite ou cole **ViewScreen**.

Selecione as reticências (...) de **Screen3**, selecione **Renomear** e, em seguida, digite ou cole **ChangeScreen**.

Configurar ícones na tela de exibição:

Na parte superior de **ViewScreen**, selecione o ícone de seta circular.



Defina a propriedade **OnSelect** desse ícone como esta fórmula:

Refresh(Schedule)

Quando o usuário seleciona esse ícone, os dados da **Agenda** são atualizados no arquivo do Excel.

No canto superior direito de ViewScreen, selecione o ícone de adição.

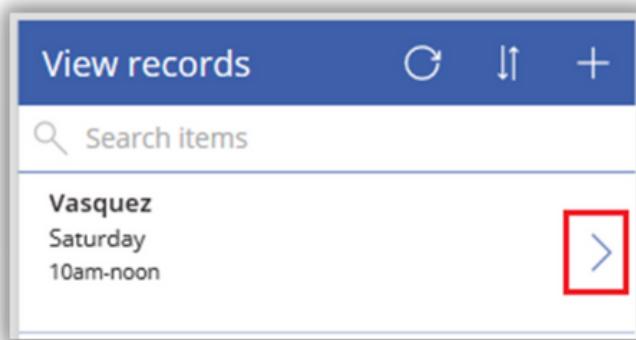


Defina a propriedade **OnSelect** desse ícone como esta fórmula:

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

Quando o usuário seleciona esse ícone, **ChangeScreen** é exibido com cada campo vazio, para que o usuário possa criar um registro com mais facilidade.

Selecione a seta que está apontando para a direita do primeiro registro na galeria.



Defina a propriedade **OnSelect** da seta como esta fórmula:

```
EditForm(EditForm1); Navigate(ChangeScreen, ScreenTransition.None)
```

Quando o usuário seleciona esse ícone, **ChangeScreen** é exibido em cada campo mostrando os dados do registro selecionado para que o usuário possa editar ou excluir o registro mais facilmente.

Configurar ícones na tela de alteração:

Em **ChangeScreen**, selecione o ícone "X" no canto superior esquerdo.



Defina a propriedade **OnSelect** desse ícone como esta fórmula:

```
ResetForm(EditForm1); Navigate(ViewScreen, ScreenTransition.None)
```

Quando o usuário seleciona esse ícone, as alterações feitas na tela pelo usuário são descartadas, e a tela de exibição é aberta.

No canto superior direito, selecione o ícone de marca de seleção.



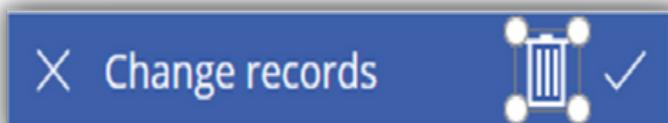
Defina a propriedade **OnSelect** da marca de seleção como esta fórmula:

```
SubmitForm(EditForm1); Navigate(ViewScreen, ScreenTransition.None)
```

Quando o usuário seleciona esse ícone, as alterações feitas por ele nesta tela são salvas, e a tela de exibição é aberta.

Na guia **Inserir**, selecione **Ícones** e, em seguida, selecione o ícone **Lixeira**.

Defina a propriedade **Color** do ícone novo como **Branco**, e mova o ícone novo para que ele apareça ao lado do ícone de marca de seleção.



Defina a propriedade **Visible** do ícone de lixeira como esta fórmula:

```
EditForm1.Mode = FormMode.Edit
```

Esse ícone aparecerá somente quando o formulário estiver no modo de **Edição**, não no modo **Novo**.

Defina a propriedade **OnSelect** do ícone de lixeira como esta fórmula:

```
Remove(Schedule, BrowseGallery1.Selected); Navigate(ViewScreen, ScreenTransition.None)
```

Quando o usuário seleciona esse ícone, o registro selecionado é excluído da fonte de dados e a tela de exibição é aberta.

ADICIONAR E CONFIGURAR CONTROLES EM APLICATIVOS DE TELA

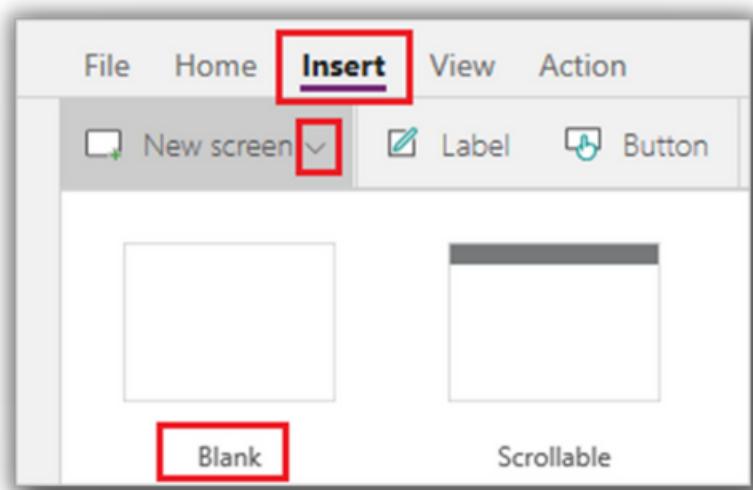
Adicione vários elementos de interface do usuário ao seu aplicativo de tela e configure aspectos de sua aparência e comportamento diretamente, na barra de ferramentas, na guia **Propriedades** ou na barra de fórmulas. Esses elementos da interface do usuário são chamados de controles e os aspectos que você configura são chamados de propriedades.

Adicionar e selecionar um controle:

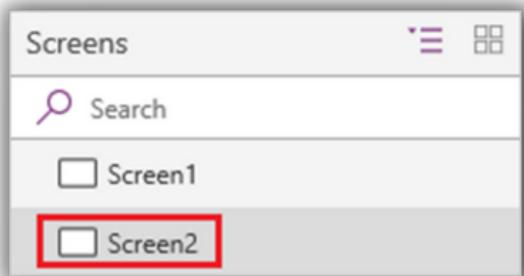
Na guia **Inserir**, execute uma destas etapas:

- Selecione **Label** ou **Button** para adicionar um desses tipos de controles.
- Selecione uma categoria de controles e, em seguida, selecione o tipo de controle que você deseja adicionar.

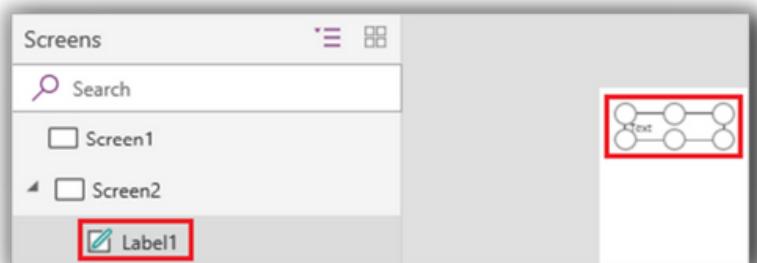
Por exemplo, selecione **Nova tela** e depois **Em branco** para adicionar uma tela em branco ao seu aplicativo. (As telas são um tipo de controle que pode conter outros tipos de controles.)



A nova tela é nomeada **Screen2** e aparece no painel de navegação esquerdo. Este painel mostra uma lista hierárquica de controles no seu aplicativo para que você possa encontrar e selecionar facilmente cada controle.



Para demonstrar como essa lista funciona, selecione **Label** na guia **Inserir**. O novo controle aparece em **Screen2** na lista hierárquica.



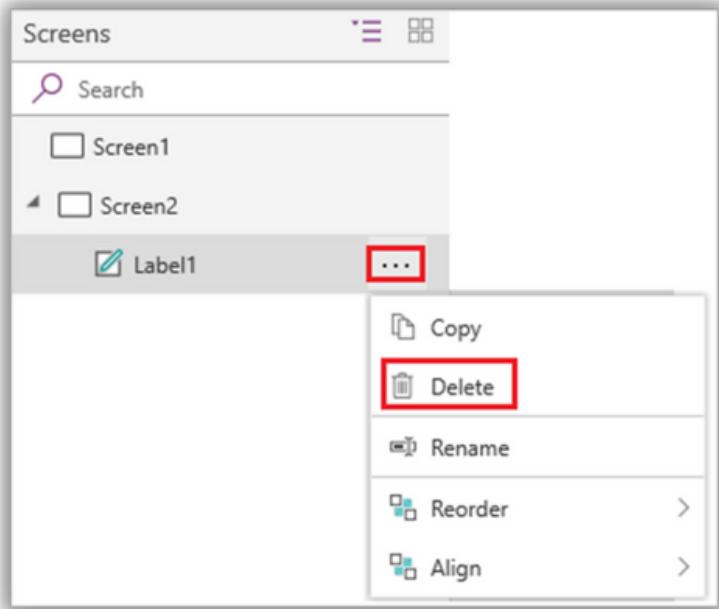
Na tela, uma caixa com seis alças envolve o rótulo por padrão. Esse tipo de caixa envolve o controle selecionado. Se você selecionar a tela clicando ou tocando nela (mas fora do rótulo), a caixa desaparecerá do rótulo. Para selecionar o rótulo novamente, clique ou toque nele ou clique ou toque no nome dele na lista hierárquica de controles.

Renomear um controle:

Na lista hierárquica de controles, passe o mouse sobre o controle que deseja renomear, selecione o botão de reticências que aparece e, em seguida, selecione **Renomear**. Você pode digitar um nome exclusivo e memorável para facilitar a criação do aplicativo.

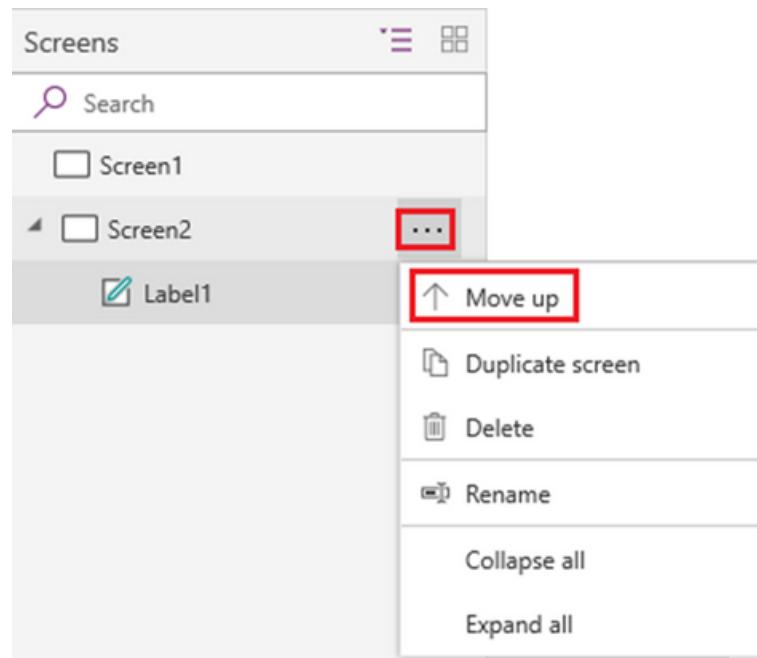
Excluir um controle:

Na lista hierárquica de controles, passe o mouse sobre o controle que deseja excluir, selecione o botão de reticências que aparece e, em seguida, selecione **Excluir**. Para excluir um controle que não é uma tela, você também pode selecionar o controle na tela e pressionar a tecla **Delete**.



Reordenar telas:

Na lista hierárquica de controles, passe o mouse sobre uma tela que você deseja se mover para cima ou para baixo, selecione o botão de reticências que aparece e selecione **Mover para cima** ou **Mover para baixo**.

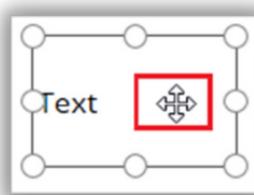


Observação:

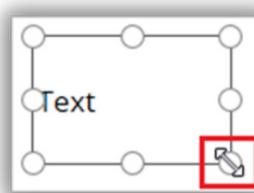
Quando o aplicativo é aberto, a tela na parte superior da lista hierárquica de controles geralmente aparece primeiro. Contudo, você pode especificar uma tela diferente definindo a propriedade **OnStart** como uma fórmula que inclui a função **Navigate**.

Mover e redimensionar um controle:

Para mover um controle, selecione-o, passe o mouse sobre o centro para que a seta de quatro pontas apareça e arraste o controle para um local diferente.



Para redimensionar um controle, selecione-o, passe o mouse sobre qualquer alça na caixa de seleção para que a seta de duas pontas apareça e arraste a alça.

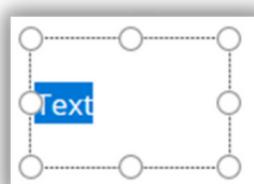


Observação:

Como este tópico descreve mais adiante, você também pode mover e redimensionar um controle modificando qualquer combinação de suas propriedades **X**, **Y**, **Height** e **Width** na barra de fórmulas.

Alterar o texto de um rótulo ou botão:

Selecione um rótulo ou botão, clique duas vezes no texto que aparece no controle e digite o texto que deseja.



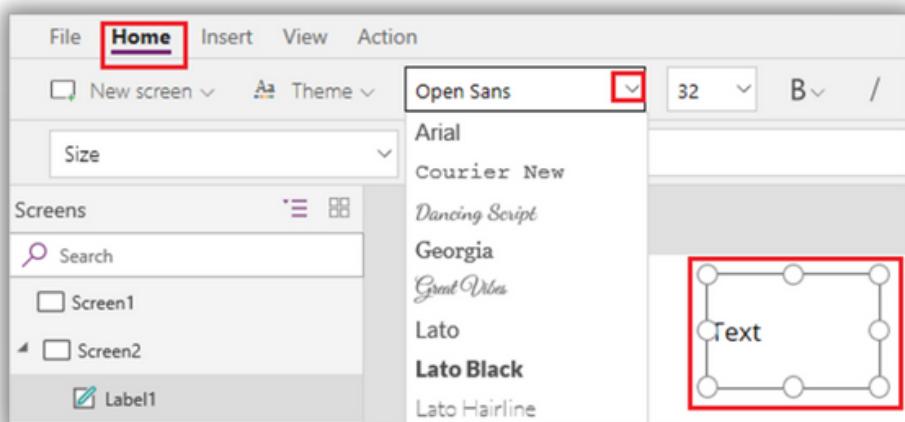
Observação:

Como este tópico descreve mais adiante, você também pode alterar esse texto modificando sua propriedade **Text** na barra de fórmulas.

Configurar um controle na barra de ferramentas:

Configurando um controle na barra de ferramentas, você pode especificar uma variedade maior de opções do que configurando um controle diretamente.

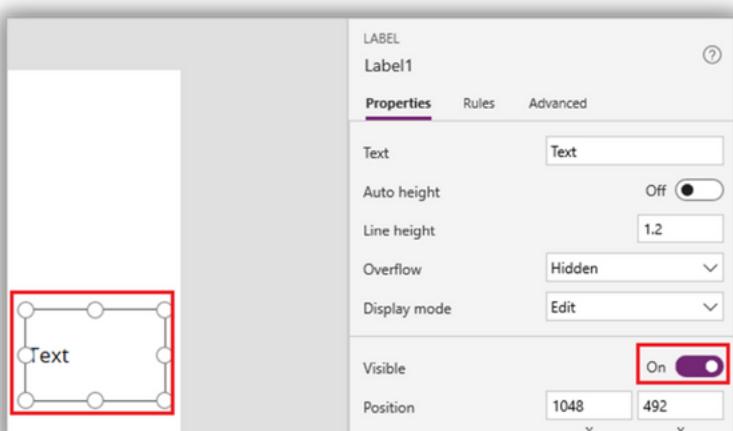
Por exemplo, você pode selecionar um rótulo, selecionar a guia **Página Inicial** e alterar a fonte do texto no rótulo.



Configurar um controle na guia Propriedades:

Usando a guia **Propriedades**, você pode especificar uma variedade maior de opções do que configurando um controle na barra de ferramentas.

Por exemplo, você pode selecionar um controle e, em seguida, mostrá-lo ou ocultá-lo, alterando sua propriedade **Visible**.

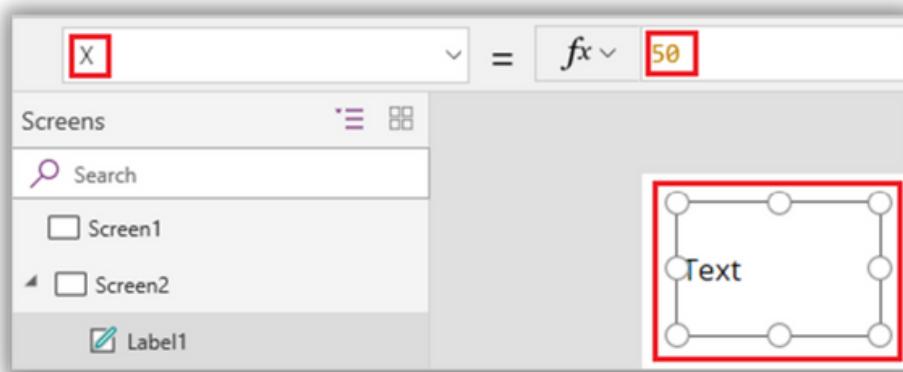


Configurar um controle na barra de fórmulas:

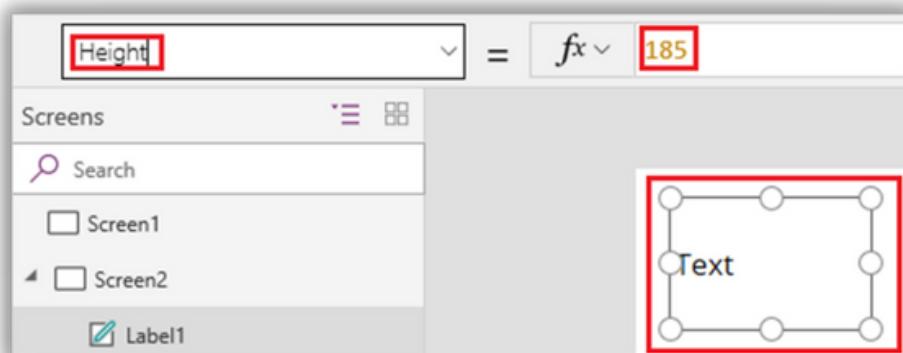
Em vez de configurar um controle diretamente, na barra de ferramentas ou na guia **Propriedades**, você pode configurar um controle selecionando uma propriedade na lista de propriedades e especificando um valor na barra de fórmulas. Ao adotar essa abordagem, é possível procurar uma propriedade em ordem alfabética e especificar mais tipos de valores.

Por exemplo, você pode selecionar um rótulo e configurá-lo das seguintes maneiras:

Mova-o selecionando **X** ou **Y** na lista de propriedades e especificando um número diferente na barra de fórmulas.

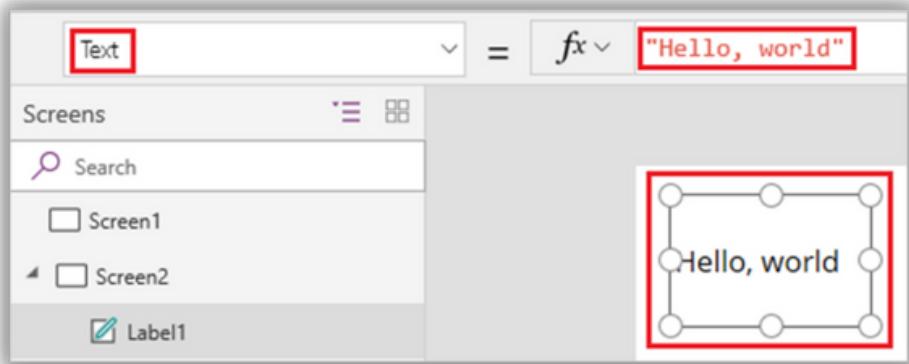


Redimensione-o selecionando **Height** ou **Width** na lista de propriedades e especificando um número diferente na barra de fórmulas.

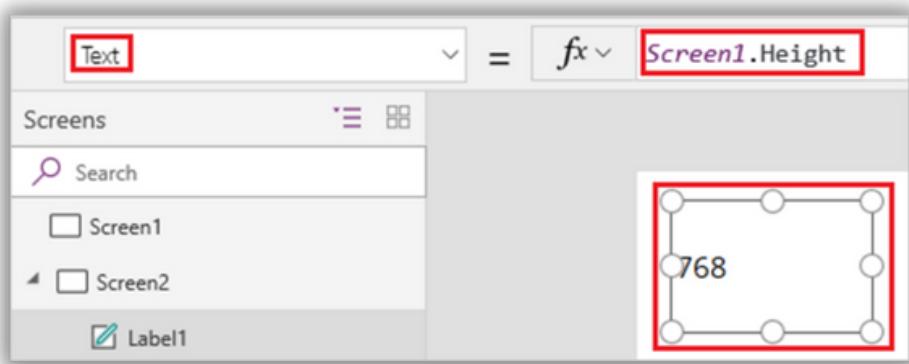


Altere seu texto selecionando **Text** na lista de propriedades e especificando qualquer combinação de uma cadeia de caracteres literal, uma expressão ou uma fórmula na barra de fórmulas.

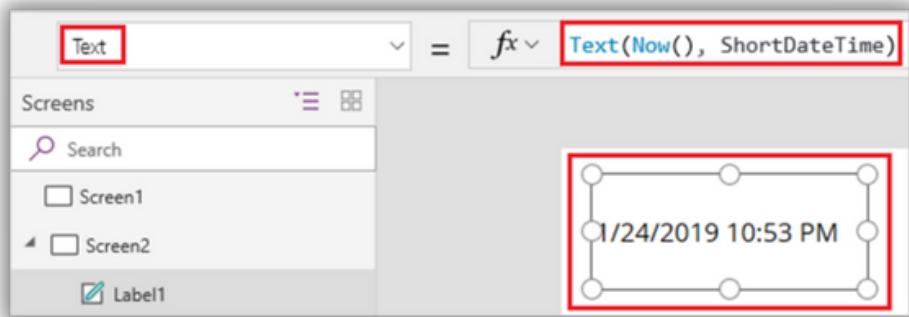
Uma cadeia de caracteres literal é colocada entre aspas e aparece conforme você a digita. "Hello, world" é uma cadeia de caracteres literal.



Uma expressão não inclui uma função e geralmente é baseada na propriedade de outro controle. **Screen1.Height** é uma expressão que mostra a altura de **Screen1**.



Uma fórmula inclui uma ou mais funções. A função **Now** retorna a data e a hora atuais no fuso horário local e a função **Text** formata valores como datas, horas e moeda.



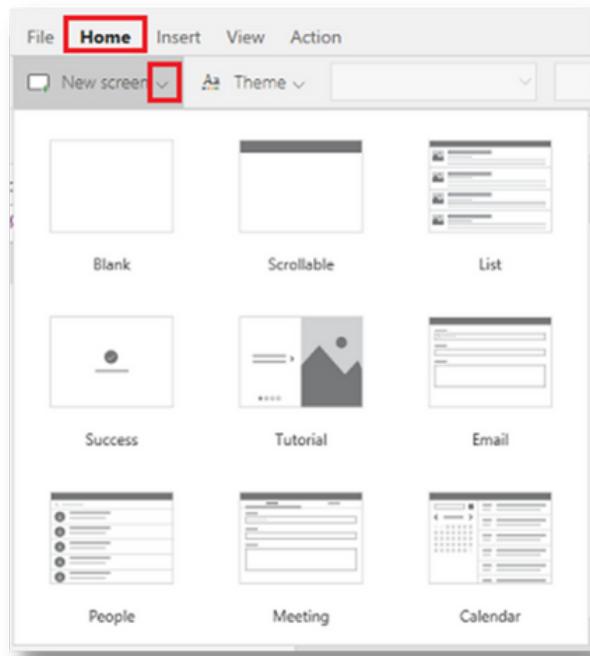
As fórmulas geralmente são muito mais complexas que esse exemplo, para que possam atualizar dados, classificá-los, filtrá-los e executar outras operações. Para obter mais informações, consulte a referência de fórmulas.

Adicionar uma tela a um aplicativo de tela e navegar entre telas:

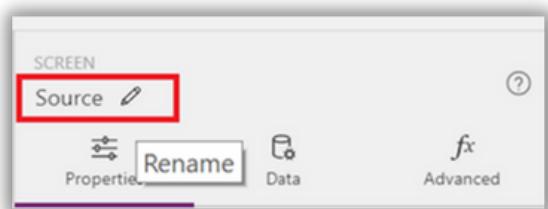
Crie um aplicativo de tela com várias telas e adicione maneiras para que os usuários naveguem entre elas.

Adicionar e renomear uma tela:

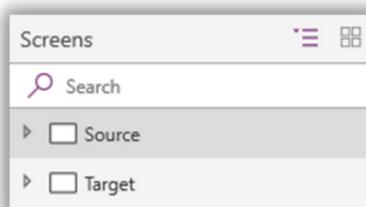
Na guia Página Inicial, selecione Nova tela e, em seguida, selecione o tipo de tela que você deseja adicionar.



No painel direito, selecione o nome da tela (imediatamente acima da guia **Propriedades**) e, em seguida, digite **Fonte**.

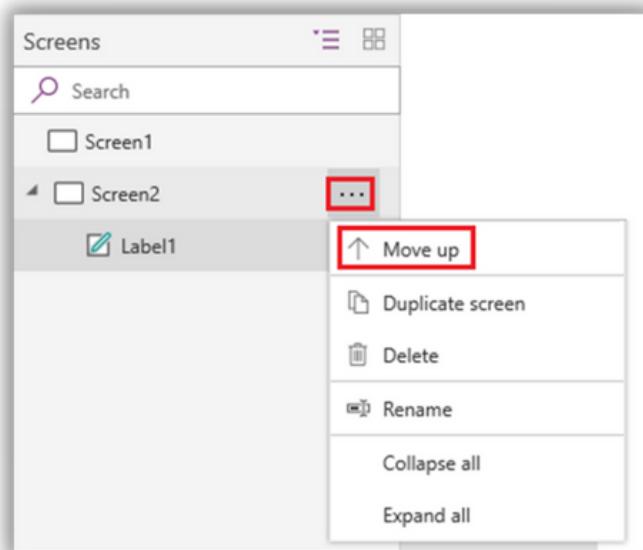


Adicione outra tela e chame-a de **Target**.



Reordenar telas:

Na barra de navegação esquerda, passe o mouse sobre uma tela que você deseja se mover para cima ou para baixo, selecione o botão de reticências que aparece e selecione **Mover para cima** ou **Mover para baixo**.

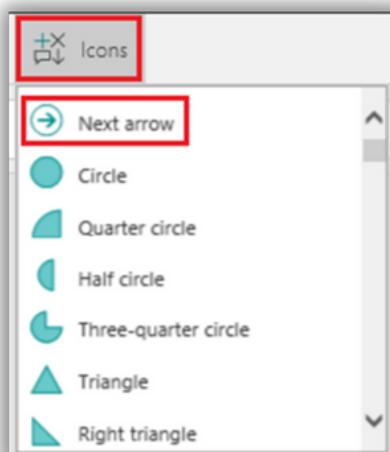


Observação:

Quando o aplicativo é aberto, a tela na parte superior da lista hierárquica de controles geralmente aparece primeiro. Contudo, você pode especificar uma tela diferente definindo a propriedade **OnStart** como uma fórmula que inclui a função **Navigate**.

Adicionar navegação:

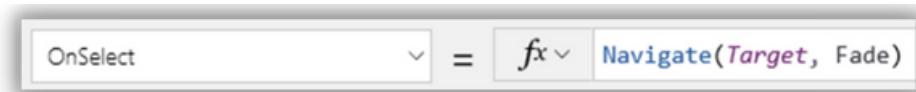
Com a tela **Fonte** selecionada, abra a guia **Inserir**, selecione **Ícones** e então selecione **Seta avançar**.



Mova a seta para que ela apareça no canto inferior direito da tela.

Com a seta ainda selecionada, selecione a guia **Ação** e, em seguida, selecione **Navegar**.

A propriedade **OnSelect** da seta é automaticamente definida como uma função **Navigate**.



Quando um usuário seleciona a seta, a tela **Target** desvanece.

Na tela **Target**, adicione uma **Seta voltar** e defina sua propriedade **OnSelect** como esta fórmula:

Navigate(Source, ScreenTransition.Fade)

Enquanto mantém pressionada a tecla Alt, alterne entre as telas selecionando a seta em cada tela.

ADICIONAR UMA CAIXA DE LISTAGEM, UMA CAIXA DE OPÇÃO OU BOTÕES DE OPÇÃO A UM APPLICATIVO DE TELA

Exiba uma única coluna de dados (por exemplo, de uma tabela de várias colunas) em um aplicativo de tela para que os usuários possam selecionar um ou mais itens em uma lista.

- Adicionar uma caixa de listagem para que os usuários possam selecionar mais de uma opção.
- Adicione uma lista suspensa ou uma caixa de combinação para ocupar menos espaço em uma tela.
- Adicionar um conjunto de botões de opção para um efeito de design específico.

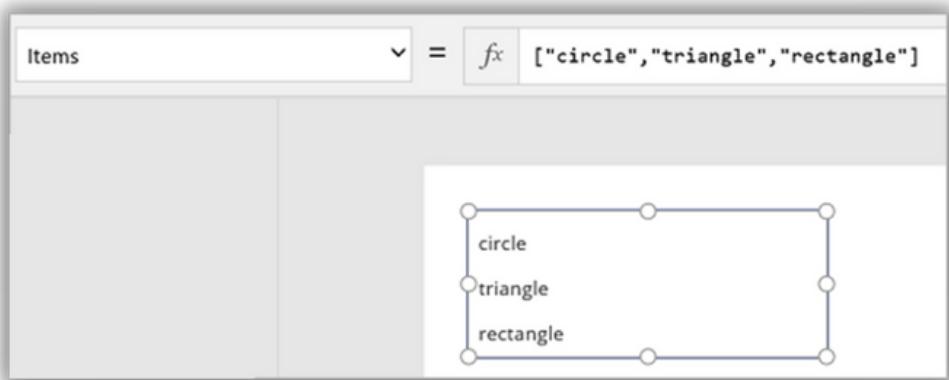
Este tópico se concentra em caixas de listagem e botões de opção, mas os mesmos princípios se aplicam a listas suspensas.

Criar uma lista simples:

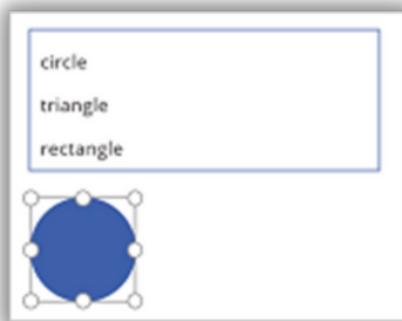
Adicione um controle **List box** chamado **MyListBox** e defina a propriedade **Items** para esta expressão:

```
["circle","triangle","rectangle"]
```

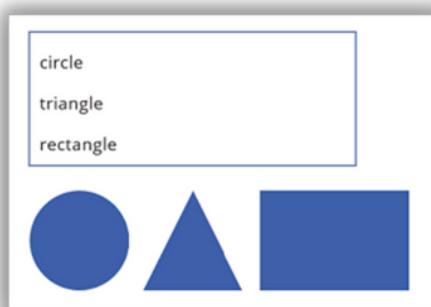
O designer é semelhante ao seguinte:



Na guia **Inserir**, selecione **Ícones**, selecione o círculo e coloque-o abaixo de **MyListBox**:



Adicione um triângulo e um retângulo e organize as formas em uma linha em **MyListBox**:



Definir a propriedade Visible das seguintes formas para as seguintes funções:

Forma	Defina a função Visible para
círculo	If("circle" in MyListBox.SelectedItems.Value, true)
triângulo	If("triangle" in MyListBox.SelectedItems.Value, true)
retângulo	If("rectangle" in MyListBox.SelectedItems.Value, true)

Enquanto mantém pressionada a tecla Alt, selecione uma ou mais formas em **MyListBox**.

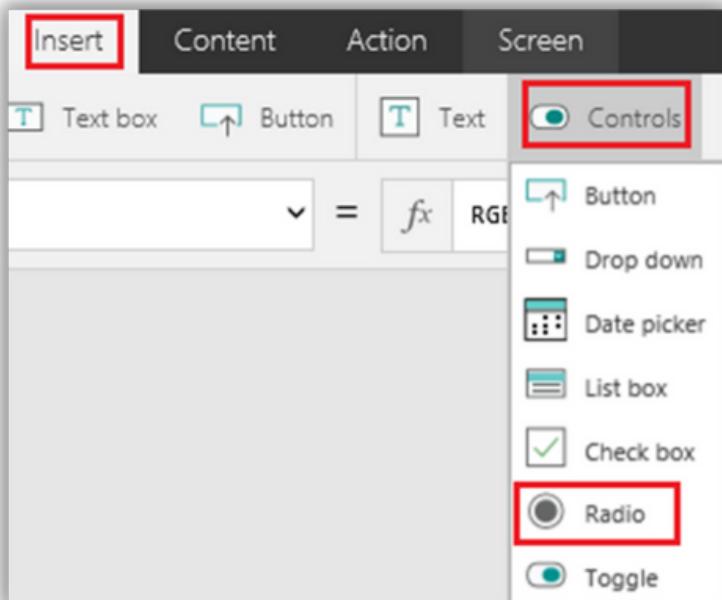
Somente a forma ou formas que você selecionou são exibidas.

Nestas etapas, você usou uma expressão para criar uma lista de itens. Você pode aplicá-la a outros elementos dentro de sua empresa. Por exemplo, você pode usar um controle **Drop down** para exibir imagens e descrições de produto e assim por diante.

Adicionar botões de opção:

Na guia **Página Inicial**, selecione **Nova tela** e, em seguida, selecione **Em branco**.

Na guia **Inserir**, selecione **Controles** e, em seguida, selecione **Radio**.



Renomeie o controle **Radio** como **Escolhas** e defina a propriedade **Items** conforme esta fórmula:

```
["red","green","blue"]
```



Se necessário, redimensione o controle para mostrar todas as opções.

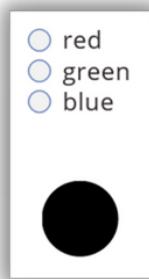
Na guia **Inserir**, selecione **Ícones** e, em seguida, selecione o círculo.

Defina a propriedade **Fill** do círculo para a função a seguir:

```
If(Choices.Selected.Value = "red", Red, Choices.Selected.Value = "green", Green,  
Choices.Selected.Value = "blue", Blue)
```

Nesta fórmula, o círculo altera sua cor dependendo de qual botão de opção que você escolher.

Mova o círculo sob o controle **Radio**, como neste exemplo:



Enquanto mantém pressionada a tecla Alt, selecione um botão de opção diferente para alterar a cor do círculo.

Adicionar um item a uma lista existente:

Adicione um controle **Button** e nomeie-o "btnReset".

Defina a propriedade **OnSelect** no **btnReset** como esta fórmula:

```
ClearCollect(MyItems, {value: "circle"},{value: "triangle"},{value: "rectangle"})
```

Defina a propriedade **Text** em **btnReset** como "Reset".

Adicione um controle **List box** nomeado **IblItems** e defina sua propriedade **Itens** como **MyItems**.

Enquanto mantém pressionada a tecla **Alt**, pressione o botão **Redefinir**.

Observação:

A caixa de lista deve ser preenchida com os itens da coleção "MyItems".

Organize a caixa de lista e o botão para que fiquem alinhados verticalmente.



Adicione um controle **Text Input** e nomeie-o como "txtAdd".

Defina a propriedade **Text** de **txtAdd** como "".

Adicione um controle **Button** e nomeie-o "btnAdd".

Defina a propriedade **Text** de **btnAdd** como "Add".

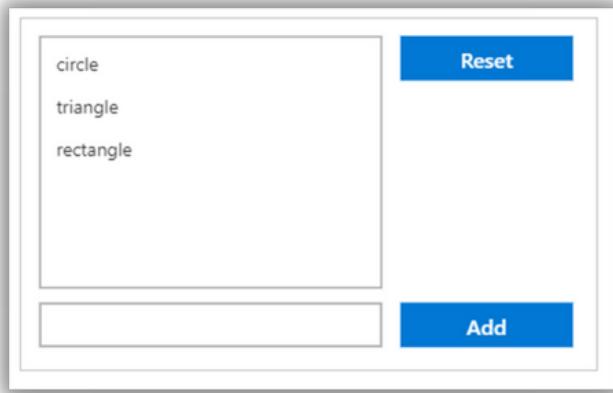
Defina a propriedade **OnSelect** de **btnAdd** como a seguinte fórmula:

`Collect(MyItems,{value: txtAdd.Text}); Reset(txtAdd)`

Observação:

- A função Collect adicionará o texto da entrada de texto como um item na coleção.
- A função de redefinição redefinirá a entrada de texto de volta ao seu estado padrão.

Organize **txtAdd** e **btnAdd** de forma que fiquem alinhados verticalmente abaixo de **lblItems** e **btnReset**.



Visualize o aplicativo pressionando **F5**.

Adicione um valor de texto ao controle de entrada de texto **txtAdd**.

Pressione o botão Adicionar.

Observação:

A caixa de lista deve ser preenchida com os itens da coleção **MyItems**.

Remover um item de uma lista existente

Adicione um controle **Button** e nomeie-o "btnDelete".

Defina a propriedade **Text** de **btnDelete** como "Delete".

Defina a propriedade **OnSelect** de **btnDelete** como a seguinte fórmula:

Remove(MyItems, lblItems.Selected)

Organize **btnDelete** de forma que ele fique alinhado verticalmente abaixo de **btnReset**

- Visualize o aplicativo pressionando F5.
- Pressione o botão Redefinir para redefinir a caixa de listagem.
- Pressione um item na caixa de lista para selecioná-lo.
- Pressione o botão Excluir para excluir o item.

MOSTRAR DADOS EM UM GRÁFICO DE LINHAS, DE PIZZA OU DE BARRAS EM APLICATIVOS DE TELA

Use gráficos de linhas, gráficos de pizza e gráficos de barras para exibir seus dados em um aplicativo de tela. Ao trabalhar com gráficos, os dados que você importa deverão ser estruturados com base nestes critérios:

- Cada série deve estar na primeira linha.
- Os rótulos devem estar na coluna mais à esquerda.

Por exemplo, seus dados devem ter uma aparência semelhante a seguinte:

Product	Revenue2012	Revenue2013	Revenue2014
Europa	21000	26000	28000
Ganymede	15000	17000	21000
Callisto	14000	19000	23000

É possível criar e usar esses gráficos dentro do Power Apps. Vamos começar.

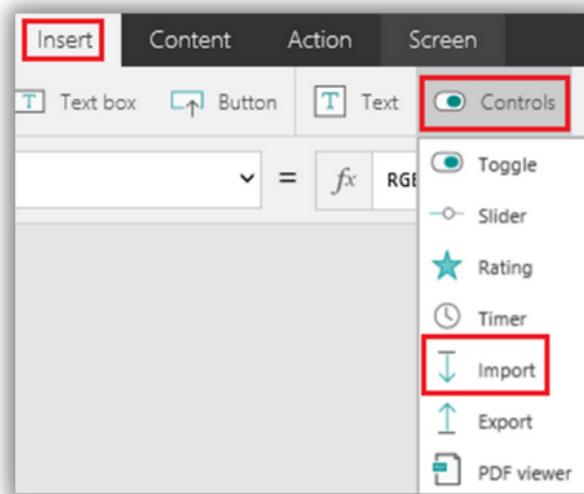
Pré-requisitos:

- Crie um aplicativo a partir de um modelo, a partir de dados ou do zero.
- Saiba como configurar um controle no Power Apps.
- Baixe [ChartData.zip](#), que contém dados de exemplo como um arquivo XML. Siga as etapas deste tópico para importá-lo diretamente para seu aplicativo. Como alternativa, descompacte o arquivo .zip, abra o arquivo XML no Excel e salve-o em uma conta de armazenamento em nuvem.

Importar os dados de exemplo:

Nestas etapas, importamos os dados de exemplo para uma coleção, denominada **ProductRevenue**.

Na guia **Inserir**, selecione **Controles** e, em seguida, selecione **Import**:



Defina a propriedade **OnSelect** do controle como a função a seguir:

```
Collect(ProductRevenue, Import1.Data)
```

Pressione F5 para abrir o modo de Visualização e, em seguida, selecione o botão **Importar Dados**.

Na caixa de diálogo **Abrir**, selecione ChartData.zip, selecione **Abrir** e, em seguida, pressione Esc.

No menu **Arquivo**, selecione **Coleções**.

A coleção ProductRevenue está listada com os dados do gráfico importado:

The screenshot shows the 'Collections' pane on the left with 'ProductRevenue' selected. On the right, the 'ProductRevenue' table is displayed with the following data:

Product	Revenue2012	Revenue2013	Revenue2014
Europa	21000	26000	28000
Ganimede	15000	17000	21000
Callisto	14000	19000	23000

Observação:

O controle Import é usado para importar dados no estilo Excel e criar a coleção. O controle Import importa dados quando você está criando seu aplicativo e visualizando-o. Atualmente, o controle Import não importa dados quando você publica seu aplicativo.

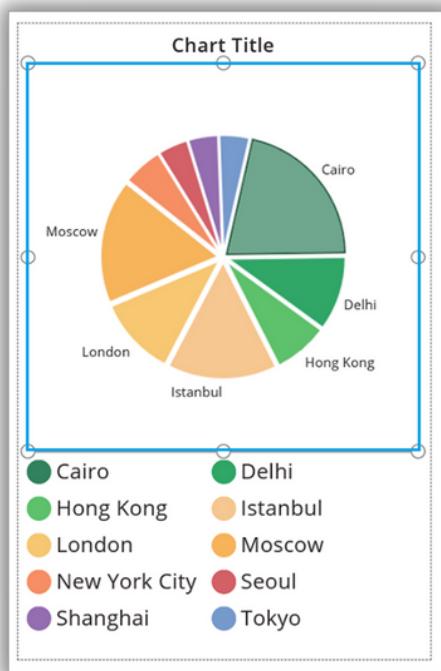
Pressione Esc para retornar ao espaço de trabalho padrão.

Adicionar um gráfico de pizza:

Na guia Inserir, selecione **Gráficos** e, em seguida, selecione **Gráfico de pizza**.

Mova o gráfico de pizza com o botão **Importar dados**.

No controle Pie chart, selecione o meio do gráfico de pizza:

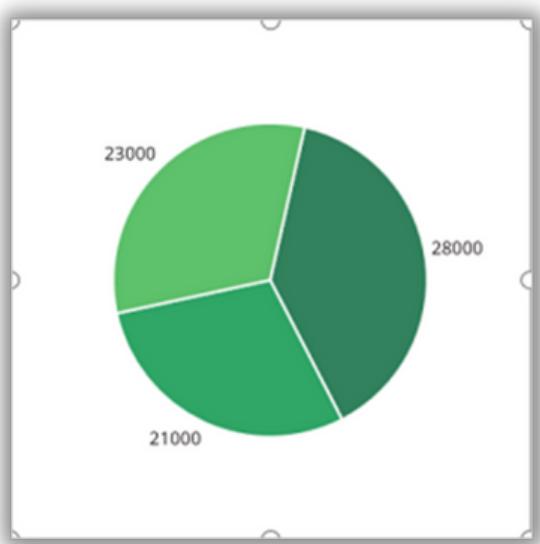


Defina a propriedade **Items** do gráfico de pizza como esta expressão:

ProductRevenue.Revenue2014



O gráfico de pizza mostra os dados de receita de 2014.



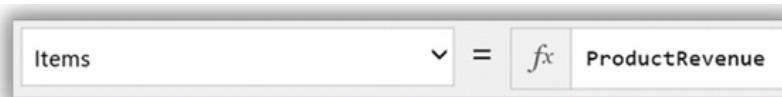
Adicionar um gráfico de barras para exibir seus dados:

Agora, vamos usar essa coleção ProductRevenue em um gráfico de barras:

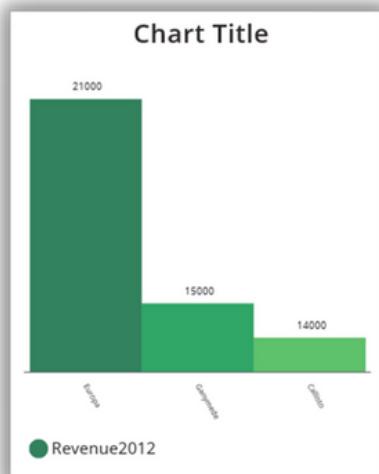
Na guia **Página Inicial**, adicione uma tela.

Na guia **Inserir**, selecione **Gráficos** e, em seguida, selecione **Gráfico de Colunas**.

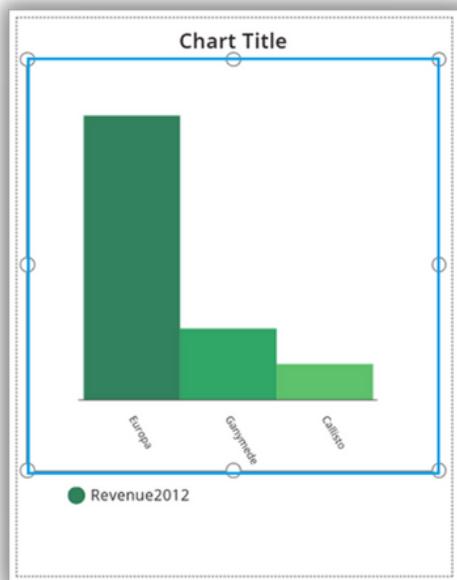
Selecione o meio do gráfico de colunas. Defina a propriedade **Items** do gráfico de colunas como ProductRevenue:



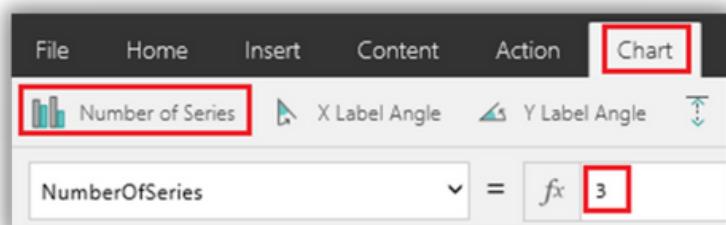
O gráfico de colunas mostra os dados de receita de 2012:



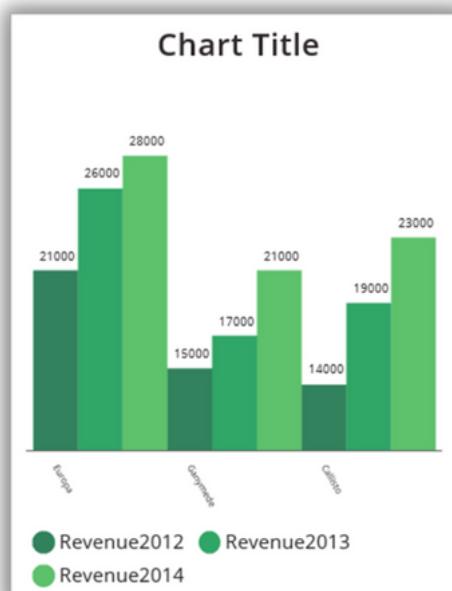
No gráfico de colunas, selecione o quadrado central:



Na guia Gráfico, selecione Número de Série e, em seguida, digite 3 na barra de fórmulas:



O gráfico de colunas mostra os dados de receita para cada produto durante três anos:



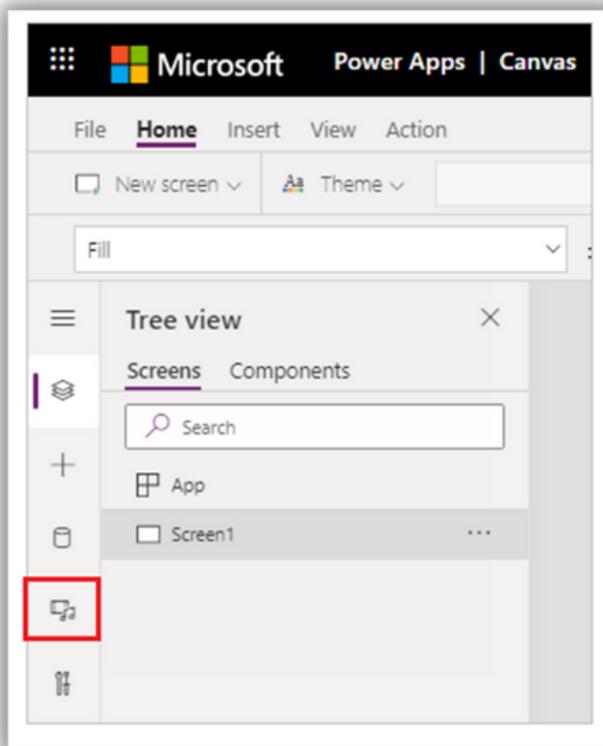
COMO USAR ARQUIVOS MULTIMÍDIA EM APLICATIVOS DE TELA

Você pode escolher o tipo de arquivo de mídia a ser adicionado (por exemplo, imagens, vídeo ou áudio). Você pode adicionar imagens usando o painel de mídia ou usando o controle de imagens.

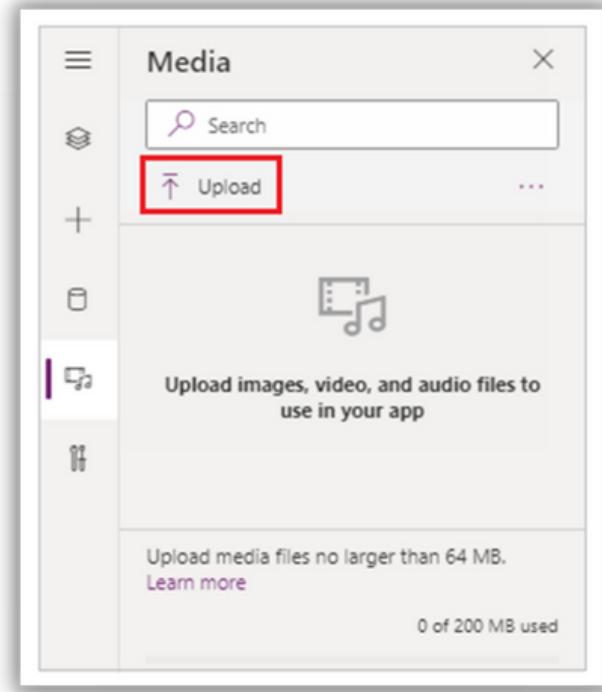
Adicione imagens, áudio ou vídeo usando o painel de mídia:

Para usar o painel de **Mídia** para adicionar, remover ou usar arquivos de mídia no seu aplicativo:

Selecione **Mídia** no painel esquerdo.

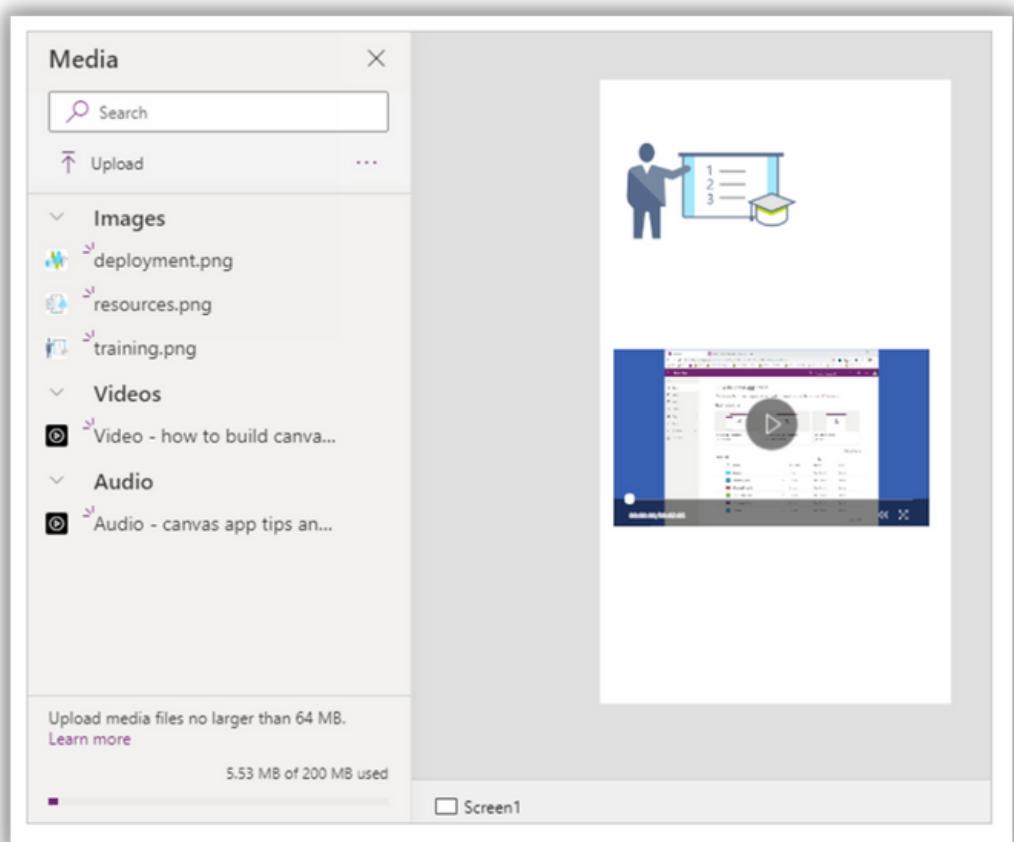


Selecione **Carregar** no painel de mídia.



Selecione os arquivos que você deseja adicionar e selecione **Abrir**.

Selecione o arquivo no painel de mídia para inserir na tela.



Salve e Publique seu aplicativo.

Compartilhe o aplicativo com outras pessoas.

Adicione imagens, áudio ou vídeo usando os controles

Para adicionar imagens, áudio ou vídeo usando os controle de Imagem, Áudio ou Vídeo:

Selecione **Inserir** no menu superior.

Selecione a lista suspensa de **Mídia**.

Escolha nos controles de **Imagen, Áudio ou Vídeo**.

Para controle de **Imagen**, atualize a propriedade Imagem com o nome do arquivo de imagem, sem extensão. Para **Áudio** ou **Vídeo** atualize a propriedade Mídia com o nome do arquivo ou o URL, como URL de vídeo do YouTube entre aspas duplas.

INTRODUÇÃO ÀS FÓRMULAS EM APLICATIVOS DE TELA

Configure seu aplicativo de tela com fórmulas que, além de calcular valores e executar outras tarefas (como no Excel), também respondem à entrada do usuário (como um aplicativo requer).

- No Excel, você cria fórmulas que, por exemplo, preenchem células e criam tabelas e gráficos.
- No Power Apps, você cria fórmulas semelhantes ao configurar controles em vez de células. Além disso, você pode criar fórmulas que se aplicam especificamente a aplicativos em vez de a planilhas.

Por exemplo, você pode criar uma fórmula para determinar como seu aplicativo responde quando os usuários selecionam um botão, ajustam um controle deslizante ou fornecem outra entrada. Essas fórmulas podem mostrar uma tela diferente, atualizar uma fonte de dados externa para o aplicativo ou criar uma tabela que contenha um subconjunto dos dados em uma tabela existente.

Você pode usar fórmulas para uma ampla variedade de cenários. Por exemplo, pode usar o GPS de um dispositivo, um controle de mapa e uma fórmula que use **Location.Latitude** e **Location.Longitude** para exibir o local atual. Quando você se deslocar, o mapa rastreará automaticamente seu local.

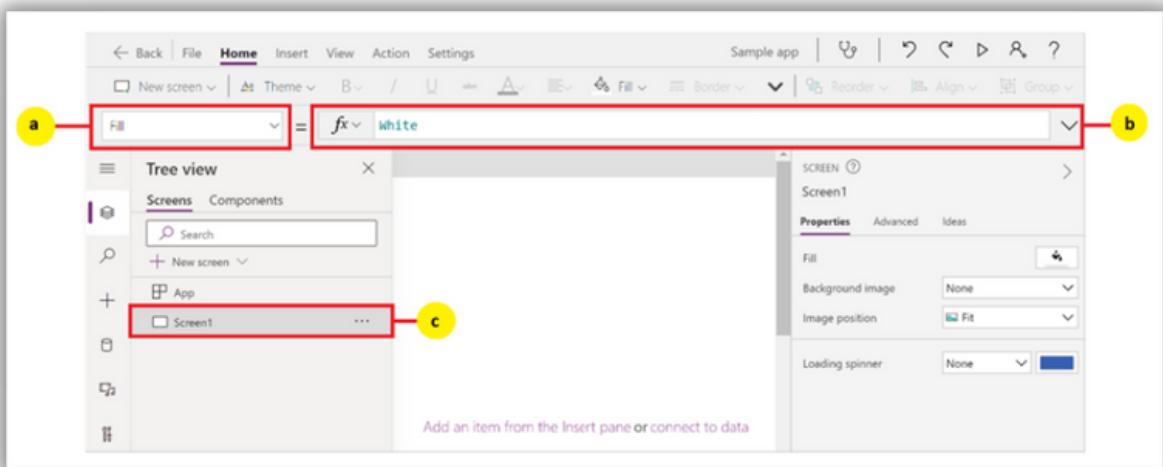
Este tópico fornece somente uma visão geral de como trabalhar com fórmulas. Procure a referência da fórmula para obter mais detalhes e a lista completa de funções, operadores e outros blocos de construção que você pode usar.

Mostrar um valor simples:

No Excel, você pode inserir dados específicos, como o número **42** ou a frase **Hello World**, digitando-os em uma célula. Essa célula sempre mostrará os dados exatamente como você digitou. No Power Apps, você pode especificar dados que não são alterados, definindo a propriedade **Text** de um rótulo para a sequência exata dos caracteres desejados, entre aspas duplas.

Criar um aplicativo de tela em branco.

A barra de fórmulas fica na parte superior da tela.

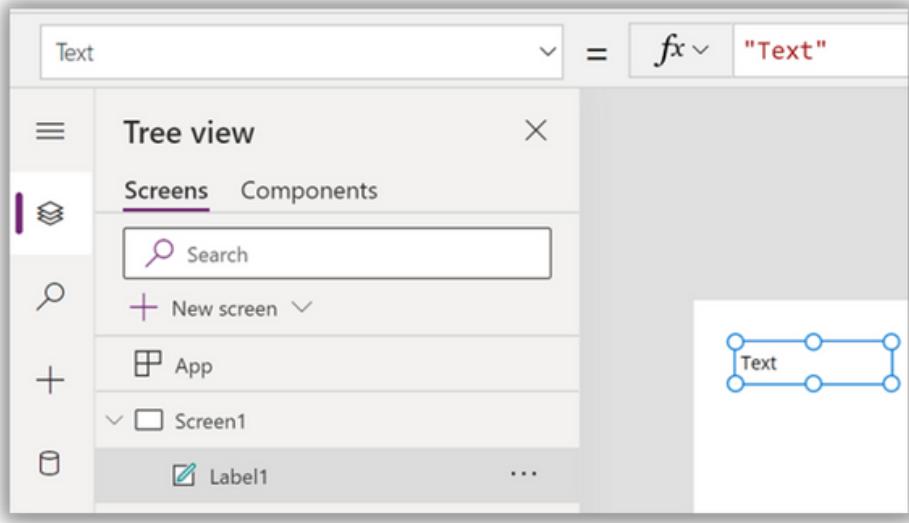


Lista de propriedades: cada controle e tela tem um conjunto de propriedades. Use essa lista para selecionar uma propriedade específica.

Fórmula: a fórmula a ser calculada para esta propriedade, composta de funções, operadores e valores.

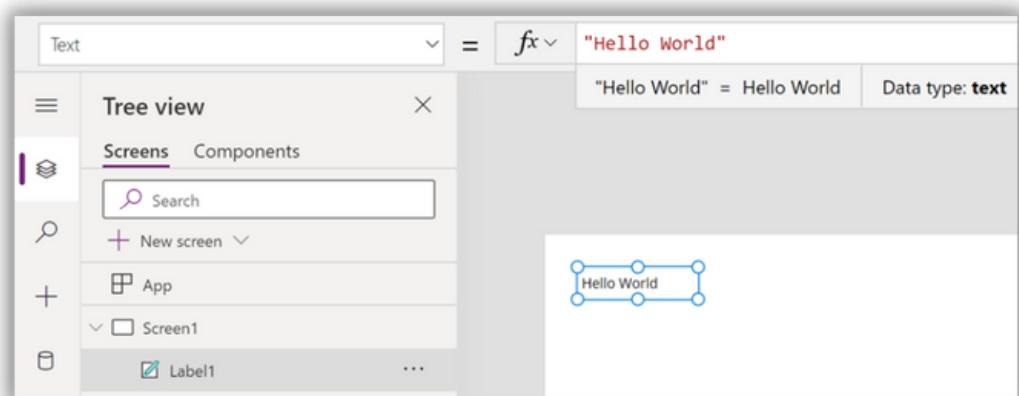
Controle selecionado: na barra de fórmulas, você pode ver e editar as propriedades para o controle selecionado ou para a tela, caso nenhum controle esteja selecionado.

Adicione um controle **Label** à tela.



Quando você adiciona um rótulo, a lista de propriedades mostra automaticamente a propriedade **Text**, que orienta o que o controle mostra. Por padrão, o valor dessa propriedade é "**Text**".

Defina o valor da propriedade **Text** como "**Hello World**", digitando essa cadeia de caracteres entre aspas duplas na barra de fórmulas:



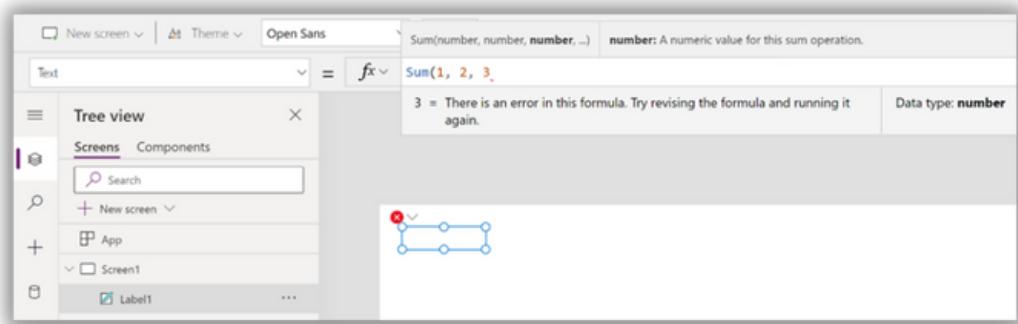
O rótulo reflete esse novo valor conforme você o digita. A tela pode mostrar ícones de ponto de exclamação amarelos enquanto você digita. Esses ícones indicam erros, mas eles desaparecem quando você termina de inserir um valor válido. Por exemplo, uma cadeia de caracteres sem aspas duplas em ambas as extremidades não é válida.

No Excel, você pode mostrar um número, como **42**, digitando-o em uma célula ou digitando uma fórmula cujo resultado seja esse número, como **=SUM(30,12)**. No Power Apps, é possível obter o mesmo efeito configurando a propriedade **Text** de um controle, como um rótulo, como **42** ou **Sum(30,12)**. A célula e o rótulo sempre mostrarão esse número, independentemente de outras alterações na planilha ou no aplicativo.

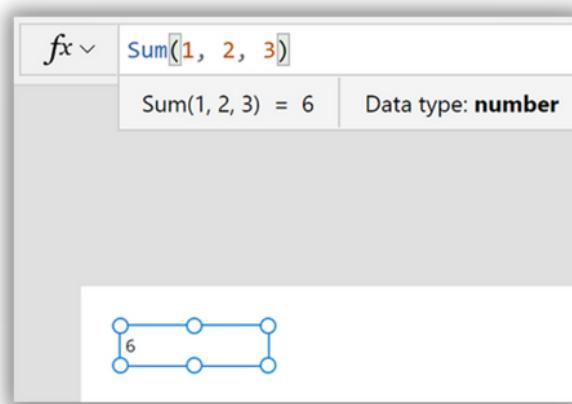
Observação:

No Power Apps, não preceda uma fórmula com um sinal de igual ou um sinal de adição, como você faz no Excel. A barra de fórmulas trata qualquer coisa que você digitar como uma fórmula por padrão. Você também não coloca uma fórmula entre aspas duplas (""), como fez anteriormente para especificar uma cadeia de texto.

Na propriedade **Text** do rótulo, substitua "Hello World" por **Sum(1,2,3)**.



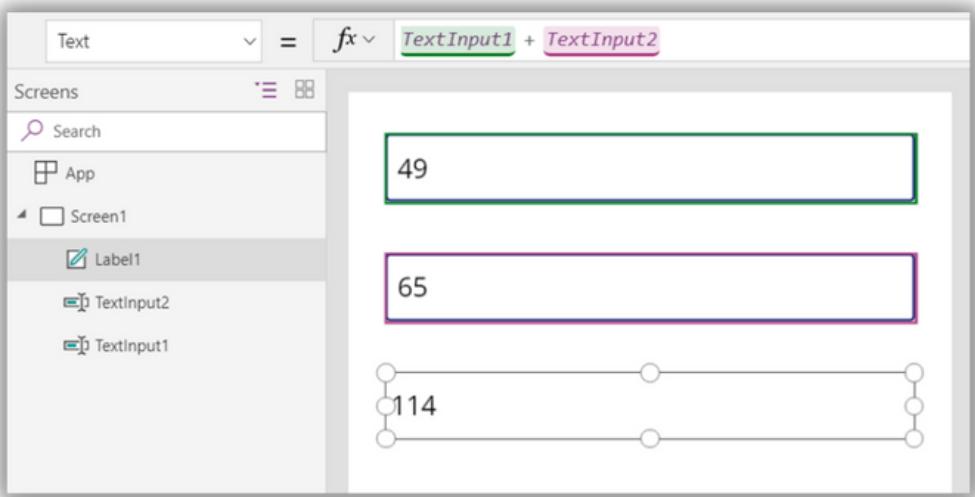
Enquanto você digita, a barra de fórmulas ajuda, mostrando a descrição e os argumentos esperados para essa função. Assim como acontece com as aspas duplas finais em "Olá, mundo", a tela mostra uma cruz vermelha para indicar um erro até que você digite o parêntese final dessa fórmula:



Alterar um valor com base na entrada:

No Excel, você digita =A1+A2 em uma célula para exibir a soma dos valores contidos nas células A1 e A2. Se um ou ambos os valores forem alterados, a célula que contém a fórmula automaticamente mostrará o resultado atualizado.

No Power Apps, você pode obter um resultado semelhante, adicionando controles a uma tela e definindo suas propriedades. Este exemplo mostra um controle Label chamado **Label1** e dois controles **Text input**, chamados **TextInput1** e **TextInput2**.



Independentemente de quais números você digitar nos controles de entrada de texto, o rótulo sempre mostrará a soma desses números, porque a propriedade **Text** está definida com esta fórmula:

TextInput1 + TextInput2

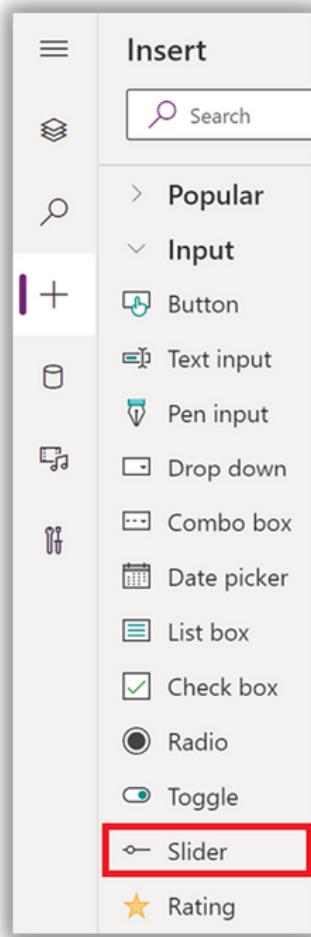
No Excel, você pode usar fórmulas de formatação condicional para mostrar, por exemplo, valores negativos em vermelho. No Power Apps, você pode usar fórmulas para determinar não apenas o valor principal de um controle, mas também propriedades como a formatação. No próximo exemplo, uma fórmula para a propriedade **Color** do rótulo mostra automaticamente os valores negativos em vermelho. A função **If** deve ser bem semelhante à do Excel:

If(Value(Label1.Text) < 0, Red, Black)

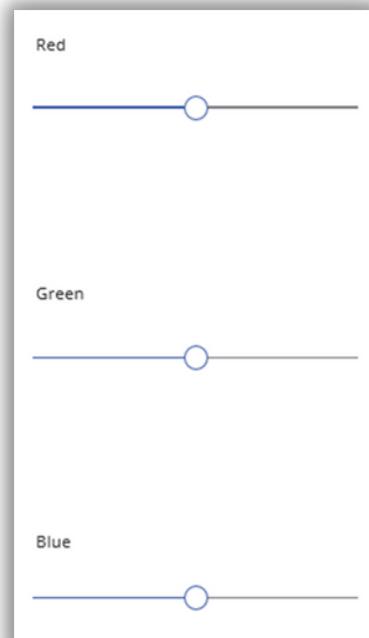
Alterar uma cor com base na entrada do usuário:

Você pode configurar seu aplicativo com fórmulas para que os usuários possam alterar a aparência ou o comportamento do seu aplicativo. Por exemplo, você pode criar um filtro para mostrar apenas os dados que contenham uma cadeia de texto especificada pelo usuário, ou pode permitir que os usuários classifiquem um conjunto de dados com base em uma determinada coluna no conjunto de dados. Neste procedimento, você permitirá que os usuários alterem a cor da tela ajustando um ou mais controles deslizantes.

Remova os controles dos procedimentos anteriores ou crie um aplicativo em branco como fez anteriormente e adicione três controles deslizantes a ele:



Organize os controles deslizantes de forma que não se sobreponham, adicione três rótulos e configure-os para mostrar Vermelho, Verde e Azul:

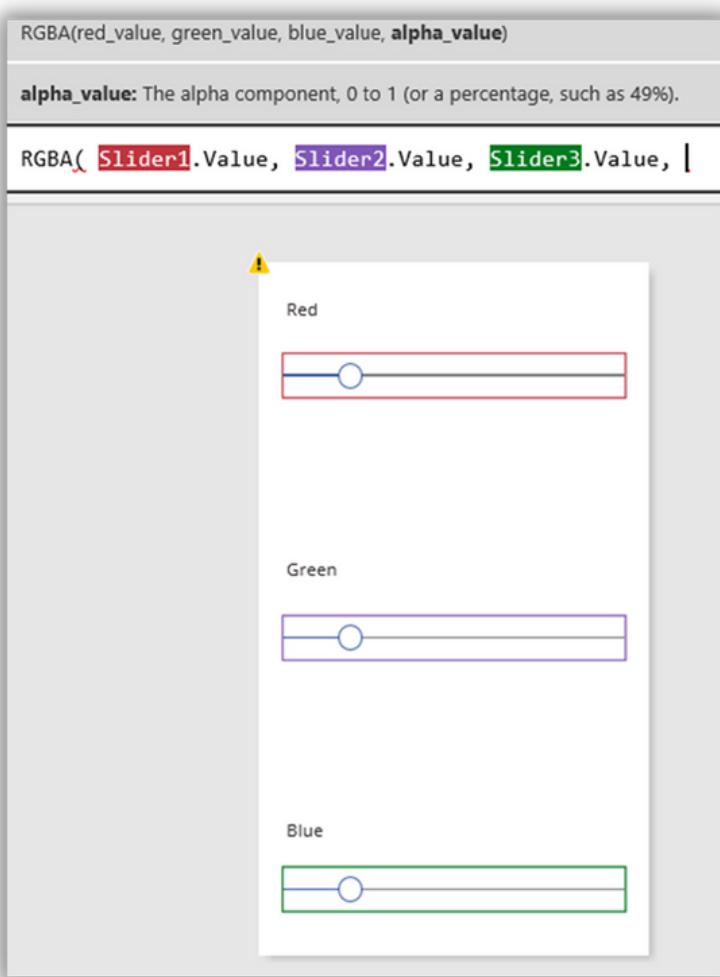


Defina a propriedade **Max** de cada controle deslizante para 255, que é o valor máximo de um componente de cor para a função **RGBA**.

Selecione a tela clicando fora de qualquer controle e defina a propriedade **Fill** da tela para esta fórmula:

`RGBA(Slider1.Value, Slider2.Value, Slider3.Value, 1)`

Como descrito anteriormente, usando o operador `.`, você acessa as propriedades de controle. **Slider1.Value** refere-se à propriedade **Value**, que reflete onde o usuário inseriu o controle deslizante entre os valores **Min** e **Max**. Conforme você digita essa fórmula, cada controle que ela contém é codificado por cor entre a tela e a barra de fórmulas:



Conforme você digita o parêntese de fechamento, o plano de fundo da tela será alterado para cinza-escuro com base no valor padrão de cada controle deslizante, que é 50. Quando você termina de digitar a fórmula, ela é calculada e usada como o valor da cor de preenchimento do plano de fundo.

Ajuste os controles deslizantes e veja como as alterações afetam a cor do plano de fundo executando o aplicativo.

À medida que cada controle deslizante é alterado, a fórmula que contém a função **RGBA** é recalculada, o que altera imediatamente a aparência da tela.

Gerenciar o comportamento do aplicativo:

Você pode usar fórmulas não apenas para executar cálculos e alterar a aparência, mas também para agir. Por exemplo, você pode definir a propriedade **OnSelect** de um botão para uma fórmula que inclua a função **Navigate**. Quando um usuário selecionar esse botão, será exibida a tela que você especificar na fórmula.

Você pode usar algumas funções, como **Navigate** e **Collect**, somente em fórmulas de comportamento. A referência de fórmula se destaca se você puder usar uma função somente neste contexto.

Você pode inserir mais de uma ação em uma fórmula de comportamento se separar as funções com um ponto e vírgula (;). Por exemplo, você talvez queira atualizar uma variável de contexto, enviar dados para uma fonte de dados e ir para outra tela.

Exibir uma lista de propriedades por categoria:

A lista de propriedades mostra as propriedades em ordem alfabética, mas você também pode exibir todas as propriedades de um controle, organizadas por categoria, se selecionar a opção **Avançado** na guia **Exibir** no lado direito da tela:



Você pode editar fórmulas diretamente nesse modo de exibição. Com o seletor de controle na parte superior do painel, você poderá localizar rapidamente um controle para trabalhar. E, com a pesquisa de propriedade, você poderá localizar rapidamente uma propriedade desse controle.

Inicialmente, essa exibição mostra as propriedades mais importantes. Para exibir todas as propriedades, clique na seta para baixo, na parte inferior do painel. Cada controle tem uma longa lista de propriedades que controlam todos os aspectos da aparência e do comportamento do controle. Você pode rolar pela lista ou procurar uma propriedade, digitando na caixa, na parte superior do painel.

Sintaxe da fórmula:

Ao digitar uma fórmula na barra de fórmulas, elementos de sintaxe diferentes aparecem em cores diferentes para melhorar a legibilidade e ajudá-lo a entender as fórmulas longas. Veja a seguir a lista de códigos de cor no Power Apps.

Function	Control	Global data source	Enumerated type
API reference	Keyword "ThisItem"	True, False	Number
String			

ENTENDER VARIÁVEIS EM APLICATIVOS DE TELA

Se você já usou outra ferramenta de programação, como Visual Basic ou JavaScript, deve estar se perguntando: onde estão as variáveis? O Power Apps é um pouco diferente e exige uma abordagem diferente. Em vez de buscar uma variável ao criar um aplicativo de tela, pergunte-se: o que eu faria no Excel?

Em outras ferramentas, você pode ter realizado explicitamente um cálculo e armazenado o resultado em uma variável. No entanto, o Power Apps e o Excel automaticamente recalculam fórmulas à medida que os dados de entrada mudam, por isso geralmente não é necessário criar nem atualizar as variáveis. Ao utilizar essa abordagem sempre que possível, é mais fácil criar, compreender e manter seu aplicativo.

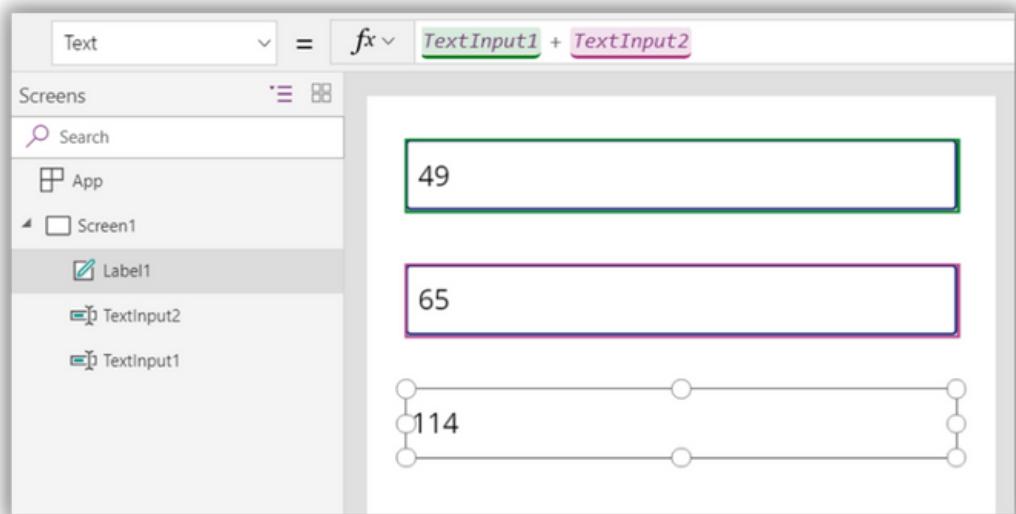
Em alguns casos, você precisará usar variáveis no Power Apps, que estende o modelo do Excel, adicionando fórmulas de comportamento. Essas fórmulas são executadas quando, por exemplo, um usuário seleciona um botão. Dentro de uma fórmula de comportamento, geralmente, é útil definir uma variável a ser usada em outras fórmulas.

Em geral, evite usar variáveis. Mas, às vezes, apenas uma variável pode habilitar a experiência que você deseja. As variáveis são implicitamente criadas e digitadas quando aparecem em funções que definem seus valores.

Power Apps:

Aplicativos que você cria no Power Apps se comportam de maneira muito parecida com o Excel. Em vez de atualizar células, é possível adicionar controles em qualquer local que desejar em uma tela e nomeá-los para uso em fórmulas.

Por exemplo, você pode replicar o comportamento do Excel em um aplicativo adicionando um controle **Label**, chamado **Label1**, e dois controles **Text input**, chamados **TextInput1** e **TextInput2**. Se você definir a propriedade **Texto** de **Label1** como **TextInput1 + TextInput2**, ela sempre mostrará a soma de quaisquer números que estiverem em **TextInput1** e **TextInput2** automaticamente.



Observe que o controle **Label1** está selecionado, mostrando sua fórmula **Text** na barra de fórmulas na parte superior da tela. Aqui, encontramos a fórmula **TextInput1 + TextInput2**. Esta fórmula cria uma dependência entre esses controles, assim como as dependências são criadas entre as células em uma pasta de trabalho do Excel. Vamos alterar o valor de **TextInput1**:

A fórmula para **Label1** foi automaticamente recalculada, exibindo o novo valor. No Power Apps, você pode usar fórmulas para determinar não apenas o valor principal de um controle, mas também propriedades como a formatação. No próximo exemplo, uma fórmula para a propriedade **Color** do rótulo mostrará automaticamente os valores negativos em vermelho. A função **If** deve ser semelhante à do Excel:

```
If( Value(Label1.Text) < 0, Red, Black )
```

Você pode usar fórmulas para uma ampla variedade de cenários:

Ao usar o GPS do seu dispositivo, um controle de mapa pode exibir seu local atual com uma fórmula que usa **Location.Latitude** e **Location.Longitude**. À medida que você se move, o mapa rastreará automaticamente seu local.

Outros usuários podem atualizar as fontes de dados. Por exemplo, outras pessoas de sua equipe podem atualizar itens em uma lista. Quando você atualiza uma fonte de dados, todas as fórmulas dependentes são recalculadas automaticamente para refletir os dados atualizados. Ao aprofundar ainda mais o exemplo, você poderá definir a propriedade **Items** de uma galeria como a fórmula **Filter(SharePointList)**, que exibirá automaticamente o conjunto de registros recém-filtrado.

Benefícios:

Usar fórmulas para criar aplicativos tem muitas vantagens:

-
- Se você conhece o Excel, também conhece o Power Apps. O modelo e a linguagem de fórmula são os mesmos.
- Se você já usou outras ferramentas de programação, pense na quantidade de códigos necessária para executar esses exemplos. No Visual Basic, você precisaria escrever um manipulador de eventos para o evento de alteração em cada controle de entrada de texto. O código para realizar o cálculo em cada um deles é redundante e pode ficar fora de sincronia, ou você precisaria escrever uma sub-rotina comum. No Power Apps, você realizou tudo com uma única fórmula de uma linha.
- Para compreender de onde o texto de Label1 é proveniente, você sabe exatamente onde procurar: a fórmula na propriedade Text. Não há nenhuma outra forma de afetar o texto deste controle. Em uma ferramenta de programação tradicional, qualquer manipulador de eventos ou sub-rotina poderia alterar o valor do rótulo, de qualquer lugar no programa. Isso pode dificultar o rastreamento de quando e onde uma variável foi alterada.
- Se o usuário altera um controle deslizante e, em seguida, muda de ideia, ele pode alterar o controle deslizante de volta para seu valor original. E é como se nada nunca tivesse sido alterado: o aplicativo mostrará os mesmos valores de controle como fazia antes. Não há nenhuma ramificação para experimentar e perguntar "e se", assim como não há nada no Excel.

Em geral, se você puder obter um efeito usando uma fórmula, será melhor. Deixe que o mecanismo da fórmula no Power Apps trabalhe para você.

Usar uma variável global:

Para criar nossa calculadora, precisamos de uma variável para manter o total acumulado. As variáveis mais simples para trabalhar com Power Apps são variáveis globais.

Como variáveis globais funcionam:

- Defina o valor da variável global com a função **Set**. **Set (MyVar, 1)** define a variável global **MyVar** como um valor 1.
- Você usa a variável global referenciando o nome usado com a função **Set**. Nesse caso, **MyVar** retornará 1.
- Variáveis globais podem conter qualquer valor, incluindo cadeias de caracteres, números, registros e tabelas.

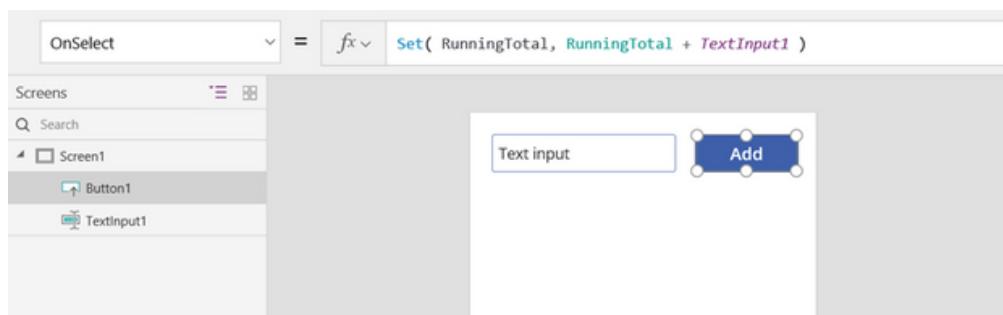
Recompilaremos nossa calculadora usando uma variável global:

- Adicione um controle de entrada de texto, chamado **TextInput1**, e dois botões, chamados **Button1** e **Button2**.
- Defina a propriedade **Text** de **Button1** como "Add" e defina a propriedade **Text** de **Button2** como "Clear".
- Para atualizar o total acumulado sempre que um usuário seleciona o botão Add, defina sua propriedade **OnSelect** como esta fórmula:

Set(RunningTotal, RunningTotal + TextInput1)

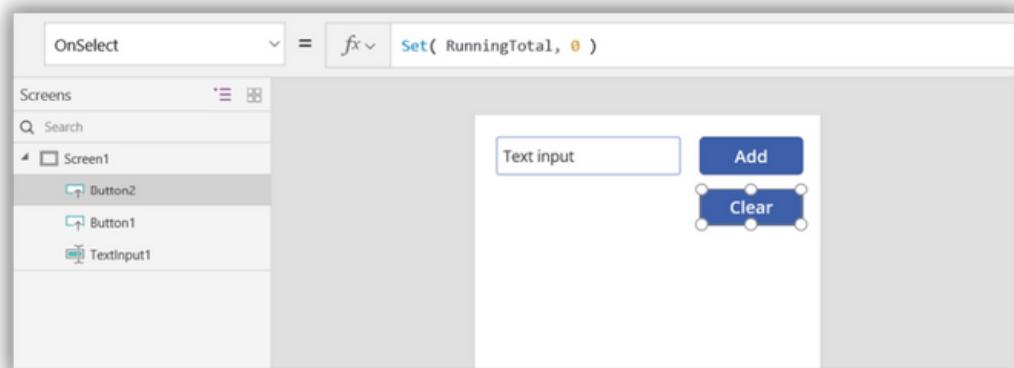
A mera existência dessa fórmula estabelece **RunningTotal** como uma variável global que contém um número por causa do operador +. Você pode referenciar **RunningTotal** em qualquer lugar do aplicativo. Sempre que o usuário abre este aplicativo, **RunningTotal** tem um valor inicial em branco.

A primeira vez que um usuário seleciona o botão **Add** e **Set** é executado, **RunningTotal** é definido como o valor **RunningTotal + TextInput1**.



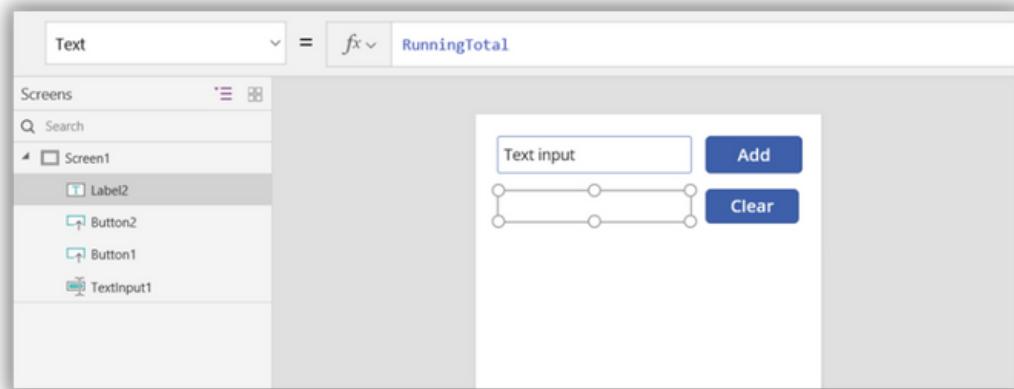
Para definir o total acumulado como **0**, sempre que o usuário seleciona o botão **Clear**, defina sua propriedade **OnSelect** como esta fórmula:

`Set(RunningTotal, 0)`



Adicione um controle **Label** e defina sua propriedade **Text** como **RunningTotal**.

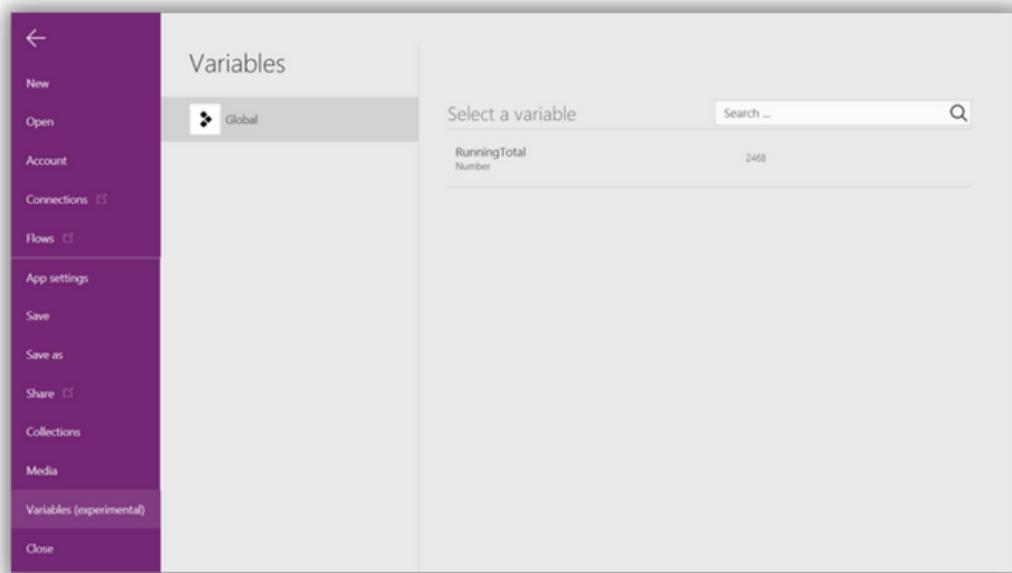
Esta fórmula será automaticamente recalculada e exibirá ao usuário o valor de **RunningTotal** à medida que ele mudar de acordo com os botões que o usuário selecionar.



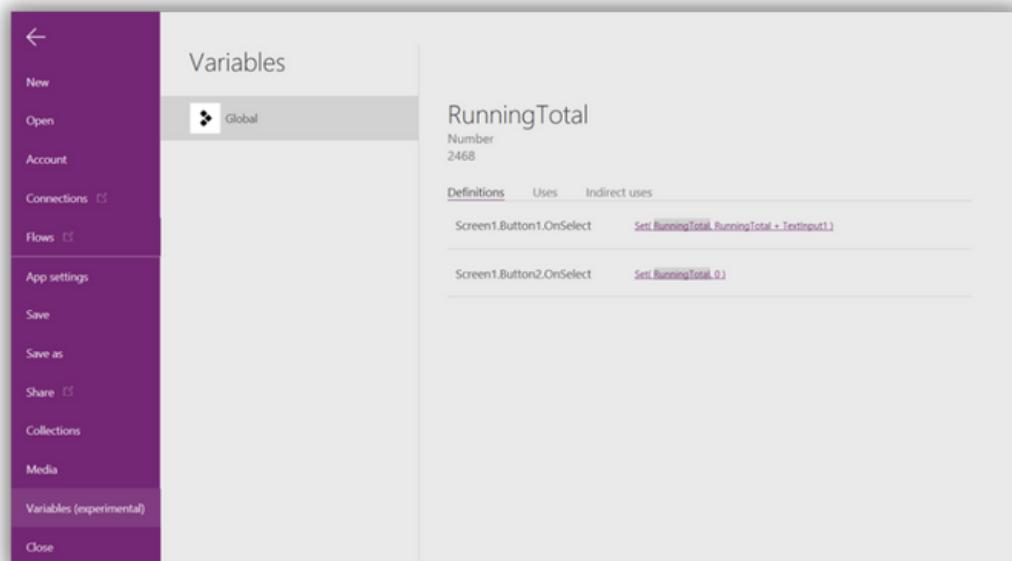
Visualize o aplicativo e nossa calculadora estará como descrito acima. Insira um número na caixa de texto e pressione o botão **Add** algumas vezes. Quando estiver pronto, retorne para a experiência de criação usando a tecla Esc.



Para mostrar o valor da variável global, selecione o menu **Arquivo** e selecione **Variáveis** no painel à esquerda.



Para mostrar todos os locais em que a variável é definida e usada, selecione-a.



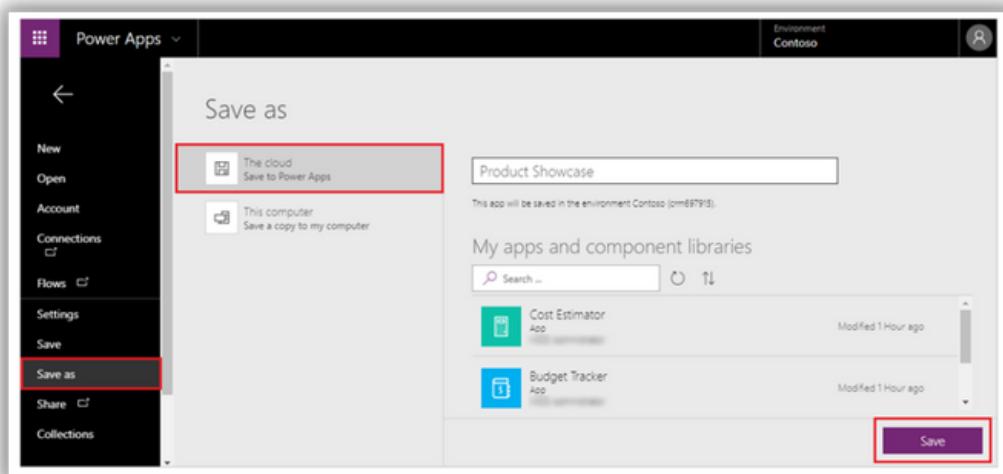
SALVAR E PUBLICAR APLICATIVOS DE TELA

Sempre que as alterações em um aplicativo de tela forem salvas, elas serão publicadas automaticamente apenas para você e qualquer outra pessoa que tenha permissões para editar o aplicativo. Ao terminar as alterações, você precisa publicá-las explicitamente a fim de disponibilizá-las para todas as pessoas com as quais compartilha o aplicativo.

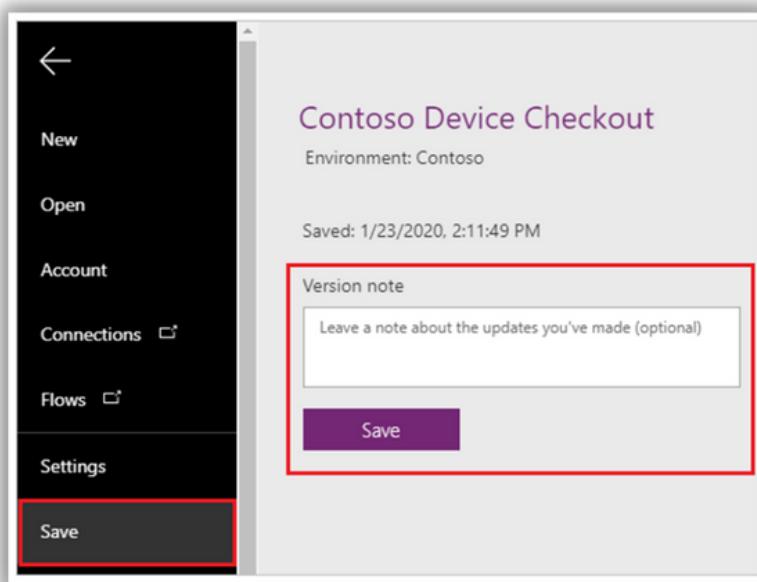
Salvar alterações em um aplicativo:

No Power Apps Studio, selecione **Salvar** no menu **Arquivo** (na borda esquerda) e, em seguida, siga uma destas etapas:

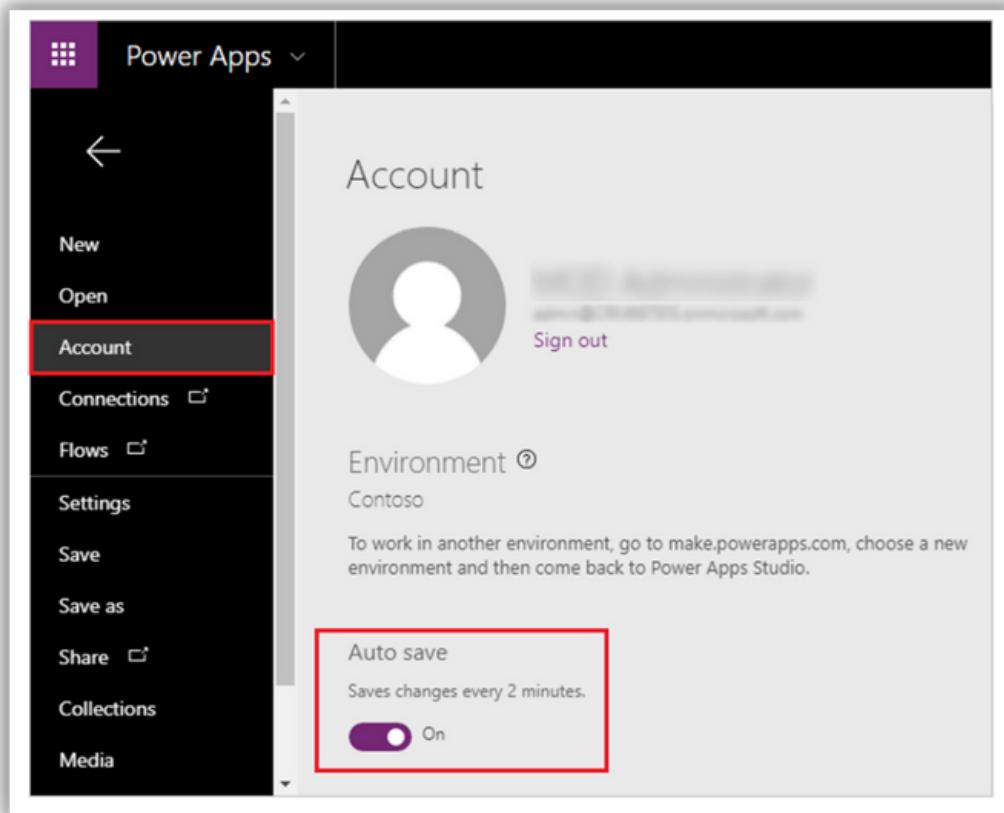
Caso o aplicativo nunca tenha sido salvo, selecionar **Salvar** no menu **Arquivo** automaticamente levará você para **Salvar como**. Selecione a localização como **A nuvem**, forneça um nome para ela e, em seguida, selecione **Salvar**.



Se o aplicativo nunca tiver sido salvo, selecione **Salvar**. Você também pode deixar anotações ou comentários específicos à versão.

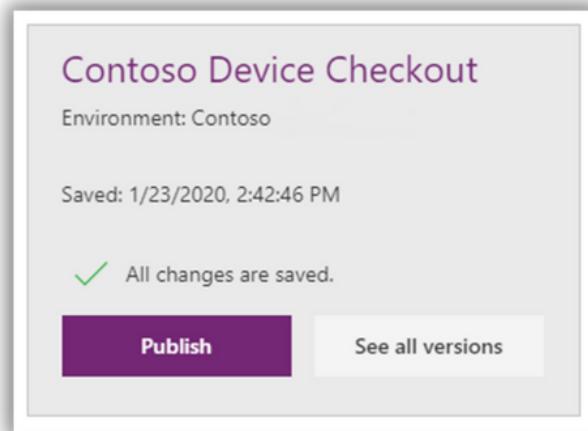


O Power Apps também pode salvar o aplicativo periodicamente a cada 2 minutos. Se você salvar o aplicativo uma vez, o Power Apps continuará salvar uma versão do aplicativo periodicamente sem exigir que o usuário pressione ou toque na ação Salvar. Os autores podem habilitar ou desabilitar a configuração **Salvamento automático** na guia **Conta** do menu **Arquivo**.

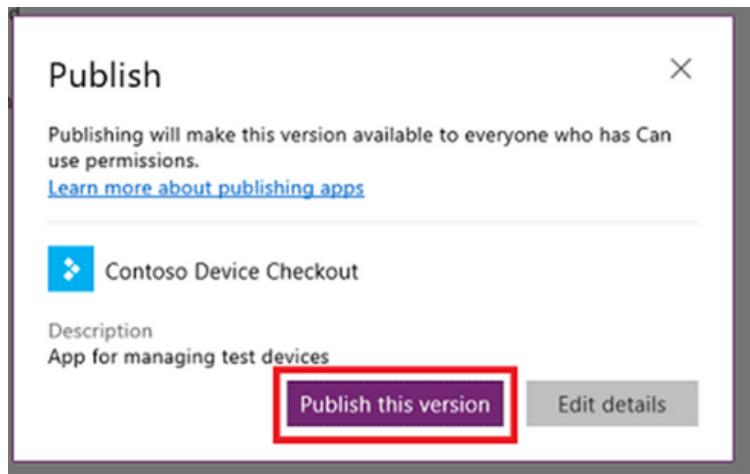


Publicar um aplicativo:

No Power Apps Studio, selecione **Salvar** no menu **Arquivo** (na borda esquerda) e, em seguida, selecione **Publicar**.



Na caixa de diálogo **Publicar**, selecione **Publicar esta versão** para publicar o aplicativo para todos os usuários com os quais ele é compartilhado.



Sempre que você publicar um aplicativo de tela, seu aplicativo será atualizado para ser executado na versão mais recente do Power Apps, o que significa que ele receberá o benefício de todos os recursos mais recentes e upgrades de desempenho adicionados desde sua última publicação. Se fizer muito tempo que não publica uma atualização, você provavelmente verá um benefício de desempenho imediatamente após a republicação.

Identificar a versão ao vivo:

Para ver todas as versões de um aplicativo, acesse Power Apps > **Aplicativos** > selecione seu aplicativo > **Detalhes** > **Versões**.

A versão **Ao vivo** é publicada para todas as pessoas com quem o aplicativo é compartilhado. A versão mais recente de qualquer aplicativo estará disponível somente para usuários que têm permissões para editá-lo.

Details	Versions		Connections	Flows	Analytics (preview)
Version	Modified	Modified by	Power Apps release	Published	Version note
Version 3	... 1/23/2020, 2:11:49 PM	Administrator	3.19123.15		Changed logo.
Version 2	... Publish this version	Administrator	3.19123.15		Updated edit form!
Version 1	... 1/14/2020, 6:51:41 PM	Administrator	3.18034.7	Live	

Para publicar a versão mais recente, realce a versão e selecione as reticências (...). Em seguida, selecione **Publicar esta versão** no menu suspenso.



O SIMPLES FUNCIONA

ACREDITE EM VOCÊ!

Não importa qual seja o tamanho da empresa que você quer trabalhar ou que você trabalha, confie em seu potencial, seja você, não tenha medo de errar, e é claro, estude muito Power Apps pois ele é fundamental.

Neste e-book você tem todos os recursos que você precisa para entender melhor sobre o Power Apps, se aprofundar em suas funcionalidades, resolver exercícios e estar preparado para qualquer situação.

É muito importante você manter contato com o Power Apps e praticar constantemente, assim como em qualquer software você acaba esquecendo e perdendo a mão se ficar muito tempo sem utilizar.

Você dá conta, confiamos em você!

Aqui na Viscari Inc. temos um prazer imenso em contribuir com a sua jornada e de milhares de outras pessoas, se o nosso E-book Power Apps Expert ou o nosso Curso de Power Apps Expert na Prática!, ou qualquer outro material nosso te ajudou de alguma maneira não deixe de falar conosco e compartilhar essa experiência em nosso instagram [@viscarinc](https://www.instagram.com/@viscarinc)

Sucesso!



VISCARI

Inc.

© VISCARI Inc - Todos os Direitos Reservados.