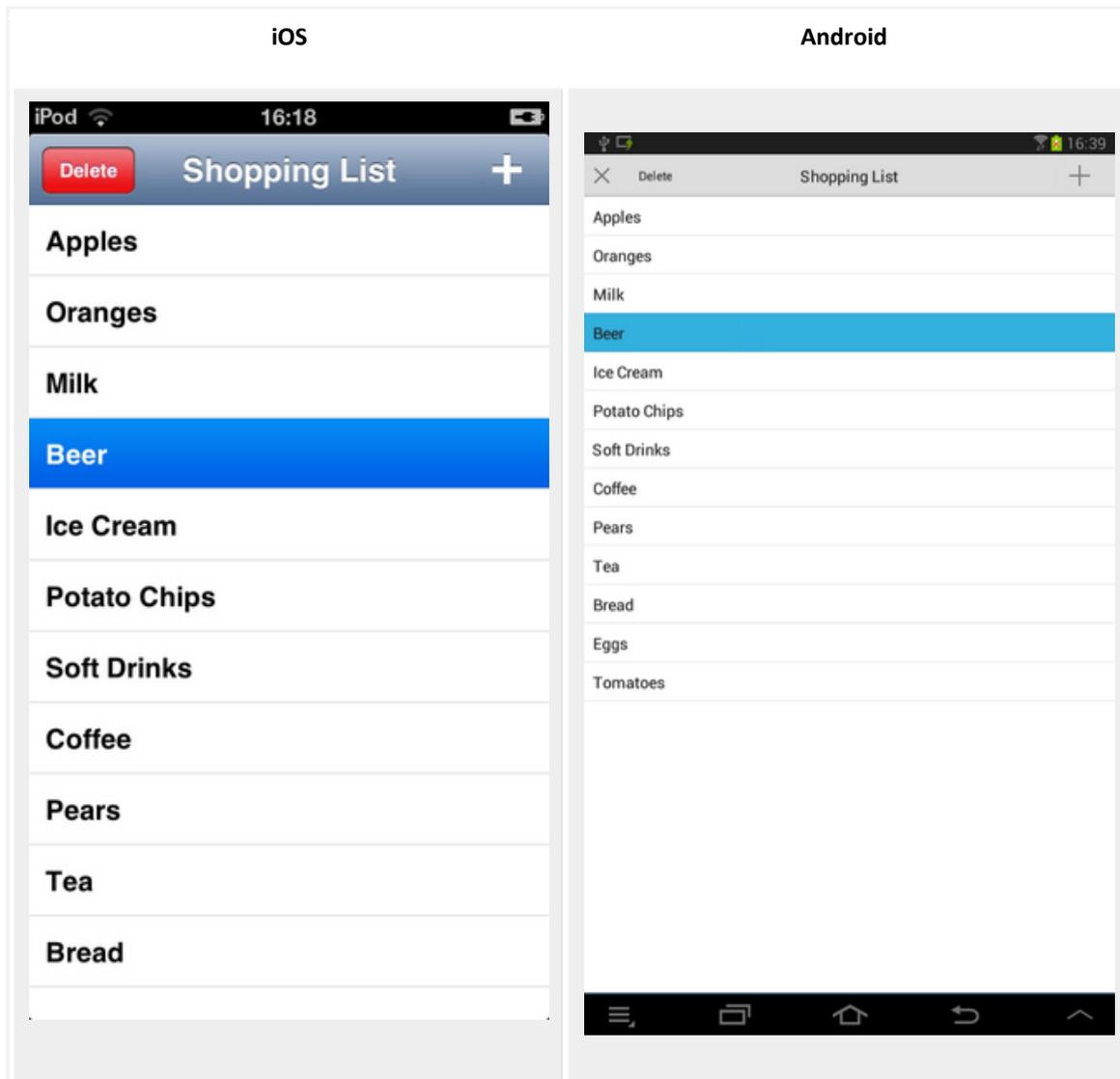


Tutorial Móvil: Usando SQLite (iOS y Android)

Antes de empezar este tutorial, usted debería leer y realizar el siguiente tutorial:

- [Tutorial Móvil: Usando Componentes ListBox para Mostrar un Table View \(iOS y Android\)](#)

Este tutorial describe los pasos básicos para usar SQLite como almacenamiento local de datos sobre su dispositivo móvil a través del framework dbExpress.



Usando dbExpress para Conectarse a la Base de Datos

dbExpress es un framework de acceso a datos muy rápido, escrito en Delphi. RAD Studio prevé drivers para bases de datos, como InterBase, Oracle, DB2, SQL Server, MySQL, Firebird, SQLite y ODBC. Usted puede acceder a diferentes bases de datos usando procedimientos similares al procedimiento descrito aquí.

- Para las plataformas móviles, dbExpress soporta **InterBase ToGo** así como **SQLite**. Esos productos de bases de datos pueden ejecutarse sobre dispositivos iOS y Android.
- Para otras bases de datos como Oracle, es necesario tener al menos una librería cliente. En plataformas Windows, la librería cliente se proporciona como un archivo DLL para conectarse. Por lo tanto, usted necesita para desarrollar aplicaciones utilizando tecnologías de nivel medio, como DataSnap para conectarse a estos productos de base de datos desde un dispositivo móvil.

Otro tutorial muestra cómo conectarse a una base de Datos Empresarial sin usar una librería cliente sobre el dispositivo móvil; vea [Tutorial Móvil: Conectándose a una Base de Datos Empresarial desde un Cliente Móvil \(iOS y Android\)](#).

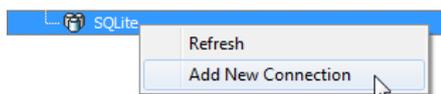
Creando la Base de Datos en el Entorno Windows para Propósitos de Desarrollo

Primero, usted necesita crear un archivo de base de datos SQLite sobre su plataforma de desarrollo Windows. Siga los siguientes pasos, puede usar el [FireMonkey Mobile Form Designer](#) para diseñar la interface de usuario de su App Móvil.

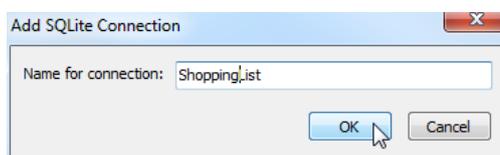
Crear la base de Datos en el Data Explorer

1. Vaya al [Data Explorer](#), click-derecho en el nodo **SQLite** y seleccione **Add New**

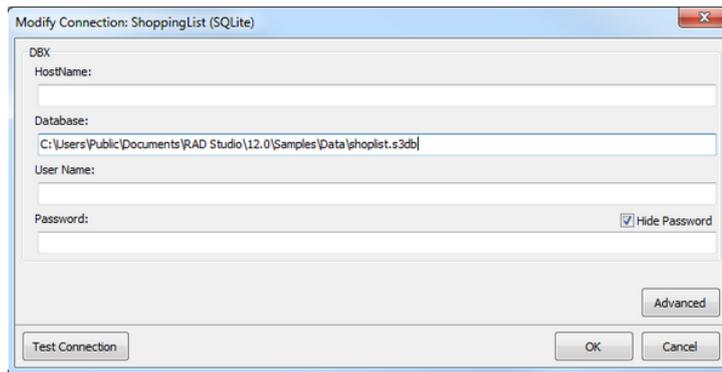
Connection:



2. Defina el nombre de la conexión, como por ejemplo **ShoppingList**.

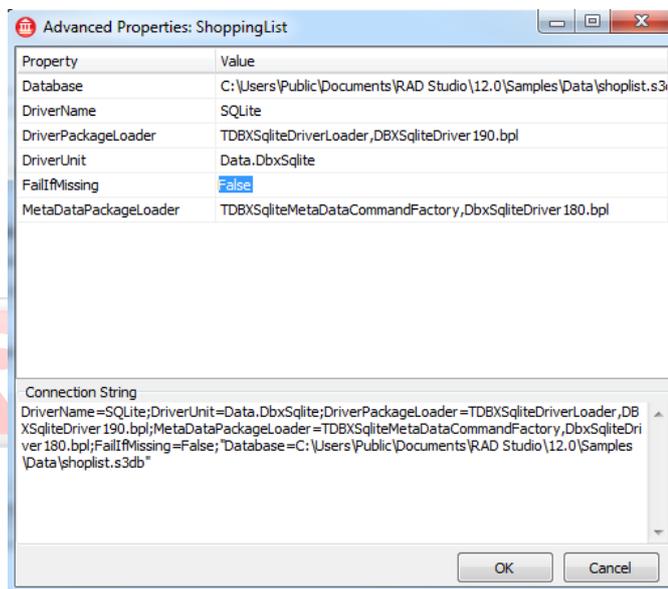


3. Especifique la ubicación del archivo de base de datos:



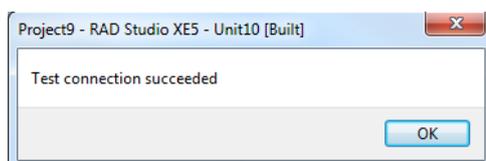
4. Click en el botón **Advanced** y abra la caja de diálogo **Advanced Properties**.
5. Cambie la propiedad **FailIfMissing** a **False** y click en OK para cerrar la caja de diálogo

Advanced Properties:



Nota: Seteando **FailIfMissing** a **False** instruye al Data Explorer a crear un nuevo fichero de base de datos si no existe.

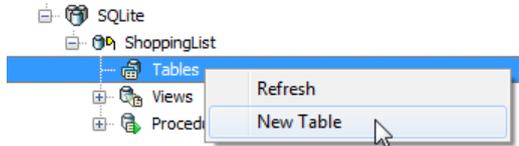
6. Regrese a la caja de diálogo en el **Modify Connection**, click en el botón **Test Connection**. Con ésta operación, el nuevo archivo de base de datos es creado si no existe.



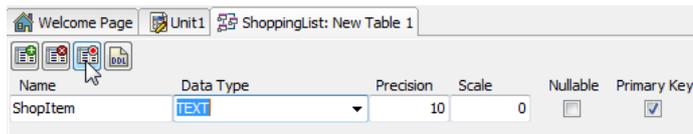
Nota: Asegúrese que **sqlite3.dll** está presente en su sistema de desarrollo. Si el archivo no está presente, descargue el **sqlite3.dll** desde <http://www.sqlite.org/download.html> a la ruta system (como **C:\Windows\SysWOW64** para 64-bit Windows).

Crear una Tabla sobre el DataExplorer

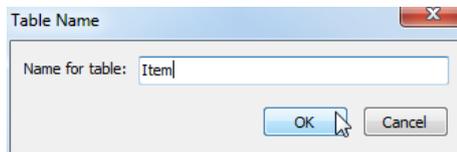
1. En el [Data Explorer](#), doble-click en el nodo **ShoppingList** bajo la sección SQLite, click-derecho en **Tables**, y luego seleccione **New Table** desde el menú contextual.



2. Setear **Data Type** para una columna **ShopItem** a **TEXT**.



3. Click el botón **Save** y especificar un nombre de tabla (por ejemplo, **Item**.)



Diseñando y Configurando la Interface de Usuario



Los componentes de Interface de Usuario visibles son cargados sobre el diseñador

Este tutorial usa un componente [TListBox](#) como el elemento de Interface de Usuario.

Para configurar un componente ListBox y otros elementos UI, use los siguientes pasos:

1. Crear una aplicación FireMonkey Mobile usando **File > New > FireMonkey Mobile Application - Delphi**.
2. Suelte un [TToolBar](#) sobre el Formulario.

3. Suelte un **TButton** sobre el componente ToolBar y establezca las siguientes propiedades en el **Object Inspector**:
 - Setear la propiedad **Name** a **ButtonAdd**.
 - Setear el **StyleLookup** a **addtoolbuttonbordered**.
4. Suelte un **TButton** sobre el componente ToolBar y establezca las siguientes propiedades en el **Object Inspector**:
 - Setear la propiedad **Name** a **ButtonDelete**.
 - Setear el **StyleLookup** a **deletetoolbutton**.
 - Setear el **Text** a **Delete**.
 - Setear el **Visible** a **False**.
5. Suelte un **TLabel** sobre el componente ToolBar y establezca las siguientes propiedades en el **Object Inspector**:
 - Setear el **Align** a **alClient**.
 - Setear el **StyleLookup** a **toollabel**.
 - Setear el **Text** a **Shopping List**.
 - Setear el **TextAlign** a **taCenter**.
6. Arrastre un componente **TListBox** sobre el formulario y setear las siguientes propiedades en el **Object Inspector**:
 - Setear la propiedad **Align** a **alClient**, entonces el componente ListBox usará el formulario completo.

Conectando a los Datos

Los siguientes son los pasos básicos para conectarse a los datos en una base de datos que ya está definida en el **Data Explorer**:

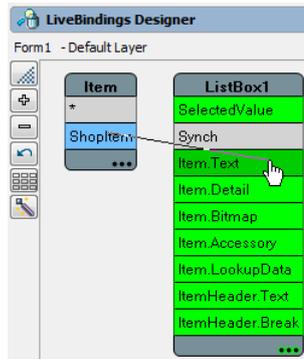
1. Seleccione la tabla **Item** sobre el **Data Explorer** y suéltelo en el Form Designer.



Nota: Esto crea dos componentes (ShoppingList: [TSQLConnection](#) e Item: [TSQLDataSet](#)) sobre el Formulario.



2. Seleccione el componente **ShoppingList** en el Formulario, y entonces cambie la propiedad **Connected** a **True**.
3. Seleccione el componente **Item** en el Formulario, y luego cambie la propiedad **Active** a **True**.
4. Seleccione **View > LiveBindings Designer** y el **LiveBindings Designer** se abre.
5. Seleccione **ShopItem** en el componente **Item** y suelte **ShopItem** a **ListBox1**.



Siguiendo esos pasos conectamos la interface de usuario de la app con la información contenida en la base de datos SQLite. Si usted ha usado una tabla con datos existentes para este tutorial, ahora podría ver la información actual con el Diseñador de Formulario.

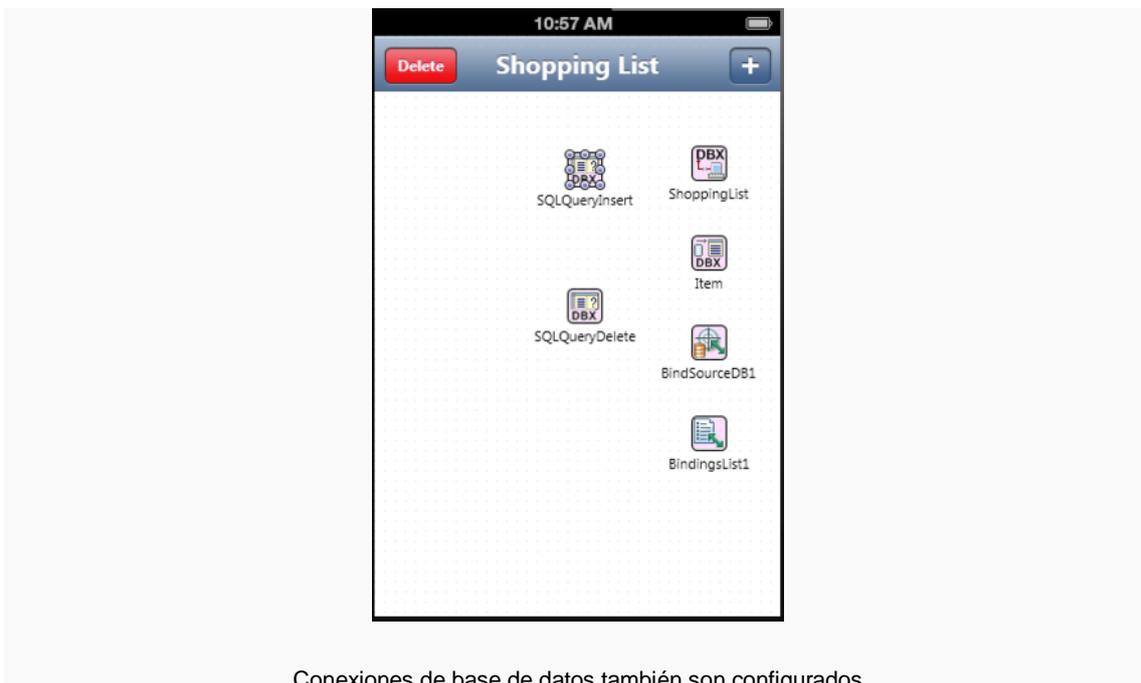
Creando el Manejador de Evento para Hacer Visible el Botón Delete Cuando el Usuario Seleccione un Item desde la Lista

La propiedad **Visible** para el botón **Delete** es seteado a **False**. Por lo tanto, por defecto, el usuario final no ve este botón. Usted puede hacerlo visible cuando el usuario seleccione un ítem de la lista, de la siguiente manera:

- Seleccione **ListBox1** y defina el siguiente manejador de evento para el evento **OnItemClick**.

```
procedure TForm1.ListBox1ItemClick(const Sender: TCustomListBox;  
  const Item: TListBoxItem);  
begin  
  ButtonDelete.Visible := ListBox1.Selected <> nil;  
end;
```

Creando el Manejador de Evento para el Botón Add para Agregar una Entrada a la Lista



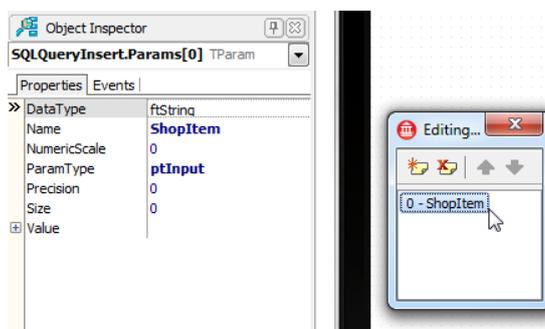
Conexiones de base de datos también son configurados

El siguiente paso es agregar una característica a esta aplicación para añadir un ítem al shopping list.

1. Suelte un componente [TSQLQuery](#) al formulario.
2. Establecer las siguientes propiedades en el [Object Inspector](#):
 - Setear la propiedad [Name](#) a **SQLQueryInsert**.
 - Setear la propiedad [SQLConnection](#) a **ShoppingList**.
 - Setear la propiedad SQL como se muestra:

```
INSERT INTO ITEM (ShopItem) VALUES (:ShopItem)
```

- Seleccione el botón **Expand (...)** sobre la propiedad [Params](#).
- Seleccione el parámetro **ShopItem** y setear [DataType](#) a **ftString**:

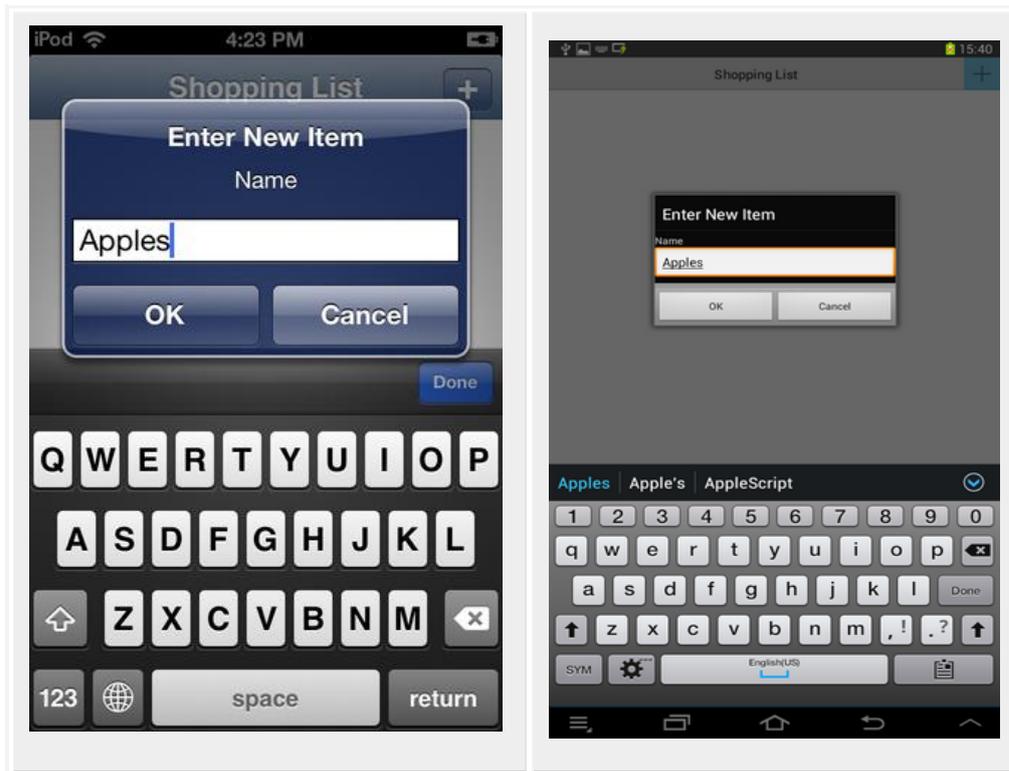


3. En el Form Designer, doble-click al componente **AddButton**. Añada el siguiente código para este manejador de evento:

```
procedure TForm1.ButtonAddClick(Sender: TObject);
var
  TaskName: String;
begin
  try
    if InputQuery('Enter New Item', 'Name', TaskName) and
(TaskName.Trim <> '') then
      begin
        SQLQueryInsert.ParamByName('ShopItem').AsString :=
TaskName;
        SQLQueryInsert.ExecSQL();
        Item.Refresh;
        ButtonDelete.Visible := ListBox1.Selected <> nil;
      end;
    except
      on e: Exception do
        begin
          ShowMessage(e.Message);
        end;
      end;
    end;
end;
```

La función [InputQuery](#) muestra un cuadro de diálogo que pide al usuario final que introduzca un texto. Esta función devuelve True cuando el usuario selecciona OK, así que usted puede agregar datos a la base de datos sólo cuando el usuario selecciona OK y el texto contenga algunos datos.

iOS	Android
-----	---------



Creando el Manejador de evento para el Botón Delete para remover una Entrada desde la Lista

El siguiente paso es agregar una característica a esta aplicación para remover un ítem desde el shopping list:

1. Suelte un componente [TSQLQuery](#) al formulario.
2. Establecer las siguientes propiedades en el [Object Inspector](#):
 - Setear la propiedad [Name](#) a **SQLQueryDelete**.
 - Setear la propiedad [SQLConnection](#) a **ShoppingList**.
 - Setear la propiedad SQL como se muestra:

```
DELETE FROM ITEM WHERE ShopItem = :ShopItem
```

- Seleccione el botón **Expand (...)** sobre la propiedad [Params](#).
 - Seleccione el parámetro **ShopItem** y setear [DataType](#) a **ftString**.
3. En el Form Designer, doble-click al componente **DeleteButton**. Agregue el siguiente código a este manejador de evento.

```
procedure TForm1.ButtonDeleteClick(Sender: TObject);
var
    TaskName: String;
begin
```

```
TaskName := ListBox1.Selected.Text;

try
    SQLQueryDelete.ParamByName('ShopItem').AsString := TaskName;
    SQLQueryDelete.ExecSQL();
    Item.Refresh;
    ButtonDelete.Visible := ListBox1.Selected <> nil;
except
    on e: Exception do
    begin
        SHowMessage(e.Message);
    end;
end;
end;
```

Configurando el Despliegue de Su Base de Datos para móviles

Hasta este punto, se ha utilizado SQLite en su escritorio. Esto significa que la base de datos real se encuentra en la unidad de su disco duro local (por ejemplo, C:\Users\Public\Documents\RAD Studio\12.0\Samples\Data\shoplist.s3db). En el dispositivo móvil, las aplicaciones son sand-boxed, y por lo general sólo se pueden leer y escribir datos que se encuentran en la carpeta **Documents** (sólo para dispositivos iOS) y **internal storage** (para dispositivos Android) en su carpeta de aplicaciones.

Para conectarse a una base de datos local en el móvil, es necesario realizar las siguientes acciones:

- Desplegar la base de datos al Dispositivo móvil.
- Cambiar la configuración (para conectarse al archivo de base de datos) a un archivo local bajo el folder **Documents** (para Dispositivos iOS) o **internal storage** (para Dispositivos Android).

Añada y Configure Su Archivo de Base de Datos en el Deployment Manager

Antes de poder ejecutar la aplicación en el móvil, es necesario configurar el despliegue de su archivo de base de datos (shoplist.s3db).

1. Abrir el [Deployment Manager](#) seleccionando **Project > Deployment**.
2. Seleccione **Add Files**, y seleccione el archivo de base de datos (por ejemplo, C:\Users\Public\Documents\RAD Studio\12.0\Samples\Data\shoplist.s3db).



3. Seleccione **shoplist.s3db**, y cambie **Remote Path** a **Startup\Documents** (para iOS platform) o **assets\internal** (para Android platform).

RemotePath sobre iOS device platform

Local Path	Local Name	Type	Platforms	Remote Path	Remote Name
C:\Users\Public\Documents\RAD Studi...	shoplist.s3db	File	[iOSDevice,iOS55m...	Startup\Documents\	shoplist.s3db

RemotePath sobre Android platform

Local Path	Local Name	Type	Platforms	Remote Path	Remote Name
C:\Users\Public\Documents\RAD Studi...	shoplist.sdb	File	[Android,Win32]	assets\internal\	shoplist.sdb

4. Seleccione la columna **Platforms** (doble-click en el ellipsis [...]) en la línea de shoplist.s3db:
 1. Asegúrese que **iOS Simulator** y **iOS Device** o **Android** están presentes para shoplist.s3db.
 2. Remueva Win32 desde la lista si está presente (usted no tiene que copiar archivos de base de datos para la plataforma Win32).
5. Seleccione **All-Configurations - iOS Device platform** o **All-Configurations - Android platform** y asegúrese que **shoplist.s3db** está seteada para ser desplegada a **Startup\Documents** o **assets\internal**.

Como usted lo ha configurado, cuando ejecute la aplicación en el dispositivo móvil, el archivo de base de datos (shoplist.s3db) esta fijada para ser desplegada en la carpeta **Documents** (para la plataforma iOS) o **internal storage** (para la plataforma Android) en el área sandbox (caja de arena) de su aplicación móvil.

Modificando Su Código para Conectarse al Archivo de Base de Datos Local en el Móvil

Las características básicas de esta aplicación ahora son implementadas. Como usted trabajó en el explorador de datos, ha creado un archivo de base de datos en Windows. El archivo de base de datos no está disponible en su dispositivo móvil a menos que usted lo copie en el dispositivo móvil o lo cree sobre la marcha.

Puede crear una base de datos SQLite y Tabla con los siguientes pasos::

Especificando la Ubicación de la base de datos SQLite sobre el Dispositivo Móvil

1. En el Form Designer, seleccione el componente **ShoppingList**.
2. En el [Object Inspector](#), doble-click al evento [BeforeConnect](#).
3. Añada el siguiente código al manejador de evento:

```
procedure TForm1.SQLConnectionSQLiteBeforeConnect(Sender:
TObject);
begin
  {$IF DEFINED(iOS) or DEFINED(ANDROID)}
  ShoppingList.Params.Values['Database'] :=
    TPath.Combine(TPath.GetDocumentsPath, 'shoplist.s3db');
  {$ENDIF}
end;
```

El registro [TPath](#) está declarado en la unidad **System.IOUtils**, solo necesita agregar **System.IOUtils** en la uses clause de su unidad.

Crear una Tabla si No Existe

Con SQLite puede crear una tabla cuando la tabla no existe, usando la sentencia `CREATE TABLE IF NOT EXISTS`. Puede crear una tabla después que el componente `SQLConnection` se conecte a la base de datos y antes que el componente `TSQLDataSet` se conecte a la tabla. Use los siguientes pasos:

1. En el Form Designer, seleccione el componente **ShoppingList**.
2. En el [Object Inspector](#), double-click en el evento [AfterConnect](#).
3. Agregue el siguiente código para manejar este evento:

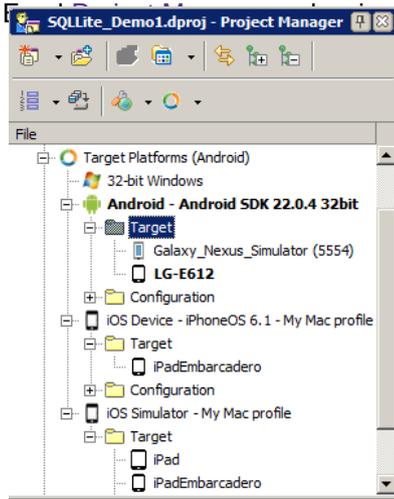
```
procedure TForm1.ShoppingListAfterConnect(Sender: TObject);
begin
  ShoppingList.ExecuteDirect('CREATE TABLE IF NOT EXISTS Item
(ShopItem TEXT NOT NULL)');
end;
```

Ejecutando Su Aplicación en el Simulador o sobre un Dispositivo Móvil

Ahora su aplicación está lista para ejecutarse sobre el simulador o su dispositivo móvil conectado.

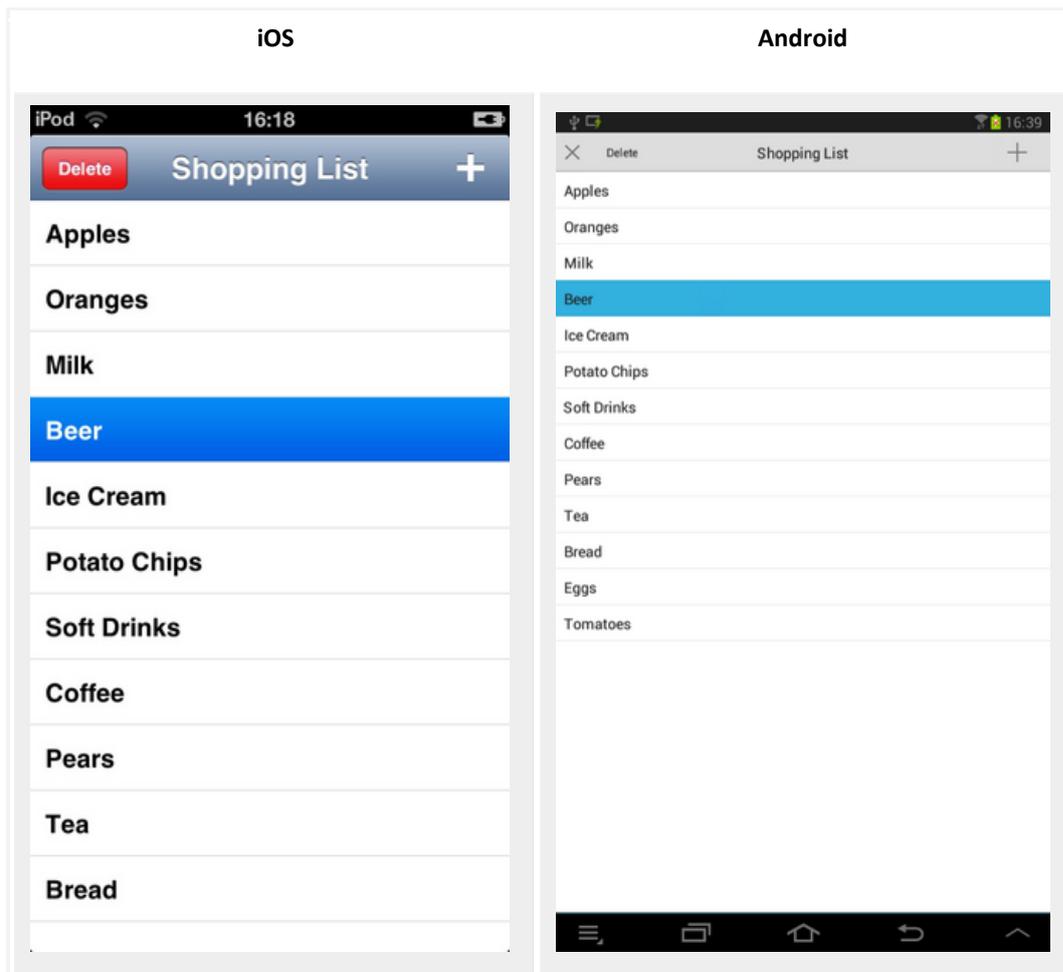
Para ejecutar su aplicación

1. Seleccione su plataforma objetivo.



2. Escoja alguno de los siguientes comandos:

- **Run > Run**
- **Run > Run Without Debugging**



Nota: Si tiene algún problema ejecutando la aplicación, siga los pasos ofrecidos en [Troubleshooting](#).

