

# Etec

Vasco Antônio  
Venchiariutti  
Jundiaí

## CRIANDO UM APLICATIVO COM ACESSO A BANCO DE DADOS LOCAL USANDO O DELPHI



RAFAEL CÉRGOLI  
FABIANO P. ORSI  
ROBERTO MELLE  
JOSÉ R. MADUREIRA JR

## Sumário:

<b>1. INTRODUÇÃO .....</b>	<b>3</b>
<b>2. ORGANIZAÇÃO DAS PASTAS PARA O APP .....</b>	<b>3</b>
<b>3. BANCO DE DADOS:.....</b>	<b>3</b>
CRIANDO O BANCO DE DADOS .....	4
CRIANDO A TABELA .....	4
INSERINDO DADOS .....	6
<b>4. CRIAÇÃO DO APP .....</b>	<b>7</b>
CRIANDO UM NOVO APLICATIVO: .....	7
ALTERANDO O NOME DO FORMULÁRIO: .....	8
SALVANDO O PROJETO:.....	8
CRIANDO O DATAMODULE:.....	9
INSERINDO OS COMPONENTES NO DATA MODULE: .....	11
CONFIGURANDO O FDConnection: .....	11
CONFIGURANDO O FDQuery:.....	12
CONFIGURANDO O FDPHYSQLITEDRIVERLINK: .....	13
DESENVOLVENDO O FORM PRINCIPAL: .....	14
USANDO LIVEBINDINGS: .....	18
ATIVANDO O FDQuery: .....	20
EXIBINDO OS DADOS: .....	20
CONFIGURANDO O LIVEBINDINGS DOS DETALHES: .....	23
FAZENDO MAIS AJUSTES: .....	24
PROGRAMANDO O LISTVIEW: .....	25
PROGRAMANDO OS BOTÕES: .....	26
OCULTANDO AS ABAS .....	27
<b>5. CONFIGURAÇÃO DO APP PARA SER ENVIADO PARA O DISPOSITIVO.....</b>	<b>28</b>
INSERINDO O BANCO DE DADOS NO APLICATIVO .....	28
CONFIGURANDO O BANCO DE DADOS .....	30
ATIVANDO A QUERY .....	31

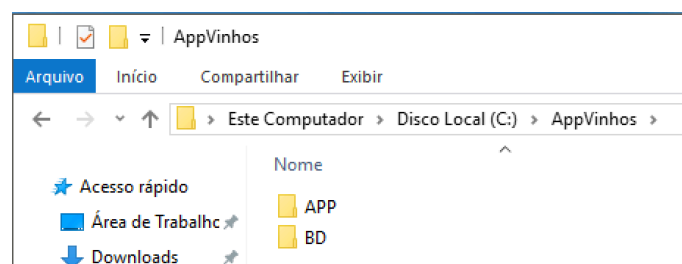
## 1. INTRODUÇÃO

Esta apostila apresenta um passo a passo para a construção de um aplicativo para dispositivo móvel com acesso a banco de dados local. Será utilizado o Delphi para a criação do aplicativo e o SQLite para a criação do Banco de Dados.

## 2. ORGANIZAÇÃO DAS PASTAS PARA O APP

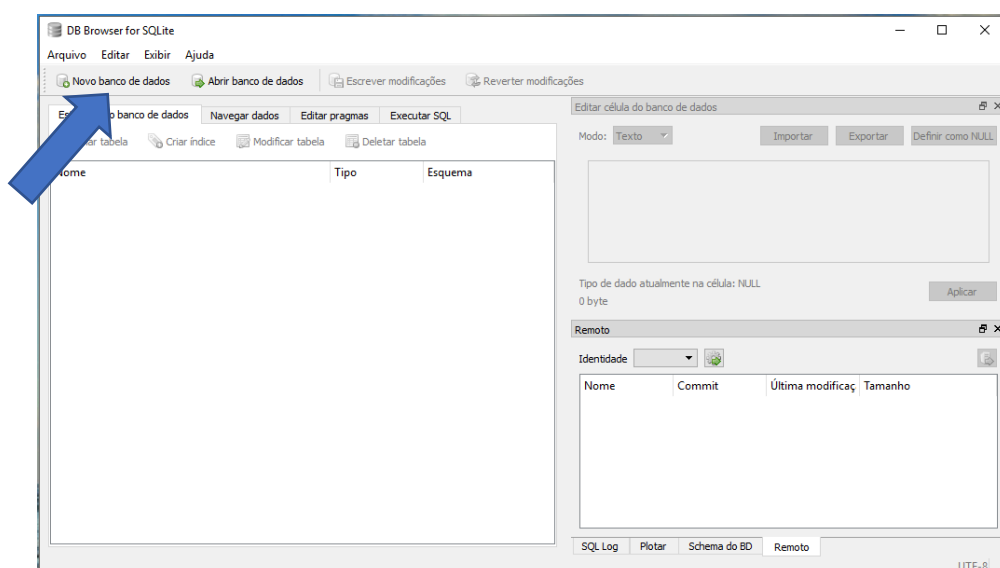
Para organizar a construção do aplicativo, primeiramente iremos criar algumas pastas no computador.

- Crie uma pasta para o projeto chamada **AppVinhos**
- Dentro desta pasta crie duas outras pastas chamadas **APP** e **BD**



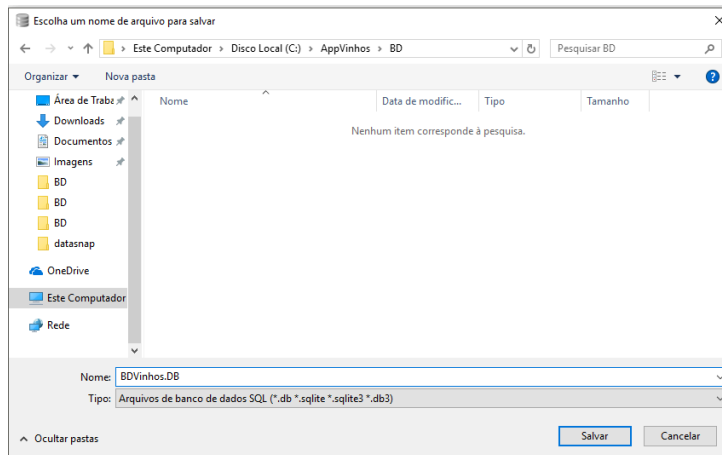
## 3. BANCO DE DADOS:

O SGBD a ser utilizado neste exemplo será o SQLite através do aplicativo DB Browser for SQLite



## Criando o banco de dados

- Clique em **Novo banco de dados**



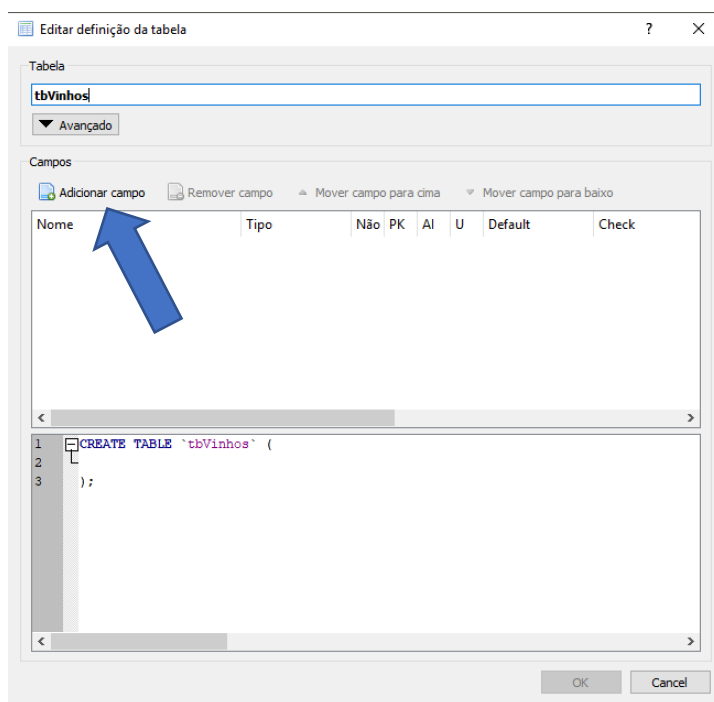
- Escolha a pasta **BD** criada dentro de AppVinhos

- Digite **BDVinhos.DB** na caixa Nome

- Clique em Salvar

## Criando a tabela

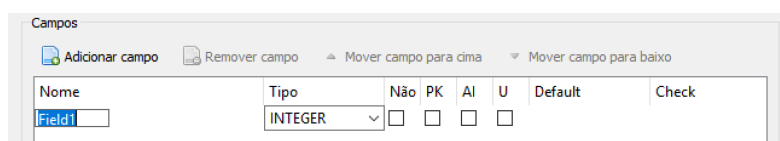
Após salvar o Banco de Dados será exibida uma tela para a criação da tabela



- Na caixa nome da tabela digite **tbVinhos**

- Para criar os campos da tabela clique no botão **Adicionar campo** indicado ao lado

Será criado o primeiro campo:



- Em nome digite **Marca**

- Em tipo digite **Varchar(50)**

O campo Marca já está definido:

Nome	Tipo	Não	PK	AI	U	Default	Check
Marca	Varchar(50)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

Agora iremos criar mais 5 campos:

Campo	Tipo
<b>Tipo</b>	Varchar(50)
<b>Uvas</b>	Varchar(50)
<b>Ano</b>	Integer
<b>PaisOrigem</b>	Varchar(50)
<b>Classificacao</b>	Integer

Editar definição da tabela

Tabela: **tbVinhos**

Avançado

Campos

Nome	Tipo	Não	PK	AI	U	Default	Check
Marca	Varchar ( 50 )	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Tipo	INTEGER	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Uvas	Varchar ( 50 )	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Ano	INTEGER	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
PaisOrigem	Varchar ( 50 )	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Classificacao	INTEGER	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

```

1 CREATE TABLE `tbVinhos` (
2   `Marca` Varchar ( 50 ),
3   `Tipo` INTEGER,
4   `Uvas` Varchar ( 50 ),
5   `Ano` INTEGER,
6   `PaisOrigem` Varchar ( 50 ),
7   `Classificacao` INTEGER
8 );

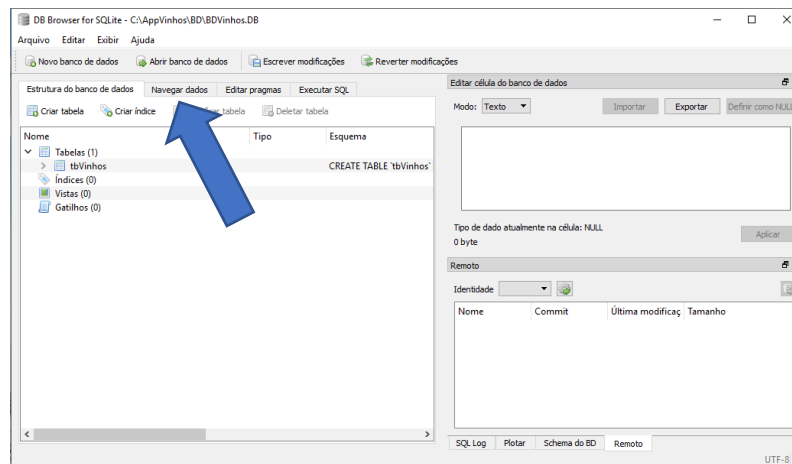
```

OK Cancel

- Clique em **OK**

## Inserindo dados

Após a tabela ter sido criada, iremos inserir alguns dados.



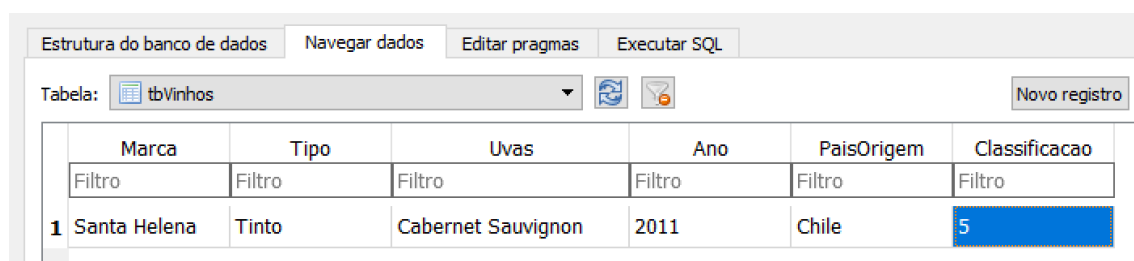
- Clique na aba **Navegar dados** indicada ao lado

Será exibida a tela para a visualização e manipulação dos dados:

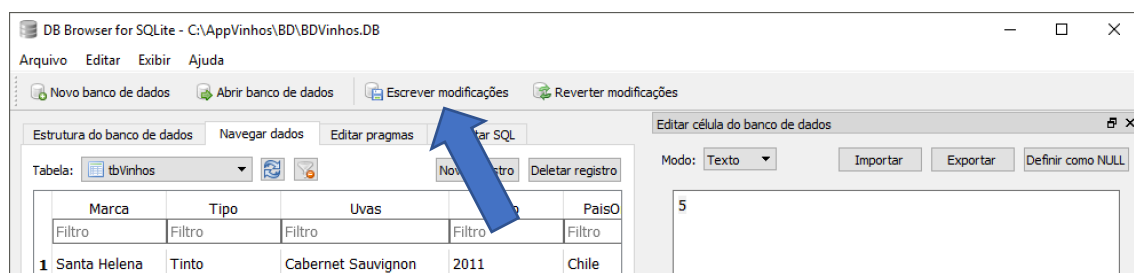


- Clique no botão **Novo registro**

- Digite os seguintes dados:



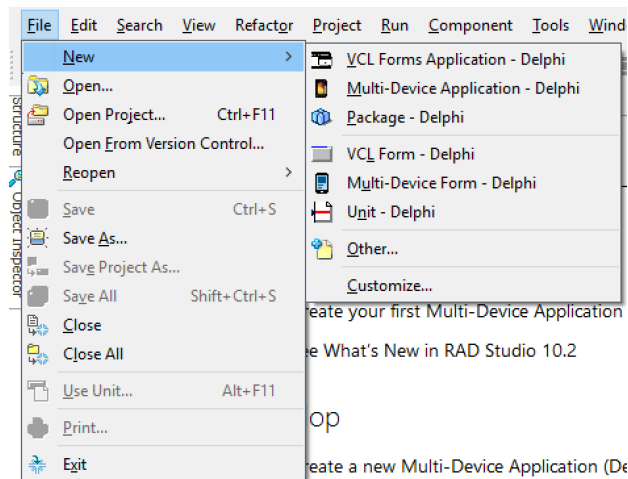
- Para finalizar, clique no botão **Escrever modificações**



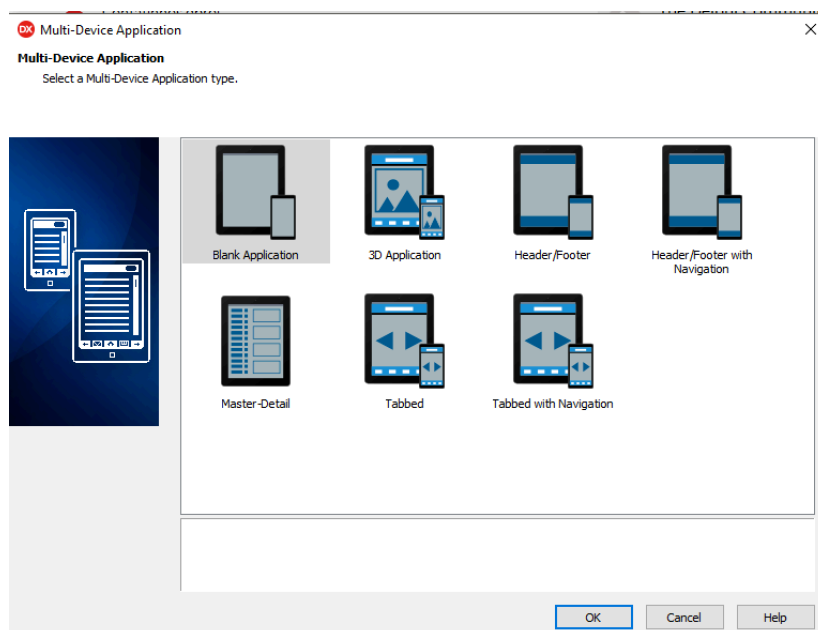
## 4. CRIAÇÃO DO APP

### Criando um novo aplicativo:

- Abra o Delphi
- Clique em **File | New**



- Clique em **Multi-Device Application – Delphi**

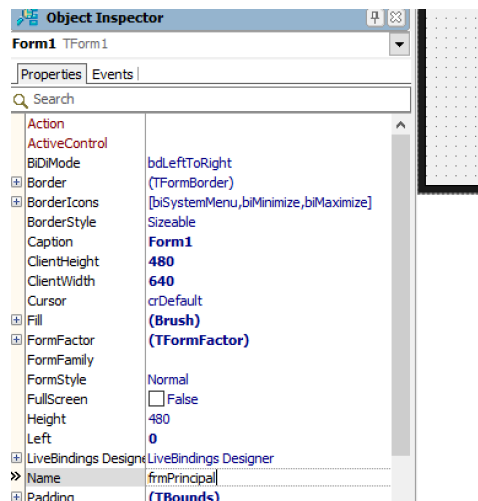


- Clique na opção **Blank Application**
- Clique em **OK**

## Alterando o nome do formulário:

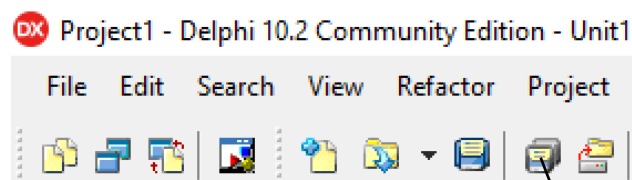
Na janela de Propriedades localize a propriedade name

- Altere a propriedade Name do form para **frmPrincipal**



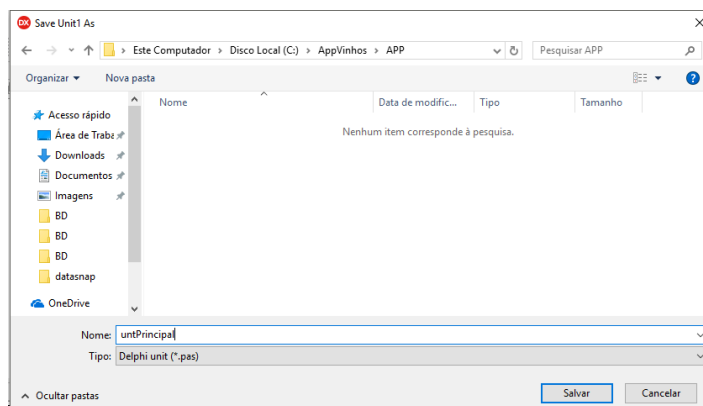
## Salvando o projeto:

Localize a seguinte barra de ferramentas:



Botão  
Save All

- Clique no botão **Save all** indicado na figura acima



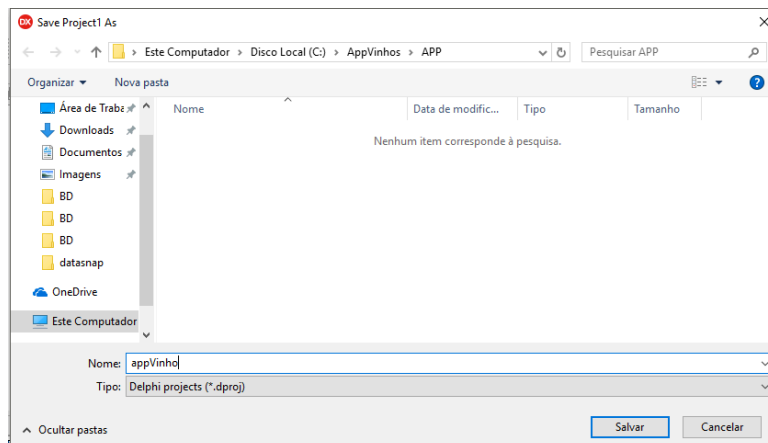
- Escolha a pasta **APP** criada anteriormente

- Digite na caixa nome: **untPrincipal**

- Clique em **Salvar**



Depois de salvar a unit, deverá ser salvo o projeto:

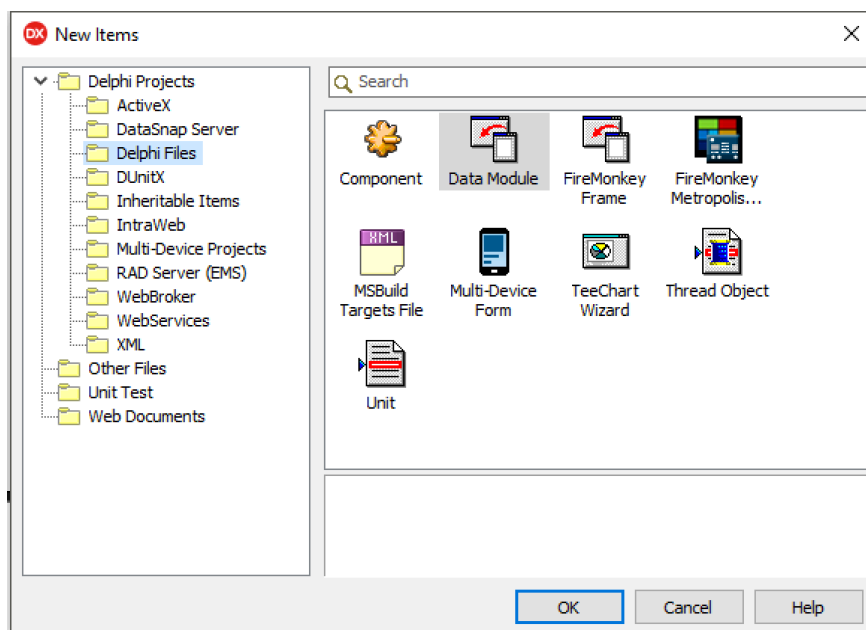


- Em nome digite:  
**appVinho**

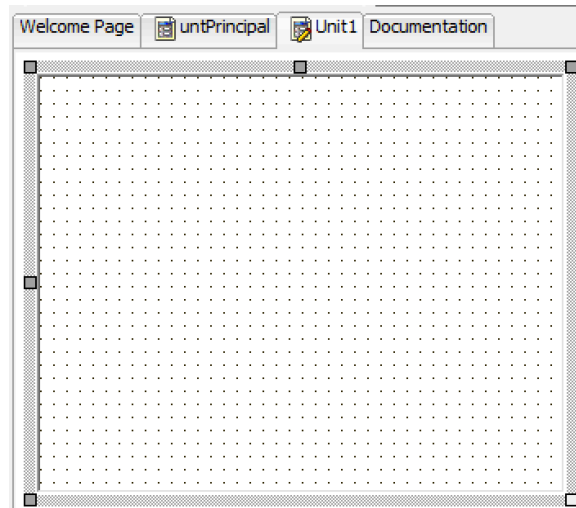
- Clique em **Salvar**

## Criando o DataModule:

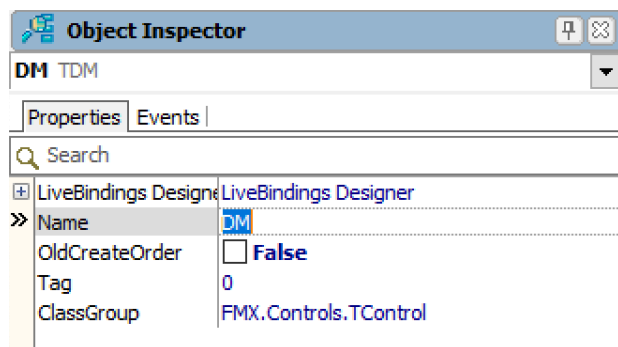
- Clique em **File | New | Other...**
- Escolha a opção **Delphi Files**



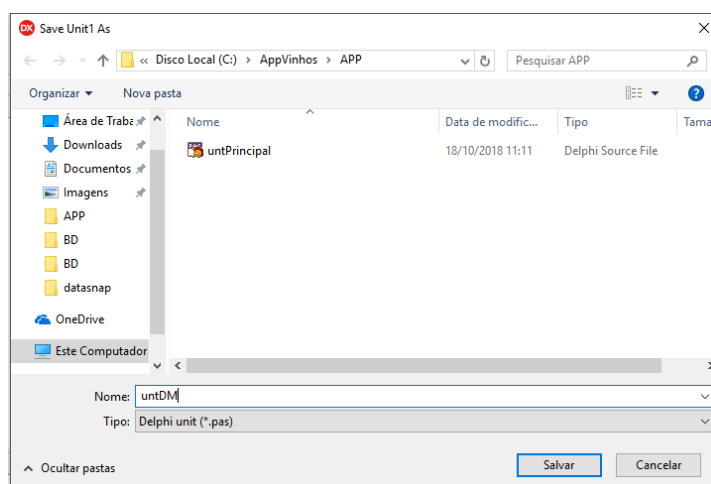
- Clique em **Data Module**
- Clique no botão **OK**



- Altere a propriedade Name para **DM**



- Clique no botão **Save All**



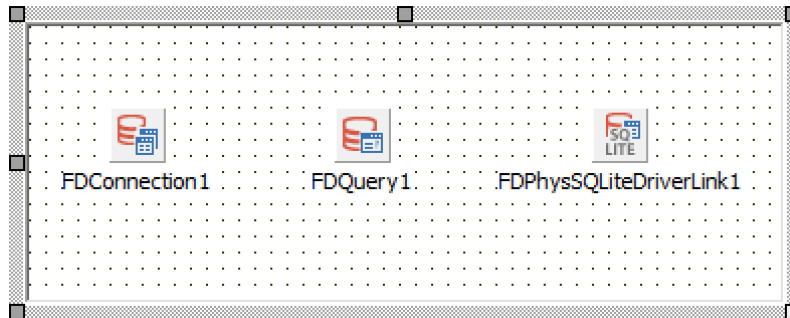
- Digite em nome: **untDM**

- Clique em **Salvar**

## Inserindo os componentes no Data Module:

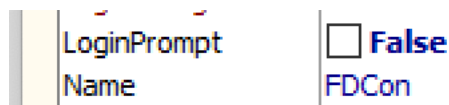
Serão inseridos 3 componentes de acesso e manipulação de dados no Data Module:

**FDConnection** e **FDQuery** presentes na categoria FireDAC e **FDPhysSQLiteDriverLink** presente na categoria FireDAC Links

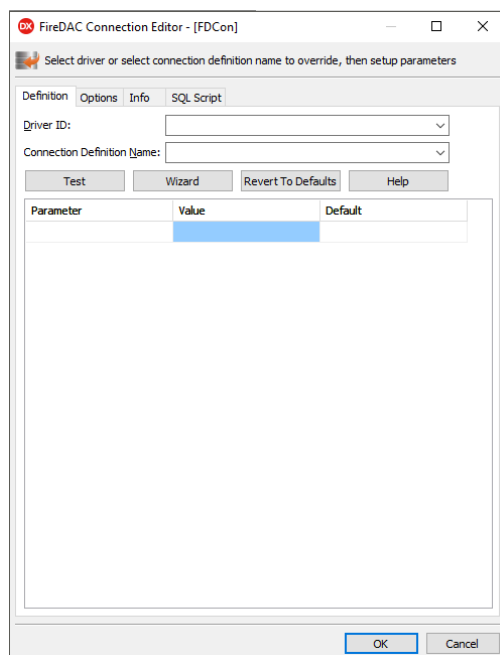


## Configurando o FDConnection:

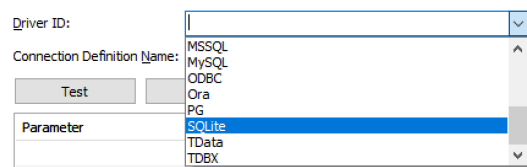
- Altere a propriedade name para **FDCon**
- Altere a propriedade Login Prompt para **False**



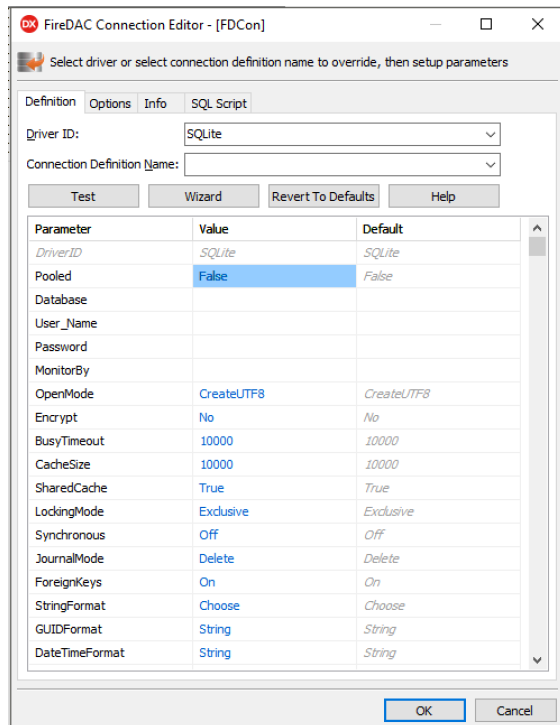
- Clique duplo sobre o componente FDCon



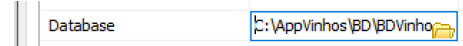
- Clique na seta para baixo da caixa Driver ID:



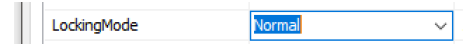
- Escolha a opção **SQLite**



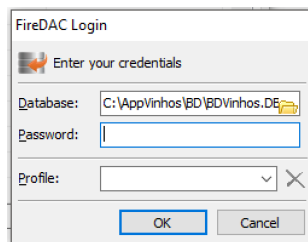
- Em **Database** escolha o banco de dados criado



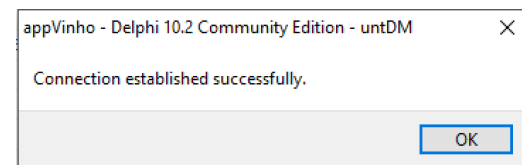
- Em **LockingMode** escolha **Normal**



- Clique no botão **Test**



- Clique em **OK**



- Clique em **OK** da caixa da mensagem

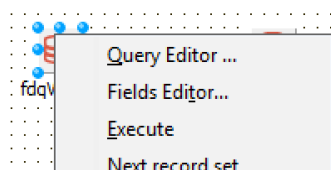
- Clique em **OK** da caixa Connection Editor

Obs: A propriedade Database será alterada futuramente via programação quando o aplicativo for enviado para o celular.

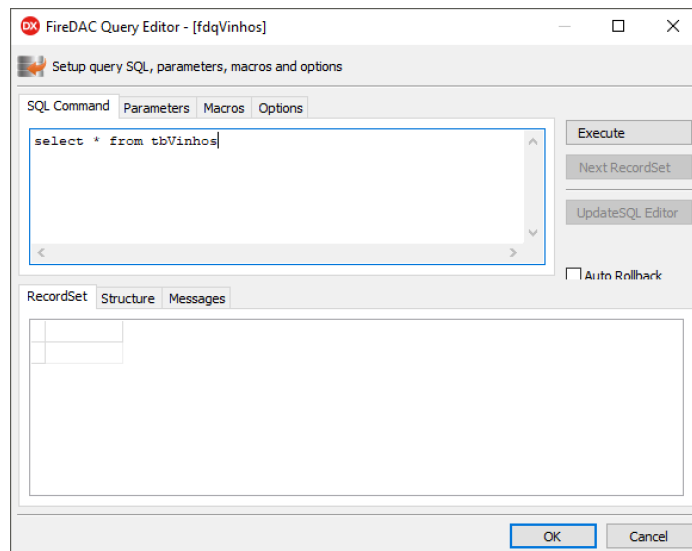
## Configurando o FDQuery:

- Altere a propriedade name para: **fdqVinhos**

- Clique com o botão alternativo sobre o componente **fdqVinhos**



- Clique em **Query Editor...**

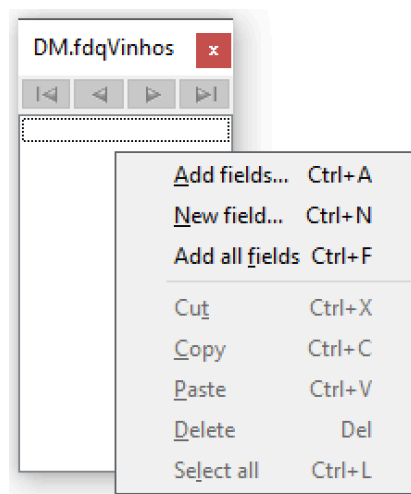


- Digite em SQL Command:

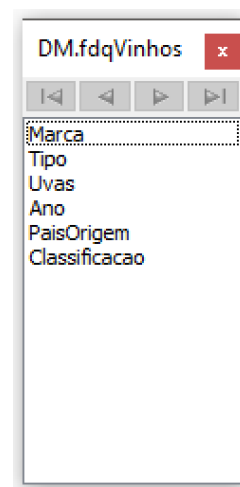
**select \* from tbVinhos**

- Clique em **OK**

- Clique novamente com o botão alternativo sobre o componente **fdqVinhos**
- Clique em **Fields Editor...**
- Clique com o botão alternativo dentro da área branca do componente



- Clique em **Add all fields**



Todos os campos são mostrados

- Feche a janela **Fields Editor**

## Configurando o FDPhysSQLiteDriverLink:

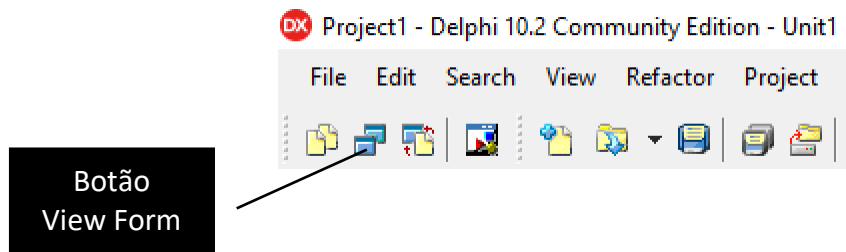
Este não precisa ser configurado. Somente iremos alterar seu nome.

- Altere a propriedade name para: **Driver**

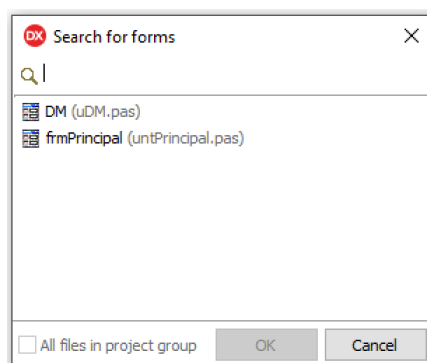
Com isso está concluída das configurações dos componentes do Data Module

## Desenvolvendo o form Principal:

Localize o botão View Form



- Clique no botão **View Form**



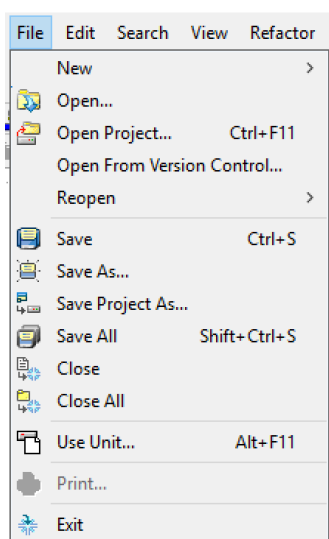
- Clique em **frmPrincipal**

- Clique em **OK**

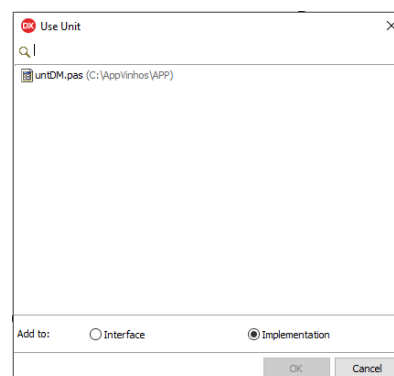
O formulário principal passa a ser exibido

Para que o frmPrincipal possa utilizar os componentes do Data Module é necessário que o untDM seja utilizada pelo form.

- Clique em **File**



- Clique em **Use Unit...**



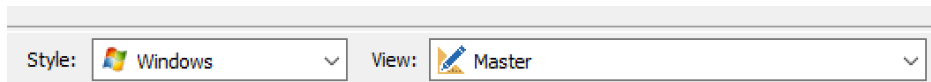
- Clique em **uDM**

- Clique em **OK**

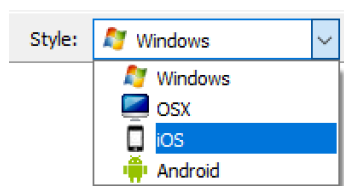
## Form

- Altere a propriedade **Width** do form para 320

Localize a caixa Style:



- Clique na seta para baixo desta caixa



- Escolha a opção **iOS** ou **Android** (Será utilizada a opção iOS nos exemplos a seguir)

## TabControl

- Insira um componente **TabControl** no form

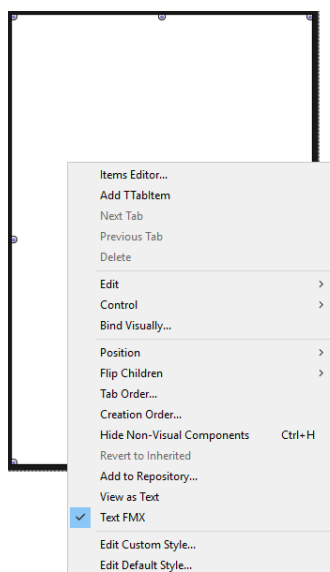
Altere as seguintes propriedades do TabControl:

*Name: tbcPrincipal*

*Align: Client*

Observe que o componente está ocupando toda a tela

- Clique com o botão alternativo sobre qualquer ponto do tabControl



- Clique em **Add TTabItem**

Observe que foi criada uma aba

- Clique novamente com o botão alternativo dentro do tabControl e acrescente mais uma aba

- Clique em **TabItem1**

Altere as seguintes propriedades:

*Name: tabVinhos*

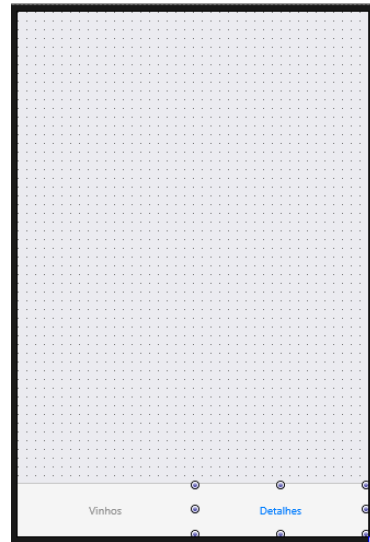
*Text para: Vinhos*

- Clique em **TabItem2**

Altere as seguintes propriedades:

*Name: tabDetalhes*

*Text para: Detalhes*



### *Aba Vinhos*

- Clique em **Vinhos** para ativar esta aba

### *ToolBar*

- Insira um componente **ToolBar** nesta aba
- Insira um componente **label** no toolbar

Altere as seguintes propriedades do label:

*Align: Contents*

*Text: Vinhos*

*StyleLookup: toollabel*

*TextSettings:*

*HorzAlign: Center*



- Insira um **botão** no ToolBar

Altere as seguintes propriedades do botão:

*Align: Right*

*Margin:*

*Right: 10*

*StyleLookup: addtoolbutton*





- Insira um componente **ListView** abaixo do ToolBar na aba Vinhos

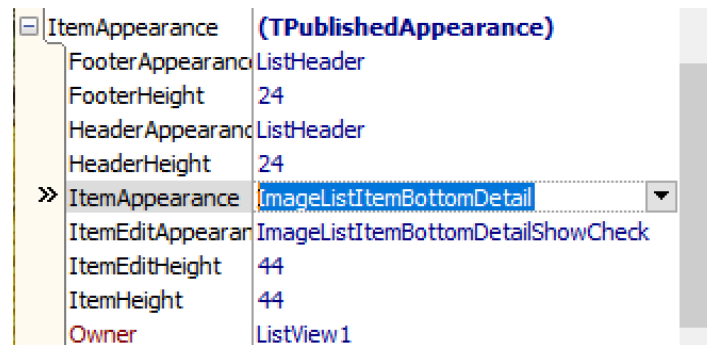
Altere as seguintes propriedades do componente ListView

*Align: Client*

*DeleteButtonText: Apagar*

*ItemAppearance:*

*ItemAppearance: ImageListItemBottomDetail*

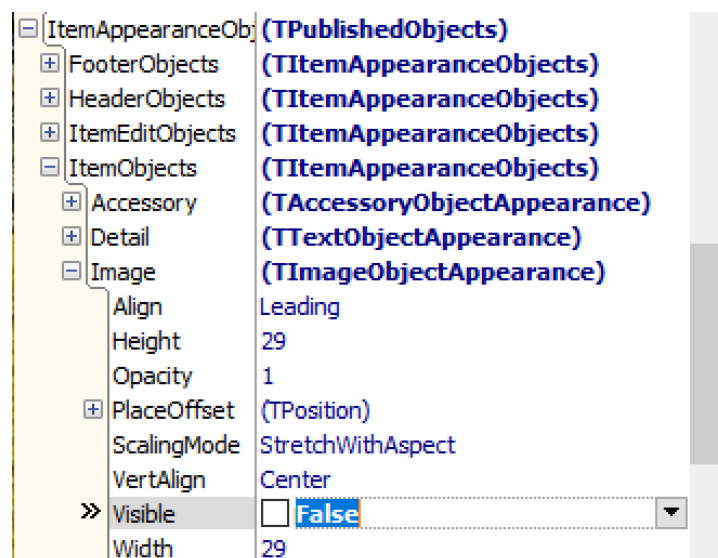


*ItemAppearanceObjects:*

*ItemObjects:*

*Image:*

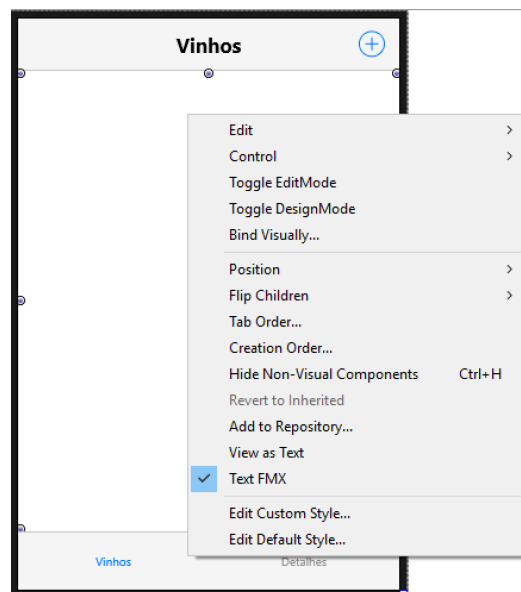
*Visible: False*



## Usando LiveBindings:

Depois de feita essas configurações desses componentes, iremos fazer a ligação do listview com os dados presente no banco de dados. Para isso iremos utilizar o recurso de LiveBindings presente no Delphi.

- Clique com o botão alternativo em qualquer parte do ListView

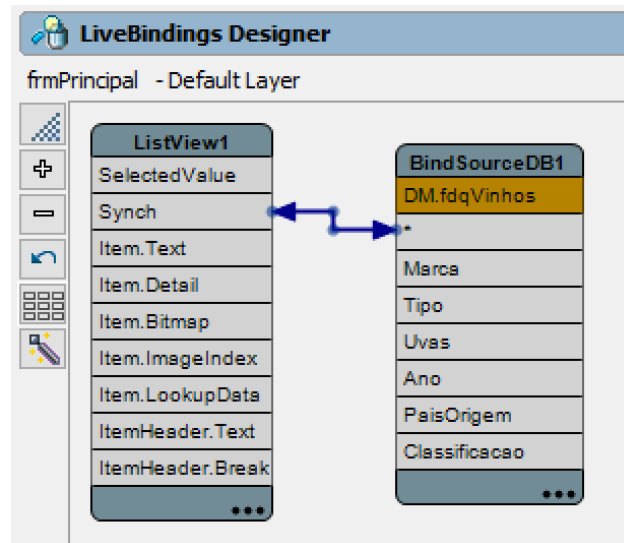


- Clique em **Bind Visually...**

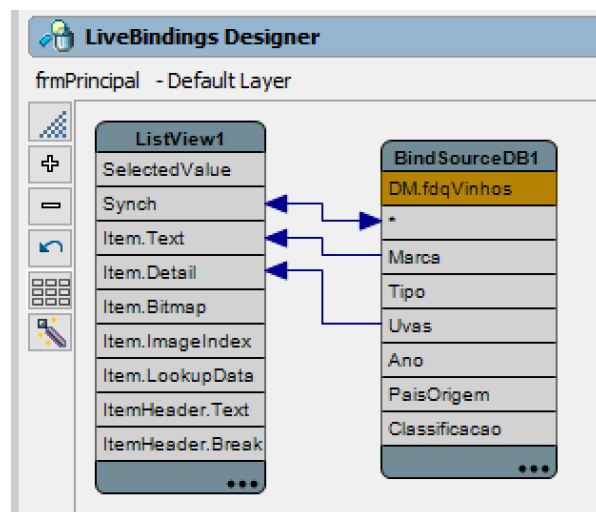
Organize os componentes da seguinte forma:



- Clique e arraste o \* presente em **Dm.fdqVinhos** para **Sync** no **ListView1**



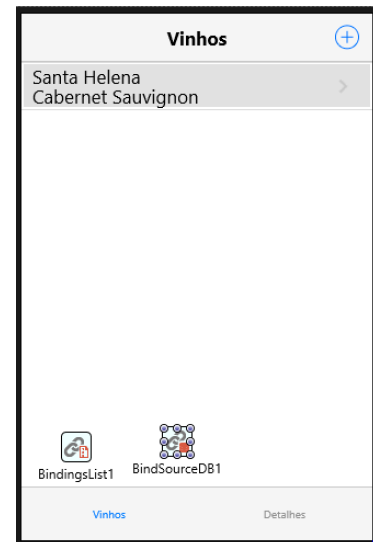
- Clique e arraste o campo **Marca** para **Item.Text**
- Clique e arraste o campo **Uvas** para **Item.Detail**



## Ativando o FDQuery:

- Reexiba o DataModule
- Clique em **fdqVinhos**
- Altere a propriedade Active para **True**
- Volte para o **form Principal**

Observe que os dados inseridos no banco estão sendo exibidos:



## Exibindo os dados:

### Aba Detalhes

- Clique na aba **Detalhes** para ativar esta aba
- Insira um componente **ToolBar** nesta aba
- Insira um componente **label** no toolbar

Altere as seguintes propriedades do label:

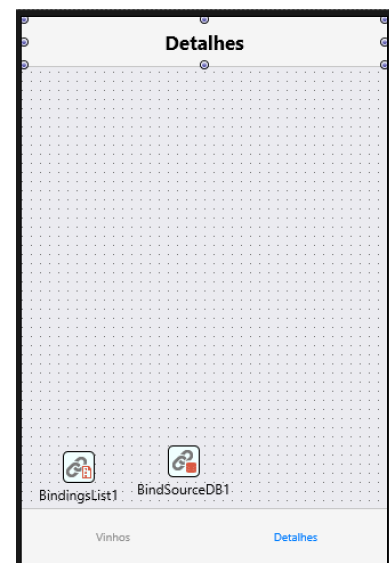
*Align: Contents*

*Text: Detalhes*

*StyleLookup: toollabel*

*TextSettings:*

*HorzAlign: Center*



Insira um botão na ToolBar

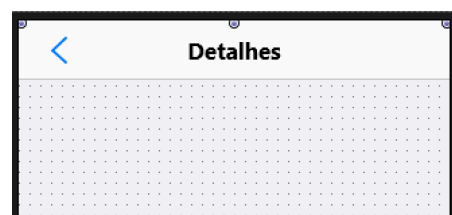
Altere as seguintes propriedades do botão:

*Align: Left*

*Margin:*

*Left: 10*

*StyleLookup: arrowlefttoolbutton*



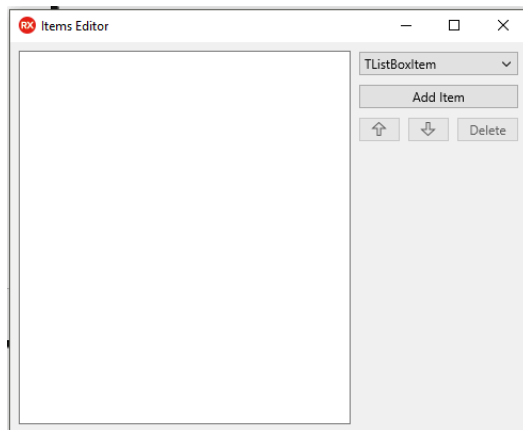
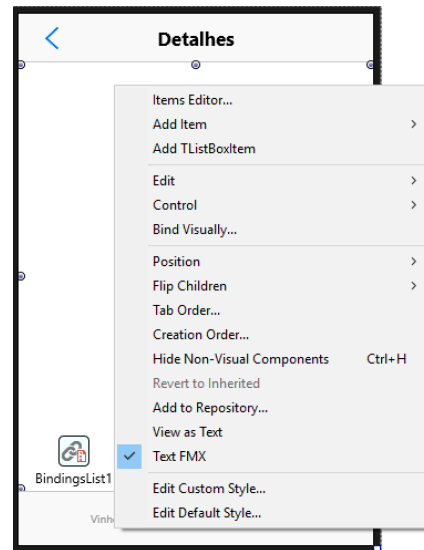
- Insira um componente **ListBox** abaixo do ToolBar na aba Detalhes

Altere as seguintes propriedades do componente ListBox

*Align: Client*

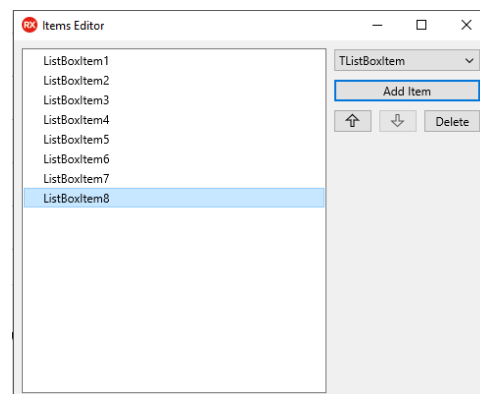
- Clique com o botão alternativo sobre o ListBox

- Clique em **Items Editor...**



- Clique em **Add Item** para adicionar itens ao ListBox

- Insira 8 itens no ListBox



- Feche a tela **Items Editor**

Selecione os 8 itens do ListBox e altere as propriedades:

*Text: (Apague o conteúdo desta propriedade para que os itens fiquem sem texto)*

- Insira um **label** e um **edit** em cada um dos cinco primeiros itens e no sexto item coloque um **label** e um **combobox**:

Altere as seguintes propriedades dos 6 labels:

*Align: left*

*Margim:*

*Left: 10*

*Text: (Cada uma das labels com Marca, Tipo, Uvas, Ano, Origem e Classificação)*

*Width: 105*

Altere as seguintes propriedades das 5 edits:

*Align: right*

*Margim:*

*right: 10*

*Width: 165*

Altere as seguintes propriedades do combobox:

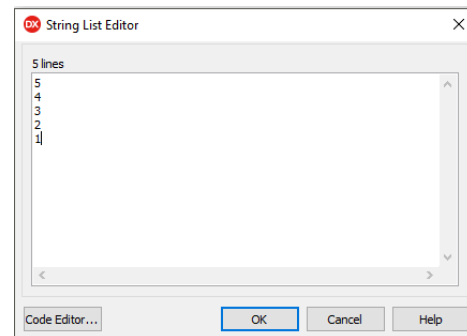
*Align: right*

*Items: 5, 4, 3, 2, 1*

*Margim:*

*right: 10*

*Width: 165*

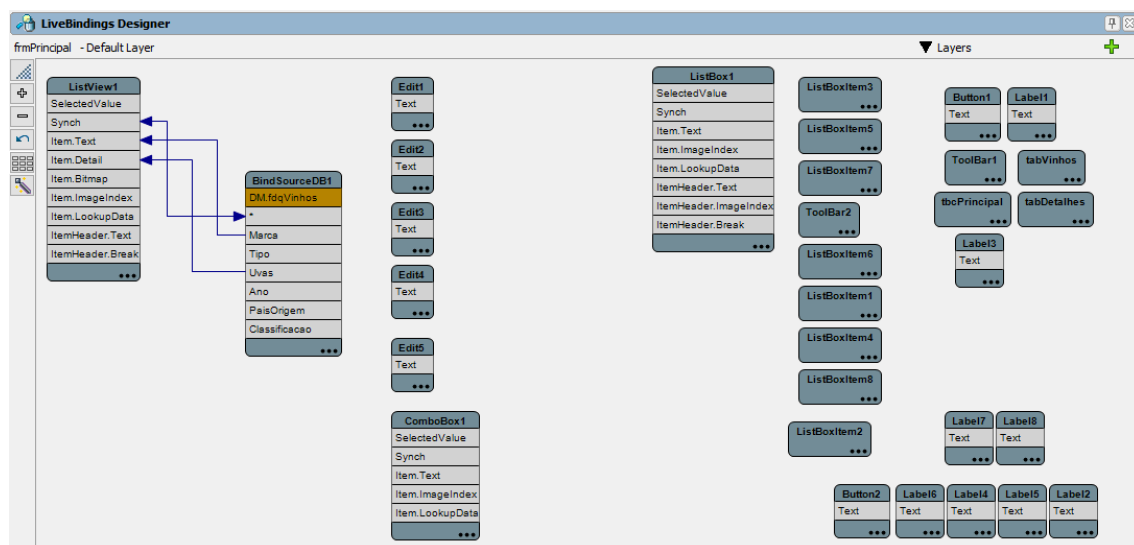


## Configurando o LiveBindings dos detalhes:

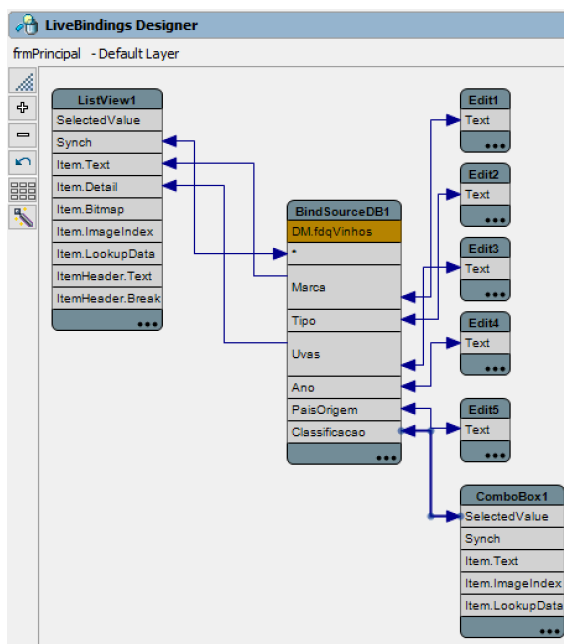
Depois de feita essas configurações desses componentes, iremos fazer a ligação das edits com os dados presente no banco de dados. Para isso iremos utilizar o recurso de LiveBindings presente no Delphi.

- Clique com o botão alternativo em qualquer parte do ListBox
- Clique em **Bind Visually...**

Organize o LiveBindings para que fique parecido com a imagem a seguir:



- Ligue o campo **Marca** com o Text do Edit1
- Ligue o campo **Tipo** com o Text do Edit2
- Ligue o campo **Uvas** com o Text do Edit3
- Ligue o campo **Ano** com o Text do Edit4
- Ligue o campo **PaisOrigem** com o Text do Edit5
- Ligue o campo **Classificação** com o SelectedValue do Combobox1



Observe que os dados são mostrados nos edits:

### Fazendo mais ajustes:

No último item do ListBox acrescente dois botões

Altere as seguintes propriedades do primeiro Botão:

*Align: left*

*Margins:*

*Left: 10*

*StyleLookup: listitembutton*

*Text: Salvar*

Altere as seguintes propriedades do primeiro Botão:

*Align: Right*

*Margins:*

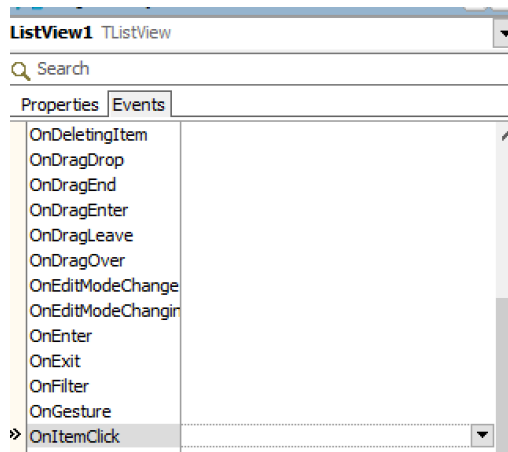
*Right: 10*

*StyleLookup: trashtoolbutton*



## Programando o listview:


- Clique na aba **Vinhos**
- Clique no **ListView**
- Clique em **Events** na caixa **Object Inspector**



- Clique duas vezes em **OnItemClick** para abrir este evento para programação:

Digite o comando indicado abaixo:

```
procedure TfrmPrincipal.ListView1ItemClick(const Sender: TObject;
  const AItem: TListViewItem);
begin
  tbcPrincipal.Next();
end;
```



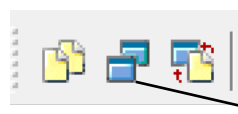
- Clique duas vezes em **OnDeleteItem** para abrir este evento para programação:

Digite o comando indicado abaixo:

```
procedure TfrmPrincipal.ListView1DeleteItem(Sender: TObject; AIndex:
  Integer);
begin
  dm.fdqVinhos.Delete;
end;
```



Volte para a edição do formulário clicando no botão Toggle Form/Unit



Botão  
Toggle  
Form/Unit


## Programando os botões:

### Botão Inserir:

- Clique duas vezes sobre o botão **Inserir**

Digite as duas linhas indicadas:

```
procedure TfrmPrincipal.Button1Click(Sender: TObject);
begin
    dm.fdqVinhos.Append;
    tbcPrincipal.Next();
end;
```




- Clique na aba **Detalhes**

### Botão Voltar:

- Clique duas vezes sobre este botão

Digite a linha indicada:

```
procedure TfrmPrincipal.Button2Click(Sender: TObject);
begin
    tbcPrincipal.Previous();
end;
```




### Botão Salvar:

- Clique duas vezes sobre este botão

Digite a linha indicada:

```
procedure TfrmPrincipal.Button3Click(Sender: TObject);
begin
    dm.fdqVinhos.Post;
    tbcPrincipal.Previous();
end;
```




### Botão Excluir:

- Clique duas vezes sobre este botão

Digite as linhas indicadas:

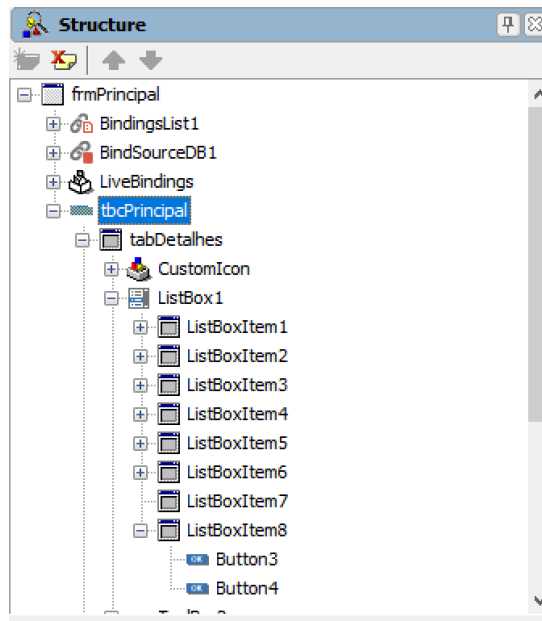
```
procedure TfrmPrincipal.Button4Click(Sender: TObject);
begin
    dm.fdqVinhos.Delete;
    tbcPrincipal.Previous();
end;
```



## Ocultando as abas

Não existe mais a necessidade das abas ficarem sendo exibidas, portanto elas serão ocultadas

- Clique em **tbcPrincipal** na caixa **Structure**



Altere as seguintes propriedades do TabControl:

*ActiveTab: tabVinhos*

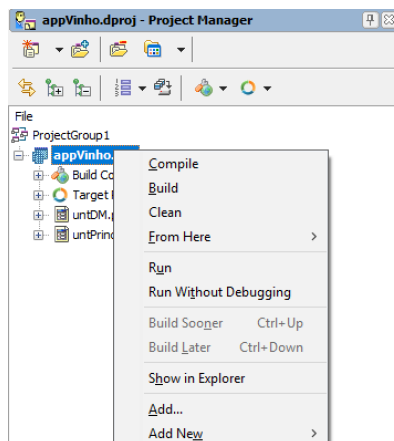
*TabPosition: Dots*

## 5. CONFIGURAÇÃO DO APP PARA SER ENVIADO PARA O DISPOSITIVO

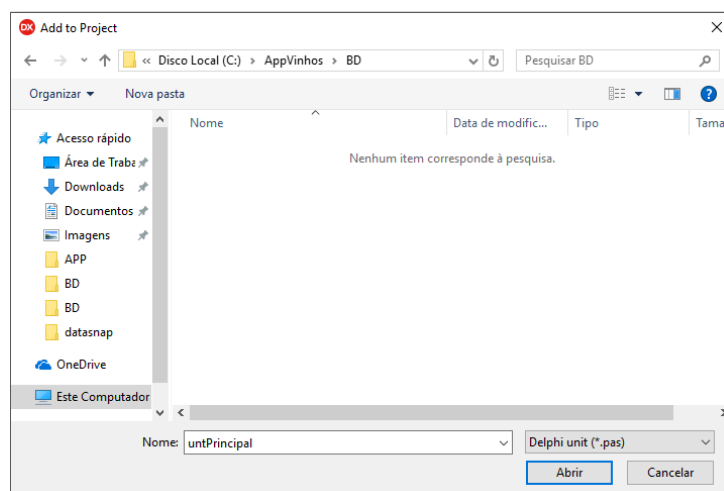
### Inserindo o banco de dados no aplicativo

Primeira ação a ser feita é adicionar o arquivo do banco de dados feito no SQLite no projeto.

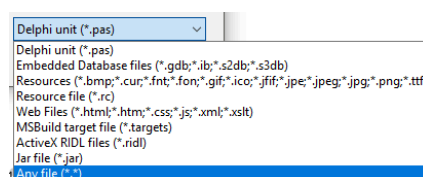
- Clique com o botão alternativo do mouse sobre o nome do projeto em Project Manager



- Clique em **Add...**

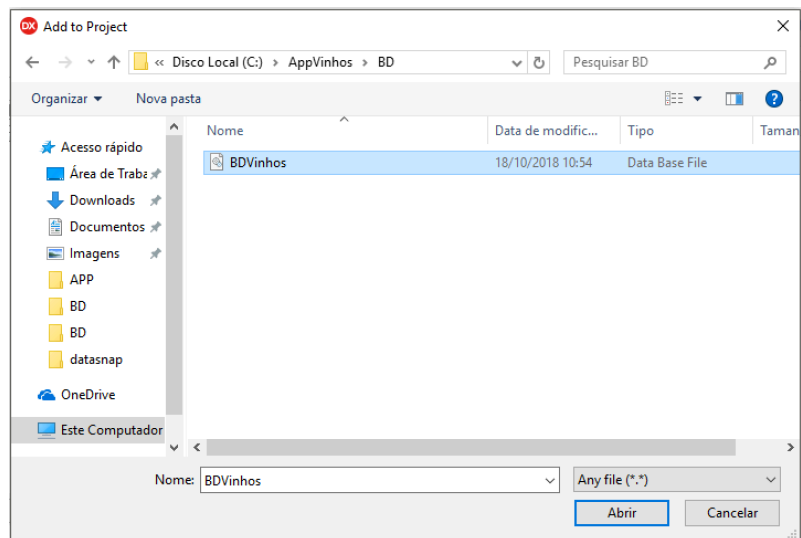


- Vá até a pasta BD criada
- Clique na seta para baixo da caixa **tipo de arquivo** ao lado da caixa nome

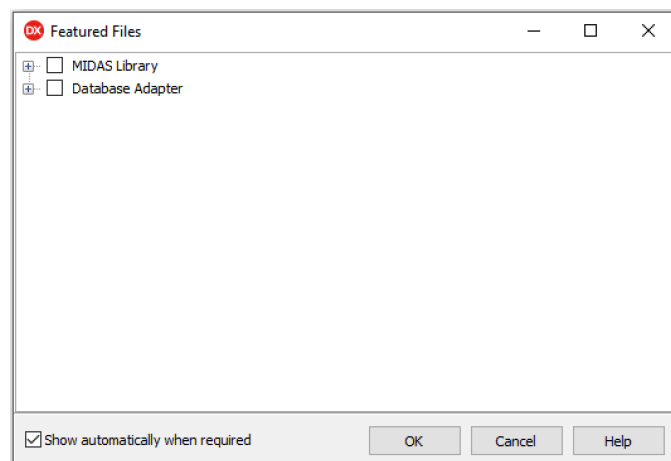


- Clique em **Any File(\*.\*)**

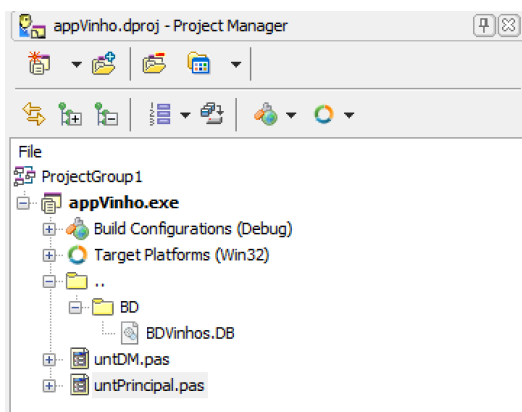
- Clique em **BDVinhos**



- Clique em **Abrir**  
Será exibida a tela **Feature Files**



- Clique no botão **Cancel**  
O arquivo BDVinhos.DB já aparece em Project Manager



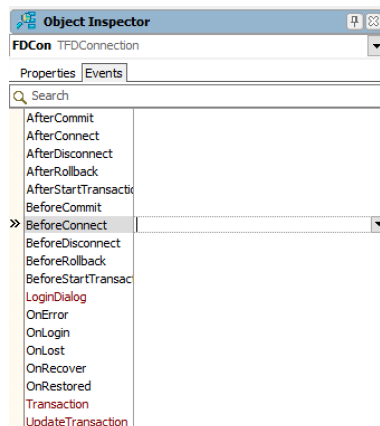
Ao ser inserido dessa forma, o banco de dados será armazenado no diretório padrão conforme o sistema operacional do dispositivo móvel.

iOS → Startup/Documents

Android → assets/Internal

## Configurando o Banco de Dados

- Abra o Data Module
- Clique em **FDCon**
- Clique em **Events**



- Clique duplo em **BeforeConnect**

Será aberto a área de código para ser feita a programação. Antes de programar este evento, será necessário acrescentar um item na cláusula uses.

### uses

```
System.SysUtils, System.Classes, FireDAC.Stan.Intf, FireDAC.Stan.Opt:
FireDAC.Stan.Error, FireDAC.UI.Intf, FireDAC.Phys.Intf, FireDAC.Stan
FireDAC.Stan.Pool, FireDAC.Stan.Async, FireDAC.Phys, FireDAC.FMXUI.W:
FireDAC.Stan.Param, FireDAC.Dats, FireDAC.DApt.Intf, FireDAC.DApt,
FireDAC.Stan.ExprFuncs, FireDAC.Phys.SQLiteDef, FireDAC.Phys.SQLite,
FireDAC.Comp.DataSet, FireDAC.Comp.Client;
```

Acrescente a seguinte biblioteca no final da cláusula uses:

**System.IOUtils**


### uses

```
System.SysUtils, System.Classes, FireDAC.Stan.Intf, FireDAC.Stan.Opt:
FireDAC.Stan.Error, FireDAC.UI.Intf, FireDAC.Phys.Intf, FireDAC.Stan
FireDAC.Stan.Pool, FireDAC.Stan.Async, FireDAC.Phys, FireDAC.FMXUI.W:
FireDAC.Stan.Param, FireDAC.Dats, FireDAC.DApt.Intf, FireDAC.DApt,
FireDAC.Stan.ExprFuncs, FireDAC.Phys.SQLiteDef, FireDAC.Phys.SQLite,
FireDAC.Comp.DataSet, FireDAC.Comp.Client, System.IOUtils;
```



No evento BeforeConnect deverá ser definido que o banco de dados estará presente na pasta padrão do sistema operacional do dispositivo, para isso, digite o código indicado:

```
procedure TDM.FDConBeforeConnect(Sender: TObject);  
begin  
    FDCon.Params.Values['Database'] :=  
        TPath.Combine(TPath.GetDocumentsPath, 'BDVinhos.DB')  
end;
```



O Código digitado acima altera o parâmetro Database do componente FDCon, para o endereço combinado entre o local padrão do sistema operacional com o nome do banco de dados.

### Ativando a query

Volte a exibir o Data Module

A query **fdqVinhos** deve estar ativa para que o aplicativo funcione

- Clique em **fdqVinhos**
- Verifique se a propriedade Active está marcada como **true**