

Adriana de Fátima Araújo

Embarcadero Delphi 2010

Desenvolvendo Aplicações



editora
VIENA

1ª Edição
Bauru/SP
Editora Viena
2012

Sumário

Lista de Siglas e Abreviaturas.....	17
1. Introdução ao Delphi 2010.....	19
1.1. O IDE do Delphi.....	21
1.1.1. Barra de Menus.....	22
1.1.2. Barras de Ferramentas.....	22
1.1.3. Formulário.....	22
1.1.4. Object Inspector	22
1.1.4.1. Guia Properties.....	23
1.1.4.2. Guia Events	24
1.1.5. Project Manager.....	25
1.1.6. To-Do List.....	26
1.2. Projetos no Delphi	27
1.2.1. Arquivo DCU (*.DCU).....	27
1.2.2. Arquivos de Formulários (*.DFM).....	27
1.2.3. Arquivos de Projeto (*.DPR)	28
1.2.4. Arquivo Executável (*.EXE)	29
1.2.5. Arquivo IDENTICACHE (*.IDENTCACHE)	29
1.2.6. Arquivo LOCAL (*.LOCAL)	29
1.2.7. Arquivos de Código (*.PAS).....	29
2. Programando no Delphi	33
2.1. Tipos de Dados.....	35
2.1.1. Dados do Tipo Inteiro (Integer)	35
2.1.2. Dados do Tipo Real (Real)	35
2.1.3. Dados do Tipo Texto (Char ou String)	36
2.1.4. Dados do Tipo Lógico (Boolean).....	36
2.2. Constantes	36
2.3. Variáveis	37
2.3.1. Variáveis Locais e Globais	37
2.4. Operadores	38
2.4.1. Tipos de Operadores	38
2.4.1.1. Operadores Aritméticos.....	38
2.4.1.2. Operador de Atribuição	39
2.4.1.3. Operador de Concatenação	39
2.4.1.4. Operadores Lógicos	39
2.4.1.5. Operadores Relacionais.....	40
2.4.2. Tipos de Expressões.....	41
2.4.2.1. Expressões Aritméticas.....	41
2.5. Condições e Decisões	42
2.5.1. Desvios Condicionais	42
2.5.1.1. Simples.....	42
2.5.1.2. Composto.....	43
2.5.1.3. Sequencial.....	44

2.6.1.4.	Encadeado	44
2.6.	Laços de Repetição	45
2.6.1.	For.....	45
2.6.2.	While	45
2.6.3.	Repeat.....	46
2.7.	Sub-Rotinas	46
2.7.1.	Procedimentos	46
2.7.1.1.	Passagens de Parâmetros	47
2.7.2.	Funções	48
3.	A Programação Orientada a Objetos (POO)	51
3.1.	Classes e Objetos.....	53
3.1.1.	Métodos.....	54
3.1.2.	Encapsulamento	54
3.1.2.1.	Operadores de Visibilidade	55
3.1.3.	Herança	55
3.1.4.	Compatibilidade de Tipos	56
3.5.	Polimorfismo	56
3.6.	Métodos.....	57
3.7.	Trabalhando com Exceções	57
3.7.1.	Try...Except...End.....	58
3.7.2.	Try...Finally...End	59
3.7.3.	Except e Finally	59
3.7.4.	Raise	60
4.	Desenvolvendo Aplicações	61
4.1.	Planejamento da Aplicação	63
4.2.	Trabalhando com as Regras de Negócio.....	63
4.3.	Modelo de Dados	63
4.3.1.	Modelo Conceitual.....	63
4.3.2.	Modelo Lógico.....	64
4.3.3.	Modelo Físico	64
4.4.	Sistemas Multicamadas.....	64
4.4.1.	Vantagens de um Sistema Multicamadas.....	65
4.4.2.	O DataSnap.....	66
5.	Criação de Componentes.....	69
5.1.	TButton.....	71
5.2.	TBitBtn	72
5.3.	TCheckBox.....	72
5.4.	TComboBox.....	73
5.5.	TEdit.....	74
5.6.	TGroupBox	75
5.7.	TImage.....	75
5.8.	TLabel	77
5.9.	TListBox.....	78
5.10.	TMainMenu.....	79
5.11.	TMaskEdit	80

5.12.	TPopupMenu	80
5.13.	TProgressBar.....	81
5.14.	TRadioGroup	83
5.15.	TShape.....	83
5.16.	TStatusBar	84
5.17.	TTimer.....	86
5.18.	TMediaPlayer.....	86
5.19.	TDBNavigator.....	87
5.20.	TPageControl	87
6.	Trabalhando com Formulários	91
6.1.	Criando Formulários.....	93
6.2.	Definindo Alinhamento, Proporções e Escala	93
6.3.	Atribuindo Estilo ao Formulário	94
7.	Manipulação de Componentes	95
7.1.	Alinhando os Componentes	97
7.2.	Configurando Propriedades	98
7.2.1.	Propriedade Name.....	98
7.2.2.	Propriedade Caption	99
7.2.3.	Propriedade Color	99
7.2.4.	Propriedades Font e Style.....	100
7.2.5.	Propriedades Enabled, Visible e ShowHint	101
7.2.6.	Propriedades de Tamanho e Posição	101
7.2.7.	Propriedade TabOrder	101
7.3.	Métodos.....	102
7.4.	Eventos.....	103
8.	Erros na Aplicação.....	107
8.1.	Tipos de Erros.....	109
8.1.1.	Erros de Compilação.....	109
8.1.2.	Erros de Execução.....	109
8.2.	Depurando Erros na Aplicação.....	109
8.2.1.	Comando Trace Into	110
8.2.2.	Comando Step Over	110
9.	Trabalhando com Banco de Dados	111
9.1.	Introdução a Banco de Dados.....	113
9.1.1.	Criando Tabelas.....	113
9.1.1.1.	Índice	113
9.1.2.	Campo	114
9.1.3.	Registro	114
9.1.4.	Chave	115
9.1.4.1.	Chave Primária.....	115
9.1.4.2.	Chave Estrangeira.....	116
9.1.4.3.	Chave Candidata.....	116
9.1.5.	Modelo Entidade Relacionamento (MER).....	116
9.1.5.1.	Entidades.....	117

9.1.5.2.	Relacionamentos	118
9.1.6.	Modelo Entidade Relacionamento Estendido	120
9.1.6.1.	Agregação	121
9.1.6.2.	Generalização/Especialização	121
9.1.6.3.	Restrições de Herança	122
9.2.	Acesso ao Banco de Dados	123
9.2.1.	Data Access	123
9.2.2.	BDE.....	124
9.2.3.	Data Controls	124
9.2.4.	Componente TDataSource.....	127
9.3.	Sistema de Gestão de Base de Dados – SGBD	127
9.3.1.	Vantagens do Uso de Banco de Dados.....	128
9.3.1.1.	Redução ou Eliminação de Redundâncias	128
9.3.1.2.	Eliminação de Inconsistências	128
9.3.1.3.	Compartilhamento dos Dados	128
9.3.1.4.	Restrições de Segurança	128
9.3.1.5.	Padronização dos Dados.....	129
9.3.1.6.	Independência dos Dados	129
9.3.1.7.	Manutenção da Integridade.....	130
10.	Gestos, Toques e Multitoques	133
10.1.	Propriedades e Componentes	135
10.1.1.	Gestos Padrão	136
10.1.2.	Gestos Personalizados	137
10.1.3.	Inserindo Gestos na Aplicação	139
10.2.	Teclado de Toque.....	141
10.2.1.	Inserindo um Teclado de Toque	141
11.	Gráficos e Relatórios	145
11.1.	Gráficos no Delphi.....	147
11.1.1.	Inserindo Gráfico	147
11.1.2.	Exportando Gráficos.....	151
11.1.3.	Imprimindo Gráfico	152
12.	Rave Reports Designer	155
12.1.	Área de Trabalho do Rave.....	157
12.1.1.	Barras de Ferramentas	157
12.1.2.	Componentes	158
12.1.2.1.	Standard	159
12.1.2.2.	Zoom.....	160
12.1.2.3.	Colors.....	160
12.1.2.4.	Lines	160
12.1.2.5.	Fills	161
12.1.2.6.	Fonts.....	161
12.1.2.7.	Alignment	162
12.1.2.8.	Drawing.....	163
12.1.2.9.	Bar Code.....	164
12.1.2.10.	Report	164

12.1.3.	Property Panel	165
12.1.4.	Barra de Status.....	166
12.1.5.	Page Designer.....	166
12.1.6.	Event Editor	167
12.1.7.	Project Tree	168
12.1.8.	Réguas	168
12.2.	Desenvolvendo Relatórios	169
12.2.1.	Criando Relatório.....	169
12.2.2.	Testando o Relatório.....	174
12.3.	Salvar Relatório.....	175
12.4.	Exportando Relatório	176
12.4.1.	Imprimindo Relatório	177
12.5.	Global Page.....	177
13.	Exercícios Práticos	181
13.1.	Calculando	183
13.2.	Formulário de Cadastro.....	184
13.3.	Formulários com Menus	188
13.4.	Relatório de Clientes	191
13.5.	Despertador Virtual	196
13.6.	Criando um Player	199
Referências.....		205
Glossário.....		207

Lista de Siglas e Abreviaturas

<i>BLOB</i>	<i>Binary Large Object - Grande Objeto Binário.</i>
<i>BMP</i>	<i>Bitmap Image File.</i>
<i>DCU</i>	<i>Digital Counting Unit.</i>
<i>DDL</i>	<i>Data Definition Language- Linguagem de Definição de Dados.</i>
<i>DER</i>	<i>Diagrama Entidade Relacionamento.</i>
<i>EMF</i>	<i>Enhanced Metafile Format.</i>
<i>HTTP</i>	<i>Hypertext Transfer Protocol - Protocolo de Transferência de Hipertexto.</i>
<i>JPEG</i>	<i>Joint Photographic Experts Group.</i>
<i>MER</i>	<i>Modelo Entidade Relacionamento.</i>
<i>MIDI</i>	<i>Musical Instrument Digital Interface.</i>
<i>MP3</i>	<i>MPEG Audio Layer-3.</i>
<i>POO</i>	<i>Programação Orientada ao Objeto.</i>
<i>RAD</i>	<i>Rapid Application Development.</i>
<i>REST</i>	<i>Representational State Transfer – Transferência de Estado Representacional.</i>
<i>SGBD</i>	<i>Sistema Gerenciador de Banco de Dados.</i>
<i>WAV</i>	<i>Waveform Audio File Format.</i>
<i>WMF</i>	<i>Windows Metafile Format.</i>

1

Introdução ao Delphi 2010

1.1. O IDE do Delphi

- 1.1.1. Barra de Menus
- 1.1.2. Barras de Ferramentas
- 1.1.3. Formulário
- 1.1.4. Object Inspector
- 1.1.5. Project Manager
- 1.1.6. To-Do List

1.2. Projetos no Delphi

- 1.2.1. Arquivo DCU (*.DCU)
- 1.2.2. Arquivos de Formulários (*.DFM)
- 1.2.3. Arquivos de Projeto (*.DPR)
- 1.2.4. Arquivo Executável (*.EXE)
- 1.2.5. Arquivo IDENTCACHE (*.IDENTCACHE)
- 1.2.6. Arquivo LOCAL (*.LOCAL)
- 1.2.7. Arquivos de Código (*.PAS)

1. Introdução ao Delphi 2010

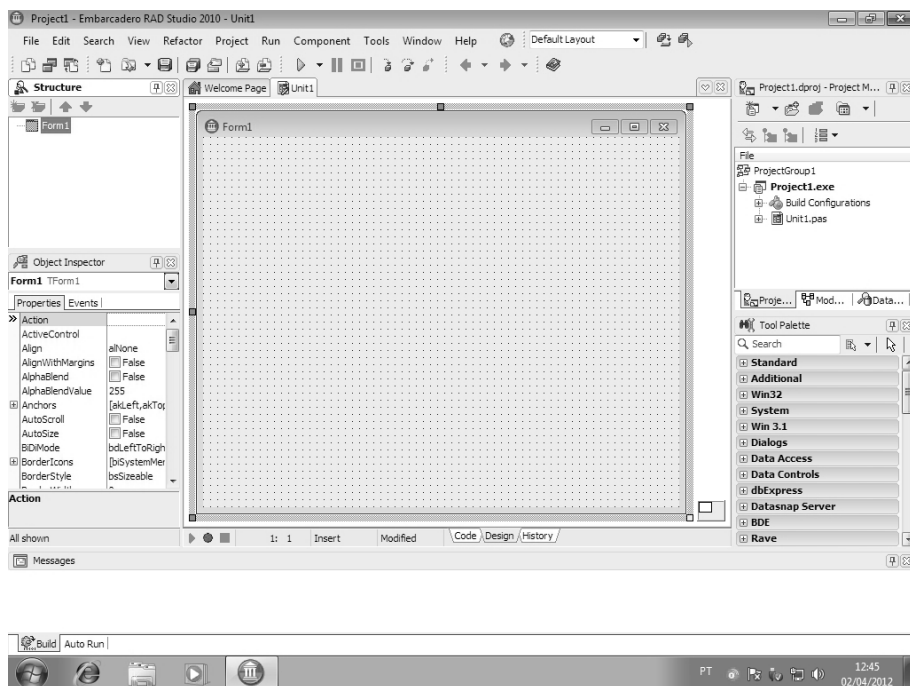
O **Delphi 2010** atualmente desenvolvido pela **Embarcadero®** apresenta um ambiente completo, com recursos e ferramentas avançadas que permitem o desenvolvimento desde aplicações simples até as mais complexas utilizando os bancos de dados mais usados atualmente.

Essa linguagem de programação possibilita o desenvolvimento de aplicações multicamadas, para o ambiente Web, cliente-servidor e para dispositivos com tela sensível ao toque (touch).

1.1. O IDE do Delphi

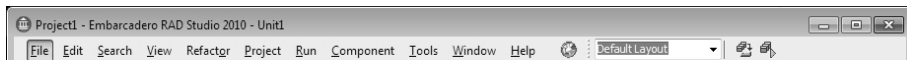
O IDE (**I**ntegrated **D**evelopment **E**nvironment ou **A**mbiente **I**ntegrado de **D**esenvolvimento) do **Delphi 2010** é composto por um conjunto de elementos que armazenam diversos recursos e ferramentas para o desenvolvimento das aplicações.

O usuário poderá personalizá-lo de acordo com suas necessidades, arrastando, ocultando ou exibindo barras e painéis.



1.1.1. Barra de Menus

A **Barra de menus** do **Delphi 2010** armazena diversas opções e comandos para serem utilizados para criar e manipular aplicações. Esta barra está localizada abaixo da **Barra de Título**.



À direita da **Barra de Menus** é exibida a barra de ferramentas **Desktop** que permite ao usuário personalizar o modo como o IDE do **Delphi** é exibido.

1.1.2. Barras de Ferramentas

As **Barras de Ferramentas** são formadas por conjuntos de ferramentas que servem de atalho para grande parte dos recursos disponibilizados pela **Barra de Menus**. Através de suas ferramentas o usuário poderá realizar diferentes tipos de tarefas mais rapidamente.



1.1.3. Formulário

Por padrão, ao criar uma nova aplicação no **Delphi 2010**, automaticamente um formulário será aberto. Nele será desenvolvido o layout da aplicação, através da inserção dos respectivos componentes.



Formulário em branco.

As propriedades do formulário como cor, nome e dimensões, por exemplo, são editadas através do painel **Object Inspector**.

Em uma aplicação cada formulário estará interligado a um código de programa, denominado **Unit** (unidade), no qual deverão estar descritos todos os eventos dos componentes.

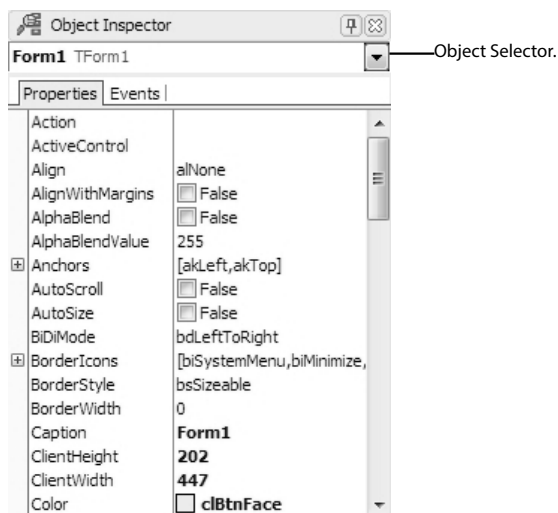
A visualização desta **Unit** é feita através da guia **Code** exibida abaixo do formulário.

1.1.4. Object Inspector

O painel **Object Inspector** permite configurar as propriedades e eventos dos componentes inseridos em uma aplicação, incluindo os formulários. Por padrão ele está fixado à esquerda da IDE do **Delphi 2010**.

O **Object Inspector** se subdivide em duas guias: **Properties** e **Events**. Além disso, no topo do painel há um menu deslizante, o **Object Selector**, onde são listados todos os componentes inseridos atualmente na aplicação.

Ao seleccionar um componente nesta lista, automaticamente serão exibidos o nome e o tipo deste componente, além de suas propriedades e as opções de eventos disponíveis para ele.



Dica: Quando este painel não estiver visível, pode-se exibi-lo acessando o menu **View** e clicando na opção **Object Inspector (F11)**.

1.1.4.1. Guia Properties

As propriedades são os atributos de um componente e determinam o modo como ele irá se comportar e ser visualizado na aplicação. A guia **Properties** do painel **Object Inspector** é o local onde são configuradas estas propriedades.



Dica: O conjunto de propriedades exibido na guia **Properties** varia de acordo com o componente selecionado.

1.1.4.2. Guia Events

Os eventos são ocorrências do sistema ou ações que um componente pode reconhecer e que poderão ser disparados através de um clique do mouse, pressionando uma tecla ou automaticamente após um determinado período de tempo. Na guia **Events** são listados todos os eventos disponíveis para um determinado componente.



Para configurar um evento, é necessário clicar duas vezes sobre o campo em branco exibido à direita do evento correspondente.

Automaticamente será exibida a procedure correspondente ao evento na guia **Code**, permitindo a programação do evento desejado.

```
procedure TForm1.FormAlignPosition(Sender: TWinControl; Control: TControl;  
  var NewLeft, NewTop, NewWidth, NewHeight: Integer; var AlignRect: TRect;  
  AlignInfo: TAlignInfo);  
30 begin  
31 |  
  end;  
  end.
```

Procedure do evento OnAlignPosition.

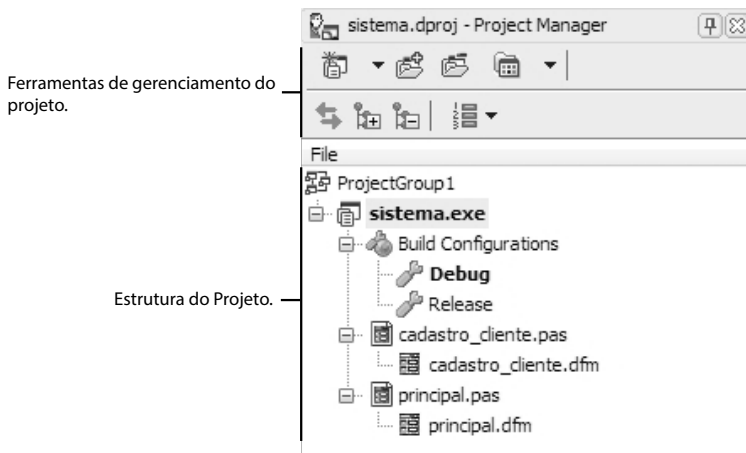
1.1.5. Project Manager

O **Project Manager** é um painel que está posicionado à direita da área de trabalho do **Delphi 2010** e permite navegar de maneira simples e rápida por projetos de grande porte. Além disso, é possível inserir uma **Unit** de outro projeto no projeto atual, incluindo o caminho do arquivo automaticamente.

Quando este painel não estiver visível, basta acessar o menu **View** e escolher a opção **Project Manager (Ctrl+Alt+F11)** para exibi-lo.

O gerenciamento de projetos no **Delphi 2010** é realizado através deste painel. Ele utiliza um modo de exibição do tipo “árvore”, permitindo a visualização de toda a estrutura hierárquica dos arquivos dentro dos projetos.

Na área superior do painel há um conjunto de ferramentas para a administração do projeto e, logo abaixo, seção **File** onde a estrutura do projeto é exibida.



Veja a seguir uma breve descrição das ferramentas que compõem estas barras:



Activate selected project: *Habilita o projeto atualmente selecionado.*



Add new project: *Adiciona um novo projeto à aplicação.*



Remove selected file or project: *Remove da aplicação atual uma unit ou projeto selecionado.*



Select project view style: *Permite selecionar um estilo de exibição para o projeto.*



Synchronize selection with editor: *Sincroniza a seleção atual com o editor.*



Expand selected nodes: *Expande todos os grupos de arquivos em um projeto, facilitando a visualização dos mesmos.*



Collapse selected nodes: *Contrai a visualização de todos os arquivos dentro de seus respectivos grupos.*



Sort all projects: *Classifica todos os projetos abertos atualmente na aplicação, por nome, data de modificação, tipo, caminho etc.*

1.1.6. To-Do List

A **To-Do List** é um painel que exibe uma lista das tarefas a serem realizadas no projeto. Ela é constituída por uma coleção de notas que ficam visíveis para o programador, tornando-se muito útil em projetos desenvolvidos por mais de um programador para facilitar a identificação das tarefas que cada um deverá realizar em um projeto.

Para exibir este painel, acesse o menu **View** e clique na opção **To-Do List**. Ele será exibido na área inferior da janela do **Delphi**, logo abaixo do formulário.

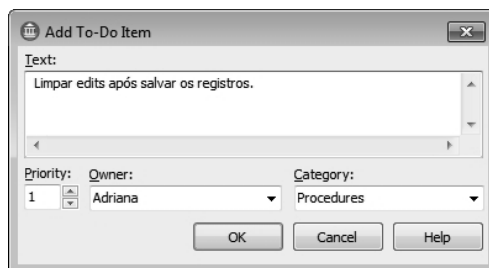
To-Do List - sistema			
Action Item	Module	Owner	Category
<input type="checkbox"/> Limpar edits após salvar os registros.	1	Adriana	Procedures
<input type="checkbox"/> Incluir botão "Salvar".	2	Adriana	Procedures

2 items (0 hidden) 2 items pending

Refactorings Find References To-Do List - sistema

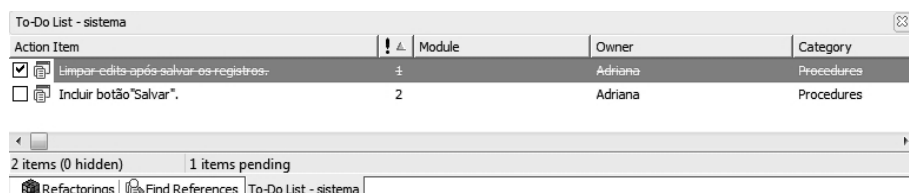
Para inserir uma tarefa no painel **To-Do List**, siga os passos:

1. Clique com o botão direito do mouse sobre o **To-Do List**;
2. No menu de contexto exibido, clique na opção **Add** (Ctrl+A);
3. Especifique na caixa de diálogo **Add To-Do Item** os seguintes itens para a nova tarefa:
 - 3.1. **Text:** Insira o texto descritivo da tarefa que deverá ser realizada;
 - 3.2. **Priority:** Determine a prioridade da tarefa determinando sua ordem na lista;
 - 3.3. **Owner:** Identifique-se digitando seu nome neste campo;
 - 3.4. **Category:** Determine uma categoria para a nova tarefa;



4. Clique no botão OK.

Quando a tarefa for realizada, deve-se marcá-la para indicar sua conclusão, clicando na caixa à esquerda da tarefa no painel. Será exibido um tachado sobre a tarefa no painel **To-Do List**.



Para remover um item desta lista, basta clicar com o botão direito do mouse sobre ele e escolher a opção **Delete (Del)** no menu de contexto.

1.2. Projetos no Delphi

Os projetos criados no **Delphi 2010** são constituídos por diferentes tipos de arquivos: *.DCU, *.DFM, *.DPR, *.EXE, *.IDENTCACHE, *.LOCAL e *.PAS.

Ao salvar um projeto, o **Delphi 2010** solicitará somente os nomes dos arquivos de unidade e projeto. Os demais arquivos são criados automaticamente.

Cada arquivo é responsável por armazenar um tipo de informação, sendo que alguns destes são imprescindíveis para o funcionamento da aplicação e outros opcionais, que podem ser excluídos posteriormente.



Atenção: Antes de fazer a exclusão de um arquivo criado com o projeto, tenha certeza que isso não comprometerá o funcionamento da aplicação.

1.2.1. Arquivo DCU (*.DCU)

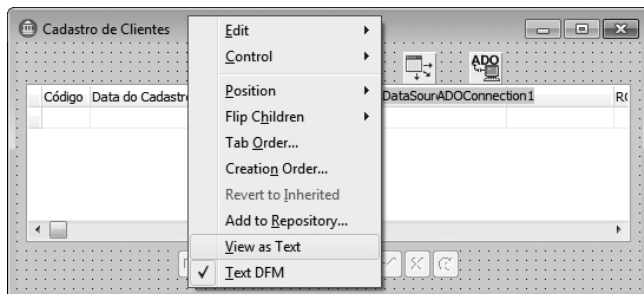
Este tipo de arquivo armazena informações sobre a última compilação realizada, de modo que o **Delphi** se encarregue de compilar apenas as alterações realizadas posteriormente no código da aplicação.

1.2.2. Arquivos de Formulários (*.DFM)

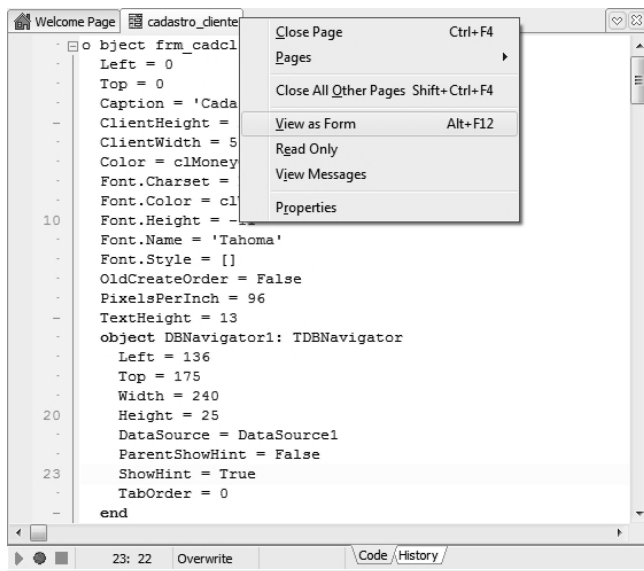
Os arquivos *.DFM são criados para cada formulário existente em uma aplicação e armazenam informações sobre o formulário e os componentes nele inseridos. Os arquivos *.DFM são gerados com o mesmo nome da **Unit** associada.

Essas informações são armazenadas em formato binário, sendo legível somente através do **Delphi** ou podendo ser convertida em formato de texto e visualizada através da guia **Code**.

A visualização deste arquivo é feita clicando com o botão direito do mouse sobre o formulário e selecionando a opção **View as Text**.



Para retornar à visualização padrão do formulário basta clicar novamente com o botão direito do mouse sobre a aba da **Unit** na guia **Code** e escolher a opção **View as Form** (**Alt+F12**).



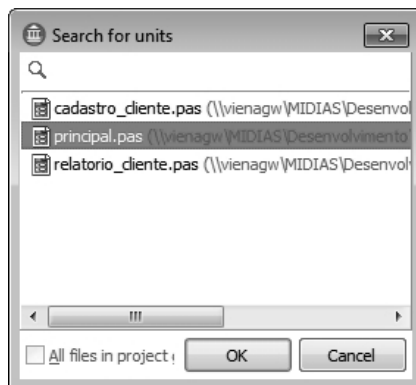
1.2.3. Arquivos de Projeto (*.DPR)

O arquivo de projeto (*.DPR) armazena o programa principal da aplicação, que armazena as informações sobre todas as **Units** e formulários associados.

Para visualizar o conteúdo deste arquivo na guia **Code**, siga os passos:

1. Acesse o menu **View** e escolha a opção **Units** (**Ctrl+F12**);

2. Na caixa de diálogo exibida serão listadas as **Units** da aplicação. Clique sobre a **Unit** que deseja visualizar os códigos para selecioná-la;



3. Clique no botão **OK**. Os códigos desta **Unit** serão exibidos na guia **Code**.

Através do arquivo ***.DPR** é possível determinar quais formulários deverão ser criados e exibidos no início da aplicação.

1.2.4. Arquivo Executável (*.EXE)

O arquivo executável (*.EXE) é criado pelo compilador e pode ser executado em computadores que não possuam o **Delphi 2010** instalado. Para criar um arquivo executável, basta pressionar as teclas <Ctrl+F9>.



Atenção: *Esse arquivo é destinado ao usuário da aplicação.*

1.2.5. Arquivo IDENTCACHE (*.IDENTCACHE)

O arquivo do tipo **IdentCache** (*.IDENTCACHE) é temporário, gerado pelo mecanismo de refatoração para aprimorar o desempenho da aplicação.

1.2.6. Arquivo LOCAL (*.LOCAL)

O arquivo do tipo **Local** (*.LOCAL) que é exibido na pasta da aplicação após a compilação da aplicação armazena as opções especificadas pelo programador durante o processo de desenvolvimento.

1.2.7. Arquivos de Código (*.PAS)

Os arquivos com extensão ***.PAS** armazenam os códigos fonte das **Units** de um projeto, onde são declarados os tipos, variáveis, procedures, eventos etc.



Lined area for notes, consisting of multiple horizontal lines.