



# ***Delphi 7***

# ÍNDICE

<b>CAPÍTULO 1</b>	
• INTRODUÇÃO .....	06
<b>CAPÍTULO 2</b>	
• INSTALANDO O DELPHI .....	10
<b>CAPÍTULO 3</b>	
• INICIANDO O Delphi .....	18
• FORM DESIGN .....	19
• CODE EDITOR .....	23
• OBJECT INSPECTOR .....	26
• OBJECT TREE VIEW .....	29
• PALHETA DE COMPONENTES .....	29
<b>CAPÍTULO 4</b>	
• OBJECT PASCAL .....	34
• ESTRUTURA DE UMA UNIT .....	34
• VARIÁVEIS .....	35
• TIPOS DE DADOS PADRÃO .....	40
<b>CAPÍTULO 5</b>	
• ESTRUTURAS DE DECISÃO .....	52
• ESTRUTURAS DE REPETIÇÃO .....	52
• TIPOS DEFINIDOS PELO USUÁRIO .....	54
• PROCEDIMENTOS, FUNÇÕES E MÉTODOS .....	56
• CRIANDO E DESTRUINDO OBJETOS .....	59
<b>CAPÍTULO 6</b>	
• PRINCIPAIS CARACTERÍSTICAS .....	62
• CARACTERÍSTICAS DA PROGRAMAÇÃO .....	62
• ELEMENTOS DA PROGRAMAÇÃO .....	62
• ARQUIVOS PRODUZIDOS PELO SISTEMA .....	63
• PROGRAMAÇÃO EM WINDOWS: JANELAS E EVENTOS .....	66
• PROGRAMAÇÃO ORIENTADA A OBJETO (POO) .....	66

## **CAPÍTULO 7**

• PRIMEIRO CONTATO .....	70
• PROPRIEDADES COMUNS .....	72
• EVENTOS .....	73
• EVENTOS COMUNS .....	73
• MÉTODOS .....	74
• MÉTODOS COMUNS .....	74
• JANELAS .....	75

## **CAPÍTULO 8**

• COMPONENTES PADRÕES .....	78
• MENUS .....	88

## **CAPÍTULO 9**

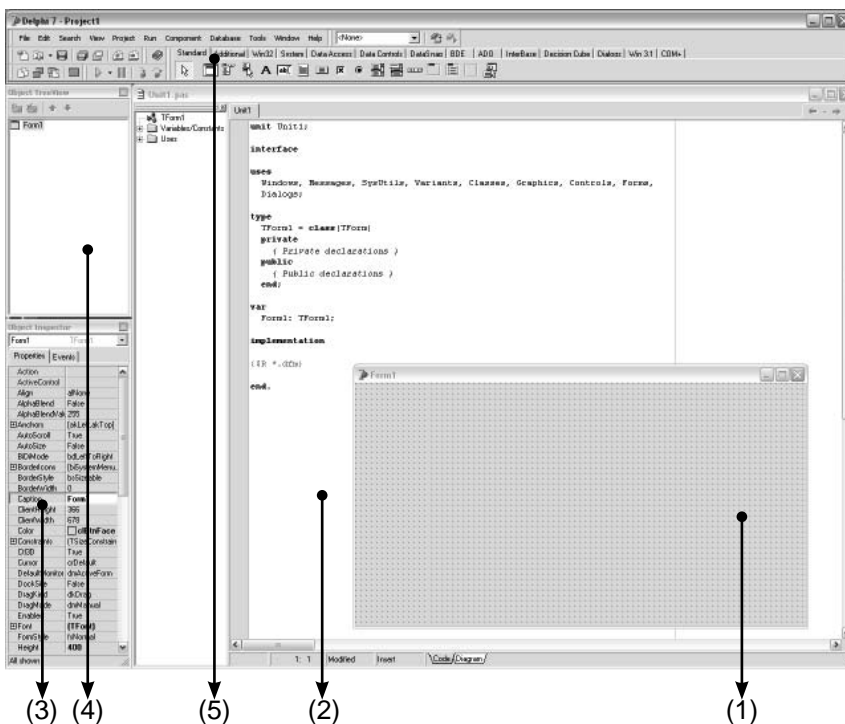
• PROJECT MANAGER E PROJECT GROUP .....	90
• SALVANDO UM LAYOUT PERSONALIZADO .....	91
• CUSTOMIZANDO AS BARRAS DE FERRAMENTAS .....	92

## **CAPÍTULO 10**

• DEBUGANDO APLICAÇÕES .....	94
• BREAK POINT .....	94
• WHATCH LIST .....	96
• EVALUATE/MODIFY .....	97
• INSPECT.....	97

## **CAPÍTULO 11**

• SGBD - SISTEMA GERENCIADORES DE BANCO DE DADOS .....	100
• BANCO DE DADOS RELACIONAIS .....	104
• MODELAGEM DE DADOS .....	106



- 1 - Form Design
- 2 - Code Editor
- 3 - Object Inspector
- 4 - Object Tree View
- 5 - Palheta de Componentes

## 3.2 - FORM DESIGN

Observe na próxima página o Form Designer. É sobre esta janela que a interface gráfica da aplicação é desenvolvida, adicionando os componentes com os quais o usuário irá interagir.

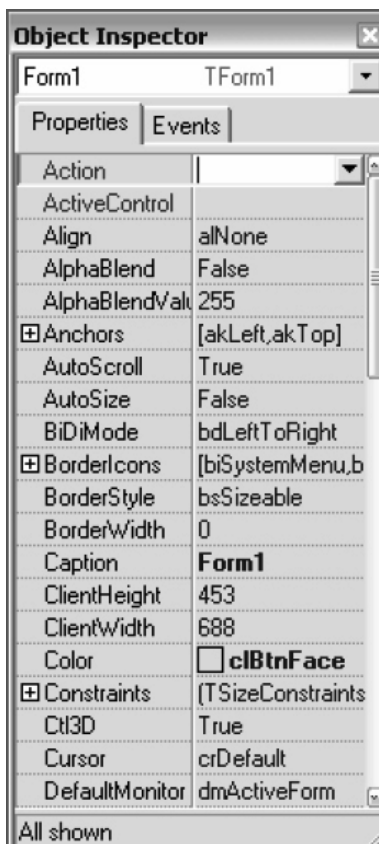
Com o botão **Add** é possível incluir um novo template, dando-lhe um **Shortcut name** e uma descrição. Este recurso é usado para encontrar o template durante a edição. Na caixa **Code** devemos inserir o código que irá aparecer quando utilizamos o template.

Os templates podem ser encontrados no interior do seguinte arquivo:

**C:\Arquivos de programas\Borland\Delphi7\Bin\Delphi32.DCI**

### 3.4 - OBJECT INSPECTOR

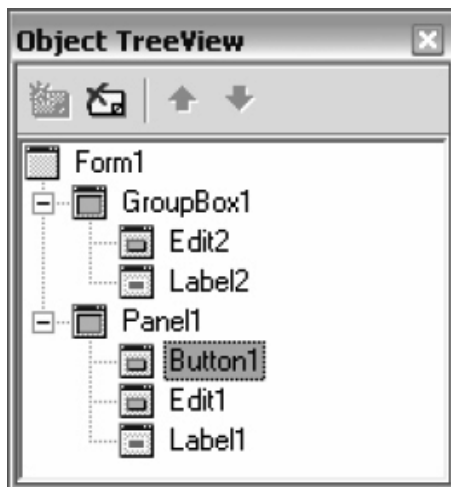
O **Object Inspector** é a janela responsável pelo controle das propriedades e eventos de todos os componentes da aplicação. Também é permitido por ele encontrar componentes de um formulário facilmente através da lista na parte superior, onde são exibidos todos os componentes do formulário ordenados alfabeticamente pelos seus nomes. Observe a figura ao lado.



### 3.5 - OBJECT TREE VIEW

---

O **Object Tree View** é a janela localizada por default logo acima do Object Inspector. É nela que temos uma visão hierárquica dos componentes do formulário, com visão em árvore. Através dele podemos ver em qual container – Panel, GroupBox, ScrollBox, o próprio Form, etc., chamados pelo Delphi de **Parents** dos componentes, cada componente está contido, podendo inclusive transferir de um parent para outro. Para fazer isto, basta arrastá-lo para cima do ícone do novo parent e ele será mudado para lá sem perder o valor de suas propriedades. É possível deletar um componente pelo Object Tree View. Outra facilidade é a localização de componentes. Geralmente, alguns componentes podem ficar de difícil acesso pelo Form Designer, tanto visualmente quanto pelo mouse, através dele é possível encontrá-lo. Veja a figura a seguir do Tree View.



### 3.6 - PALHETA DE COMPONENTES

---

A **palheta de componentes** é uma barra de ferramentas onde se encontram todos os componentes que o desenvolvedor irá utilizar para programar. Ela está dividida em várias palhetas onde os componentes são agrupados. A seguir está uma visão das palheta de componentes.

## 4.1 - OBJECT PASCAL

---

Este capítulo tem por objetivo apresentar a linguagem que o Delphi utiliza que é o Object Pascal. Lembrando que o Delphi não é uma linguagem, mas sim uma Ferramenta de Desenvolvimento que utiliza a linguagem Object Pascal.

### Conceitos de Programação Orientada a Objetos

Antes de aprendermos a utilizar a ferramenta propriamente dita, veremos alguns conceitos básicos de Programação Orientada a Objetos.

**Classe:** Definição de tipo dos objetos, modelo de objeto.

**Objeto:** Instância de classe, variável cujo tipo é uma classe.

**Atributos:** Variáveis de instância. São os dados de um objeto.

**Métodos:** Funções e procedimentos de um objeto.

**Propriedades:** Nome usado para evitar o acesso direto aos atributos de um objeto, onde é possível especificar métodos que serão usados para ler e atribuir seus valores a esses atributos.

**Mensagens:** Chamada de métodos, leitura e atribuição de propriedades.

**Encapsulamento:** Conjunto de técnicas usadas para limitar o acesso aos atributos e métodos internos de um objeto.

**Herança:** Possibilidade de criar uma classe descendente de outra, aproveitando seus métodos, atributos e propriedades.

**Ancestral:** Super classe ou classe de base, a partir da qual outras classes podem ser criadas.

**Descendente:** Subclasse.

**Hierarquia de Classes:** Conjunto de classes ancestrais e descendentes, representada em uma árvore hierárquica.

**Polimorfismo:** Capacidade de redefinir métodos e propriedades de uma classe em seus descendentes.

## 4.2 - ESTRUTURA DE UMA UNIT

---

Examinando um código gerado para um novo Form, identificando as principais seções de uma Unit típica.

**Unit:** A primeira declaração de uma unit é seu identificador, que é o nome do arquivo.

**Interface:** Seção interface, é onde fica as declarações que podem ser

## 5.1 - ESTRUTURAS DE DECISÃO

---

### 1 - If

O if é uma estrutura de decisão utilizada para realizar instruções em determinadas condições. Ele é considerado uma só instrução, então, é encontrado o ponto-e-vírgula no final. O else é opcional.

```
if Opn.Execute then  
  Img.Picture.LoadFromFile(Opn.FileName);
```

```
if Media < 5 then  
  ShowMessage('Inapto')  
else  
  ShowMessage('Apto');
```

### 2 - Case

Permite que o fluxo da execução seja desviado em função de várias condições de acordo com o valor do argumento, que obrigatoriamente tem que ser ordinal, caso o valor do argumento não corresponda a nenhum dos valores listados, é possível incluir um else.

```
case Ch of  
  ' ': ShowMessage('Espaço');  
  '0'..'9': ShowMessage('Número');  
  '+', '-', '*', '/': ShowMessage('Operador');  
else  
  ShowMessage('Caracter especial');  
end;
```

```
case CbbBorda.ItemIndex of  
  0: BorderStyle := bsDialog;  
  1: BorderStyle := bsSingle;  
  2: BorderStyle := bsSizeable;  
end;
```

## 5.2 - ESTRUTURAS DE REPETIÇÃO

---

### 1 - While

Executa uma instrução até que uma condição seja falsa.



## **6.1 - PRINCIPAIS CARACTERÍSTICAS**

---

Neste capítulo vamos se familiarizar com o Delphi, abordando os seus principais componentes e as suas principais propriedades.

### **Características**

- É Gerado um executável verdadeiro, independente de run-time.
- Utiliza-se da linguagem Object Pascal na programação dos procedimentos do programa.
- Utiliza-se do processo de desenvolvimento Two-Way, permitindo tanto escrever o código em Object Pascal gerando assim os objetos visuais, como utilizar os métodos visuais gerando código em Object Pascal.
- Os componentes são como objetos, o que permite a herança.
- Permite criar novos componentes na própria linguagem.
- Possui acesso facilitado a banco de dados.
- Possui ambiente de depuração integrado.
- Possui componentes para a internet.

## **6.2 - CARACTERÍSTICAS DA PROGRAMAÇÃO**

---

- Um programa Delphi é uma estrutura de aplicativo orientada ao desenho de formulários ou janelas.
- A interface com usuário é confeccionada através de componentes.
- Possui um conjunto de controles pré-desenvolvidos que dão acesso às características do sistema.
- Os componentes são objetos.
- Cada controle ou componente possui propriedades, métodos e pode responder a eventos.
- As propriedades podem ter seus valores definidos em tempo de desenvolvimento e alterados em tempo de execução.
- Os eventos são as mensagens de resposta de cada componente, tendo associado a eles um procedimento de evento.

## **6.3 - ELEMENTOS DA PROGRAMAÇÃO**

---

A seguir vamos ver alguns elementos da programação feito através do Delphi.