**Curso de Delphi**

**Resumo 1**

**Capítulo 2: O Ambiente de Desenvolvimento Integrado do Delphi 7**

Fundamentos em: Composição do Ambiente de Desenvolvimento  
Apresentação do Ambiente de Desenvolvimento Integrado do Delphi 7

*Principais janelas:*

*Object inspector: exibe as propriedades e eventos do objeto selecionado no formulário. O valor definido para as propriedades será o valor inicial, que poderá ser alterado em tempo de execução, via código.*

*Paleta de componentes: exibe todos os componentes visuais(controles) e não visuais disponíveis.*

*Project manager: Lista todos os projetos no grupo de projetos atual e os arquivos de cada projeto. Um grupo de projetos pode ter quantos projetos quantos forem necessários. Todo projeto novo sempre é criado dentro de um grupo de projetos.*

*Structure: mostra a estrutura hierárquica dos componentes dentro do formulário ativo. Alguns componentes podem conter outros (****containers****)****.***

Formulários – Os Elementos de Criação da Interface com o Usuário.

*Todos os outros elementos serão colocados dentro de um formulário, que representa uma janela.*

Controles e Componentes.

*Componentes são objetos que fornecem uma funcionalidade específica para o aplicativo.*

*Existem componentes visuais e não visuais.*

*Componentes não visuais pode ser utilizados simplesmente adicionando o seu código fonte ao projeto. Se estiverem instalados na paleta de componentes, podem ser adicionados ao formulário e ficam representados por um ícone e podem ter suas propriedades e eventos configurados pelo object inspector. Este ícone não estará visível quando o programa for executado.*

*Componentes visuais são também chamados de controles. O uso comum é adicionar os controles ao formulário para projetar a aparência da interface e manipular as propriedades pelo código, embora qualquer controle possa ser criado e adicionado ao form em tempo de execução.*

Propriedades

*Propriedades são valores que controlam o acesso aos atributos dos objetos.*

*Ex: Classe TForm 🡪 propriedade Caption: quando atribuída, automaticamente muda o título da janela. Também pode ser lida e manipulada como qualquer string.*

*As propriedades podem ser de qualquer tipo, inclusive de outras classes.*

*Exemplo: TForm.ActiveControl, que determina qual o controle ativo quando o formulário é aberto. É do tipo TWinControl, ou seja, qualquer controle pode pode ser atribuído a esta propriedade.*

Métodos

*(Explicado mais a frente ao introduzir o conceito de classes e objetos)*

Eventos

*Eventos são “modelos” de métodos que podem ser chamados em determinadas situações para informar ao aplicativo da ocorrência de uma determinada ação.*

*Exemplo: TButton tem o evento OnClick que é disparado sempre que o usuário clica com o mouse ou pressiona Enter sobre o botão (possivelmente também em interfaces TouchScreen).*

*Dizemos modelo porque cada evento define o tipo de método que pode ser associado a ele. Este tipo se refere ao número e tipo dos parâmetros.*

*Quando um método é associado a um evento, dizemos que o método* ***trata*** *o evento, ou seja, executa algo em resposta à notificação enviada pelo componente.*

*Todos os eventos tem por padrão um argumento Sender: TObject, que representa o objeto que disparou o evento. Isto significa que um único método pode tratar eventos de vários componentes, diferenciando um do outro pelo parâmetro Sender, desde que os eventos sejam compatíveis quanto aos parâmetros. A maioria dos eventos tem apenas este parâmetro.*

*No Delphi, apenas um método pode ser associado a um evento de cada vez.*

*É padrão todos os eventos terem um nome começado com “On” seguido da ação que ele representa.*

O Object Inspector

*O Object Inspector é dividido em 2 guias: uma para as propriedades e outra para os eventos.*

Alterando o Valor de uma Propriedade no Object Inspector

*Na guia Properties, aparece a lista das propriedades do objeto seleciona no form. As propriedades de tipos primitivos (integer, double, char, string, boolean, etc) e alguns outros tipos permitem a digitação diretamente na linha correspondente. Outros tipos apresentam uma lista de valores ou abrem uma interface específica para entrada de valores. Exemplo: fontes, cores.*

Definindo Procedimentos Associados a Eventos.. 20  
O Conceito de Projeto de uma Aplicação. 22  
Iniciando um Novo Projeto. 28  
Salvando o Projeto Recém-criado. 29  
Fechando um Projeto 31  
Manipulando Grupos de Projetos. 32  
O Code Explorer. 35  
Desabilitando o Recurso de Ancoragem de Janelas. 36  
Garantindo a Visibilidade de uma Janela. 37

**Capítulo 3: Fundamentos da Linguagem Object Pascal 39**

Fundamentos em: Estrutura de uma Unidade de Código (Unit) 40  
Examinando o Código de uma Unidade de Código (Unit) Gerado Pelo Delphi 7. 40  
Examinando as Seções de uma Unit.. 41  
Acessando Tipos e Variáveis Definidos em Outras Units. 42  
Fundamentos em: Declaração de Variáveis.. 42  
O Conceito de Variáveis.. 42  
Atribuindo um Valor a uma Variável.. 43  
Tipos de Dados Predefinidos na Linguagem Object Pascal. 43  
Tipos de Variáveis Inteiras 43  
Tipos de Variáveis Reais 44  
Tipos de Variáveis Booleanas.. 44  
Tipos de Variáveis Para Manipulação de Caracteres.. 44  
Tipos de Variáveis Para Manipulação de Arquivos 45  
Tipo Genérico de Variáveis. 46  
Comentários. 46  
Comentários de uma única linha.. 46  
Comentários de múltiplas linhas.. 46  
Definindo Novos Tipos de Dados. 46  
Tipos de Dados Enumerados 47  
Conjuntos.. 47  
Variáveis Compostas.. 49  
Vetores (Arrays) 49  
Operadores Aritméticos 51  
Tipos Ordinais 52  
Escopo e Tempo de Vida das Variáveis 52  
Variáveis Locais. 52  
Variáveis Globais a uma Unidade de Código 53  
Criação de Variáveis Globais a uma Aplicação.. 53  
Adicionando uma Nova Unit ao Projeto Para Armazenamento de Variáveis Globais 53  
Fundamentos em: Blocos de Comandos, Estruturas Condicionais e de Repetição 54  
Alterando o Fluxo de Execução do Programa. 54  
Blocos de Comandos 55  
Estruturas Condicionais. 55  
Estrutura Condicional if-then-else 55  
Estrutura Condicional Case Of.. 56  
Testes Condicionais. 56  
Os Operadores Relacionais.. 56  
Estruturas de Repetição 57  
Laços For.. 57  
Laços While 58  
Laços Repeat 58  
Condições Compostas 59  
Operadores Lógicos da Linguagem Object Pascal. 59  
Fundamentos em: Funções e Procedimentos 59  
A Estratégia de Dividir Para Conquistar 60  
Procedimentos (Procedures). 60  
Funções 61