

Revista The Club Megazine - 05/2002

A utilização, reprodução, apropriação, armazenamento em banco de dados, sob qualquer forma ou meio, de textos, fotos e outras criações intelectuais em cada publicação da revista "The Club" são terminantemente proibidos sem autorização escrita dos titulares dos direitos autorais.

Copyright© The Club® 2002



THE CLUB

THE CLUB

Rua Acre, 950 - Avaré - SP - CEP 18.700-260

Informações: (0xx14) 3732-3689

Suporte: (0xx14) 3733-1588

Fax: (0xx14) 3732-0987

Internet

<http://www.theclub.com.br>

Cadastro: cadastro@theclub.com.br

Suporte: suporte@theclub.com.br

Informações: info@theclub.com.br

Dúvidas

Correspondência ou fax com dúvidas devem ser enviados ao - THE CLUB, indicando "Suporte".

Opinião

Se você quer dar a sua opinião sobre o clube em geral, mande a sua correspondência para a seção "Tire sua dúvida".

Reprodução

A utilização, reprodução, apropriação, armazenamento em banco de dados, sob qualquer forma ou meio, de textos, fotos e outras criações intelectuais em cada publicação da Revista "The Club" são terminantemente proibidos sem autorização escrita dos titulares dos direitos autorais.

Copyright® The Club® 2001

Impressão e acabamento:

Impressos Gril - Gril Gráfica e Repr. Ind. Ltda.

Tel.: (0xx14) 3762.1345 - Fax: (0xx14) 3762.1259

Rua São Paulo, 447 - Cep 18.740-000

Taquarituba - SP

Tiragem: 5.000 exemplares

Diretor - Presidente

Celso Jefferson M. Paganelli

Diretor Técnico

Mauro Sant'Anna

Colaboradores

Mário Camilo Bohm

Editora Relativa

Delphi é marca registrada da Borland International, as demais marcas citadas são registradas pelos seus respectivos proprietários.

Editorial

Olá amigos,

Primeiramente eu gostaria de agradecer a todos pelos e-mails com sugestões e palavras de incentivo que temos recebido. Neste mês estamos trazendo matérias muito interessantes a respeito de um novo componente que veio junto com o Delphi 6, o ActionBar. Outra matéria muito interessante e que auxiliará muitos sócios é a matéria sobre impressora fiscal e também as dicas e truques. Caso você queira ver alguma matéria publicada em nossa revista, envie um e-mail para info@theclub.com.br.

Até a próxima

CJPaganelli

Celso Jefferson Paganelli
Presidente - The Club

Editorial	03
News	04
Criando e usando DLLs na Plataforma .NET	08
Dicas de instruções Oracle... ..	13
Criando uma barra de ferramentas no estilo Outlook	16
Como obter uma ROM a partir de seu Palm	20
Page Size - O Mistério	22
Controlando o Preview do QuickReport em Run-Time	23
Tratamento do Repique e 'gorjeta' após e antes da emissão do Cupom Fiscal	26
Perguntas & Respostas	29

IBExpert Personal Edition - versão Freeware

A linha de produto IBExpert da HK-Software, Alemanha, tem um novo membro. **IBExpert Personal Edition.**

O IBExpert é um ambiente de desenvolvimento integrado para trabalhar com o Borland Interbase (tm)/Firebird Database Server.

Depois da versão comercial e a versão free para propósitos educacionais, esta nova versão é uma ferramenta de lançamento para o Interbase e o Firebird.

Esta versão é dedicada para pessoas que trabalham com o Interbase ou Firebird e não estão satisfeitos com o IBConsole.

O IBExpert Personal Edition cobre todas as funções do IBConsole e adiciona mais algumas características de scripts SQL, administração do banco de dados e programação.

Você pode fazer o download do IBExpert no endereço <http://www.hksoftware.net/download>

A versão comercial tem mais características, como diagramas com engenharia reversa, debugger de stored procedures e trigger, ferramenta de performance, gerenciamento de permissões e muito mais.

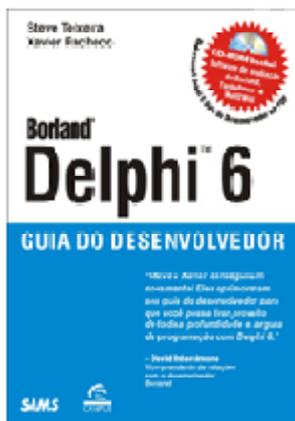
Você pode fazer o download desta versão no endereço <http://www.ibexpert.com>.

Você também pode encontrar mais informações, como pegar a versão educacional.

O IBExpert suporta o Interbase(tm) Open Edition 6.X, Interbase(tm) Desktop e Server Edition 6.X e o Firebird 1.X.

Para maiores informações sobre as ferramentas SQL-Database Systems for Windows, Linux, Solaris e outro sistema operacional, você pode encontrar nos endereços:

<http://www.borland.com/interbase> and <http://www.firebirdsql.org>



Delphi 6 Guia do Desenvolvedor

Steve Teixeira • Xavier Pacheco

Lançado em 04/2002

740 pág

ISBN 85-352-0955-7.

Escrito por desenvolvedores para desenvolvedores, Delphi 6 continua a oferecer a referência completa e o guia definitivo para a versão mais recente do Delphi. Inclui CD-Rom.

Onde comprar:

<http://www.campus.com.br>

Ligue Grátis: 0800-265340

Rio de Janeiro:(0xx21) 3970-9300

São Paulo:(0xx11) 3259-9944

Intel revela detalhes do seu computador do futuro

Durante os próximos dois anos, os consumidores e a indústria de PCs testemunharão mudanças significativas na forma pela qual os PCs são projetados, revelaram os executivos da Intel no evento Intel Developer Forum (IDF) organizado pela própria fabricante de chips.

No coração dessa mudança está a iniciativa Big Water da companhia, criada para determinar como os futuros PCs podem ser projetados para incorporar os avanços recentes e futuros da tecnologia.

As especificações técnicas do Big Water ainda estão guardadas nas mangas da Intel. A empresa, no entanto, resolveu revelar alguns detalhes sobre o projeto, que será finalizado em 2003 e deverá chegar ao mercado somente em 2004.

O Big Water, segundo ela, contará com a tecnologia PCI (Peripheral Component Interconnect) Express, uma especificação capaz de agilizar a transferência de dados entre os componentes de um PC. Essa

tecnologia oferece um canal de comunicação de até 200 Mbps - velocidade quase duas vezes maior que o sistema PCI existente. Para a Intel, esse aumento é indispensável para que os futuros computadores acompanhem o avanço dos processadores e das redes, e suportem uma quantidade maior de capacidades gráficas.

Uma outra característica do Big Water será a flexibilidade. A Intel, em sua apresentação, demonstrou como um PC baseado na sua nova tecnologia vai incluir módulos adicionais, de forma que os usuários poderão atualizar de forma simples vários componentes do PC, como drives de disco rígido. O design do Big Water também será bem menor que os PCs disponíveis atualmente no mercado. Como consequência, a Intel quer que seu novo computador seja projetado para ocupar lugares onde os consumidores raramente pensariam em colocar suas máquinas, como na parede, por exemplo. Fonte: IDGNOW

Imation lança player MP3 com mini CD-R

Conhecida pela sua linha de mídias de armazenamento, a Imation ampliou sua família de soluções com o **RipGo!**, um curioso player MP3 que também funciona como gravador de CD-R.

O conceito lembra vagamente o HipZip da Iomega, porém, em vez de utilizar uma mídia proprietária, o RipGo! funciona com pequenos CDs de 8 cm, chamados CD singles. A única diferença em relação aos CDs convencionais de 12 cm é a menor capacidade de armazenamento – 185 MB –, algo como 125 disquetes de 3 ½ polegadas, seis horas de música no formato WMA ou três horas em MP3. Uma das grandes vantagens dessa solução é que esses pequenos discos podem ser lidos em praticamente qualquer PC equipado com unidade de CD-ROM.

Na cor cinza metálico, o RipGo! lembra um pequeno CD player portátil que, apesar de leve (230 gramas) e relativamente compacto (9,8 cm por 12,6 cm por 2,4 cm – LxAxP), não entra facilmente em qualquer bolso de camisa. O compartimento do CD fica embaixo do painel de controle, que também faz o papel de tampa. Essa solução apresenta uma pequena desvantagem, já que, ao pressionarmos os botões, sentimos a tampa ceder um pouco. Apesar de ser normal, essa sensação pode passar uma idéia de fragilidade do equipamento.

Como player, o RipGo! possui controles relativamente simples e diretos, porém, sem dispensar a leitura do manual do usuário.

Depois de inserir o CD e apertarmos a tecla *Play*, o RipGo! entra em ação, varrendo todo o disco – processo que leva algum tempo – à procura de músicas no formato de áudio MP3 ou WMF (Windows Media File). O nome e o autor das músicas podem ser identificados por meio de seus rótulos (padrão id3), formando listas que são apresentados no visor LCD do player.

Os recursos do player são os mesmos disponíveis em outros modelos, como volume, avanço, retrocesso, programação e repetição de trilhas, reprodução aleatória, equalização, etc. Uma chave seletora que permite que a saída do fone de ouvido seja usada como uma conexão do tipo line-out para equipamentos de som.

Acompanham o RipGo! uma bateria recarregável de íon-lítio com autonomia estimada de 5 horas de uso contínuo, fone de ouvido, carregador, cabo de comunicação USB, um mini CD de instalação e dois mini CD-R. A Imation oferece essa mídia compacta na sua linha Neon Colors ao preço sugerido de R\$ 2,30 cada (cerca de R\$ 0,012 por megabyte gravado).

CD-R de bolso

O RipGo! se comunica com qualquer PC (e até Mac) por meio

da porta USB. Depois de devidamente instalado (o que inclui o uso de conversor de rede elétrica) e reconhecido pelo sistema operacional, o equipamento se comporta como uma unidade de CD, não havendo praticamente diferenças no seu modo de ler e gravar dados. Acompanham o produto o Nero Burning ROM versão 5, Media Jukebox, além de drivers e versões atualizadas do Windows Media Player, documentação em PDF e amostras de músicas em MP3.

Nos testes realizados, o RipGo! teve um desempenho modesto – em parte causado pelo uso da interface USB –, alcançando taxas de transferência de até 583 KB/s (no modo *buffered*) e tempo de acesso de 735 ms. Com sua velocidade de gravação limitada pelo barramento USB – 600 KB por segundo ou 4X –, o RipGo!

gravou um mini CD com aproximadamente 18min14s de música em aproximadamente 5min52s. Repetimos a experiência com um volume de dados de formatos diversos, totalizando 185 MB em 3.480 arquivos em 109 pastas. O tempo gasto para gravar a mídia foi de 7min7s.

Como player, o RipGo! mostrou ser bastante resistente a saltos na música causados por trepidações. O som é de boa qualidade, apesar de poder não ser do agrado de apreciadores de volume alto.

Após algum tempo de uso, notamos um aquecimento demasiado na sua base durante a reprodução de músicas, fato também observado durante o processo de gravação de discos.

Além de um simples reproduzidor de músicas, o RipGo! pode ser uma solução muito interessante para armazenar e distribuir dados, principalmente entre aqueles que acham um desperdício de mídia gravar poucos dados em um CD-R convencional. Entretanto, o equipamento perdeu a oportunidade de ser uma solução perfeita de armazenamento removível por ainda não ser compatível com o padrão CD-RW.

Imation : (11)3901-7026 - <http://www.imation.com.br>



Pró	Versátil, utiliza mídia de baixo custo.
Contra	Esquenta muito após algum tempo de uso; não grava CD-RW.
Resumo	Player MP3 capaz de gravar mini CD-R.
Preço	US\$ 320.
Onde comprar	http://www.imation.com.br

Microsoft lança mouse sem fio operado por rádio homologado pela Anatel

Controlado por rádio, o Wireless IntelliMouse Explorer é o único homologado pela Anatel para operar no Brasil

A Microsoft Brasil anunciou o lançamento do Wireless IntelliMouse Explorer, o mouse sem fio mais avançado da linha de hardware da empresa. O produto, que opera por rádio e possui homologação da Anatel para ser usado no Brasil chega às lojas no final de maio com preço sugerido de R\$ 265. "A homologação é importante porque o sinal de rádio de equipamentos sem fio pode interferir em outras transmissões, como toda a comunicação de tráfego aéreo, por exemplo. Além disso, é a garantia de que o usuário está comprando um produto seguro, testado segundo normas internacionais", explica Maria Carolina Braga Vianna, gerente de marketing de consumo da empresa.

Com alcance de até 1,8 metros de distância do sensor, o novo mouse da Microsoft possui dois canais separados de transmissão que permitem ao usuário escolher a frequência que utilizará. O equipamento funciona a pilha, modelo AA, e vem com um sofisticado sensor que indica a vida útil da bateria e quando ela deve ser trocada por meio do software IntelliPoint.

Outra vantagem do produto é a tecnologia IntelliEye, que foi reformulada e está em uma nova versão. A plataforma substitui a tradicional bolinha por sensores infra-vermelhos com precisão de movimento 140% superior aos mouses ópticos existentes no mercado. O mouse conta ainda com cinco botões programáveis, que podem ser usados para gravar comandos nos aplicativos mais utilizados pelo usuário. "O Wireless IntelliMouse Explorer traduz os esforços da Microsoft em desenvolver equipamentos que agreguem valor à vida dos usuários, seja por meio de novas tecnologias, preocupação com a saúde, ou seja pelo design ergonômico", destaca Maria Carolina.

Microsoft Hardware

A linha de hardware da Microsoft é voltada para usuários de computador e fãs de games de PC. Além de dez tipos de mouses e

teclados com funcionalidades específicas para cada pessoa, a empresa possui ainda uma ampla família de joysticks, volantes e gamepads ergonômicos que potencializam a experiência de jogos. Todos os produtos são desenvolvidos com base em extensas pesquisas com usuários.

Perfil Microsoft

A Microsoft Corporation ocupa a primeira posição no ranking mundial das empresas de software. Historicamente, a empresa tem investido US\$ 5,4 bilhões em pesquisa e desenvolvimento, um dos maiores orçamentos do mundo. A subsidiária brasileira foi inaugurada em setembro de 1989 e, nestes 12 anos, conseguiu atingir a marca de 45 mil pessoas que, de alguma forma, desenvolvem software, prestam serviços, treinam e capacitam profissionais na plataforma. Este contingente está espalhado por 10 mil empresas brasileiras, gerando transferência de tecnologia e capacitação profissional para a população. Somente nos últimos três anos, a Microsoft Brasil investiu US\$ 15 milhões em projetos sociais no país para levar a tecnologia a comunidades carentes, em parceria com organizações como CDI - Comitê para Democratização da Informática, Instituto Ayrton Senna e Projeto Aprendiz.



foto ilustrativa

Atendimento ao cliente Microsoft: (11) 3444-6844
Para mais informações à imprensa, contate:

S2 Comunicação Integrada S/C Ltda.

www.s2.com.br

José Luiz Schiavoni (MTb 14.119) - joseluiz@s2.com.br

Priscila Rocha (MTb 19.977) - prirocha@s2.com.br

Tel: (11) 3457-0200 - Fax: (11) 3457-0222

Áreas de atendimento:

Varejo e Games

Ana Carolina Fullen, acarol@s2.com.br - (11) 3457-0204

Corporativo

Saulo Filho, saulof@s2.com.br - (11) 3457-0202

Roberta M. Pagnaca, robertam@s2.com.br - (11) 3457-0223

Microsoft oferece descontos para pequenas empresas

Pacote de aplicativos Office XP terá descontos de até 43% negociados em dez vezes sem juros. Sistema operacional Windows XP entra na promoção 10% mais barato

A Microsoft Brasil acaba de lançar mais uma promoção para o mercado de pequenas empresas. Na compra de quatro ou mais licenças da versão completa do Office XP Standard (Open NL), a empresa ganha três pacotes adicionais do programa. Open NL é um contrato de licenciamento para no mínimo cinco licenças de software da Microsoft, válido exclusivamente para empresas com CNPJ.

Com essa oferta, que vai até 28 de junho, o cliente Microsoft pode obter descontos de até 43%. A iniciativa faz parte da estratégia de atender as pequenas empresas que necessitam de mais infra-estrutura de tecnologia e de um maior número de PCs para agilizar seus negócios. Os pacotes de Office XP podem ainda ser negociados em dez vezes sem juros.

“É uma forma de oferecer o produto por um preço atrativo visando atender às necessidades de aumento de produtividade nas pequenas empresas. A oferta também ajudará as companhias no processo de legalização do pacote de escritório da Microsoft, que pode ser adquirido por uma oferta dentro da realidade do pequeno empresário”, afirma Carlos Arantes, gerente de marketing pequenas empresas da Microsoft.

Para divulgar o Office promocional, a Microsoft iniciou uma grande campanha publicitária de alto impacto na mídia impressa, em emissoras de TV e no cinema. Com o conceito “Uma empresa não precisa ser grande para parecer grande”, a campanha tem o objetivo de informar os pequenos empresários, em uma linguagem simples e clara, como o pacote de aplicativos pode garantir melhor controle de seus negócios, atendimento personalizado e aperfeiçoamento do gerenciamento das vendas.

Descontos para Windows XP

Além do pacote Office XP, a promoção oferece 10% de desconto para a compra de cinco

licenças, no mínimo, da versão completa do sistema operacional Windows XP Professional (Open NL), que também será vendido em dez vezes sem juros, até o final de junho.

Mais informações e um exemplo fictício ilustrativo podem ser encontrados no site www.microsoft.com/brasil/office/pequenaempresa.

Nesse endereço, o empresário encontra modelos de folhetos, convites, catálogos de produtos e boletins informativos, prontos para serem adaptados e utilizados.

Mais informações sobre as promoções podem ser encontradas no site: www.microsoft.com/brasil.

Atendimento ao cliente Microsoft: (11) 3444-6844
Para mais informações à imprensa, contate:

S2 Comunicação Integrada S/C Ltda.

www.s2.com.br

José Luiz Schiavoni (MTb 14.119) - joseluiz@s2.com.br

Priscila Rocha (MTb 19.977) - prirocha@s2.com.br

Tel: (11) 3457-0200 - Fax: (11) 3457-0222

Áreas de atendimento:

Varejo e Games

Ana Carolina Fullen, acarol@s2.com.br - (11) 3457-0204

Corporativo

Saulo Filho, saulof@s2.com.br - (11) 3457-0202

Robertta M. Pregnaca, robertam@s2.com.br - (11) 3457-0223

Windows e Office

Silvia Angerami, (MTb 16.182), sangerami@s2.com.br

(11) 3457-0225

Ana Paula Aido (MTb 30.184), palmeida@s2.com.br

(11) 3457-0204



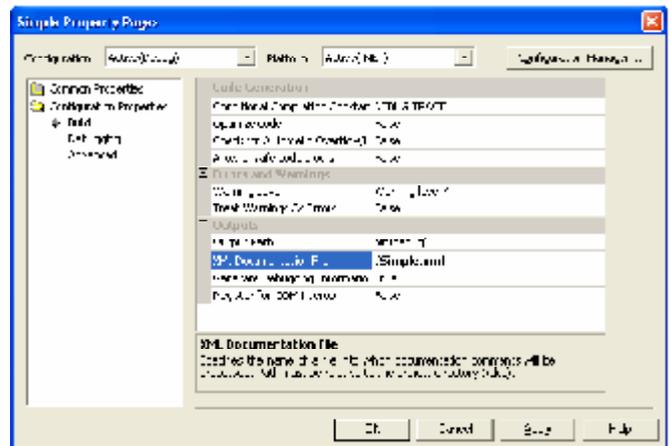
Digite o código da classe, mostrado a seguir:

```
namespace BizObj {
    /// <summary>
    /// Do simple arithmetic
    /// </summary>
    public class Contas {
        double _N1;
        double _N2;
        /// <summary>
        /// Default constructor
        /// </summary>
        public Contas() {
            _N1 = 0;
            _N2 = 0;
        }
        /// <summary>
        /// Assign two values
        /// </summary>
        /// <param name="x">One value</param>
        /// <param name="y">Another value
        /// </param>
        public void Assign
            (double x, double y) {
            _N1 = x;
            _N2 = y;
        }
        /// <summary>
        /// Add the numbers
        /// </summary>
        /// <returns></returns>
        public double Add() {
            return _N1 + _N2;
        }
        /// <summary>
        /// Multiply the numbers
        /// </summary>
        /// <returns></returns>
        public double Multiply() {
            return _N1 * _N2;
        }
        /// <summary>
        /// Get the first number
        /// </summary>

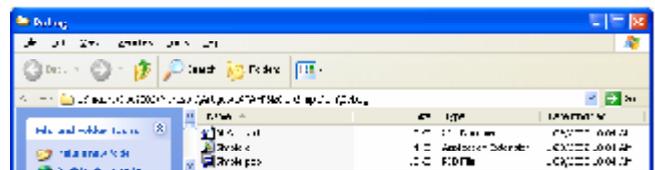
```

```
        public double N1 {
            get {
                return _N1;
            }
        }
        /// <summary>
        /// Get the second number
        /// </summary>
        public double N2 {
            get {
                return _N2;
            }
        }
    }
}
```

Queremos que usuários da classe tenham acesso à documentação on-line digitada no fonte acima. Para isto, clique com o botão direito sobre o projeto no "Solution Explorer" e digite o nome do arquivo XML no campo "XML Documentation File". Este arquivo deve ter o mesmo nome que a DLL com a extensão trocada de "DLL" para "xml":

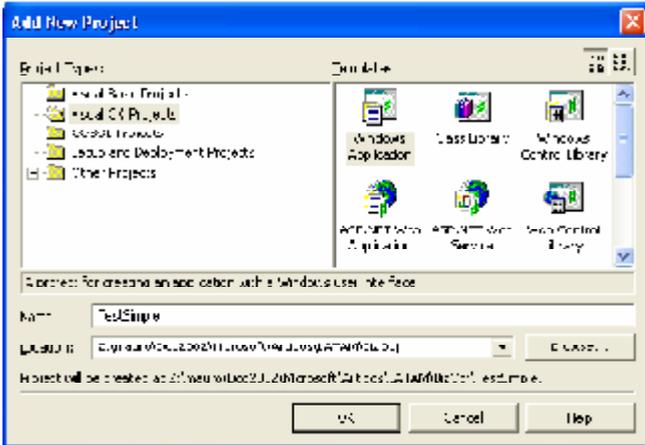


Compile o programa acima. Observe que são criados dois arquivos principais: uma DLL e um XML (o arquivo PDB contém informações de depuração e pode ser apagado):

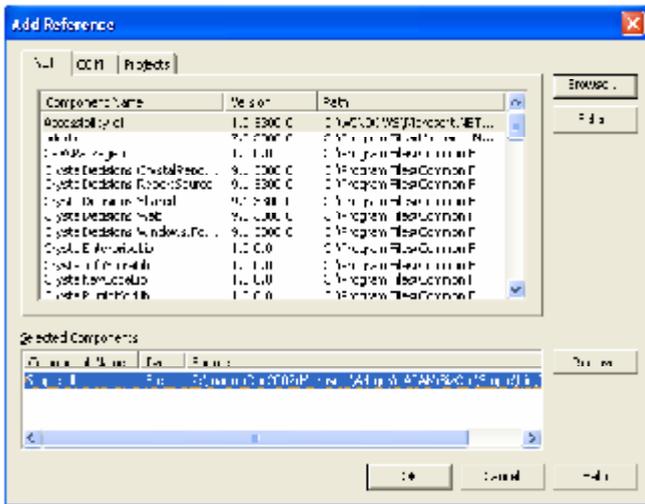


Usando a DLL

Para usar a DLL, crie um projeto do tipo executável, por exemplo, um aplicativo Windows:



Selecione "Project | Add Reference..." e selecione a DLL criada anteriormente:

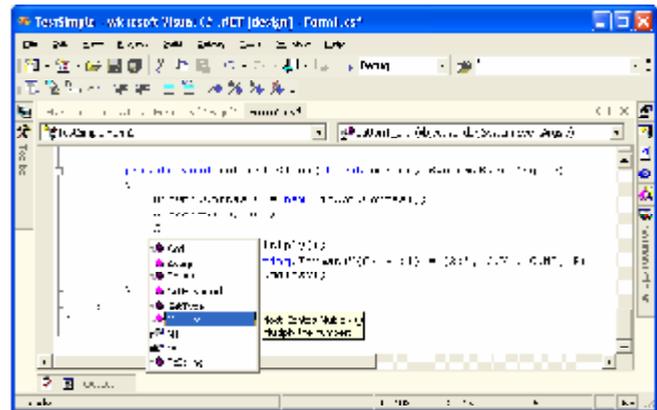


Podemos agora usar o componente normalmente. No nosso caso, adicionamos um *Button* e um *ListBox* ao formulário. O código do evento *Click* do *Button* é o seguinte:

```
private void button1_Click(object sender,
System.EventArgs e)
{
    BizObj.Contas C = new BizObj.Contas();
    C.Assign(10, 40);
    float R = C.Multiply();
    string Msg = string.Format("{0} + {1} = {2}",
```

```
C.N1, C.N2, R);
    listBox1.Items.Add(R);
}
```

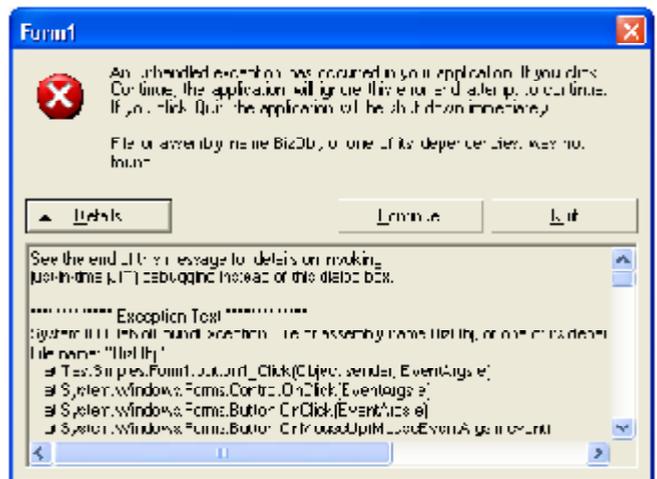
Observe que a documentação XML colocada no fonte do componente estará disponível como "dica":



Observe o seguinte:

- NÃO foi necessário registrar o componente, como teríamos que fazer no COM;
- Quando você adicionou a referência, a DLL foi copiada para o diretório de saída do executável. Este comportamento é, a princípio, necessário.

A DLL não precisa estar presente no momento da carga do programa, como no Windows. Ela é necessária apenas quando for usado algum tipo de dentro dela. Experimente apagar o arquivo ".DLL" e observar que o programa ainda é carregado. No momento em que você clicar no botão para criar o objeto "Contas" é que a DLL será carregada. Caso ela não exista, será gerada uma exception:



O seu programa pode interceptar a exceção acima se quiser. Isto dá uma grande flexibilidade na distribuição de novas versões ou funcionalidade adicional.

Observe que a única informação presente no nosso programa que identifica a DLL é seu nome. Caso você ou alguém crie uma DLL com o mesmo nome, o seu programa irá carregá-la e, caso os tipos dentro dela tenham sido definidos com os mesmos nomes e assinaturas, irá usá-los. Ou seja, é possível usar uma versão diferente do planejado ou mesmo uma DLL “falsificada”, desde que ela tenha o nome esperado e esteja no mesmo diretório do programa.

Por questões de segurança, no .NET não podemos carregar DLLs a partir de locais arbitrários do disco. Esta é a razão de termos que copiar a DLL para o mesmo diretório do programa de testes ou ainda em um sub-diretório deste, mas não em um local arbitrário do disco.

Caso você deseje colocar a DLL em um subdiretório, você deve criar um arquivo de configuração para o aplicativo com a entrada “probing privatePath”. Este arquivo, no formato XML, tem o mesmo nome do executável mais a extensão “.config”, por exemplo, “TestSimples.exe.config”. Veja um exemplo que indica para serem procurados vários subdiretórios:

```
<?xml version="1.0"?>
<configuration>
  <runtime>
    <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
      <probing privatePath="bin;bin2\subbin;bin3"/>
    </assemblyBinding>
  </runtime>
</configuration>
```

Nomes fortes

Para evitar usar uma versão diferente do planejado, é possível dar à DLL um “nome forte”, composto de:

- Nome da DLL;
- Versão;
- Assinatura criptográfica;
- Cultura (opcional), que identifica o idioma.

Para criar um assembly com “nome forte” é necessário fazer o seguinte:

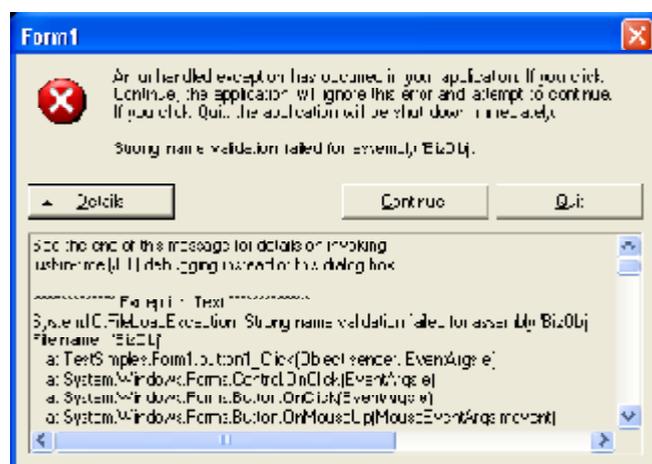
1. Criar uma chave criptográfica com o comando “sn -k MyKeyFile.snk”. O arquivo contendo a chave criada DEVE SER ARMAZENADO COM CUIDADO E NUNCA DESCARTADO. Esta chave criptográfica pode ser útil para melhorar a segurança do seu site. Além disto, é absolutamente necessária para a criação de uma versão mais nova da DLL;

2. Colocar uma diretiva no fonte “AssemblyInfo.cs” para referenciar o arquivo com a chave: [assembly: AssemblyKeyFile(“MyKeyFile.snk”)];
3. Definir um número de versão específico no fonte “AssemblyInfo.cs” de forma a desligar o mecanismo de numeração automática do Visual Studio com a chave [assembly: AssemblyVersion(“1.0.0.0”)];
4. Compilar o projeto;

Ao compilar o programa, a DLL gerada inclui a sua chave pública e um “hash” criptograficamente seguro do arquivo. Além disto, ele é assinado com a chave privada. Por estas razões, uma DLL com nome forte tem as seguintes vantagens:

- Não pode ser falsificada; apenas o possuidor da chave privada pode assinar a DLL com aquela assinatura;
- Não pode ser alterada; caso um bit apenas mude, o sistema não irá carregá-la; isto confere uma proteção tanto “anti-falsificação” como “anti-virus”;
- Nosso programa usará sempre a versão para qual foi desenvolvido; todos programas que usam a DLL incluem o “public key token”, um “hash” de 64 bits da chave pública que indica o autor da DLL;
- O administrador pode indicar uma versão diferente do originalmente desejado no arquivo “.config” associado ao aplicativo, no elemento “bindingRedirect”;
- Podemos carregar a DLL de outros diretórios arbitrários, inclusive via http (Web);
- Podemos registrar a DLL em um local comum chamado “GAC – Global Assembly Cache” para uso por vários programas com o comando “gacutil /i BizObj.dll”, caso deseje que ele seja usado por vários programas.

Se você alterar um bit sequer na DLL do programa (por exemplo, com o “Binary Editor” do Visual Studio .NET), a seguinte mensagem aparecerá quando a DLL for carregada:



A princípio, a DLL com “nome forte” está sujeita às mesmas regras dos assemblies “private” quanto à sua localização, ou seja, no mesmo diretório do programa ou um subdiretório deste.

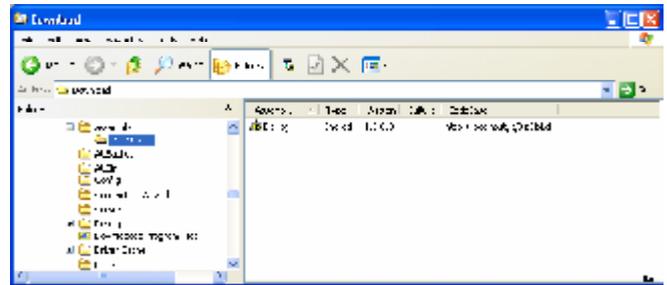
Adicionalmente, um assembly com “strong name” pode ser carregado a partir de um diretório arbitrário. Este comportamento é especificado no elemento “codebase” do arquivo “.config”, como no exemplo a seguir:

```
<?xml version="1.0"?>
<configuration>
  <runtime>
    <gcConcurrent enabled="true" />
    <assemblyBinding xmlns=
      "urn:schemas-microsoft-com:asm.v1">
      <dependentAssembly>
        <assemblyIdentity name="BizObj"
          publicKeyToken="AD61E477AD3C16AC" />
        <codeBase version="1.0.0.0" href=
          "file:///c:/temp/BizObj.dll"/>
        </dependentAssembly>
      </assemblyBinding>
    </runtime>
  </configuration>
```

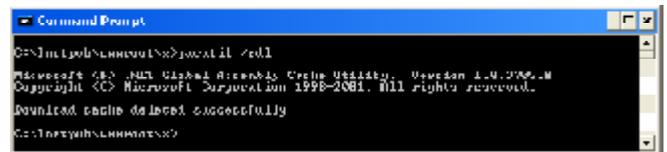
A entrada “href” acima pode indicar um documento baixado via da Web com “<http://>”, como no exemplo abaixo:

```
<?xml version="1.0"?>
<configuration>
  <runtime>
    <gcConcurrent enabled="true" />
    <assemblyBinding xmlns=
      "urn:schemas-microsoft-com:asm.v1">
      <dependentAssembly>
        <assemblyIdentity name="BizObj"
          publicKeyToken="AD61E477AD3C16AC" />
        <codeBase version="1.0.0.0" href=
          "http://localhost/x/BizObj.dll"/>
        </dependentAssembly>
      </assemblyBinding>
    </runtime>
  </configuration>
```

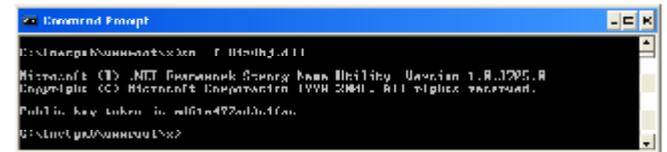
No caso de especificar um arquivo da Web, a DLL irá ser baixada para um cache, no diretório “\WINDIR\assembly\download”:



Podemos apagar entradas neste “cache de download” com o comando “gacutil /cdl”:



Note nos exemplos acima que precisamos do valor do “public key token” para indicar precisamente a DLL no arquivo “.config”. Você pode obter este valor com a opção “-T” (maiúsculo) do utilitário “sn”, por exemplo: “sn -T BizObj.dll”:



Conclusão

Criar e usar DLL não só é fácil como também existem vários recursos para permitir um bom controle de versão e impedir alterações ou atualizações indesejadas.

Mauro Sant'Anna

Mauro é um “MSDN Regional Director”, consultor e instrutor da MAS Informática (www.mas.com.br), tendo ministrado treinamentos na arquitetura .NET desde outubro de 2000 para diversas empresas, dentre as quais a própria Microsoft.

E-mail: mas_mauro@hotmail.com

MONTAGEM DINÂMICA DE SELECTS E “CRITÉRIOS DE SELEÇÃO” NO DELPHI

Ao contrário do que acontece num componente TTable, em que você seleciona uma tabela específica na propriedade TableName, no componente Tquery você irá “montar” a tabela através de um SELECT, dentro da propriedade SQL do componente. Vamos então mostrar algumas técnicas que irão facilitar sua vida...



Note que, sendo um SELECT, você não precisa selecionar obrigatoriamente todos os campos da tabela. No caso do exemplo, que acessa a tabela CLIENTES constante do projeto exemplo do curso, podemos tanto selecionar todos os campos como selecionar apenas alguns campos. Para selecionar todos os campos da tabela CLIENTES utilizamos o SELECT da seguinte forma:

```
SELECT * FROM CLIENTES  
ORDER BY NOME;
```

Já se quisermos apenas os campos CHAVE, NOME e CGC devemos utilizar o SELECT desta outra forma (neste caso somente poderão ser acessados, entretanto, os campos incluídos no SELECT):

```
SELECT CHAVE, NOME, CGC FROM CLIENTES  
ORDER BY NOME;
```

Durante a operação vão surgir necessidades de segmentar consultas por Cliente, por CGC, por Cidade e por muitas outras condições. Como fazer para tornar o SELECT “versátil” e com pouco código? Simples!

Defina 3 variáveis para manipular o SELECT conforme a necessidade do usuário, que poderá ter opções para mudar a seleção, o ordenamento, etc. Assim, no caso do exemplo teríamos as variáveis **clientesBase**, **clientesFiltro** e **clientesOrdem** declaradas como string na Unit principal da aplicação (porque assim podemos enxergá-las de qualquer outro Form).

Se quisermos que todos os clientes sejam selecionados quando o usuário abrir o Cadastro de Clientes por exemplo, iremos trabalhar com as variáveis da seguinte forma:

```
sqlClientes.close;  
ClientesBase := 'SELECT * FROM CLIENTES `';  
ClientesFiltro := ` `;  
ClientesOrdem := 'ORDER BY NOME `';  
sqlClientes.sql.text := clientesBase + clientesFiltro  
+ clientesOrdem;  
sqlClientes.open;
```

Já se o usuário quiser selecionar as informações de um determinado cliente apenas, a utilização das variáveis nos dará flexibilidade para fazê-lo, alterando apenas o conteúdo da variável **clientesFiltro**:

```
sqlClientes.close;  
ClientesFiltro := 'WHERE CHAVE = 3415 `';  
sqlClientes.sql.text := clientesBase + clientesFiltro  
+ clientesOrdem;  
sqlClientes.open;
```

Da mesma forma poderemos proceder com o conteúdo da variável clientesOrdem, sempre que quisermos ordenar a consulta por qualquer outro campo, sem precisar ficar redefinindo o SELECT inteiro:

```
sqlClientes.close;  
ClientesOrdem := 'ORDER BY CGC `';  
sqlClientes.sql.text := clientesBase + clientesFiltro  
+ clientesOrdem;  
sqlClientes.open;
```

O que fizemos foi apenas separar o SELECT em 3 partes:
- a parte básica, que é o código obrigatório para definirmos no SELECT que tabela e que campos queremos selecionar,
- a parte relativa aos critérios de seleção dos registros e
- a parte relativa à ordenação.

Com um pouco de criatividade, à partir deste modelo podemos fazer qualquer mudança no SELECT com muita simplicidade e pouco código. **O grande ganho é que todas as atualizações do SELECT serão resolvidas no servidor.**

DICA: Não utilize, para montar SELECTS, os parâmetros definidos na **propriedade Params** do componente TQuery. Essa técnica torna as consultas extremamente lentas e pesadas, com enorme perda de performance, pelo menos em aplicações que acesses bases Oracle.

RDprint 3.0 Componente de Impressão para Matriciais e Jato de Tinta

Característica e Recursos

- * Preview real com Zoom
- * Setup para Usuário Final
- * Impressão Rápida em Matriciais
- * Montagem dinâmica de Página
- * Fontes Expandida, Normal
- * Comprimento em 12, 17 e 20 cpm
- * Negrito, Itálico e sublinhado
- * Imprime em 6 e 8 LPD
- * Seleção de Página p/ impressão
- * Seleção de Número de Cópias
- * Impressão Local e em REDE

Novos Recursos da Versão 3.0

- * Impressão Gráfica Melhorada
- * Impressora Personalizada
- * Opção para Salvar em Arquivo
- * Desenho Box e Linhas
- * Impressão de Códigos ASCII
- * Impressão alinhada de Valores
- * Compatível com Windows NT/2000

"Mais do que desenvolver sistemas, criamos Soluções"

RDprint Setup

Configuração da Impressão

Local da Impressão:
HP DeskJet 3700 Series em LFT1:

Modelo da impressora:
Linha Epson 9 e 24 agulhas/Job/Layer

Intervalo de Páginas:
 Tudo
 Páginas de 1 até 9999

Quantidade de Cópias:
1

Visualizar

www.deltress.com.br
Componentes e Ferramentas para programadores Delphi
Fone/Fax (14) 427-5322

NVE

AUTHORISED TESTING CENTRE

THE CLUB

Microsoft
CERTIFIED
 Partner

<http://www.theclub.com.br>

Rua Acre, 950 - Cep 18.700-260
 Tel.: (0xx14) 3732-3689 - Fax: (0xx14) 3732-0987

As certificações Microsoft valorizam o profissional, e com certeza facilitam a entrada no mercado de trabalho. A maioria das empresas hoje pedem que o profissional possua alguma certificação, principalmente MCSA, para administradores de rede. Por isso mesmo, não perca tempo. Peça maiores informações agora e venha fazer o seu exame. Entre para o seleto grupo de profissionais certificados, que são reconhecidos mundialmente. Se você está fazendo curso, este é o momento ideal para fazer os seus exames de certificação. Seja um profissional certificado e enriqueça o seu currículo.

Criando uma barra de ferramentas no estilo do Outlook

O ActionBand é um conjunto de componentes que veio para simplificar o processo de desenvolvimento da interface com o usuário. Você pode organizar ações e imagens e adicioná-los a menus e barra de ferramentas no estilo do Microsoft Office. Estes componentes podem ser acessados na palheta de componentes Additional. Nesta palheta você irá encontrar os seguintes **TActionManger**, **TActionMainMenubar**, **TActionToolBar** e **TCustomizeDlg**. Para você ter uma idéia do poder destes componentes, vamos criar uma barra de ferramentas no estilo do Outlook. Chame o Delphi 6 e crie um novo projeto. Agora para criarmos a nossa barra de ferramentas siga os passos a seguir:

Arraste os seguintes componentes:

- ImageList
- ActionManager. Altere as propriedades Images para ImageList1 e a propriedade FileName para Config
- ControlBar. Altere a propriedade Align para alTop e propriedade AutoSize para True.
- ActionMainMenuBar. Arraste este componente para dentro do componente ControlBar.
- ActionToolBar. Arraste este componente para dentro do componente ControlBar.
- Panel. Arraste este componente para o form e altere as propriedades Align = alLeft, Name = pnlMenu, BevelOuter = bvNone
- Splitter. Arraste este componente para o form e altere as propriedades Width = 5, Beveled = True.
- Panel. Arraste este componente para o form e altere as propriedades Align = alClient, Name = pnlBase, BevelOuter = bvNone
- Panel. Arraste este componente dentro do pnlBase e altere

as propriedades Align = alTop, Name = pnlTitle, BevelOuter = bvNone, Color = clGray, Fonte.Size = 14, Font.Color = clWhite, Alignment = taLeftJustify, Caption = Exemplo de utilização do ActionBar

- TreeView. Arraste este componente dentro do pnlBase e altere a propriedade Align = alLeft.
- Splitter. Arraste este componente para dentro do pnlBase e altere as propriedades Width = 5, Beveled = True.
- RichEdit. Arraste este componente para dentro do pnlBase e altere as propriedades Align = alClient.

Agora nós vamos criar a barra de ferramentas vertical.

1. No componente pnlMenu que está no lado esquerdo do form arraste os seguintes componentes:

- 1.1 Panel. Altere a propriedade Name = pnlItem1, BevelOuter = bvLowered, Width = 100, Height = 100
- 1.2 Panel. Arraste este componente para dentro do componente pnlItem1. Altere a propriedade Name = pnlItem1Title, Align = alTop, Caption = Item1, Height = 18.
- 1.3 ActionToolBar. Arraste este componente para dentro do componente pnlItem1. Altere a propriedade Align = alClient, Orientation = boTopToBottom, Color = clGray

2. Vá até o Object Treeview e selecione o componente pnlItem1. Clique com o botão direito do mouse sobre este componente. No popmenu clique em Edit -> Copy.

3. Agora selecione o componente pnlMenu. Clique com o botão direito do mouse sobre este componente. No popmenu clique em

Edit -> Past. Altere a propriedade Name para pnlItem2. Dentro deste componente você tem outro componente Panel. Altere a propriedade para pnlItem2Title.

4. Repita as operações 2 e 3 mais três vezes. Não esqueça de alterar o nome dos componentes conforme mencionado no item 3. O próximo componente panel terá o nome de pnlItem3 e o componente panel que está dentro deste outro panel terá o nome de pnlItem3Title e assim por diante.

Agora nós vamos criar e adicionar as ações ao componente ActionBar.

1. Dê um duplo clique sobre o componente ActionManager.
2. Na aba Action clique na seta ao lado do botão New Action. Clique sobre o item New Standard Action.
3. Clique sobre todas as ações do item File, como mostrado na figura 1, e clique em OK.

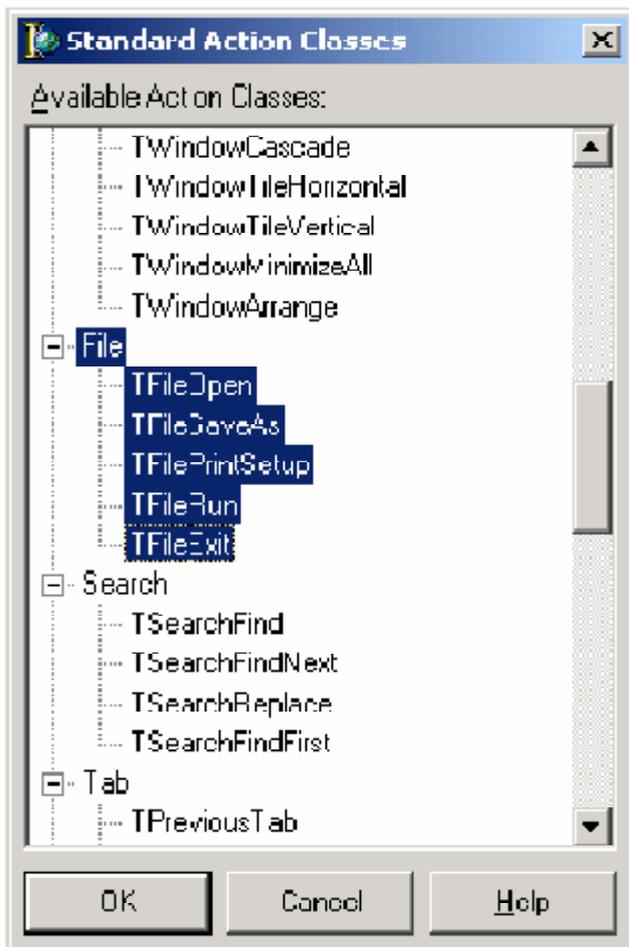


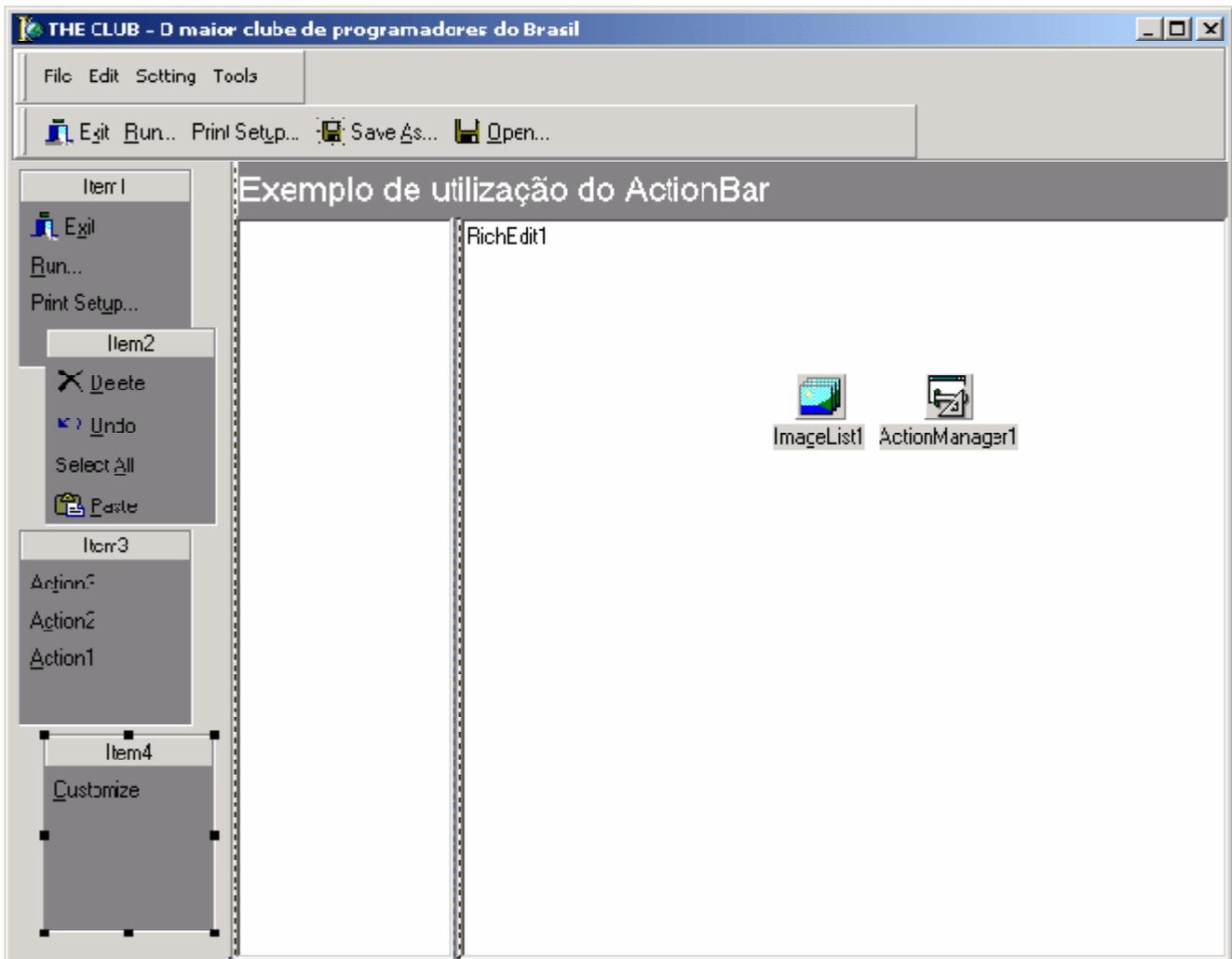
Figura 1: Selecionando ações.

1. Repita os passos 2 e 3 para os itens Edit e Tools.
2. Clique no item (No Category) e clique no botão New Action três vezes.
3. Pressione o botão CTRL e clique sobre os três itens para que eles fiquem selecionados.
4. Vá até o Object Inspector e altere a propriedade Category para Setting.
5. Clique sobre o item File e arraste-o para o componente ActionMainMenuBar que está no topo do seu form.
6. Repita esta última operação para os outros itens.
7. Agora nós vamos adicionar os itens para os cinco componentes ActionToolBar. Você verá uma tela semelhante a mostrada na figura 2



Figura 2: Criando itens

8. Clique sobre o item File e arraste-o para o componente ActionToolBar que está no topo do seu form logo abaixo do componente ActionMainMenuBar
9. Clique sobre o item File e arraste-o para o componente ActionToolBar que está no pnlItem1 que está no lado esquerdo do seu form.
10. Agora arraste cada item para cada ActionToolBar que está no lado esquerdo do seu form.
11. Vá até o Object Treeview e selecione o item ActionManager1.
Ligado a este item você poderá ver o ActionBars e ligado ao ActionBars você poderá ver o item 0 - ActionBar -> ActionMainMenuBar1.
Selecione este item e vá até o Object Inspector e altere a propriedade AutoSize para True.
12. Volte ao Object Treeview e selecione o item 1 - ActionBar -> ActionToolBar1 e vá até o Object Inspector e altere a propriedade AutoSize para True.
13. Volte ao Object Treeview e selecione o item 2 - ActionBar -> ActionToolBar2 e vá até o Object Inspector e altere a propriedade AutoSize para False.
14. Repita esta última operação para os itens 3,4,5.
Feito isto você terá uma tela semelhante a tela a seguir:



Agora nós vamos trabalhar com a programação.

1. Defina uma variável global com o nome TamanhoLista do tipo TObjectList.
2. Declare a unit Contrns.pas na clausula uses.
3. Inclua o código abaixo no evento OnCreate do form:

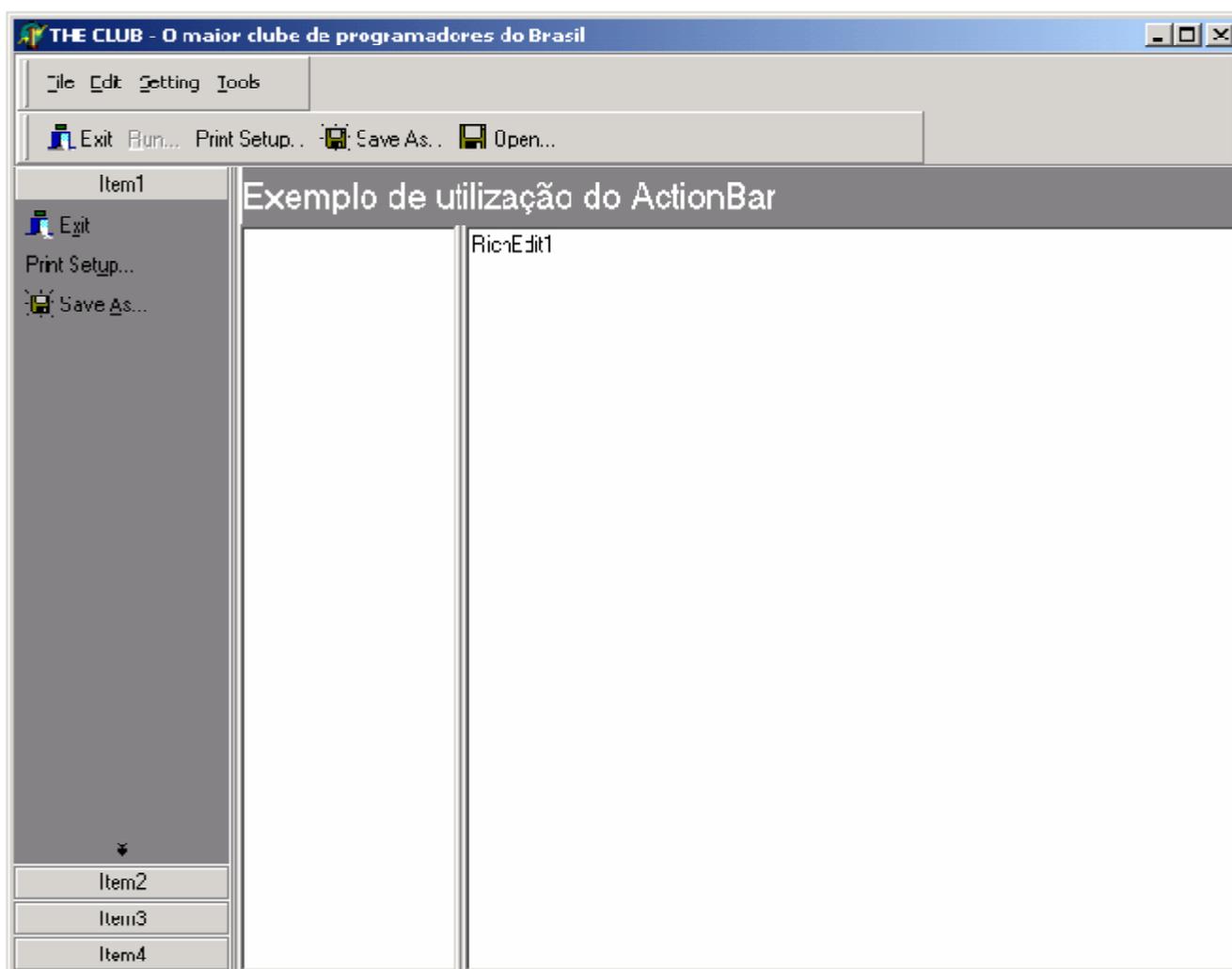
```
procedure TForm1.FormCreate(Sender: TObject);
begin
  TamanhoLista := TObjectList.Create;
  TamanhoLista.OwnsObjects := False;
  TamanhoLista.Add(pnlItem1);
  TamanhoLista.Add(pnlItem2);
  TamanhoLista.Add(pnlItem3);
```

```
TamanhoLista.Add(pnlItem4);
pnlItem1Title.OnClick(pnlItem1Title);
end;
```

4. Vá até o Object Treeview e selecione os itens pnlItem1Title, pnlItem2Title, pnlItem3Title, pnlItem4Title. Vá até o Object Inspector e dê um duplo clique sobre o evento OnClick. Fazendo isto você terá acesso ao editor de código. Então inclua o código abaixo neste evento.

5. Volte ao Object Treeview e selecione o Form1. Vá até o Object Inspector e inclua o código abaixo no evento OnDestroy.

Agora você já pode compilar o seu projeto e ver o resultado como mostrado na figura a seguir:



Exemplo da tela do projeto.

Com a aplicação rodando clique em Tools|Customize. Isto lhe mostrará a tela de customização.

Para adicionar um novo item no menu ou um botão barra de ferramentas, basta você selecionar um item nesta tela e arrastá-lo até ao menu ou a barra de ferramentas.

Quando você fechar o projeto, o sistema automaticamente gravará as configurações e quando você chamar o projeto novamente ele estará da mesma forma que você o configurou.

Conclusão.

Este é um exemplo simples da utilização do ActionBars disponível no Delphi 6.

Como você pôde ver ele é muito fácil e prático, permitindo a você criar aplicações com um visual muito bonito com poucas linhas de programação.

Você pode fazer o download deste arquivo no endereço www.theclub.com.br/revista/actionbar0502.zip

Caso você tenha algum assunto que gostaria de ver em nossa revista, envie um e-mail para info@theclub.com.br. A sua participação é muito importante para nós.

Como obter uma ROM a partir de seu Palm (incluindo via USB)

por: Ricardo Gruber Bernstein
Colaboração: Cláudio Akerman

Para montar um bom ambiente de programação/depuração na plataforma PalmOS, o desenvolvedor normalmente precisará das seguintes ferramentas de apoio:

1) O Emulador atualizado (obter em <http://www.palmos.com/dev/tools/emulator/>).

A versão atual (em 02/05/2002) é a 3.5. (Figura 1)

2) As Skins (capas) do Emulador atualizadas (obter também em <http://www.palmos.com/dev/tools/emulator/>). A versão atual (em 02/05/2002) é a 1.8. As Skins facilitam a visualização de um Palm real, a medida que determinam para o emulador o modelo de moldura desejada pelo desenvolvedor.

3) As ROMs do PalmOS.

O Emulador consegue simular os diversos modelos de Palm disponíveis no mercado, e cada um desses modelos tem sua ROM. Assim, o Emulador só funciona após a escolha de uma ROM pertencente ao modelo que se deseja emular.



Figura 1: Tela do emulador Palm

O grande problema é que a Palm não disponibiliza as ROMs em seu site, e tradicionalmente há todo um processo extremamente burocrático, realizado através de assinatura de contratos e envios pelo correio, de se obter as ROMs.

Recentemente, o ClubePalm (www.clubepalm.com.br) obteve autorização da Palm para disponibilizar em seu site as ROMs do PalmOS. Mesmo assim, eventualmente o programador pode desejar obter a ROM específica de seu dispositivo Palm real, por exemplo, uma ROM não disponível ainda para download. No trecho abaixo explicaremos como obter a ROM de seu dispositivo Palm, e mais, um macete para obter a ROM via porta USB para os modelos mais novos.

Como obter a ROM de seu Palm (via serial e USB)

Seguindo as instruções contidas no próprio emulador:

1) Primeiramente você precisa do software Palm Desktop instalado em seu computador.

Este programa vem junto com seu Palm e também pode ser baixado em www.palm.com.

2) Instale o programa "Rom Transfer.prc" (vem junto com o download do emulador) em seu dispositivo Palm através do Palm Install Tool e após o HotSync certifique-se que foi corretamente instalado.

3) Se o seu Palm tiver conexão USB, copie o arquivo **USBPort.dll** que está no diretório **C:\Palm** para o diretório em que está o executável do emulador.

4) Execute o emulador.

5) Clique com o botão direito do mouse no Emulador e escolha a opção Transfer-ROM.

6) Escolha a porta serial apropriada e a velocidade de transmissão no menu abaixo. As velocidades altas nem sempre funcionam, ou seja, se tiver problemas, escolha a velocidade mais baixa possível.

7) Posicione o dispositivo Palm no cradle e execute a aplicação "ROM Transfer" no Palm.

8) Selecione a mesma velocidade no menu "ROM Transfer".

9) Desabilite o HotSync ou qualquer aplicação que utilize a porta Serial ou USB escolhida.

10) Clique no botão "Begin Transfer" da aplicação "ROM Transfer" de seu dispositivo Palm.

11) Pressione o botão "Begin" na caixa de diálogo. O emulador aguardará a resposta do dispositivo Palm.

12) A transferência durará alguns minutos, aguarde. Se ocorrer algum erro durante a transferência, a aplicação não mais responder ou parecer "travada" espere alguns minutos para dar à aplicação uma chance de time-out.

13) Depois de transferida a ROM para o desktop, grave o arquivo com o nome desejado, por exemplo, "Palm_m130.rom". Agora basta criar uma nova sessão do emulador utilizando essa ROM.

Ricardo Gruber Bernstein

ClubePalm – A Referência do Desenvolvedor PalmOS.

Vejam os cursos de Desenvolvimento para Palm baseados na Ferramenta PocketStudio - Básico e Avançado - oferecidos pelo ClubePalm em www.clubepalm.com.br



Page Size – O Mistério

Introdução:

Neste artigo mostrarei o que é e para que serve a opção Page Size, presente em todos os SGBD “Sistema Gerenciadores de Banco de Dados” do mercado.

Page Size

Um SGBD é formado por várias páginas de dados, isto é, quando um SGBD procura por registros, ele não sai a procura de “registros*”, e sim, a procura de qual página de dados se encontra uma determinada linha. Isto serve para qualquer manipulação no Banco, isto é, um SELECT e ou comandos DML (Data Manipulation Language), os famosos: UPDATE, DELETE.

*Nota: É errado usarmos o conceito de registros em Banco de Dados. Para bancos de dados SGBD, devemos usar a palavra: Linha.

A procura das linhas no Banco de Dados, é feita através de cálculos matemáticos/álgebra relacional, e desempenha um papel fundamental nessa procura as Chaves Primárias (PK) , Chaves Estrangeiras (FK) e os Índices.

Tamanho Correto

O que me levou escrever este artigo, foi exatamente este tópico, a dúvida de qual o tamanho correto a usar.

Se existisse uma regra para o tamanho da página, com certeza a mesma seria fixa, e não é o caso. O tamanho de page size normalmente são múltiplos de 1024.

Mas, a regra é a seguinte para a criação correta do tamanho da página:

Para uma boa performance em consultas, deve-se usar o maior tamanho possível... Mas, com isso temos um problema,

quanto maior o Tamanho da Página, menor performance de atualização teremos, por que, estamos acumulando muitos registros numa mesma página.

Então, qual seria o tamanho correto?

Analise o seu sistema e veja qual a preferência que você deseja dar ao mesmo. Se for SELECT, páginas maiores, se for UPDATE páginas menores..... temos um problema. E se o SELECT e o UPDATE se equivalem para o meu sistema, dê preferência ao SELECT, ou então, utilize o mesmo tamanho de página que seu File System “FAT32, NTFS...” usa!

Conclusão:

A escolha correta de Page Size ajuda e muito na performance do seu sistema, mas, não podemos levar em conta que apenas o tamanho de página correto fará com que nossos sistemas voem. Na edição de Dezembro/2001 da The Clube Magazine foi publicado um artigo sobre Performance, Índice e Estatística que também lhe ajudará a saber como deixar os seus sistemas mais rápidos.

Sucesso a todos e até a próxima,

Anderson Haertel Rodrigues - AHR.

**Desenvolvedor/Analista de Sistemas.
Consultor especializado em desenvolvimento
de Sistemas Client/Server - MultiCamadas/
DataSnap.**

**Oferece Treinamentos para empresas em:
InterBase, SQL-Server, Delphi e COM+/MTS
Recuperação de Banco de Dados InterBase.**

anderson.hr@bol.com.br -

anderson.hr@zipmail.com.br.

Florianópolis - Santa Catarina - Brasil.

Controlando o Preview do QuickReport em Run-Time

Por Alessandro Ferreira
Suporte Técnico The Club, suporte@theclub.com.br

Neste pequeno artigo iremos mostrar como controlar o Preview do QuickReport via programação e em tempo de execução. Esta vem sendo uma solicitação que há muito tempo os programadores Delphi que utilizam o QuickReport como sua principal ferramenta de desenvolvimento de relatórios têm nos solicitado.

O Preview nada mais é que um Form!

Isso mesmo, o Preview do QuickReport nada mais é que um Form igualzinho aos demais Forms de sua aplicação e como os demais, também pode ser controlado por você, porém com um pouquinho mais de trabalho. A idéia é bem simples, primeiro temos que conseguir o Handle do Form de Preview e atribuí-lo a uma variável TForm e a partir da mesma começar os controles. Veja o código abaixo:

```
Var
  PrevFrm: TForm;
begin
  { Pega o Form do Preview Ativo }
  PrevFrm := nil;
  for i := 0 to Screen.FormCount-1 do
    with Screen.Forms[i] do
      if ClassName = 'TQRStandardPreview' then
        PrevFrm := Screen.Forms[i];

  { Se não encontrou o Form, abandona }
  if PrevFrm = nil then
    Exit;
end;
```

Bem, a partir daqui temos um objeto TForm com a referência do Form do Preview do QuickReport. Um detalhe importante, é que o Preview do QuickReport é "Modal" e com isso não temos

acesso para executar o código acima antes de fechá-lo, porém, se fecharmos o Preview o código acima perde a razão de existir. O que fazer? Felizmente temos um componente para salvar nosso código e conseguir executá-lo durante a execução do Preview do QuickReport. Para quem não conhece, este componente é o TTimer que fica na palheta System de seu Delphi, em resumo todo o código deverá ficar no evento onTimer do componente TTimer.

Como descobrir os componentes no Form de Preview?

Isso será bem fácil, bastará fazer um Looping "varrendo" os componentes existentes em nosso "Form" e verificar qual nos interessa a partir disso alterar suas propriedades.

Primeiramente vamos relacionar o nome dos componentes existentes em um Preview padrão do QuickReport:

Nome do Objeto	Classe	Descrição
ToolBar1	TToolBar	Barra de ferramentas no Preview
Print	TToolButton	Botão imprimir
ZoomFit	TToolButton	Botão "Zoom To Fit"
Zoom100	TToolButton	Botão "Zoom 100%"
ZoomToWidth	TToolButton	Botão "Zoom To Width"
FirstPage	TToolButton	Botão página inicial

Continua...

Nome do Objeto	Classe	Descrição
PreviousPage	TToolButton	Botão página anterior
ToolButton2	TToolButton	Botão próxima página
LastPage	TToolButton	Botão página final
PrintSetup	TToolButton	Botão Setup de impressão
SaveReport	TToolButton	Botão Salvar relatório
LoadReport	TToolButton	Botão Ler relatório
ExitButton	TSpeedButton	Botão Fechar Preview

Agora que temos os nomes de todos os objetos que nos interessam no Preview do QuickReport, vamos por a mão na massa e criar os controles, veja o código abaixo, onde podemos desabilitar a Barra totalmente e com isso todos os componentes que estão sobre a mesma ou ainda escolher qual componente iremos desabilitar:



Um detalhe importante para o exemplo abaixo funcionar, é que nomeamos cada CheckBox com o nome de seu objeto correspondente, por exemplo, o CheckBox "Mostrar botão FirstPage" possui o nome "FirstPage", veja o código completo do evento onTimer do componente QuickTimer:

```

procedure TfmPersonal.QuickTimerTimer(Sender:
TObject);
var
  tBar, tBut, tPrev: TComponent;
  j, k, i: integer;
  PrevFrm: TForm;
begin
  { Pega o Form do Preview Ativo }
  PrevFrm := nil;
  for i := 0 to Screen.FormCount-1 do
    with Screen.Forms[i] do
      if ClassName = 'TQRStandardPreview' then
        PrevFrm := Screen.Forms[i];
  { Se não encontrou o Form, abandona }
  if PrevFrm = nil then
    Exit;
  { Desabilita componentes }
  for j := 0 to PrevFrm.ControlCount-1 do
    begin
      tBar := PrevFrm.Controls[j];
      if tBar is TToolBar then
        begin
          { desabilita a barra inteira }
          if not Check_Barra.Checked then
            begin
              TToolBar(tBar).Visible := False;
              { Cancela o For, pois a barra toda foi
              desabilitada }

              Break;
            end;
          { Percorre os componentes da barra e verifica o
          checkbox correspondente para habilitar/desabilitar os
          botões. }
          for k := 0 to TToolBar(tBar).ControlCount-1 do
            begin
              tBut := TToolBar(tBar).Controls[k];
              if tBut is TToolbutton then
                if FindComponent(TToolButton(tBut).Name)
                <> nil then
                  if FindComponent(TToolButton(tBut).Name)
                  is TCheckBox then
                    TToolButton(tBut).Enabled :=
                    TCheckBox(FindComponent
                    (TToolButton(tBut).Name)).Checked;

```

```

end;
end;
end;
{ Botão Close do Preview }
tBar := PrevFrm.FindComponent('ExitButton');
if tBar <> nil then
  if FindComponent(TSpeedButton(tBar).Name) is
    TCheckBox then
    TSpeedButton(tBar).Enabled :=
      TCheckBox(FindComponent(TSpeedButton
        (tBar).Name)).Checked;
end;

// Código do Botão Preview;
procedure TfmPersonal.Button1Click(Sender: TObject);
begin
  try
    { Habilita Timer para configuração da barra }
    QuickTimer.Enabled := TRUE;
    { Chama o Preview }
    fmQuickRep.QuickRep1.PreviewModal;
  finally
    { Desliga o Timer }
    QuickTimer.Enabled := FALSE;
  end;
end;
end;

```

A primeira vista o código acima pode parecer um pouco complicado, mas na realidade é bem simples. Quando percorremos os componentes através de um Looping, conseguimos obter o nome do mesmo e apenas fazemos uma verificação se existe algum CheckBox correspondente ao componente atual no Looping e se existe, habilitamos ou desabilitamos de acordo com o estado do CheckBox.

Porém, caso você necessite desabilitar apenas um, dois ou três determinados componentes, podemos simplificar o código, indo diretamente aos componentes que nos interessam, veja o código abaixo:

```

var
  tBar: TComponent;
  PrevFrm: TForm;
begin
  { Pega o Form do Preview Ativo }
  PrevFrm := nil;
  for i := 0 to Screen.FormCount-1 do
    with Screen.Forms[i] do
      if ClassName = 'TQRStandardPreview' then
        PrevFrm := Screen.Forms[i];

```

```

{ Se não encontrou o Form, abandona }
if PrevFrm = nil then
  Exit;
{ Botão Close do Preview }
tBar := PrevFrm.FindComponent('ExitButton');
if tBar <> nil then
  TSpeedButton(tBar).Enabled := False;
{ Botão Print do Preview }
tBar := PrevFrm.FindComponent('Print');
if tBar <> nil then
  TToolButton(tBar).Enabled := False;
{ Botão PrintSetup do Preview }
tBar := PrevFrm.FindComponent('PrintSetup');
if tBar <> nil then
  TToolButton(tBar).Enabled := False;
end;

```

Lembrando sempre que essas instruções deverão ser codificadas no evento onTimer de um componente TTimer para funcionarem, caso contrário não terão nenhum efeito sobre o Preview.

Essa é a única forma de controlar o Preview?

Não! Esta é a única forma de controlar o Preview “nativo” do QuickReport, porém junto ao QR poderá encontrar um componente chamado QRPreview que lhe permitirá criar um Preview totalmente personalizado, contendo apenas os componentes que você determinar e com o formato que você desejar, porém todos os controles deverão ser implementados pelo programador e consegue-se um ótimo resultado com o mesmo, deixando-o bem com “a cara” de sua aplicação.

Não vamos nos estender no assunto pois não é exatamente o foco deste artigo, mas caso queira se aventurar neste componente, bastará nos solicitar um projeto de exemplo.

O projeto de exemplo referente a este artigo está disponível para download em:

www.theclub.com.br/revista/QRPersonaliza.Zip.

Conclusão.

Mais uma vez utilizamos a RTTI (Run-Time Type Information) para nos auxiliar em tarefas que a princípio poderiam parecer bem difíceis de serem executadas, mas felizmente o Delphi nos oferece inúmeros recursos à explorar e facilitar o nosso dia-a-dia.

Até a próxima.

Tratamento do Repique e 'gorjeta' após e antes da emissão do Cupom Fiscal

Por Claudenir C. Andrade

Este assunto é polêmico em muitos estados, pois o repique ou a devolução dos valores após o fechamento do cupom é considerado valor de compra para alguns estados e por isso deve ser tributado.

O repique é aplicado na seguinte situação:

Vamos imaginar que a compra de um cliente foi o total de R\$ 47,00 e muito generosamente ele indica para o caixa que os R\$ 3,00 de troco podem ficar de gorjeta. Isso ainda não é o repique!! A diferença entre o repique e a gorjeta é que a gorjeta é informada antes de fechar o cupom fiscal; o repique é informado após o fechamento do cupom fiscal.

Vamos considerar os dois casos:

Caso 1: Valor da Compra de R\$ 47,00 – O Cliente pagou R\$ 50,00 e indicou que o troco será a gorjeta. O que deve fazer o software? Deve fazer o seguinte:

```
Bematech_FI_AbreCupom(Pchar('038.64.7265-17'));
 Bematech_FI_VendeItem('7896834200030','Jantar Menu 3'
 , '1800', 'I', '3', '2', '15,00', '$', '0');
 Bematech_FI_VendeItem('7896834200029','Cerveja'
 , '1800', 'I', '2', '2', '1,00', '$', '0');
 Bematech_FI_IniciaFechamentoCupom('A','$','3,00');
 Bematech_FI_EfetuaFormaPagamento('Dinheiro','50,00');
 Bematech_FI_TerminaFechamentoCupom(
 'Obrigado, volte Sempre!!');
```

No código apresentado, estamos efetuando a venda de um item cujo total é de R\$ 3,00, 3 refeições de R\$15,00 = mais 2 cervejas de 1,00 = R\$ 47,00. Em nosso exemplo o cliente esta

deixando R\$ 3,00 de gorjeta e informou isso antes da emissão do término do cupom fiscal. Indicamos que estes R\$ 3,00 foram contabilizados na Opção Acréscimo do Cupom Fiscal e por sua vez estes R\$ 3,00 serão tributados em forma de 'Rateio' nas alíquotas utilizadas no cupom, em nosso caso R\$ 18,00. Se tivéssemos utilizado mais de uma alíquota no cupom fiscal, este valor seria rateado entre estas duas alíquotas, parte em 18,00% e parte na outra alíquota utilizada.

Caso 2: Este segundo caso se aplica se em seu Estado for obrigada a tributação da gorjeta em alíquota de serviço ou em alguma outra alíquota definida pela Secretaria Estadual da Fazenda de seu estado.

```
Bematech_FI_AbreCupom('038.64.7265-17');
 Bematech_FI_VendeItem('7896834200030','Jantar Menu 3'
 , '1800', 'I', '3', '2', '15,00', '$', '0');
 Bematech_FI_VendeItem('7896834200029','Cerveja'
 , '1800', 'I', '2', '2', '1,00', '$', '0');
 Bematech_FI_VendeItem('7896834203029','Serviço'
 , '0500', 'I', '1', '2', '3,00', '$', '0');
 Bematech_FI_IniciaFechamentoCupom('A','$','0');
 Bematech_FI_EfetuaFormaPagamento('Dinheiro','50,00');
 Bematech_FI_TerminaFechamentoCupom(
 'Obrigado, volte Sempre!!');
```

Neste caso, vendemos o item 'Serviço' com o valor da gorjeta, permitindo assim uma tributação exclusiva neste item e possibilitando uma contabilidade mais transparente do tributo que está sendo oferecido ao governo. Como a gorjeta foi 'ofertada' para tributação na venda do item, não mais é necessária a

inclusão do acréscimo no cupom fiscal.

Caso 3: Neste caso se aplica o 'Repique'. O cliente não informa que serão deixados os R\$ 3,00 de troco e por sua vez o cupom fiscal já foi encerrado. Como proceder? Bem, vamos às linhas de código do cupom fiscal:

```
Bematech_FI_AbreCupom('038.64.7265-17');
 Bematech_FI_VendeItem('7896834200030','Jantar Menu 3'
    , '1800', 'I', '3', '2', '15,00', '$', '0');
 Bematech_FI_VendeItem('7896834200029','Cerveja'
    , '1800', 'I', '2', '2', '1,00', '$', '0');
 Bematech_FI_IniciaFechamentoCupom('A','$','0');
 Bematech_FI_EfetuaFormaPagamento('Dinheiro','50,00');
 Bematech_FI_TerminaFechamentoCupom(
    'Obrigado, volte Sempre!!!');
```

Neste caso a impressora vai indicar no cupom fiscal **TOTAL R\$ 47,00**, e o fechamento das formas de pagamento, antes da mensagem promocional – **TROCO R\$ 3,00**. Fechará o cupom normalmente e esse troco já foi debitado do Totalizador Dinheiro de sua impressora.

Existem duas maneiras de proceder neste caso:

1) Criar um Totalizador Não Fiscal, chamado 'Repique'. Vimos isso nos capítulos anteriores, mas é só após a redução Z ou no começo do dia chamar a função - Bematech_FI_NomeiaTotalizadorNaoSujeitoIcms (Índice, Nome) - Exemplo : **Bematech_FI_NomeiaTotalizadorNaoSujeitoIcms('09','Repique')**. Uma vez criado esse totalizador, você deve utilizá-lo após o fechamento do cupom como recebimento do valor de R\$ 3,00 deixado pelo cliente, pois ele já não pode ser colocado como acréscimo do cupom fiscal, uma vez que o cupom já foi fechado.

Nosso código continuaria então da seguinte maneira:

```
Bematech_FI_RecebimentoNaoFiscal
    ('09', '3,00', 'Dinheiro');
```

As vantagens de tratar assim o repique são várias. Uma delas é a transparência na contabilidade, pois você terá um totalizador exclusivo para este tipo de recebimento, conseguindo controlar todo o recebimento em seu caixa, incluindo as gorjetas e os repiques que não são os mais fáceis de contabilizar e no final do mês seu negócio estará perdendo uma quantia considerável de arrecadação.

Outra vantagem é que seu totalizador dinheiro não sofreu alteração, uma vez que teve a saída de R\$ 3,00 no cupom como

TROCO que foi impresso e a entrada desses R\$ 3,00 por meio do totalizador Dinheiro, refletindo assim um espelho ou a realidade do fluxo de sua gaveta ou caixa.

2) Caso você já tenha fechado o cupom, conforme vimos anteriormente, e o cliente tenha deixado R\$ 3,00 como gorjeta, que para nós é repique, e você necessita oferecer isso à tributação, então a coisa complicou.

Mas dentro desta situação existem outras duas maneiras de se trabalhar; na verdade, três maneiras. São elas:

2.1) Você efetua este mesmo procedimento e por sua vez no final do dia efetua uma sangria de caixa, para dar a saída de dinheiro, e emite um cupom fiscal com o total do valor 'sangrado'.

Vamos imaginar que em todo o dia você teve de repique o valor de R\$ 35,00 (Lembrando sempre que o repique é apenas quando o cupom já foi fechado e o cliente está deixando a gorjeta), então você sangra a impressora fiscal e emite um cupom oferecendo este valor à tributação respectiva, assim:

```
Bematech_FI_Sangria ('35,00');
 Bematech_FI_AbreCupom(' ');
 Bematech_FI_VendeItem('Sangria1','Repique' , '0500',
    'I', '1', '2', '35,00', '$', '0');
 Bematech_FI_IniciaFechamentoCupom('A','$','0');
 Bematech_FI_EfetuaFormaPagamento('Dinheiro','35,00');
 Bematech_FI_TerminaFechamentoCupom(
    'Obrigado, volte Sempre!!!');
```

2.2) A segunda possibilidade é após a emissão do cupom fiscal você emitir um outro cupom com o valor do repique, com isso você estará oferecendo à tributação imediatamente o valor do repique, em sua respectiva alíquota.

2.3) A outra e não muito agradável possibilidade é você cancelar o cupom emitido, com a função **Bematech_FI_CancelaCupom(void)**, e reimprimir o cupom fiscal, incluindo a gorjeta que agora não é mais repique pois você está reemitindo o cupom fiscal para adicioná-lo como gorjeta, podendo colocar o valor da gorjeta como acréscimo ou em sua respectiva alíquota na venda do item.

Claudenir é Gerente de Desenvolvimento de Software na Bematech e autor do Livro "Automação comercial com C#, VB.NET e Delphi 6", lançado pela Editora Érica www.ERICA.com.br. Pode ser contatado pelo e-mail: claudenir@bematech.com.br

propaganda da mas



Perguntas Respostas

Pergunta: Como fazer para verificar o tamanho de um campo numérico em um arquivo DBF e poder modificar o tamanho do mesmo?

Resposta: Isso é possível de ser feito através de APIs do BDE, veja abaixo um exemplo para retornar as informações de um campo:

```
uses BDE;

// pegar o valor de uma campo numérico com decimais no
DBase.
procedure GetFieldAttrib(Field: TField; var Len, Dec:
Integer);
var
  CursorProps: CurProps;
  FieldDescs: PFieldDescList;
  FieldDescCount, ItrDsc: Integer;
begin
  Len := 0;
  Dec := 0;
  DbiGetCursorProps((Field.DataSet as TTable).
  Handle, CursorProps);
  FieldDescCount := CursorProps.iFields;
  FieldDescs := AllocMem
    (FieldDescCount * SizeOf(FLDDesc));
  try
    Check(DbiGetFieldDescs((Field.DataSet as TTable)
    .Handle, PFldDesc(FieldDescs)));
    for ItrDsc := 0 to FieldDescCount - 1 do
```

```
    if StrIComp(FieldDescs^[ItrDsc].szName,
      PChar(Field.FieldName)) = 0 then
    begin
      Len := FieldDescs^[ItrDsc].iUnits1;
      Dec := FieldDescs^[ItrDsc].iUnits2;
      Break;
    end;
  finally
    FreeMem(FieldDescs, FieldDescCount *
      SizeOf(FLDDesc));
  end;
end;

// Verifica
var
  Len, Dec: Integer;
begin
  // pega
  GetFieldAttrib(tabTeste.FieldByName('Valor'), Len, Dec);
  // mostra no Caption.
  Caption := 'Tamanho: ' + IntToStr(Len) +
    ' Decimais: ' + IntToStr(Dec);
end;
```

Para modificar as definições de um campo, voce pode consultar nossa revista de Janeiro/1999 e Agosto/1999 na sessão Dicas & Truques onde publicamos rotinas para manutenção em tabelas dBase e Paradox.

Dúvida enviada por Assinfor Inf. Ltda., Três Coroas/RS



PERGUNTAS & RESPOSTAS

Pergunta: Gostaria de tirar uma dúvida em relação ao componente SQLStoredProc.

Quando tento gravar no Interbase um valor Decimal ou Numeric e está apresentando o erro "Stack OverFlow", gostaria de saber se tenho que usar outro componente ou outro tipo de dado?

Resposta: Esse erro ocorre na seguinte situação:

Quando criamos um parâmetro na StoredProcedure do tipo Numeric(18,2) e depois selecionamos essa StoredProcedure no componente SQLStoredProc o tipo desse parâmetro para o Delphi fica como BCD, no momento em que passamos o valor para o parâmetro utilizando o AsFloat ele gera o erro, veja abaixo:

```
procedure TForm1.Button1Click(Sender: TObject);
begin
    SQLStoredProc1.ParamByName('NVALOR').AsFloat :=
    CurrencyEdit1.Value;
    SQLStoredProc1.ExecProc;
end;
```

Neste caso a solução para o problema e colocá-lo como AsBCD conforme exemplo abaixo:

```
procedure TForm1.Button1Click(Sender: TObject);
begin
    SQLStoredProc1.ParamByName('NVALOR').AsBCD :=
    CurrencyEdit1.Value;
    SQLStoredProc1.ExecProc;
end;
```

{ Dica The Club }

Se utilizar o parâmetro como Double Precision dentro da StoredProcedure, o Delphi irá reconhecer o tipo do parâmetro no componente SQLStoredProc como Float, neste caso poderá utilizar o ParamByName com .AsFloat sem gerar problemas.

Dúvida enviada por Alessandro Martins Silva, São Paulo/SP.

Pergunta: Tenho um "Set" de imagens em um TImageList. Tenho como acessar uma determinada imagem desse TImageList e jogar para um Bitmap de um botão por exemplo?

Resposta: O TImageList possui um método chamado GetBitmap, veja abaixo:

```
procedure TForm1.BitBtn1Click(Sender: TObject);
begin
    ImageList.GetBitmap(0, seu_botao_1.Glyph);
```

```
ImageList.GetBitmap(1, seu_botao_2.Glyph);
end;
```

Dúvida enviada por App Sistemas Ltda., São José do Rio Preto/SP.

Pergunta: Como eu faço para gerar um "TGUID" no Delphi. Existe alguma função ou API que me retorne este "número"?

Resposta: Sim, veja abaixo um simples exemplo:

Veja abaixo simples exemplo:

```
implementation
uses ActiveX;
{$R *.DFM}

procedure TForm1.Button2Click(Sender: TObject);
var
    G: TGUID;
begin
    CoCreateGuid(G);
end;
```

Dúvida enviada por Marcus Paulo Carneiro Araújo, Brasília/DF.

Pergunta: Posso um formulário de uso genérico. Neste formulário, existe um componente TQuery, que pode estar conectado a qualquer banco ou tabela Desktop.

Como eu posso fazer para identificar, em tempo de execução, qual o banco, que este objeto esta conectado?

Resposta: Uma forma para isso é obter o DRIVER ao qual o Alias esta "amarrado", veja abaixo um simples exemplo:

```
procedure TForm1.Button1Click(Sender: TObject);
begin
    Caption := Session.GetAliasDriverName
    (Table1.DataBaseName);

end;
// no caso de Paradox, irá retornar STANDARD.
```

Dúvida enviada por Mauro Mâncio da Silva, São Paulo/SP.