

[BETA] - GExperts

GExperts é uma ferramenta de auxílio ao desenvolvedor, voltada a atalhos que facilitam a localização de componentes e funções, além de outras utilidades.

- [Instalação e configurações](#)
- [Atalhos](#)

Instalação e configurações

GExperts é uma ferramenta de auxílio ao desenvolvedor, voltada a atalhos que facilitam a localização de componentes e funções, além de outras utilidades.

Instalação

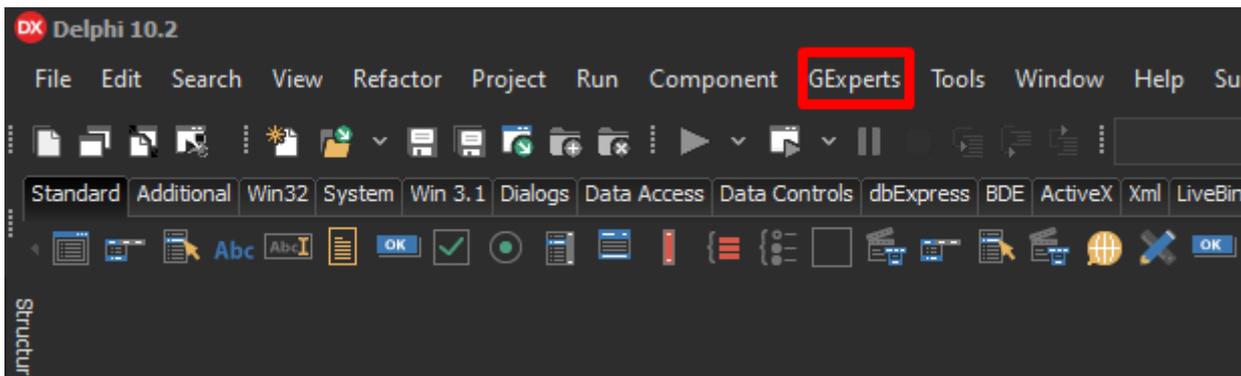
A versão utilizada, será a do Delphi 10.2 Tokyo.

Antes de começar a instalação deve-se fechar o Delphi.

Execute o [GExperts 10.2.exe](#).

A instalação é bem simples, sendo necessário apenas dar next, install e finish.

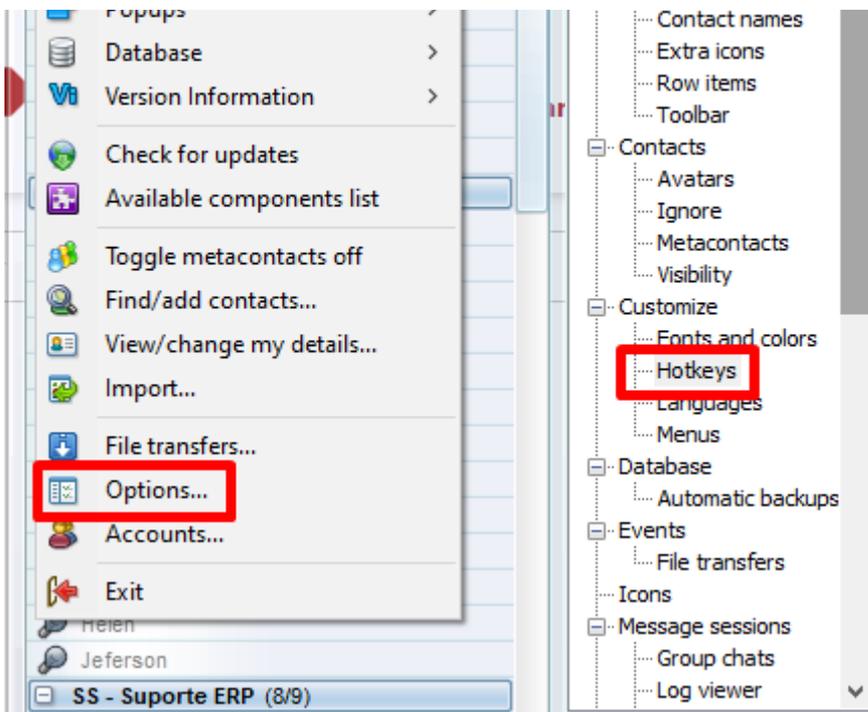
Feito isso, ao abrir o Delphi, o menu já estará disponível na IDE



Configurações

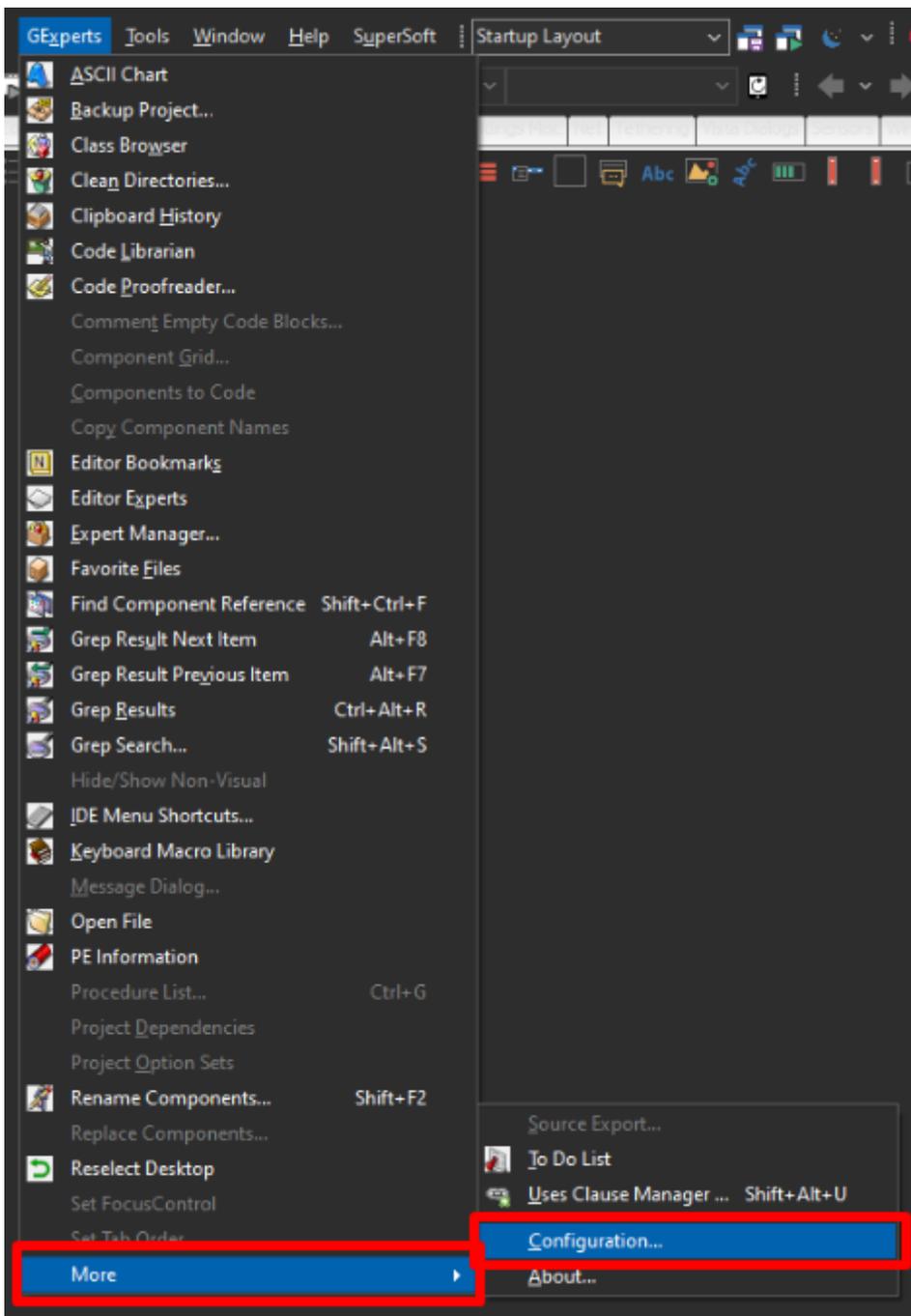
Atualmente, utilizamos o comunicador Miranda NG, que utiliza atalhos com teclas iguais ao GExperts. Como o comunicador tem prioridade, as teclas de atalho poderão acionar o Miranda, impedindo que alguns atalhos do GExperts sejam utilizados. Fica a critério do desenvolvedor decidir quais atalhos irá mudar.

No Miranda NG:



Caso não utilize nenhum dos atalhos, eles poderão ser desabilitados para não ocorrerem conflitos.

No Delphi - GExperts:



As teclas de atalho das funções que serão apresentadas na página [atalhos](#) poderão ser alteradas através desse menu de acordo com a necessidade do desenvolvedor.

Atalhos

Os atalhos que serão apresentados a seguir são padrões do GExperts. Caso necessário, eles poderão ser alterados a critério do desenvolvedor de acordo com seus respectivos nomes.

A ferramenta oferece vários atalhos ao desenvolvedor, facilitando a localização de componentes, métodos, palavras-chave, e mais algumas utilidades.

Dentre eles, os de mais destaque são:

Adição de uses (Ctrl + Shift + A) - Delphi

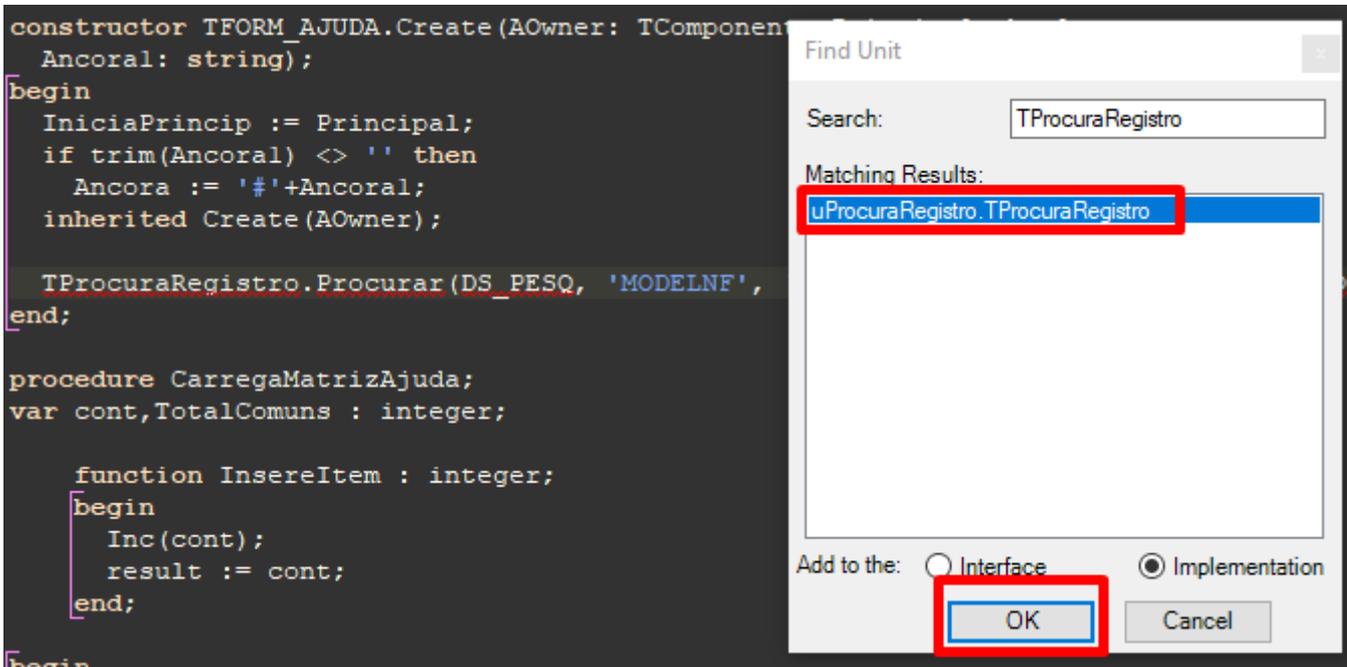
Esse é um atalho próprio do Delphi, mas tem grande valia e poderá ser utilizado ao adicionar um método do sistema a uma unit que não tenha sua *use* declarada. O atalho irá buscar e apresentar a *use* necessária para a utilização do mesmo.

Exemplo:

```
constructor TForm_AJUDA.Create(AOwner: TComponent; Principal: boolean;
  Ancoral: string);
begin
  IniciaPrincip := Principal;
  if trim(Ancoral) <> '' then
    Ancora := '#' + Ancoral;
  inherited Create(AOwner);

  TProcuraRegistro.Procurar(DS_PESQ, 'MODELNF', 'Modelo', ['55'], ['S'], C_ACAO_PROCURA) then
end;
```

Durante uma implementação, o método do sistema "TProcuraRegistro" foi adicionado à uma unit que ainda não tem informação sobre sua função, ficando sublinhado em vermelho. Para adicionar sua *use*, basta utilizar as teclas de atalho, selecionar a *unit* e confirmar.



Por padrão, as *uses* podem ser adicionadas em "Implementation". Caso ocorra algum erro, passá-la para "Interface".

Modo de edição sincronizada (Ctrl + Shift + J) - Delphi

Mais um atalho nativo do Delphi, será muito útil na hora de refatorar trechos de código, podendo mudar nomes de métodos, propriedades e variáveis dentro do código selecionado

Exemplo:

```
procedure TForm_PEDIDOS_EDIT.DBE_PEDIDOSNUMPEDCLIExit(Sender: TObject);
begin
  inherited;
  if (Trim(DM_PEDIDOS.QMANUT.FieldName('NumPedCli').AsString) <> '') and (not FGerandoReplicaPh
begin
  if ((ExisteProcesso('PHARMAPRESS')) or (ExisteProcesso('CANDELARIA'))) and (Modo = atEdita) t
    Exit;
  DM_PEDIDOS.QMANUT.FieldName('NumPedCli').AsString := Trim(DM_PEDIDOS.QMANUT.FieldName('Nu
  DM_PEDIDOS.QPESQ.sql.Clear;
  DM_PEDIDOS.QPESQ.sql.add('select count(*) as quantos from PEDIDOS where Abrev = ' + #39 + TG1
  DM_PEDIDOS.QPESQ.sql.add('and Modelo = ' + #39 + 'PE' + #39 + ' and NumeroCli = ' + #39 + DM_
  DM_PEDIDOS.QPESQ.sql.add('and NumPedCli = ' + #39 + DM_PEDIDOS.QMANUT.FieldName('NumPedCli'
  if Modo <> AtNovo then
    DM_PEDIDOS.QPESQ.sql.add('and NumPedido <> '+#39+DM_PEDIDOS.QMANUT.FieldName('NumPedido')
  DM_PEDIDOS.QPESQ.abrir;
  if (DM_PEDIDOS.QPESQ.FieldName('Quantos').AsInteger <> 0) and (MessageDialog.Show('Número d
    DBE_PEDIDOSNUMPEDCLI.enabled then
      DBE_PEDIDOSNUMPEDCLI.setfocus;
end;
end;
```

O trecho apresentado se encontra fora dos padrões de código, sendo necessário refatora-lo.

Selecione o trecho de código desejado, e utilize o atalho para que o código fique selecionado da seguinte forma:

```

procedure TForm_PEDIDOS_EDIT.DBE_PEDIDOSNUMPEDCLIExit(Sender: TObject);
begin
  inherited;
  if (Trim(DM_PEDIDOS.QMANUT.FieldByName('NumPedCli').AsString) <> '') and (not FGerandoReplicaPh
  begin
    if ((ExisteProcesso('PHARMAPRESS')) or (ExisteProcesso('CANDELARIA'))) and (Modo = atEdita) t
      Exit;
    DM_PEDIDOS.QMANUT.FieldByName('NumPedCli').AsString := Trim(DM_PEDIDOS.QMANUT.FieldByName('Nu
    DM_PEDIDOS.QPESQ.sql.Clear;
    DM_PEDIDOS.QPESQ.sql.add('select count(*) as quantos from PEDIDOS where Abrev = ' + #39 + TGL
    DM_PEDIDOS.QPESQ.sql.add('and Modelo = ' + #39 + 'PE' + #39 + ' and NumeroCli = ' + #39 + DM
    DM_PEDIDOS.QPESQ.sql.add('and NumPedCli = ' + #39 + DM_PEDIDOS.QMANUT.FieldByName('NumPedCli'
    if Modo <> AtNovo then
      DM_PEDIDOS.QPESQ.sql.add('and NumPedido <> '+#39+DM_PEDIDOS.QMANUT.FieldByName('NumPedido')
    DM_PEDIDOS.QPESQ.abrir;
    if (DM_PEDIDOS.QPESQ.FieldByName('Quantos').AsInteger <> 0) and (MessageDialog.Show('Número d
      DBE_PEDIDOSNUMPEDCLI.enabled then
        DBE_PEDIDOSNUMPEDCLI.setfocus;
    end;
  end;
end;

```

Após isso, basta reescrever a palavra desejada, e as demais palavras em comum dentro do trecho selecionado serão alteradas.

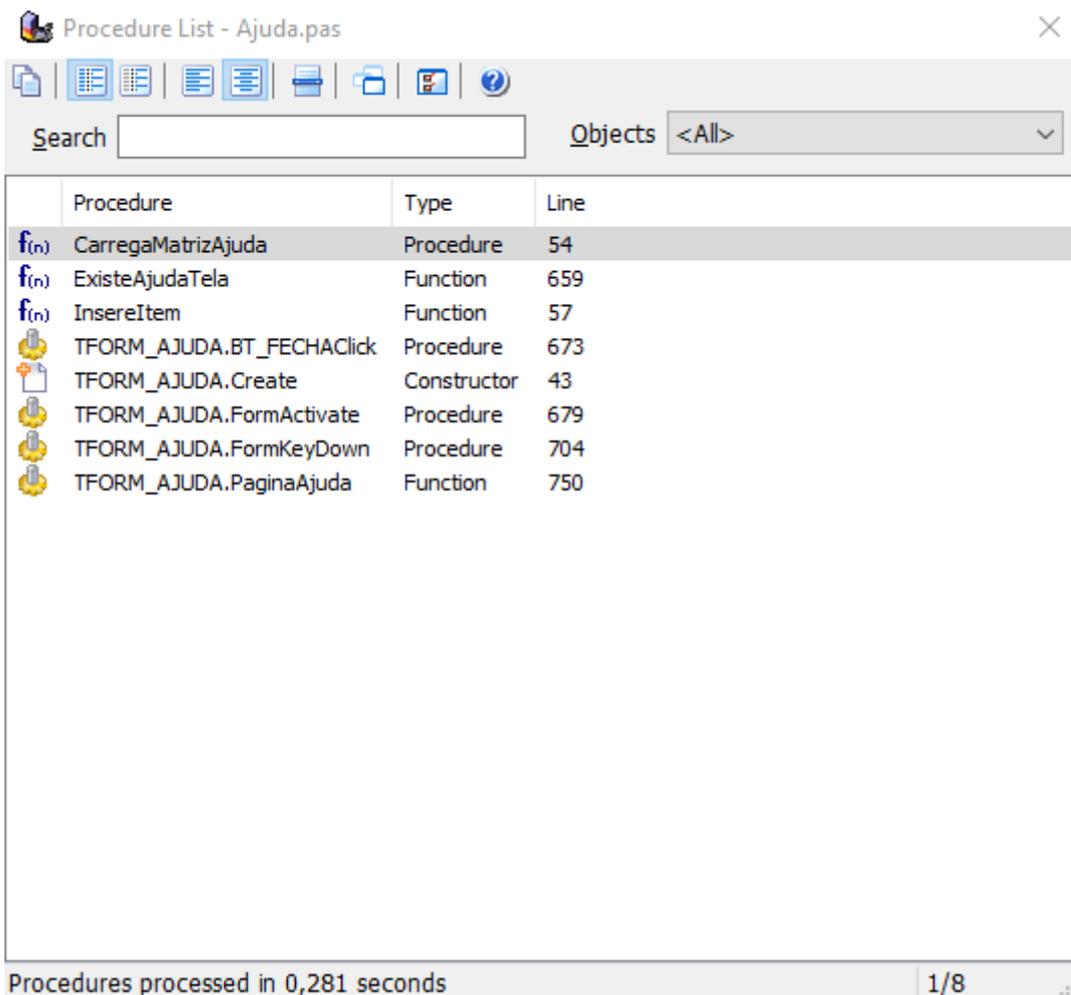
```

procedure TForm_PEDIDOS_EDIT.DBE_PEDIDOSNUMPEDCLIExit(Sender: TObject);
begin
  inherited;
  if (Trim(DM_PEDIDOS.QMANUT.FieldByName('NumPedCli').AsString) <> '') and (not FGerandoReplicaPh
  begin
    if ((ExisteProcesso('PHARMAPRESS')) or (ExisteProcesso('CANDELARIA'))) and (Modo = atEdita) t
      Exit;
    DM_PEDIDOS.QMANUT.FieldByName('NumPedCli').AsString := Trim(DM_PEDIDOS.QMANUT.FieldByName('Nu
    DM_PEDIDOS.QPESQ.SQL.Clear;
    DM_PEDIDOS.QPESQ.SQL.add('select count(*) as quantos from PEDIDOS where Abrev = ' + #39 + TGL
    DM_PEDIDOS.QPESQ.SQL.add('and Modelo = ' + #39 + 'PE' + #39 + ' and NumeroCli = ' + #39 + DM
    DM_PEDIDOS.QPESQ.SQL.add('and NumPedCli = ' + #39 + DM_PEDIDOS.QMANUT.FieldByName('NumPedCli'
    if Modo <> AtNovo then
      DM_PEDIDOS.QPESQ.SQL.add('and NumPedido <> '+#39+DM_PEDIDOS.QMANUT.FieldByName('NumPedido')
    DM_PEDIDOS.QPESQ.abrir;
    if (DM_PEDIDOS.QPESQ.FieldByName('Quantos').AsInteger <> 0) and (MessageDialog.Show('Número d
      DBE_PEDIDOSNUMPEDCLI.enabled then
        DBE_PEDIDOSNUMPEDCLI.setfocus;
    end;
  end;
end;

```

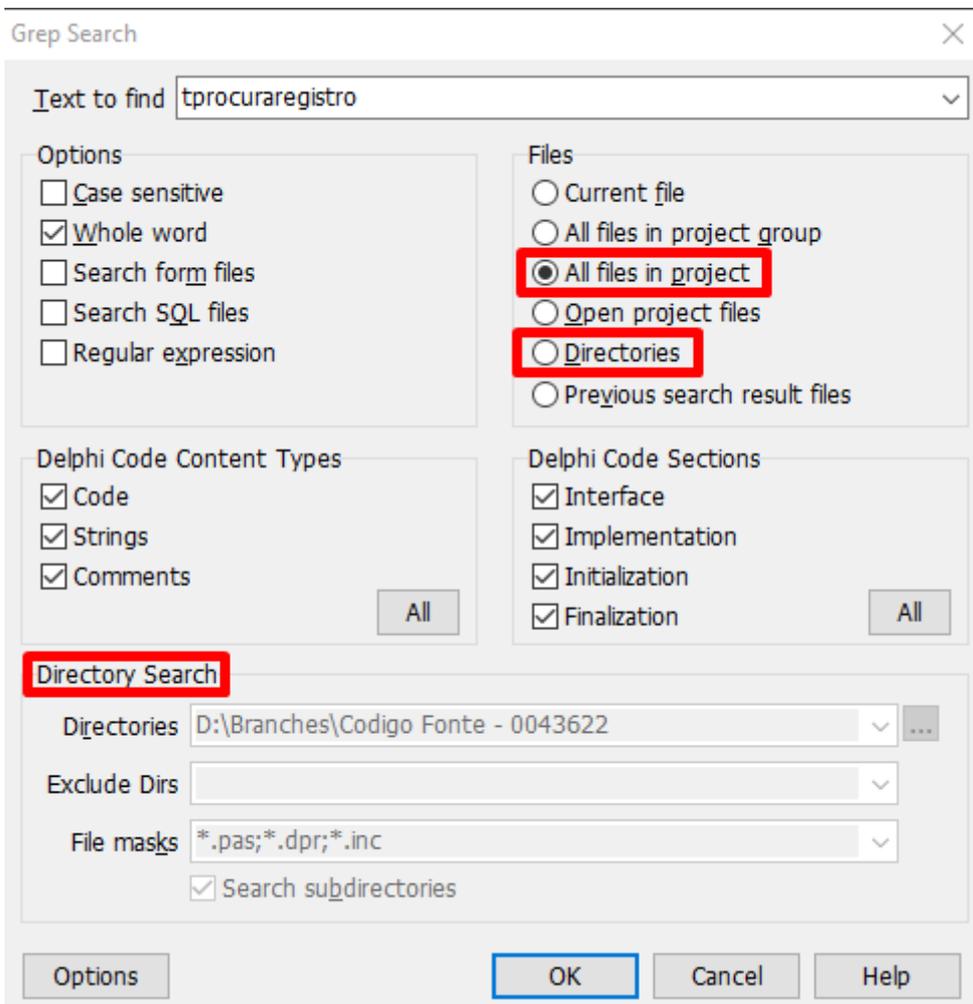
Procedure List (Ctrl + G)

Esse atalho é utilizado para visualizar todos os métodos vigentes na *unit*, dando mais agilidade na hora de encontrar o método desejado.



Grep Search (Shift + Alt + S)

Esse atalho faz uma busca em determinado local, procurando pela palavra-chave digitada, sendo muito útil para pesquisas.



Em "Files", o alcance da pesquisa pode ser alterado. Na maior parte das vezes, a opção "All files in project" será suficiente.

Caso queira fazer uma pesquisa mais centrada, utilizar a opção "Directories". Em "Directory Search", selecionar o caminho da pasta que será utilizada na pesquisa.

Find Component Reference (Ctrl + Shift + F)

Esse atalho pesquisará no formulário da *unit* o componente selecionado e, caso encontre, mostrará sua localização.

Exemplo:

Localizando o componente "DBE_CLIENTESCODREG" no formulário de Clientes

```
TS_IDENTIFICACAO: TTabSheet;
GB_INFORMACOESGERAIS: TGroupBox;
Label1: TLabel;
LB_DATACAD: TLabel;
DBE_NOMFANT: TDBEdit;
GB_CONTABIL: TGroupBox;
LB_CONTACLI: TLabel;
ME_CONTACLI: TMacDbEditDlg;
DBCD_CLIENTESDTCAD: TMacDbEditDlg;
DBR_CLIENTESSITUACAO: TDBRadioGroup;
DBR_CLIENTESCLASSIF: TDBRadioGroup;
TS_LOCALIZACAO: TTabSheet;
GroupBox4: TGroupBox;
Label4: TLabel;
Label5: TLabel;
Label6: TLabel;
Label7: TLabel;
Label8: TLabel;
Label23: TLabel;
Label10: TLabel;
Label11: TLabel;
Label12: TLabel;
LB_EMAIL: TLabel;
LB_HOMEPAGE: TLabel;
DBE_CLIENTESENDERECO: TDBEdit;
DBE_CLIENTESCOMPLEMENTO: TDBEdit;
DBE_CLIENTESCID: TDBEdit;
DBC_CLIENTESEST: TDBComboBox;
DBE_CLIENTESCODREG: TMacDbEditDlg;
DBE_CLIENTESCAPOS: TDBEdit;
DBE_CLIENTESTEL: TDBEdit;
DBE_CLIENTESFAX: TDBEdit;
DBE_CLIENTESEMAIL: TDBEdit;
DBE_CLIENTESHOMEPAGE: TDBEdit;
GB_TRANSP: TGroupBox;
Label43: TLabel;
Label44: TLabel;
Label15: TLabel;
Label22: TLabel;
Bevell: TBevel;
```

The image shows a screenshot of a software application window titled "Clientes". The window has a menu bar with options: "Identificação", "Contatos", "Telefones / Observações", "Localização", "Dados Complementares/Referências", "Cobrança/Entrega", "Padrões", and "Proc". The "Localização" tab is active. The form contains several sections:

- Localização:**
 - CEP: DBE_CLIENTESCEP
 - Endereço: DBE_CLIENTESENDERECO
 - Código da Cidade: DBE_CODCIDADE
 - Cidade: DBE_CLIENTESCID
 - Estado: [Dropdown]
 - Região: DBE_CLIEN [Dropdown] (highlighted with a red box)
 - Complemento: DBE_CLIENTESCOMPLEMENTO
 - Bairro: [Dropdown]
 - País: [Dropdown]
 - Caixa Postal: DBE_CLIENTESCXPO
 - Tempo de residência: DBE_TEMPOI [Dropdown]
 - Casa própria:
 - Telefone: DBE_CLIENTESTEL
 - Fax: DBE_CLIENTESFAX
 - Homepage: DBE_CLIENTESHOMEPAGE
 - E-mail: [Text Field] por e-mail.
 - E-mail 2: DBE_CLIENTESEMAIL2
- Transportadora/Redespacho:**
 - Código: DBE_CL [Dropdown]
 - Nome: [Text Field]
 - Código: DBE_CL [Dropdown]
 - Nome: [Text Field]

Ao utilizar o atalho, o formulário será aberto selecionando o componente marcado.

Identifier Reference (Ctrl + Alt + Setas cima/baixo)

Esse atalho serve para localizar o próximo local onde aparece a palavra selecionada. Útil para acompanhar atribuições de variáveis.

Observação:

Em algumas máquinas, esse atalho rotacionará a tela, sendo necessário alterá-lo através do menu "Configuration..." visto anteriormente.

Move to Matching Delimiter (Ctrl + Alt + Seta direita)

Encontrará o delimitador correspondente. Útil na hora de encontrar o *end* de um *begin*, o *finally* de um *try*, ou até mesmo para selecionar os parênteses do método externo.

Exemplo:

Localizando o *end* de um *begin*

```
FieldByName( Variavel ).AsInt := FieldByName( Variavel ).AsInteger);
9367 begin
//tratamento para inclusão de registros simultaneamente
if (TParametroDoSistema.Ler('UsaNotasSimultaneas', 'VD', 'PARAMVD2') = 'N') and (not ChecaTransSIS('NFMODEL1', TGlobalInfoSi
exit;
try
//TrocaNum := False;
if (Modo = AtNovo) and TProcuraRegistro.Procurar(DM_NFMODEL1.DS_PESQ, 'NFMODEL1', 'Abrev;NumLancto',
[TGlobalInfoSistema.AbrevEmp, FieldByName('NumLancto').AsString],
['S','S'], C_ACAO_PROCURA, 'NumLancto') then
begin
TrocaNum := True;
if (SistemaImpressao = 'P') then
FieldByName('NumNota').AsString := AtribuiNovoCodigo('NFMODEL1', 'NumNota', MTipoDoc, 6, ' ', ' ', ' ', UsaVariasSeries, 'S
if ExisteProcesso('SMS') then
begin
NumeroTemporario := TPreencheComBranco.Preencher(6, IntToStr(NumDocs.GetNext('NFMODEL1', TGlobalInfoSistema.AbrevEmp,
while TExisteRegistro.Verificar('NFMODEL1', 'Abrev;Modelo;NumLancto', [TGlobalInfoSistema.AbrevEmp, DM_NFMODEL1.QMANUI
NumeroTemporario := TPreencheComBranco.Preencher(6, IntToStr(NumDocs.GetNext('NFMODEL1', TGlobalInfoSistema.AbrevEm
DM_NFMODEL1.QMANUT.FieldByName('NumLancto').AsString := NumeroTemporario;
end
else
FieldByName('NumLancto').AsString := AtribuiNovoCodigo('NFMODEL1', 'NumLancto', ' ', 6);
end;
if (Self.ExisteProcesso(
['CENTRALDEVENDAS', 'UNIAODESERVICOS', 'ISTECH']) and
(SameText(MTipoDoc, 'NS')) then
begin
@SetNumRps :=
GetProcAddress(DLLCustom, PChar('SeExisteInsereNovoRps'));
if @SetNumRps <> nil then
SetNumRps(DM_NFMODEL1);
@SetNumRps := nil;
9390
9367: 7 | Insert | Code Design History
```

Posicionar o mouse no *begin* e utilizar o atalho

```
VD Cliente2 NFMMod12
TFORM_NFMODEL1_EDIT TFORM_NFMODEL1_EDIT.Gravar
Q_ITENSINF.BeforeInsert := Q_ITENSINFBeforeInsert;
Q_ITENSINF.BeforePost := Q_ITENSINFBeforePost;
DM_NFMODEL1.Q_ITENSINF.FieldByName('Lote').ReadOnly := False;
end;
if ListaLotesGerados.Count > 0 then
begin
try
FORM_INCONSISTENCIA := TFORM_INCONSISTENCIA.Create(nil);
FORM_INCONSISTENCIA.Caption := 'Lotes Gerados';
FORM_INCONSISTENCIA.RE_ERROS.Text := ListaLotesGerados.Text;
9640 FORM_INCONSISTENCIA.Hide;
FORM_INCONSISTENCIA.ShowModal;
finally
FORM_INCONSISTENCIA.Free;
end;
end;
9649 end;
9650
if (Copy(ED_CODFISCAL.Text,1,1) = '3') and (MTipoDoc = 'NE') and (Modo <> AtVisualiza) and (not(NFeAutorizada)) then
begin
DM_NFMODEL1.Q_DADOSIMP.AplicarAtualizacoes;
DM_NFMODEL1.Q_DADOSIMP.CommitUpdates;
end;
if (ParametroDoSistema('COMISSVD', 'VD') = 'S') and (ParametroDoSistema('ITCOMISS', 'VD2') = 'S') and
((ParametroDoSistema('LOCALPG', 'VD2') = 'N') or (ParametroDoSistema('ImpComisPE', 'VD2') = 'S')) then
begin
DM_NFMODEL1.Q_ItComiss.AplicarAtualizacoes;
DM_NFMODEL1.Q_ItComiss.CommitUpdates;
end;
if HabilitaNotasRef then
GravarNotasReferenciadas;
if not nfeAutorizada and (MTipoDoc <> 'CF') then
begin
```

A marcação será redirecionada até a linha em que se encontra o *end*.

Locate matching delimiter (Ctrl + Alt + Seta esquerda)

Seleciona todo o código que está dentro do delimitador selecionado. Útil ao adicionar uma nova condição, onde é necessário mover o código dois espaços a frente.

Exemplo:

Adicionando uma nova condição ao código

```
// FORM_DESDOBNF_V1.Show();
10580 if (ImportouPedido) and (ImportouApenasUmPedido) and
      (SameText(ParametroDoSistema('DESDOBPE', 'VD', 'PARAMVD'), 'S')) and (not SimpNac) then
10582 begin
      LBuscaInfoPedido := TBuscaInformacoesPedido.Create(NumPedidos, DM_NFMODELL.QMANUT.FieldByName('NumeroCli').AsString,
      'DESDOBNF', 'CodSituacao', GetCodigoDaSituacaoPadrao);
      try
      LBuscaInfoPedido.Buscar;
      FORM_DESDOBNF_V1.Q_DESDOBNF.Primeiro;
      while not FORM_DESDOBNF_V1.Q_DESDOBNF.Eof do
10590 begin
      FORM_DESDOBNF_V1.Q_DESDOBNF.Edit;
      FORM_DESDOBNF_V1.Q_DESDOBNF.FieldByName('CodSituacao').AsString :=
      LBuscaInfoPedido.QInfoPedido.FieldByName('CodSituacao').AsString;
      FORM_DESDOBNF_V1.Q_DESDOBNF.FieldByName('BancoBoleto').AsString :=
      LBuscaInfoPedido.QInfoPedido.FieldByName('BancoBoleto').AsString;
      FORM_DESDOBNF_V1.Q_DESDOBNF.FieldByName('BancoIdentificadorApelido').AsString :=
      LBuscaInfoPedido.QInfoPedido.FieldByName('BancoIdentificadorApelido').AsString;

      if LBuscaInfoPedido.PermiteAtribuirValorDasDuplicatasNoPedido(DM_NFMODELL.QMANUT.FieldByName('ValLiquido').AsFloat) an
      FORM_DESDOBNF_V1.NotaParcelouIguamentePedidos and
10600 (DM_NFMODELL.QMANUT.FieldByName('ValTotal').AsFloat = LBuscaInfoPedido.QInfoPedido.FieldByName('ValorTotal').AsFloa
      FORM_DESDOBNF_V1.Q_DESDOBNF.FieldByName('Valor').AsFloat := LBuscaInfoPedido.QInfoPedido.FieldByName('Valor').AsFloa

      FORM_DESDOBNF_V1.Q_DESDOBNF.Post;
      FORM_DESDOBNF_V1.Q_DESDOBNF.Next;
      LBuscaInfoPedido.QInfoPedido.Next;
      end;
      finally
      FreeAndNil(LBuscaInfoPedido);
      end;
10610 end;
// FORM_DESDOBNF_V1.Hide;
      FORM_DESDOBNF_V1.ShowModal;
      PassouDesdob := True;
end;
end
```

```
// FORM_DESDOBNF_V1.Show();
if (ImportouPedido) and (ImportouApenasUmPedido) and
(SameText(ParametroDoSistema('DESDOBPE', 'VD', 'PARAMVD'), 'S')) and (not SimpNac) then
begin
LBuscaInfoPedido := TBuscaInformacoesPedido.Create(NumPedidos, DM_NFMODELL.QMANUT.FieldByName('NumeroCli').AsString,
'DESDOBNF', 'CodSituacao', GetCodigoDaSituacaoPadrao);
if (A = B) then
begin
try
LBuscaInfoPedido.Buscar;
FORM_DESDOBNF_V1.Q_DESDOBNF.Primeiro;
while not FORM_DESDOBNF_V1.Q_DESDOBNF.Eof do
begin
FORM_DESDOBNF_V1.Q_DESDOBNF.Edit;
FORM_DESDOBNF_V1.Q_DESDOBNF.FieldByName('CodSituacao').AsString :=
LBuscaInfoPedido.QInfoPedido.FieldByName('CodSituacao').AsString;
FORM_DESDOBNF_V1.Q_DESDOBNF.FieldByName('BancoBoleto').AsString :=
LBuscaInfoPedido.QInfoPedido.FieldByName('BancoBoleto').AsString;
FORM_DESDOBNF_V1.Q_DESDOBNF.FieldByName('BancoIdentificadorApelido').AsString :=
LBuscaInfoPedido.QInfoPedido.FieldByName('BancoIdentificadorApelido').AsString;

if LBuscaInfoPedido.PermiteAtribuirValorDasDuplicatasNoPedido(DM_NFMODELL.QMANUT.FieldByName('ValLiquido').AsFloat) an
FORM_DESDOBNF_V1.NotaParcelouIguamentePedidos and
(DM_NFMODELL.QMANUT.FieldByName('ValTotal').AsFloat = LBuscaInfoPedido.QInfoPedido.FieldByName('ValorTotal').AsFloa
FORM_DESDOBNF_V1.Q_DESDOBNF.FieldByName('Valor').AsFloat := LBuscaInfoPedido.QInfoPedido.FieldByName('Valor').AsFloa

FORM_DESDOBNF_V1.Q_DESDOBNF.Post;
FORM_DESDOBNF_V1.Q_DESDOBNF.Next;
LBuscaInfoPedido.QInfoPedido.Next;
end;
finally
FreeAndNil(LBuscaInfoPedido);
end;
end;
end;
// FORM_DESDOBNF_V1.Hide;
FORM_DESDOBNF_V1.ShowModal;
PassouDesdob := True;
```

Posicionar o mouse no delimitador *end* adicionado e utilizar o atalho

```
'DESDOBNF', 'CodSituacao', GetCodigoDaSituacaoPadrao);
if (A = B) then
begin
try
LBuscaInfoPedido.Buscar;
FORM_DESDOBNF_V1.Q_DESDOBNF.Primeiro;
while not FORM_DESDOBNF_V1.Q_DESDOBNF.Eof do
begin
FORM_DESDOBNF_V1.Q_DESDOBNF.Edit;
FORM_DESDOBNF_V1.Q_DESDOBNF.FieldName('CodSituacao').AsString :=
LBuscaInfoPedido.QInfoPedido.FieldName('CodSituacao').AsString;
FORM_DESDOBNF_V1.Q_DESDOBNF.FieldName('BancoBoleto').AsString :=
LBuscaInfoPedido.QInfoPedido.FieldName('BancoBoleto').AsString;
FORM_DESDOBNF_V1.Q_DESDOBNF.FieldName('BancoIdentificadorApelido').AsString :=
LBuscaInfoPedido.QInfoPedido.FieldName('BancoIdentificadorApelido').AsString;

if LBuscaInfoPedido.PermiteAtribuirValorDasDuplicatasNoPedido(DM_NFMODEL1.QMANUT.FieldName('Valliquido').AsFloat) and
FORM_DESDOBNF_V1.NotaParcelouIgualmentePedidos and
(DM_NFMODEL1.QMANUT.FieldName('ValTotal').AsFloat = LBuscaInfoPedido.QInfoPedido.FieldName('ValorTotal').AsFloa
FORM_DESDOBNF_V1.Q_DESDOBNF.FieldName('Valor').AsFloat := LBuscaInfoPedido.QInfoPedido.FieldName('Valor').AsFloa

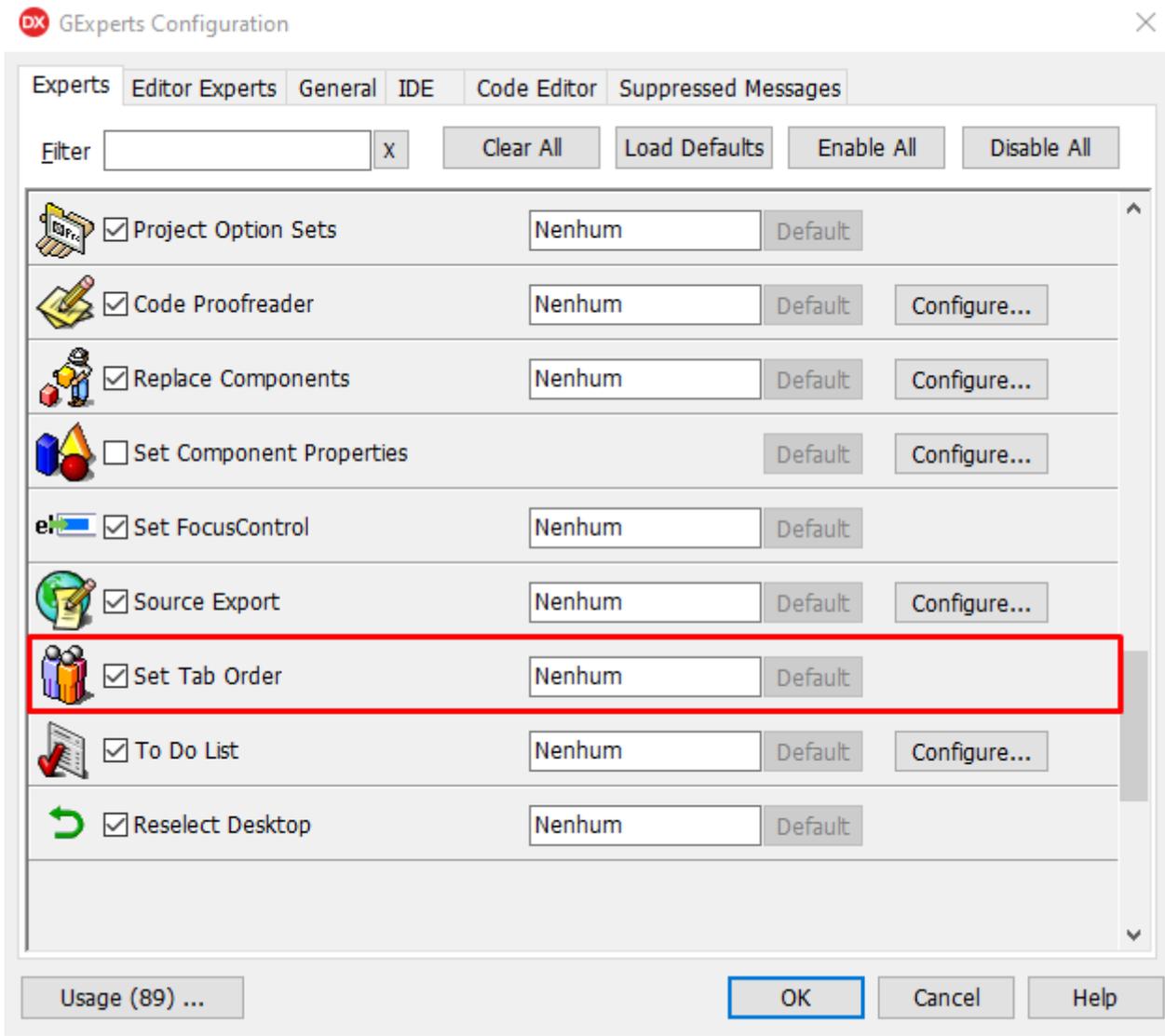
FORM_DESDOBNF_V1.Q_DESDOBNF.Post;
FORM_DESDOBNF_V1.Q_DESDOBNF.Next;
LBuscaInfoPedido.QInfoPedido.Next;
end;
finally
FreeAndNil(LBuscaInfoPedido);
end;
end;
end;
//
FORM_DESDOBNF_V1.Hide;
FORM_DESDOBNF_V1.ShowModal;
PassouDesdob := True;
```

Assim, todo o código poderá ser facilmente reposicionado, principalmente ao se tratar de trechos maiores.

Set tab order

Esse atalho, por padrão, não vem configurado.

Acessar menu "configuration..." visto na página anterior



Adicionar teclas de atalho. **Recomendação: Ctrl + Shift + T**

Apesar de pouco comum, convencionalmente poderá ser necessário adicionar componentes e alterar o "tab order" dos mesmos.

Com esse atalho, os componentes podem ser organizados apenas arrastando-os, não sendo necessário alterar as propriedades de cada um.

Arrange Components (Drag and Drop)

- ▼ FORM_NFMODEL1_EDIT: TForm_NFMODEL1_EDIT
 - ▼ Panel1: TPanel
 - ▼ PageControl1: TPageControl
 - ▼ TS_IDENT: TTabSheet
 - ▼ GroupBox2: TGroupBox
 - ▼ GB_CLIENTE: TGroupBox
 - DBR_NFMODEL1CLIFOR: TRadioGroup
 - RG_TIPO: TRadioGroup
 - DBE_NFMODEL1NUMEROFOR: TMacDbEdit
 - DBE_NFMODEL1NUMEROCLI: TMacDbEdit
 - ME_DESCRICA0: TKWRMaskEditDlg
 - DBE_CODENDENTR: TMacDbEditDlg
 - ED_CCM: TEdit
 - DBE_NFMODEL1INSCRMUNICIPAL: TMacDbEdit
 - DBE_NFMODEL1NUMNOTA: TDBEdit
 - DBE_NFMODEL1SERIE: TDBEdit
 - CB_TIPOIMPORTA: TComboBox
 - LB_DADmis: TListBox
 - LB_OS: TListBox
 - LB_PED: TListBox
 - LB_EMB: TListBox
 - LB_ORC: TListBox
 - LB_RecebMer: TListBox
 - ED_COPIAORCPED: TEdit
 - DBE_NFMODEL1NUMEROOS: TMacDbEdit
 - DBE_NFMODEL1TIPOOS: TMacDbEditD

OK

Cancel

Help

Order by Position

Reset Order