



## TREPORT

<b>Produto</b>	: Microsiga Protheus, versão 10	<b>Chamado</b>	: SCFHBT
<b>Data da criação</b>	: 18/01/10	<b>Data da revisão</b>	: 13/05/10.
<b>País(es)</b>	: Brasil	<b>Banco de Dados</b>	: Todos
<b>FNC</b>	: 00000026652/2009		

Este boletim tem o objetivo de informar a utilização do objeto TReport para criação de relatórios, detalhando as propriedades e métodos de cada classe utilizada no TReport.

### TREPORT

Classe de impressão que substitui as funções SetPrint, SetDefault, RptStatus e Cabec.

A classe TReport permite que o usuário personalize as informações que serão apresentadas no relatório, alterando fonte (tipo, tamanho, etc.), cor, tipo de linhas, cabeçalho, rodapé, etc.

Estrutura do componente TReport:

- O relatório (TReport) contém uma ou mais seções (TRSection);

- Uma seção (TRSection) pode conter uma ou mais seções;

- A seção (TRSection) contém células pré-definidas e células selecionadas pelo usuário;

- A seção (TRSection) também contém as quebras (TRBreak) para impressão de totalizadores (TRFunction);

- Os totalizadores são incluídos pela seção que automaticamente inclui no relatório (TReport).

### Propriedades

aBreak	Array com todas as quebras totalizadoras do relatório. Elemento: 1-Objeto TRBreak.
aCollection	Array com todos totalizadores do tipo TRCollection do relatório. Elemento: 1-Objeto TRCollection.
aCustomText	Array contendo a customização para impressão do cabeçalho padrão. Elementos: 1=Texto a ser impresso, no qual, um elemento por linha. Existem algumas strings que pode auxiliar na criação do cabeçalho: __NOLINEBREAK__ - Não quebra linha __NOTRANSFORM__ - Imprime sem nenhum tratamento __LOGOEMP__ - Imprime o logo da empresa __FATLINE__ - Imprime um linha grossa __THINLINE__ - Imprime uma linha fina
aBmps	Array com as imagens dos gráficos enviadas por email. Elemento: 1- Caminho da imagem.
aFontSize	Array com as fontes do sistema. Elementos: 1-Fonte, 2-Tamanho, 3- Tamanho em pixel.

aFunction	Array com todos totalizadores do tipo TRFunction do relatório. Elemento: 1-Objeto TRFunction.
aHeaderPage	Array com todas as seções que imprimem cabeçalho no topo da página.
aSection	Array com todas as seções do relatório. Elemento: 1-Objeto TRSection.
bAction	Bloco de código executado quando o usuário confirmar a impressão do relatório.
bCustomText	Bloco de código para atualização da propriedade aCustomText.
bOnNumberPage	Bloco de código para atualização do número da página atual.
bOnPageBreak	Bloco de código para tratamentos na inicialização de cada página.
bTotal	Compatibilidade – Não utilizado.
bTotalCanPrint	Bloco de código utilizado para validar a impressão dos totalizadores.
bTotalPos	Bloco de código utilizado para localizar a posição do totalizador a ser impresso.
bTotalPrint	Bloco de código utilizado para imprimir os totalizadores.
bTotalReset	Bloco de código utilizado para limpar os totalizadores.
bTotalRSize	Bloco de código utilizado para definir o tamanho das Collections.
bTotalText	Bloco de código utilizado na impressão do texto do totalizador.
cClassName	Nome da classe. Exemplo: TREPORT.
cDate	Data da impressão do relatório.
cDescription	Descrição do relatório.
cDir	Diretório selecionado para geração do relatório.
cEmail	E-mail utilizado na geração do relatório via e-mail.
cFontBody	Fonte definida para impressão do relatório.
cFile	Nome do arquivo que será gerado.
cID	ID do component. Exemplo: TREPORT.
cLogo	Logo da empresa/filial.
cMsgPrint	Mensagem apresentada durante a geração do relatório.
cPrinterName	Nome da impressora selecionada para impressão.
cReport	Nome do relatório. Exemplo: MATR010.
cTime	Hora da impressão do relatório.
cTitle	Título do relatório.
cRealTitle	Título padrão do relatório definido pelo criador do relatório.
cUserObs	Observação do usuário.
cXlsTHStyle	Estilo do cabeçalho padrão utilizado na geração da planilha.
cXlsSHStyle	Estilo do cabeçalho utilizado na geração da planilha.
cXmlDefault	Arquivo XML contendo Informações do relatório padrão.
cXlsFile	Nome do arquivo que será gerado em planilha.
IBold	Aponta que as Informações serão impressas em negrito.
ICanceled	Aponta que o relatório foi cancelado.
IClrBack	Define que a cor de fundo deverá ser atualizada.
IClrFore	Define que a cor da fonte deverá ser atualizada.
IDisableOrientation	Orientação (Retrato/Paisagem) não poderá ser modificada.
IDynamic	Aponta que o relatório é dinâmico, permitindo imprimir as seções conforme a ordem de impressão selecionada.
IEdit	Relatório não poderá ser configurado pelo usuário.

IEnabled	Impressão do relatório foi desabilitada.
IEmptyLineExcel	Suprime as linhas em branco e os totais na geração em planilha.
IFooterVisible	Habilita a impressão do rodapé.
IFunctionBefore	Imprime os totalizadores do tipo TRFunction antes dos totalizadores do tipo TRCollecions.
IHeaderVisible	Habilita a impressão do cabeçalho.
Italic	Aponta que as informações serão impressas em itálico.
IONPageBreak	Cabeçalho das seções impressas após a quebra de página.
IPageBreak	Quebra página antes da impressão dos totalizadores.
IParamPage	Existe parâmetros para impressão.
IParamReadOnly	Parâmetros não poderão ser alterados pelo usuário.
IPixColSpace	Espaçamento das colunas serão calculadas em pixel.
IPreview	Visualização do relatório antes da impressão física.
IPrinting	Relatório esta em processo de impressão.
IPrtParamPage	Aponta que serão impressos os parâmetros do relatório.
IStartPage	Aponta que uma nova página deverá se inicializada.
ITotalInLine	Imprime as células no formato linha.
ITPageBreak	Quebra página após a impressão do totalizador.
IUnderline	Aponta que as Informações serão impressas sublinhadas.
IUserAccess	Valida permissão para geração dos gráficos do relatório.
IUserInfo	Imprime Informações do usuário na página de parâmetros.
IUserFilter	Permite a utilização de filtros na personalização do relatório.
IXIsHeader	Imprime informações do cabeçalho padrão na geração em planilha.
INoPrint	Aponta que nenhuma informação foi impressa.
IXmlEndRow	Aponta fim de linha na geração em planilha.
IXIsParam	Aponta a existência de parâmetros na geração em planilha.
IEndReport	Imprime total geral do relatório.
nBorderDiff	Tamanho da borda utilizado para cálculo da altura de uma linha.
nClrBack	Cor de fundo.
nClrFore	Cor da fonte.
nCol	Coluna posiciona na impressão.
nColSpace	Espaçamento entre as colunas.
nDevice	Tipo de impressão selecionado. Opções: 1-Arquivo,2-Impressora,3-email,4-Planilha e 5-Html.
nEnvironment	Ambiente selecionado. Opções: 1-Server e 2-Cliente.
nFontBody	Tamanho da fonte definida para impressão do relatório.
nHeaderDiff	Tamanho do cabeçalho utilizado para cálculo do altura da página.
nLeftMargin	Tamanho da margem a esquerda.
nLineHeight	Altura da linha.
nLogPxYDiff	Utilizado no cálculo para geração da visualização do relatório.
nLogPxXDiff	Utilizado no cálculo para geração da visualização do relatório.
nMeter	Posição da régua de progressão.
nOrder	Ordem de impressão selecionada.
nPageWidth	Largura da página.

nPxColSpace	Espaçamento da coluna em pixel.
nPxBase	Tamanho da base em pixel.
nPxDate	Tamanho da sistema operacional em pixel.
nPxLeftMargin	Tamanho da margem a esquerda em pixel.
nPxPage	Tamanho da numeração da página em pixel.
nPxTitle	Tamanho do título em pixel.
nRemoteType	Aponta de que forma o Server está gerando o relatório. Opções: 1-Sem Remote, 2-Remote Delphi,3-Remote Windows e 4-Remote Linux.
nRow	Linha posicionada na impressão.
nXlsCol	Coluna posicionada na geração em planilha.
nXlsRow	Linha posicionada na geração em planilha.
nXlsStyle	Estilo utilizado na geração em planilha.
nExcel	Número do arquivo na geração em planilha.
nColumnPos	Posicionamento no arquivo gerado em planilha.
oBrdBottom	Objeto TRBorder com a borda Inferior .
oBrdLeft	Objeto TRBorder com a borda à esquerda.
oBrdRight	Objeto TRBorder com a borda à direita.
oBrdTop	Objeto TRBorder com a borda superior.
oHBrdBottom	Objeto TRBorder com a borda Inferior no cabeçalho.
oHBrdLeft	Objeto TRBorder com a borda à esquerda no cabeçalho.
oHBrdRight	Objeto TRBorder com a borda à direita no cabeçalho.
oHBrdTop	Objeto TRBorder com a borda superior no cabeçalho.
oClrBack	Objeto TBrush com a cor de Fundo.
oFontBody	Objeto TFont com a fonte do relatório.
oFontHeader	Objeto TFont com a fonte do cabeçalho.
oMeter	Objeto TMeter com a régua de progressão.
oMsg	Objeto TSAY com a mensagem apresentada durante a impressão do relatório.
oPage	Objeto TRPage com a configuração da página de impressão.
oParamPage	Objeto TRParamPage com a configuração da página de parâmetros.
oPrint	Objeto TMSPrinter.
oReport	Componente de impressão.
oXlsCell	Compatibilidade – Não utilizado.
oXlsRow	Compatibilidade – Não utilizado.
oXlsStyles	Compatibilidade – Não utilizado.
oXlsWorksheet	Compatibilidade – Não utilizado.
uParam	Parâmetros do relatório cadastrado no Dicionário de Perguntas (SX1). Também pode ser utilizado bloco de código para parâmetros customizados.

## MÉTODOS

### AddBreak(oBreak)

Adiciona a quebra de impressão na propriedade aBreak.

oBreak      Objeto TRBreak

### AddCollection(oCollection)

Adiciona o totalizador na propriedade aCollection.

oCollection      Objeto TRCollection

### AddFunction(oFunction,oParent)

Adiciona o totalizador na propriedade aFunction.

oFunction      Objeto TRFunction

oParent      Objeto TRSecion que aponta a seção que pertence o totalizador

### AddHeaderPage(oSection)

Adiciona a seção que imprime cabeçalho no topo da página.

oSection      Objeto TRSection

### **AddSection(oSection)**

Adiciona a seção na propriedade aSection.

oSection      Objeto TRSection

### **Border(uBorder,IHeader)**

Retorna a borda do relatório.

uBorder      Tipo Caracter: "TOP","BOTTOM","LEFT","RIGHT","ALL"

                Tipo Numérico: 1-Superior,2-Inferior,3-Esquerda,4-Direita,5-Todas

IHeader      Borda do cabeçalho

Retorno      Objeto do tipo TRBorder

### **Box(nRow,nCol,nBottom,nRight,oPen)**

Desenha uma caixa, utilizando as especificações do objeto TPen.

nRow          Linha no qual inicia o desenho da caixa

nCol          Coluna no qual inicia o desenho da caixa

nBottom      Linha no qual finaliza o desenho da caixa

nRight       Coluna no qual finaliza o desenho da caixa

oPen          Objeto da classe TPen

### **Cancel()**

Retorna se o usuário cancelou a impressão do relatório.

### **CancelPrint()**

Cancela a impressão do relatório.

### **Char2Pix(nSize,cFont,nFontSize)**

Retorna o tamanho de acordo com a fonte informada.

nSize          Tipo Caracter: Texto a ser considerado no cálculo

                Tipo Numérico: Valor a ser considerado no cálculo

cFont          Fonte do relatório

nFontSize     Tamanho da fonte

Retorno      Tamanho calculado em pixel

### **ChkIncRow(nInc,ILine)**

Verifica a necessidade de inicializar uma nova página antes da inclusão da linha.

nInc          Quantidade de linhas a serem impressas

ILine         Considera o tamanho da linha no cálculo

Retorno      Lógico

### **ClassName()**

Retorna o nome da classe. Exemplo: TREPORT.

### **ClrBack(IObject)**

Retorna a cor de fundo do relatório.

IObject       Aponta que deve ser retornado o objeto TBrush

Retorno      Caso o parâmetro IObjeto seja verdadeiro o retorno será o objeto TBrush, caso contrário será o número da cor RGB.

**ClrFore()**

Retorna a cor de fonte do relatório.

Retorno          Número da cor RGB

**Col()**

Retorna a coluna posicionada na impressão.

**ColSpace()**

Retorna o espaçamento entre as colunas.

**Description()**

Retorna a descrição do relatório.

**Disable()**

Desabilita a impressão do relatório.

**DisableOrientation()**

Desabilita a seleção da orientação (Retrato/Paisagem).

**Enable()**

Habilita a impressão do relatório.

**Enabled()**

Retorna se a impressão do relatório esta habilitada.

**EndPage(IFooter)**

Finaliza a página na impressão.

IFooter          Imprime rodapé na finalização da página

**EvalBreak(IForce,IPrintHeader,oSection)**

Executa a quebra na impressão do relatório.

IForce          Força a execução do método OnBreak da classe TRBreak

IPrintHeader   Imprime cabeçalho da seção

oSection        Seção considerada na quebra

Retorno          Lógico. Se verdadeiro, quebrou a impressão

**EndReport()**

Retorna se imprime o total geral do relatório.

**FatLine()**

Desenha uma linha com altura grossa, iniciando posição da linha atual de impressão com a largura da página.

**FillRect(aRect,oBrush)**

Preenche um retângulo na impressão utilizando as especificações do objeto TBrush.

aRect          Vetor com coordenadas no formato: linha inicial, coluna inicial, linha final,  
coluna final

oBrush        Objeto da classe TBrush

**Finish()**

Finaliza a impressão do relatório, imprime os totalizadores, fecha as queries e índices temporários, entre outros tratamentos do componente.

Não é necessário executar o método Finish se for utilizar o método Print, já que este faz o controle de inicialização e finalização da impressão.

### **FreeAllObjs()**

Elimina os objetos da memória no servidor.

### **FunctionBefore(IFunctionBefore)**

Define se a impressão dos totalizadores do tipo TRFunction será realizada antes dos totalizadores do tipo TRCollection.

### **GetAction()**

Retorna o bloco de código definido para a propriedade bAction.

### **GetBreak(uBreak)**

Retorna a quebra do relatório.

uBreak           Tipo Caracter: Nome da quebra  
                  Tipo Numérico: Número da quebra no array aBreak

Retorno          Objeto TRBreak

### **GetDynamic()**

Retorna se o relatório é do tipo dinâmico.

Retorno          Lógico

### **GetEdit()**

Retorna se é permitida a configuração do relatório pelo usuário.

Retorno Lógico

### **GetFontSize(cFont,nFontSize,IBold,Italic,IUnderline)**

Retorna o tamanho da fonte.

cFont            Fonte a ser considerada no cálculo  
nFontSize        Tamanho da fonte  
IBold            Aponta se a fonte é negrita  
Italic            Aponta se a fonte é itálica  
IUnderline       Aponta se a fonte é sublinhada

Retorno          Array com informações do tamanho da fonte. Elementos: 1-Fonte, 2-Tamanho, 3-Tamanho em pixel

### **GetFunction(uFunction)**

Retorna objeto da classe TRFunction (totalizadores).

uFunction        Tipo Caracter: Nome da Function  
                  Tipo Numérico: Índice do objeto segundo a ordem de criação dos componentes TRFunction

Retorno          Objeto da classe TRFunction (totalizadores)

### **GetOrder()**

Retorna a ordem de impressão selecionada.

Retorno          Ordem selecionada

### **GetOrientation()**

Retorna a orientação de página (Retrato ou Paisagem) selecionada pelo usuário.

Retorno        1=Retrato ou 2=Paisagem

### **GetParam()**

Retorna a pergunta ou bloco de código utilizado como parâmetros do relatório.

### **GetPassword()**

Não utilizado.

### **GetWidth()**

Retorna a largura da página.

### **HideFooter()**

Define que não será impresso o rodapé padrão da página.

### **HideHeader()**

Define que não será impresso o cabeçalho padrão da página.

### **HideParamPage()**

Define se será permitida a alteração dos parâmetros do relatório.

### **IncMeter(nInc)**

Incrementa a régua de progressão do relatório.

nInc            Quantidade a incrementar na régua. Padrão: 1

### **IncRow(nInc)**

Incrementa linhas na impressão.

nInc            Quantidade de linhas que deverão ser incrementadas

Retorno        Linha atualizada.

### **Init()**

Inicializa as configurações e define a primeira página do relatório.

Não é necessário executar o método Init se for utilizar o método Print, já que estes fazem o controle de inicialização e finalização da impressão.

### **LeftMargin(IPixel,IForce)**

Retorna o tamanho da margem à esquerda do relatório.

IPixel        Considera cálculo em pixel

IForce        Força o recálculo da margem à esquerda

Retorno        Tamanho da margem à esquerda

### **Line(nTop,nLeft,nBottom,nRight,oPen)**

Desenha uma linha, utilizando as especificações da classe TPen.

nTop        Linha no qual inicia o desenho da linha

nLeft        Coluna no qual inicia o desenho da linha

nBottom    Linha no qual finaliza o desenho da linha

nRight      Coluna no qual finaliza o desenho da linha

oPen        Objeto da classe TPen

### **LineHeight()**

Retorna a altura da linha.



Retorno          Altura da linha

### **LoadDefault()**

Carrega o relatório padrão para personalização.

### **LoadLayout(cLayout)**

Carrega um layout customizado para impressão.

cLayout          Layout a ser carregado

### **New(cReport,cTitle,uParam,bAction,cDescription,ILandscape,uTotalText,ITotalInLine,cPageTText,IPageTInLine,ITPageBreak,nColSpace)**

Método construtor da classe TReport.

cReport	Nome do relatório. Exemplo: MATR010
cTitle	Título do relatório
uParam	Parâmetros do relatório cadastrado no Dicionário de Perguntas (SX1) Também pode ser utilizado bloco de código para parâmetros customizados.
bActionBloco	de código que será executado quando o usuário confirmar a impressão do relatório
cDescription	Descrição do relatório
ILandscape	Aponta a orientação de página do relatório como paisagem
uTotalText	Texto do totalizador do relatório, podendo ser caracter ou bloco de código
ITotalInLine	Imprime as células em linha
cPageTText	Texto do totalizador da página
IPageTInLine	Imprime totalizador da página em linha
ITPageBreak	Quebra página após a impressão do totalizador
nColSpace	Espaçamento entre as colunas

Retorno          Objeto

### **NoUserFilter()**

Desabilita a utilização de filtros na personalização do relatório.

### **NoCell()**

Verifica a existência de seção sem células.

Retorno          Lógico

### **NoPrint()**

Retorna se não foram impressos registros.

Retorno          Lógico

### **OnPageBreak(bOnPageBreak,IONPageBreak)**

Atualiza a propriedade bOnPageBreak utilizada para tratamentos na inicialização de cada página.

bOnPageBreak          Bloco de código para tratamentos na inicialização de cada página

IONPageBreak          Cabeçalho das seções impressas após a quebra de página

### **Page()**

Retorna o número da página atual na impressão.

### **PageBreak()**

Retorna se existe quebra de página.

Retorno          Lógico

#### **PageHeight(IHeaderDiff)**

Retorna a altura da página baseado no tamanho da folha (A4, Carta, etc), conforme seleção do usuário.

IHeaderDiff      Desconsidera o tamanho reservado para impressão do cabeçalho

#### **PageTotalBefore()**

Define se o total da página será impresso antes do total geral do relatório.

#### **PageTotalInLine(ITotalInLine)**

Define se o total da página será impresso em linha ou coluna.

ITotalInLine      Se verdadeiro será impresso em linha

#### **PageTotalText(cText)**

Define o texto do totalizador da página.

cText              Texto do totalizador da página

#### **PageWidth()**

Retorna a largura da página baseado no tamanho da folha (A4, Carta, etc), conforme seleção do usuário.

#### **ParamReadOnly(IParamReadOnly)**

Define se o usuário terá acesso aos parâmetros do relatório.

IParamReadOnly      .T. – Não permite acesso aos parâmetros

                         .F. – Permite acesso aos parâmetros

#### **Preview()**

Apresenta a visualização do relatório antes da impressão física.

#### **Print(IDlg)**

Executa a impressão do relatório conforme o bloco de código da propriedade bAction.

IDlg                  Aponta se deverá apresentar a tela de configuração do relatório ou se imprime em segundo plano.

#### **PrintCollection(IPrintHeader,IFinish)**

Imprime os totalizadores do tipo Collections.

IPrintHeader      Imprime cabeçalho da seção

IFinish              Aponta que é o encerramento do relatório

Retorno          Lógico. Se verdadeiro, o total foi impresso

#### **PrintDialog(bAction)**

Exibe a tela de configuração para a impressão do relatório.

bActionBloco de código que será executado quando usuário confirmar a impressão do relatório

Retorno          Lógico

#### **PrintFooter()**

Imprime o rodapé do relatório.

#### **PrintFunction(IPrintHeader)**

Imprime os totalizadores do tipo TRFunctions.

IPrintHeader      Imprime cabeçalho da seção

**PrintGraphic()**

Imprime os gráficos do relatório.

**PrintHeader(IHeaderSection,IPage)**

Imprime o cabeçalho padrão do relatório.

IHeaderSection      Imprime cabeçalho da seção

IPage                  Imprime Informações sobre a página, como por exemplo, número da página

**Printing()**

Retorna se relatório esta em processo de impressão.

O status de impressão é definido pelos métodos Init(), que indica o início da impressão, e o método Finish, que indica a finalização.

**PrintSHeader()**

Imprime o cabeçalho da seção.

**PrintText(cText,nRow,nCol,nClrText,cStyle,nCells,IUpdateRow)**

Imprime um texto no relatório.

cText                Texto que será impresso

nRow                Linha em que o texto será impresso. Caso não informada, será considerada a linha posicionada na impressão

nCol                Coluna em que o texto será impresso. Caso não informada, será considerada a coluna posicionada na impressão

nClrText            Cor do texto

cStyle              Utilizado internamente para geração do relatório em planilha

nCells              Utilizado internamente para geração do relatório em planilha

IUpdateRow        Atualiza o posicionamento da linha quando informado o parâmetro nRow

**PrintTHeader(nWidth,nCells)**

Imprime o cabeçalho dos totalizadores.

nWidth              Largura do cabeçalho

nCells              Quantidade de células considerada na geração em planilha

**PrintTotal(IFinish)**

Imprime os totalizadores do relatório.

IFinish              Aponta que é o fim do relatório

Retorno            Se verdadeiro, os totalizadores foram impressos

**PrtCenter(cText)**

Imprime um texto centralizado.

cText                Texto a ser centralizado

**PrtLeft(cText)**

Imprime um texto à esquerda.

cText                Texto a ser impresso à esquerda

**PrtLogo()**

Imprime o logo da empresa/filial.

**PrtRight(cText)**

Imprime um texto à direita.

cText                Texto a ser impresso à direita

**ReportName()**

Retorna o nome do relatório. Exemplo: MATR010.

**Row()**

Retorna a linha posicionada na impressão.

**SaveAsHTML()**

Salva o relatório em HTML.

**SaveDefault()**

Salva o relatório padrão sem as customizações do usuário.

**Say(nRow,nCol,cText,oFont,nWidth,nClrText,nBkMode,nPad)**

Imprime um texto no relatório.

nRow	Linha para impressão do texto
nCol	Coluna para impressão do texto
cText	Texto que será impresso
oFont	Objeto da classe TFont
nWidth	Tamanho em pixel do texto para impressão
nClrText	Cor da fonte
nBkMode	Compatibilidade – Não utilizado
nPad	Compatibilidade – Não utilizado

**SayBitmap(nRow,nCol,cBitmap,nWidth,nHeight,nRaster)**

Imprime uma imagem no relatório.

nRow	Linha para impressão da imagem
nCol	Coluna para impressão da imagem
cBitmap	Nome da imagem, podendo ser path de um arquivo ou resource compilado no repositório
nWidth	Largura da imagem
nHeight	Altura da imagem
nRaster	Compatibilidade – Não utilizado

**Section()**

Retorna objeto da classe TRSection (seção).

uSection	Tipo Caracter: Título da seção
	Tipo Numérico: Índice da seção segundo a ordem de criação dos componentes TRSection

**SendMail()**

Envia o relatório através do e-mail.

**SendToPrinter()**

Direciona o relatório para o tipo de impressão selecionada.

**SetAction(bAction)**

Define o bloco de código que será executado ao usuário confirmar a impressão do relatório.

bActionBloco de código que será executado na confirmação

**SetBorder(uBorder,nWeight,nColor,lHeader)**

Define as bordas do relatório.

uBorder	Tipo Caracter: "TOP","BOTTOM","LEFT","RIGHT","ALL"
	Tipo Numérico: 1-Superior,2-Inferior,3-Esquerda,4-Direita,5-Todas
nWeight	Largura da borda
nColor	Cor da borda
lHeader	Aponta se é borda de cabeçalho

Retorno Objeto do tipo TRBorder

### **SetClrBack(nClrBack)**

Define a cor de fundo do relatório.

nClrBack Número da cor de fundo no format RGB

### **SetClrFore(nClrFore)**

Define a cor da fonte do relatório.

nClrFore Número da cor da fonte no format RGB

### **SetCol(nCol)**

Define a coluna de impressão.

### **SetColSpace(nColSpace,lPixel)**

Define o espaçamento entre as colunas.

nColSpace Tamanho do espaçamento

lPixel Aponta se o tamanho será calculado em pixel

### **SetCustomText(uCustomText)**

Define que a impressão do cabeçalho padrão será customizado.

uCustomText Tipo Array: Array contendo a customização para impressão do cabeçalho padrão.

Elementos: 1=Texto a ser impresso, no qual, um elemento por linha. Existem algumas strings que pode auxiliar na criação do cabeçalho:

\_\_NOLINEBREAK\_\_ - Não quebra linha

\_\_NOTTRANSFORM\_\_ - Imprime sem nenhum tratamento

\_\_LOGOEMP\_\_ - Imprime o logo da empresa

\_\_FATLINE\_\_ - Imprime um linha grossa

\_\_THINLINE\_\_ - Imprime uma linha fina

Tipo Bloco de código: Contendo Informações para geração do array descrito acima

### **SetDescription(cDescription)**

Define a descrição detalhada do relatório.

cDescription Descrição do relatório

### **SetDevice(nDevice)**

Define o tipo de impressão selecionado. Opções: 1-Arquivo,2-Impressora,3-email,4-Planilha e 5-Html.

nDevice Tipo de impressão selecionada

### **SetDynamic()**

Define que o relatório será do tipo dinâmico, no qual, permite imprimir as seções conforme a ordem de impressão selecionada.

### **SetEdit(lEdit)**

Define se o relatório poderá ser configurado pelo usuário no caso de verdadeiro.

**SetEnvironment(nEnv)**

Define o ambiente para impressão.

nEnv                Ambiente: 1-Server e 2-Cliente

**SetFile()**

Define o nome do arquivo temporário utilizado para geração dos relatórios.

**SetLandscape()**

Define orientação de página do relatório como paisagem.

**SetLeftMargin(nLeftMargin)**

Define a margem à esquerda do relatório.

nLeftMargin      Tamanho da margem à esquerda

**SetLineHeight(nLineHeight)**

Define a altura da linha na impressão.

nLineHeight      Altura da linha

**SetLogo()**

Define o logo da empresa na impressão do cabeçalho padrão.

**SetMeter(nTotal)**

Define o limite da régua de progressão do relatório.

nTotal            Limite da régua

**SetMsgPrint(cMsg)**

Define a mensagem apresentada durante a geração do relatório.

cMsg              Texto da mensagem

**SetOnPageNumber(bOnNumberPage)**

Define o bloco de código utilizado para manipular o número da página atual.

**SetPageNumber(nPage)**

Define o número da página atual.

**SetPageFooter(nLinesFooter,bPageFooter,IFooterBefore)**

nLinesFooter    Linhas reservada para impressão do rodapé

bPageFooter    Bloco de código utilizado para imprimir Informações customizadas, no qual, deverá ser utilizado métodos do TReport para impressão. Exemplo: Say, PrintText.

IFooterBefore   Imprime rodapé antes dos totalizadores

**SetParam(uParam)**

Define os parâmetros que serão utilizados pelo usuário.

uParam           Parâmetros do relatório cadastrado no Dicionário de Perguntas (SX1).

Também pode ser utilizado bloco de código para parâmetros customizados

**SetPortrait()**

Define orientação de página do relatório como retrato.

**SetPreview()**

Define se será apresentada a visualização do relatório antes da impressão física.

**SetPxLeftMargin(nPxLeftMargin)**

Define a margem à esquerda em pixel do relatório.

nPxLeftMargin Tamanho da margem à esquerda em pixel

**SetRow()**

Define a linha de impressão.

**SetStartPage(IStartPage)**

Define que deverá ser gerada uma nova página no relatório.

IStartPage Se verdadeiro, aponta a necessidade de uma nova página

**SetTitle(cTitle)**

Define o título do relatório.

cTitle Título do relat

**SetTotalInLine(ITotalInLine)**

Define se os totalizadores serão impressos em linha ou coluna.

**SetTotalPageBreak(ITPageBreak)**

Define que será quebrada página após a impressão do totalizador.

ITPageBreak Se verdadeiro, aponta a quebra de página após a impressão do totalizador

**SetTotalText(uText)**

Define o texto que será impresso antes da impressão dos totalizadores.

uText Texto a ser impresso no formato caracter ou bloco de código

**ShowFooter()**

Define que será impresso o rodapé do relatório.

**ShowHeader()**

Define que será impresso o cabeçalho do relatório.

**ShowParamPage()**

Define que será impressa a página de parâmetros do relatório.

**SkipLine(nSkip)**

Salta linhas na impressão.

nSkip Quantidade de linhas a serem saltadas

**StartPage()**

Inicializa uma nova página para impressão.

**ThinLine()**

Desenha uma linha simples, iniciando posição da linha atual de impressão com a largura da página.

**Title()**

Retorna o título do relatório.

**TotalText()**

Retorna o texto que será impresso no totalizador geral do relatório.

**TotalInLine()**

Retorna se a impressão dos totalizadores será em linha.

**XmlLoad(cXml)**

Carrega um layout personalizado pelo usuário.

cXml                Arquivo do tipo XML com as personalizações do usuário

**XmlSave()**

Salva um layout personalizado pelo usuário.

**XlsNewCell(c, lIndex, nCol, cStyle, nCells, nWidth, cType)**

Adiciona uma célula na geração em planilha.

c                    Informação a ser inserido na célula  
lIndex            Controla numeração de colunas na planilha  
nCol               Coluna posicionada  
cStyle            Estilo da célula  
nCells            Quantidade de células utilizadas para a informação ( Mesclar )  
nWidth            Largura da célula  
cType              Tipo de dado. C-Character, N-Numérico, L-Lógico

**XlsNewCol(nCol, nWidth)**

Adiciona uma coluna na geração em planilha.

nCol               Coluna posicionada  
nWidth            Largura da coluna

**XlsNewRow(lIncRow)**

Adiciona uma linha na geração em planilha.

lIncRow           Aponta se deverá ser adicionada uma linha

**XlsNewStyle(cID, cName, nAlign, aFont, aBorder, lWrapText, cType)**

Cria um estilo para utilizar nas criações das células.

cID                ID do estilo  
cName            Nome do estilo  
nAlign            Tipo de alinhamento. 1-Esquerda, 2-Centralizado e 3-Direita  
aFont            Array contendo a fonte. Elementos: 1-Fonte e 2-Tamanho  
aBorder           Array contendo a borda. Elementos: 1-"Bottom", "Left", "Right", "Center" e 2=Largura da borda  
lWrapText       Aponta quebra de linha automática  
cType              Tipo de dado. C-Character, N-Numérico, L-Lógico

**XlsSave()**

Compatibilidade – Não utilizado.



## TSECTION

Seção de um relatório que utiliza a classe TReport.

A classe TRSection pode ser entendida como um layout do relatório, por conter células, quebras e totalizadores que darão um formato para sua impressão.

Com a classe TRSection é possível definir uma query, filtro ou índice com filtro (IndRegua) que será utilizada por ela para processamento do relatório, através do método Print e utilizando as células de posicionamento (TRPosition) e componente de posicionamento de seção (TRPosSection), sem a necessidade de escrever um fonte para processar os resultados da query, filtro ou índice com filtro.

Esta classe herda as propriedades e métodos da classe TREPORT.

### Propriedades

aCell	Array contendo as células da seção. Elemento: 1-Objeto TRCell.
aCellPos	Array contendo as células reposicionadas da seção. Elemento: 1-Objeto TRCell.
aFilter	Array contendo os filtros da seção: Elementos: 1-Tabela, 2-Filtro, 3-Chave de Índice, 4-Ordem.
aOrder	Array contendo as ordens do relatório: Elementos: 1-Objeto TROrder.
aPosCell	Array com as células da seção na ordem de impressão.
aTable	Array com as tabelas utilizadas na seção.
aTCFields	Array com os campos que possuem o tipo de dados diferente de caracter e que devem ser tratados para apresentar os resultados na query. Elementos: 1-Campo, 2-Tipo, 3-Tamanho e 4-Decimal.
aTCMemo	Array com os campos do tipo de dados Memo a serem desconsiderados na query.
aTCTables	Array com as tabelas utilizadas na query.
aLoadCells	Array com as tabelas que executaram o carregamento de Informações das células através do Dicionário de Dados (SX3).
aUserFilter	Array com os filtros de usuários: Elementos: 1-Tabela, 2-Expressão ADVPL, 3-Expressão SQL e 4-Filtro adicionado na query principal.
aNoFilter	Array com as tabelas que não poderão aplicar filtros de usuário. Elemento: 1-Tabela.
aSection	Array com as seções filhas. Elemento: 1-Objeto TRSection.
bCompQuery	Bloco de código utilizado na montagem da query através de compilação em tempo real.
bLineCondition	Bloco de código utilizado na validação do registro.
bOnPrintLine	Bloco de código com os tratamentos a serem realizados antes da impressão do registro da seção.
bParentFilter	Bloco de código com a regra para saída do loop.
bParentParam	Bloco de código com a expressão que retorna o valor que é enviado como parâmetro para a regra de saída do loop da seção.
bRealQuery	Bloco de código utilizado para montar a query da seção.
cAlias	Tabela principal da seção.
cAdvplExp	Filtro do usuário em forma de expressão ADVPL.
cDynamicKey	Chave que identifica a seção na impressão dinâmica.
cFilter	Filtro da tabela principal da seção.
cIdxFile	Índice temporário utilizado na filtro da tabela principal.
cName	Nome da seção.
cQuery	Query da seção com os tratamentos de adição de campos e filtros.
cRealFilter	Filtro da tabela principal da seção.
cRealQuery	Query sem os tratamentos de adição de campos e filtros.
cCharSeparator	Caracter que separa as Informações na impressão em linha.
cSqlExp	Filtro do usuário em forma de expressão SQL.

IAutoSize	Ajusta o tamanho das células para que caiba em uma página.
ICellPos	Ajusta o cabeçalho das células.
IChangeQuery	Tratamento para utilizar a query em diversos Banco de Dados.
ICheckFilters	Compatibilidade – Não utilizado.
IEdit	Aponta se a seção poderá ser personalizada pelo usuário.
IEditCell	Aponta se o usuário poderá personalizar as células da seção.
IForceLineStyle	Força a impressão em linha.
IHeaderBreak	Imprime cabeçalho da seção na quebra de impressão (TRBreak).
IHeaderPage	Imprime cabeçalho da seção no topo da página.
IHeaderSection	Imprime cabeçalho da seção na quebra de seção.
IIdxOrder	Utiliza ordem do Dicionário de Índices (SIX) na impressão da seção.
INit	Aponta que a impressão da seção não foi iniciada.
INitFilter	Aponta que os filtros da seção não foram iniciados.
ILineBreak	Aponta que a impressão da seção quebra linhas no caso das colunas não couberem em uma linha.
ILineStyle	Impressão em linhas.
ISkipped	Aponta que a seção saltou o registro da seção pai.
IParentQuery	Utiliza Informações da query da seção pai para impressão dos registros.
IParentRecno	Utiliza Informações do registro da seção pai.
IPrintHeader	Aponta impressão do cabeçalho da seção.
IPrintLayout	Aponta que é impressão de visualização do layout.
IReadOnly	Define se o usuário pode personalizar informações da seção.
ITCFields	Define que deverá ser efetuado tratamento na query de campos com tipo de dado diferente de caracter.
IVisible	Aponta que a seção será impressa.
IUserVisible	Aponta que a seção será impressa na personalização do usuário.
ICellUseQuery	Utiliza query na impressão de células da seção.
nCols	Quantidade de colunas a serem impressas.
nIdxOrder	Índice utilizado na impressão da seção.
nLineCount	Quantidade de linhas a serem impressas para o registro.
nLinesBefore	Quantidade de linhas a serem saltadas antes da impressão da seção.
nOrder	Ordem de impressão da seção.
nPercentage	Percentual da largura da página a ser considerada.
nRow	Linha posicionada na impressão da seção.
nWidth	Largura da seção.
oCBrdBottom	Objeto TRBorder com a borda Inferior.
oCBrdLeft	Objeto TRBorder com a borda à esquerda.
oCBrdRight	Objeto TRBorder com a borda à direita.
oCBrdTop	Objeto TRBorder com a borda superior.
oParent	Seção pai.
oRelation	Objeto TRRelation com informações do relacionamento entre as seções.

## MÉTODOS

### AddCell(oCell)

Adiciona a célula na propriedade aCell da seção.

oCell            Objeto TRCell

### AddOrder(oOrder)

Adiciona a ordem da seção na propriedade aOrder.

oOrder            Objeto TROrder

### AddPosCell(oCell)

Adiciona a célula, na sequência de impressão da seção, na propriedade aPosCell.

oCell            Objeto TRCell

### **AddTable(cTable)**

Adiciona a tabela, que será utilizada na impressão da seção, na propriedade aTable.

cTable            Tabela utilizada na impressão da seção

### **Alias()**

Retorna o alias da tabela utilizado pela query da seção, definida pelo Embedded SQL com os métodos BeginQuery e EndQuery.

Retorno            Alias da tabela posicionada

### **AutoSize()**

Retorna se a impressão da seção será ajustada automaticamente.

Retorno            Se verdadeiro, ajusta automaticamente a impressão das células da seção

### **BeginQuery()**

Indica que será utilizado o Embedded SQL para criação de uma query para a seção.

### **Cell(uCell)**

Retorna o objeto da classe TRCell (célula) baseado.

uCell            Tipo Caracter: Nome ou título do objeto  
                  Tipo Numérico: Índice do objeto segundo a ordem de criação dos componentes  
                  TRCell

Retorno            Objeto da classe TRCell

### **CellBorder(uBorder,IHeader)**

Retorna a borda da seção.

uBorder            Tipo Caracter: "TOP", "BOTTOM", "LEFT", "RIGHT", "ALL"  
                  Tipo Numérico: 1-Superior, 2-Inferior, 3-Esquerda, 4-Direita, 5-Todas  
IHeader            Borda do cabeçalho

Retorno            Objeto do tipo TRBorder

### **CellPos(IReset)**

Ordena as células para impressão.

IReset            Força a inicialização do tamanho das células

### **CharSeparator()**

Retorna o caracter que separa as Informações na impressão das células em linha.

Retorno            Caracter utilizado para separar as informações

### **CheckFields(aTables,aMemo,IEnabled,aHasArea,nTotCell,cSelect)**

Retorna as células que poderão ser adicionadas na query, ou seja, retira os campos do tipo Memo e as células que não foram selecionadas pelo usuário.

aTables            Tabelas utilizadas na query  
aMemoCampos      memo da tabela  
IEnabled           Define se considera as células selecionadas pelo usuário  
aHasArea           Valida se as tabelas utilizadas na query são padrão do sistema  
nTotCell           Quantidade maxima de células a serem consideradas

cSelectTexto contendo a select da query

Retorno Campos a serem considerados na query

### **ChkMainFilter(cAlias)**

Retorna o filtro da tabela.

cAlias Tabela que contém o filtro

Retorno Filtro da tabela

### **ChkTcMemo(aTables,aMemo)**

Retorna os campos do tipo Memo das tabelas.

aTables Tabelas a serem verificadas

aMemoArray contendo os campos do tipo Memo

Retorna Array contendo os campos do tipo Memo

### **CheckWidth(nWidth)**

Retorna a largura da seção.

nWidth Caso a largura da seção for inferior a largura informada, a seção assume a informada no parâmetro

Retorno Largura da seção

### **CloseFilter()**

Finaliza todos os filtros da seção.

### **CloseQuery(ISections)**

Finaliza todas as queries da seção.

ISections Finaliza as queries de todas seções

### **DelUserCell()**

Exclui as células adicionadas pelo usuário.

### **EndQuery(aParam)**

Indica a query criada utilizando o Embedded SQL para a seção.

O método EndQuery ira juntar na query as células selecionadas e o filtro criado pelo usuário, mais a expressão das perguntas do tipo Range, que foram convertidas anteriormente pela função MakeSQLExpr.

Após sua execução, a query esta pronta para ser utilizada tanto pela classe TRSection como para o programa que esta criando os componentes, sem a necessidade de utilizar TCGenQuery, ChangeQuery e TCSetField.

aParam Tipo Caracter: Pergunta. Exemplo: mv\_par01 do tipo Range  
Tipo Array: Lista de perguntas. Exemplo: {mv\_par01, mv\_par02} do tipo Range

### **EndBorder()**

Finaliza a impressão das bordas.

### **EvalCell()**

Atualiza o conteúdo de todas as células da seção.

### **EvalFunction()**

Atualiza o conteúdo dos totalizadores da seção.

**EvalPosition()**

Atualiza o conteúdo das células utilizadas na impressão da seção.

**ExecSql()**

Executa a query da seção.

**ExUserFilter()**

Executa os filtros definidos pelo usuário.

Retorno          Retorno da execução do filtro ( Verdadeiro ou Falso )

**Finish()**

Finaliza a impressão da seção, imprime os totalizadores, tratamentos de quebras das seções, entre outros tratamentos do componente.

Não é necessário executar o método Finish se for utilizar o método Print, já que este faz o controle de inicialização e finalização da impressão.

**ForceLineStyle()**

Força a impressão da seção em linhas.

**GetAdvplExp(cAlias)**

Retorna o filtro do usuário em forma de expressão ADVPL.

cAlias          Tabela a ser consultada

Retorno          Filtro do usuário em forma de expressão ADVPL

**GetDynamicKey()**

Retorna a chave dinâmica utilizada para geração de relatórios dinâmicos.

Retorno          Chave dinâmica

**GetIdxOrder()**

Retorna o índice utilizado pela tabela principal da seção.

Retorno          Índice utilizado pela tabela principal

**GetOrder()**

Retorna a ordem selecionada.

Retorno          Ordem selecionada pelo usuário

**GetParentQuery()**

Atualiza as Informações de query da seção atual com as Informações da seção pai.

**GetQuery()**

Retorna a query, que foi definida pelo Embedded SQL com os métodos BeginQuery e EndQuery, com as células selecionadas, o filtro criado pelo usuário e as perguntas do tipo Range inclusas.

Retorno          String com a query

**GetSqlExp()**

Retorna o filtro do usuário em forma de expressão SQL.

cAlias          Tabela a ser consultada

Retorno          Filtro do usuário em forma de expressão SQL

#### **GetUseQuery()**

Retorna se a seção utiliza query para impressão.

Retorno          Se verdadeiro, a query é utilizada para impressão

#### **GetUserExp()**

Retorna o filtro de usuário.

Retorno          Filtro do usuário

#### **GetUserFilter()**

Retorna os filtros de usuário.

Retorno          Array contendo os filtros da seção

#### **GetWidth()**

Retorna a largura da seção.

Retorno          Largura da seção

#### **HeaderBreak()**

Retorna se o cabeçalho das células será impresso após uma quebra (TRBreak).

Retorno          Se verdadeiro, aponta que será impresso o cabeçalho

#### **HeaderSection()**

Retorna se o cabeçalho das células será impresso na quebra de seção.

Retorno          Se verdadeiro, aponta que será impresso o cabeçalho

#### **Hide()**

Desabilita a impressão da seção, porém todas as linhas serão processadas, somente não aparecerão no relatório.

#### **Init()**

Executa as quebras de seções, imprime cabeçalhos entre outras configurações do relatório.

Não é necessário executar o método Init se for utilizar o método Print, já que estes fazem o controle de inicialização e finalização da impressão.

#### **IniRow()**

Inicializa a linha posicionada nas seções.

#### **LineCount()**

Quantidade de linhas a serem impressas para cada registro da seção.

Retorno          Quantidade de linhas a serem impressas

#### **LoadCells(cTable,aCells,IDisableAll,IMarkAsUser)**

Carrega campos do Dicionário de Campos (SX3) como células da seção, respeitando nível, uso e contexto do campo.

cTable          Tabela que será utilizada para carregar os campos. Se não for informada será utilizada todas as tabelas informadas no método New

aCells	Array com nomes dos campos que serão carregados como células da seção
IDisableAll	Desabilita todas as células
IMarkAsUser	Apona que as células foram definidas pelos usuários

### **LoadOrder()**

Carrega índices do Dicionário de Índices (SIX) como ordem da seção.

### **IReadOnly()**

Retorna se o usuário pode personalizar informações da seção.

Objeto Caso verdadeiro, aponta que o usuário não pode alterar informações da seção

### **Name()**

Retorna o nome da seção.

Retorno Nome da seção

### **New(oParent,cTitle,uTable,aOrder,ILoadCells,ILoadOrder,uTotalText,ITotalInLine,IHeaderPage,IHeaderBreak,IPageBreak,ILineBreak,nLeftMargin,ILineStyle,nColSpace,IAutoSize,cCharSeparator,nLinesBefore,nCols,nClrBack,nClrFore,nPercentage)**

Método construtor da classe TRSection.

No parâmetro onde é informado as tabelas utilizadas pela seção, a primeira será a principal, sendo utilizada para o processamento pelo método Print. As outras tabelas serão utilizadas para que o usuário possa incluir os campos como célula.

O relacionamento entre as tabelas poderá ser feito pela query ou então pelo uso dos objetos da classe TRPosition.

Dos elementos do parâmetro onde são informadas as ordens utilizada pela seção serão criados os objetos da classe TROrder automaticamente, utilizando como índice da tabela principal a ordem do elemento no vetor.

oParent	Objeto da classe TReport ou TRSection que será o pai da classe TRSection
cTitle	Título da seção
uTable	Tipo Caracter: Tabela que será utilizada pela seção Tipo Array: Lista de tabelas que serão utilizadas pela seção
aOrder	Array contendo a descrição das ordens. Elemento: 1-Descrição, como por exemplo, Filial+Código
ILoadCells	Carrega os campos do Dicionário de Campos (SX3) das tabelas da seção como células
ILoadOrder	Carrega os índices do Dicionário de Índices (SIX)
uTotalText	Texto do totalizador da seção, podendo ser caracter ou bloco de código
ITotalInLine	Imprime as células em linha
IHeaderPage	Cabeçalho da seção no topo da página
IHeaderBreak	Imprime cabeçalho na quebra da seção
IPageBreak	Imprime cabeçalho da seção na quebra de página
ILineBreak	Quebra a linha na impressão quando as informações não caber na página
nLeftMargin	Tamanho da margem à esquerda da seção
ILineStyle	Imprime a seção em linha
nColSpace	Espaçamento entre as colunas
IAutoSize	Ajusta o tamanho das células para que caiba em uma página
cCharSeparator	Define o caracter que separa as informações na impressão em linha
nLinesBefore	Apona a quantidade de linhas a serem saltadas antes da impressão da seção
nCols	Quantidade de colunas a serem impressas
nClrBack	Cor de fundo das células da seção
nClrFore	Cor da fonte das células da seção

nPercentage    Tamanho da página a ser considerada na impressão em percentual

### **NoCell()**

Verifica se não existem células configuradas para a seção.

Retorno        Se verdadeiro, não existem células configuradas

### **OnPrintLine(bOnPrintLine)**

Permite efetuar tratamentos antes da impressão do registro da seção.

bOnPrintLine    Bloco de código com os tratamentos a serem realizados antes da impressão do registro da seção

### **Order(uOrder)**

Retorna objeto da classe TROrder (ordem).

uOrder        Tipo Caracter: Nickname da ordem

                Tipo Numérico: Índice do objeto segundo a ordem de criação dos componentes  
TROrder

Retorno        Objeto da classe TROrder

### **PageBreak()**

Retorna se salta a página na quebra de seção.

Retorno        Se verdadeiro, aponta que quebra página na seção

### **PageWidth()**

Retorna a largura da página.

Retorno        Largura da página

### **Parent()**

Retorna o objeto pai da classe TRSection.

Retorno        Objeto TRSection ou TReport

### **Print(IFromParent)**

Realiza a impressão baseada na tabela ou query principal, executando o método PrintLine para cada linha e o método Print de todas as seções filhas. Nas seções filhas, o posicionamento inicial pode ser baseado na query da seção pai (através do método SetParentQuery) ou através da regra informada pelo método SetRelation, junto com a regra informada pelo método SetParentFilter que controla o fim da impressão.

IFromParent    Aponta que é impressão de uma seção filha

### **PrintHeader(ICellPos,IVisible,cStyle,IExcel)**

Imprime o cabeçalho da seção.

ICellPos        Ajusta o cabeçalho das células

IVisible        Aponta que a seção está habilitada para impressão

cStyle         Utilizado internamente para geração do relatório em planilha

IExcel         Geração em planilha

### **PrintLine(IEvalPosition,IParamPage,IExcel)**

Imprime a linha baseado nas células existentes.

IEvalPosition   Força a atualização do conteúdo das células

IParamPage     Aponta que é a impressão da página de parâmetros



IExcel            Aponta que é geração em planilha

### **ResetCellPos(IResetAll)**

Define que as células deverão ser ajustadas para impressão.

IResetAll        Reavalia o posicionamento das células

### **Report()**

Retorna o objeto da classe TReport que a classe TRSection pertence.

Retorno        Objeto TReport

### **SetAutoSize(IAutoSize)**

Define que as células serão ajustadas automaticamente na seção.

IAutoSize        Ajuste automático das células

### **SetCellBorder(uBorder,nWeight,nColor,IHeader)**

Define a borda da seção.

uBorder        Tipo Caracter: "TOP","BOTTOM","LEFT","RIGHT","ALL"

                 Tipo Numérico: 1-Superior,2-Inferior,3-Esquerda,4-Direita,5-Todas

nWeight        Largura da borda

nColor         Cor da borda

IHeader        Borda do cabeçalho

Retorno        Objeto do tipo TRBorder

### **SetCharSeparator(cCharSeparator)**

Retorna o caracter que separa as Informações na impressão das células em linha.

cCharSeparator    Caracter utilizado para separar as informações

### **SetCols(nCols)**

Define a quantidade de colunas a serem impressas.

nCols    Quantidade de colunas a serem impressas

### **SetDynamicKey(cDynamicKey)**

Define a chave que identifica a seção na impressão dinâmica.

cDynamicKey    Chave que identifica a seção na impressão dinâmica

### **SetEdit(IEdit)**

Define se a seção poderá ser personalizada pelo usuário.

IEdit    Se verdadeiro, o usuário poderá personalizar a seção

### **SetEditCell(IEditCell)**

Define se o usuário poderá personalizar as células da seção.

IEditCell        Se verdadeiro, o usuário poderá personalizar as células

### **SetFilter(cFilter,cIndexKey,cOrdem,cAlias,nIdxOrder)**

Define um filtro para a tabela principal da seção.

Se não informado o parâmetro com a chave de índice, será executado SET FILTER TO com o primeiro parâmetro, senão será criado um índice com filtro (IndRegua).

cFilter        Expressão do filtro no format ADVPL

cIndexKey     Chave de índice

cOrdem            Ordem para a criação do índice  
cAlias            Tabela a ser filtrada  
nIdxOrder        Ordem no Dicionário de Índices (SIX)

### **SetHeaderBreak(IHeaderBreak)**

Define se imprime cabeçalho das células após uma quebra (TRBreak).

IHeaderBreak    Se verdadeiro, aponta que salta página na quebra

### **SetHeaderPage(IHeaderPage)**

Define que imprime cabeçalho das células no topo da página.

IHeaderPage    Se verdadeiro, aponta que imprime o cabeçalho no topo da página

### **SetHeaderSection(IHeaderSection)**

Define que imprime cabeçalho das células na quebra de seção.

IHeaderSection        Se verdadeiro, aponta que imprime cabeçalho na quebra da seção

### **SetIdxOrder(nIdxOrder)**

Define uma ordem de índice para a tabela principal.

nIdxOrder        Ordem de índice para a tabela principal

### **SetLineBreak(ILineBreak)**

Define que a impressão poderá ocorrer em uma ou mais linhas no caso das colunas excederem o tamanho da página.

ILineBreak        Se verdadeiro, imprime em uma ou mais linhas

### **SetLineCondition(bLineCondition)**

Permite validar a impressão do registro.

bLineCondition    Bloco de código utilizado na validação

### **SetLineStyle(ILineStyle)**

Define se imprime as células da seção em linhas.

ILineStyle        Imprime as células da seção em linhas

### **SetLinesBefore(nLinesBefore)**

Define a quantidade de linhas que serão saltadas antes da impressão da seção.

nLinesBefore    Quantidade de linhas

### **SetName()**

Compatibilidade – Não utilizado.

### **SetNoFilter(cAlias)**

Define que a tabela não poderá receber filtros de usuário.

cAlias            Tabela a ser considerada

### **SetUseQuery(ICellUseQuery)**

Define que a seção utilize query na impressão.

ICellUseQuery    Se verdadeiro, utiliza query na impressão

### **SetOrder(nOrder)**

Define a ordem (TROrder) que será utilizada pela seção.

nOrder            Ordem da seção

### **SetPageBreak(IPageBreak)**

Define se salta a página na quebra de seção.

IPageBreak Se verdadeiro, aponta que salta página na quebra de seção

### **SetParentFilter(bFilter,bParam)**

Define a regra de saída do loop de impressão das seções filhas.

bFilter Bloco de código com a regra para saída do loop

bParam Bloco de código com a expressão que retorna o valor que é enviado como parâmetro para a regra de saída do loop

### **SetParentQuery()**

Define que a seção filha utiliza a query da seção pai na impressão da seção.

### **SetParentRecno(IParentRecno)**

Define se a seção utiliza o registro da seção pai.

IParentRecno Utiliza o registro da seção pai

### **SetPercentage(nPercentage)**

Define o tamanho da página a ser considerada na impressão em percentual.

nPercentage Tamanho da página a ser considerada na impressão em percentual

### **SetPrintLayout()**

Compatibilidade – Não utilizado.

### **SetQuery(cAlias,cQuery,IChangeQuery,aParam,aTCFields)**

Aplica os tratamentos necessários para execução das queries, tais como, adicionar células de usuários, aplicação de filtros entre outros.

cAlias Tabela principal da query

cQuery Query da seção

IChangeQuery Se verdadeiro, realiza tratamentos para outros bancos de dados

aParam Parâmetros do tipo Range a serem utilizados no filtro da query

aTCFields Lista de campos com tipo de dados diferente de character que devem ser tratados para apresentar os resultados na query. Elementos: 1-Campo, 2-Tipo, 3- Tamanho e 4-Decimal

### **SetReadOnly(IReadOnly)**

Define que o usuário não poderá alterar informações da seção, ou seja, não poderá remover as células pré-definidas.

IReadOnly Se verdadeiro, aponta que o usuário não poderá alterar Informações da seção

### **SetRelation(bFormula,cAlias,uOrder,ISeek)**

Define a fórmula de relacionamento de uma seção filha com sua seção pai, caso não utilize a query da seção pai através do método SetParentQuery.

bFormula Bloco de código com a expressão para relacionamento entre as seções

cAlias Tabela utilizada pela fórmula

uOrder Ordem utilizada na tabela

Tipo Character: Nickname da ordem de índice

Tipo Numérico: Ordem do índice

ISeek Indica se será executado DbSeek com o conteúdo da fórmula

### **SetRow(nRow)**

Define a linha de impressão.

nRow Linha atual

### **SetWidth(nWidth)**

Define a largura da seção.

nWidth Largura da seção

**Show()**

Habilita a impressão da seção.

**UseFilter()**

Retorna se existem células personalizadas pelo usuário.

Retorno            Se verdadeiro, existe célula personalizada

**Visible()**

Retorna se a seção esta habilitada para impressão.

Retorno            Caso verdadeiro, a seção está habilitada

**XmlLoad(oXml)**

Carrega uma seção do layout personalizado pelo usuário.

oXml                Objeto do tipo XML com as personalizações da seção

**XmlSave()**

Salva a seção no layout personalizado pelo usuário.

## TRCELL

Célula de impressão de uma seção (TRSection) de um relatório que utiliza a classe TReport.

Esta classe herda as propriedades e métodos da classe TRSECTION.

### Propriedades

aCBox	Array com os possíveis textos a serem impressos na célula. Elemento: 1- Conteúdo. Exemplo: 1=Sim.
aFormatCond	Array com as condições do usuário para impressão de forma variável da cor da célula: Elementos: 1-Condição, 2-Cor de fundo e 3-Cor da fonte.
bCanPrint	Bloco de código que valida a impressão da célula.
bCellBlock	Bloco de código que retornará o conteúdo de impressão da célula.
cFormula	Fórmula para impressão da célula.
cOrder	Ordem de impressão da célula.
cPicture	Máscara da célula.
cRealFormula	Fórmula em forma de expressão ADVPL.
cType	Tipo de dado da célula.
cUserFunction	Tipo de acumulador: "MIN" – Menor valor, "MAX" – Maior valor, "SUM" – Soma, "COUNT" – Contador ou "AVERAGE" – Média.
cXlsHStyle	Estilo do cabeçalho padrão utilizado na geração da planilha.
cXlsStyle	Estilo utilizado na geração da planilha.
IBold	Aponta que a célula será impressa em negrito.
IHeaderSize	Aponta que o tamanho a ser considerado na impressão é do cabeçalho.
IPixelSize	Aponta que o tamanho da célula está calculada em pixel.
IPrintCell	Aponta que a célula está habilitada para impressão.
ICellBreak	Compatibilidade – Não utilizado.
IUserEnabled	Aponta que a célula foi habilitada para impressão pelo usuário.
IUserField	Aponta que a célula foi personalizada pelo usuário.
IUserAccess	Aponta que o usuário tem acesso a impressão desta célula, no caso de falso, o usuário não possui o nível de campo ou acesso definido no cadastro de usuários.
nAlign	Alinhamento da célula. 1-Esquerda, 2-Center ou 3-Direita.
nAutoWidth	Largura gerada automaticamente quando excedida a largura da página.
nCellPixel	Largura da célula em pixel.
nHeaderAlign	Alinhamento do cabeçalho 1-Esquerda, 2-Center ou 3-Direita.
nHeaderPixel	Tamanho do cabeçalho da célula em pixel.
nHeaderSize	Tamanho do cabeçalho da célula.
nLineStart	Aponta a primeira linha da célula a ser impressa no caso de quebra de linha.
nNegative	Número do item de sinal negativo no array aNegative.
nPixelSize	Tamanho da célula em pixel.
nRowDiff	Quantidade de linhas a serem consideradas na impressão das bordas.
nSize	Tamanho da célula.
nType	Tipo da célula. 1-Celula, 2-Formula, 3-Acumulador ou 4-Célula de usuário.
nUserValue	Auxilia no controle do totalizador do tipo MAX e MIN.
nUserCount	Contador de itens impressos quando utilizados totalizadores.
nLevel	Nível de campo da célula.
uValue	Valor da célula a ser impresso.
uPrint	Conteúdo da célula a ser impresso.
oFontBody	Objeto TFont com Informações da fonte da célula.

### MÉTODOS

#### CanPrint()

Valida se a célula pode ser impressa.

**Col()**

Retorna a coluna da célula.

**ColPos()**

Retorna a coluna que a célula será impressa pelo método PrintLine da seção que a célula pertence.

Retorno          Coluna da célula na impressão

**CellBreak()**

Compatibilidade – Não utilizado.

**ClrBack(IObject)**

Retorna a cor de fundo do relatório.

IObject          Aponta que deve ser retornado o objeto TBrush

Retorno          Caso o parâmetro IObjeto seja verdadeiro o retorno será o objeto TBrush, caso contrário será o número da cor RGB.

**ClrFore(IPrintHeader)**

Retorna a cor de fonte do relatório.

IPrintHeader    Aponta impressão do cabeçalho da célula

Retorno          Número da cor RGB

**Disable()**

Desabilita a impressão da célula.

**Enable()**

Habilita a impressão da célula.

**Enabled()**

Retorna se a célula está habilitada para impressão.

Retorno          Se verdadeiro, a célula está habilitada

**EvalFunction()**

Atualiza o conteúdo dos totalizadores da seção.

**Execute(IPrintLayout)**

Atualiza o conteúdo da célula a ser impresso.

IPrintLayout    Aponta visualização de layout

Retorno          Conteúdo a ser impresso

**GetCBox()**

Retorna o conteúdo da lista de dado a ser impresso. Exemplo: 1=Sim ou 2=Não.

Retorno          Conteúdo da lista

**GetCellSize()**

Retorna o tamanho da célula.

Retorno          Tamanho da célula

**GetCellWidth()**

Retorna a largura da célula.

Retorno      Largura da célula

**GetFieldInfo(cField)**

Carrega informações (título, picture, tamanho, etc) do campo baseado no Dicionário de Campos (SX3).

cField      Campo a ser consultado no dicionário

Retorno      Se verdadeiro, conseguiu coletar informações do dicionário

**GetHeaderSize()**

Retorna o tamanho do cabeçalho da célula.

Retorno      Tamanho do cabeçalho

**GetHeaderWidth()**

Retorna a largura do cabeçalho da célula.

Retorno      Largura do cabeçalho

**GetSize()**

Retorna o tamanho da célula considerando o maior tamanho entre a célula e o cabeçalho da célula.

Retorno      Tamanho da célula

**GetText()**

Retorna o texto que será impresso.

Retorno      Conteúdo da célula a ser impresso

**GetValue()**

Retorna o valor a ser impresso na célula.

Retorno      Valor da célula

**GetWidth(IAutoWidth)**

Retorna a largura da célula.

IAutoWidth      Largura automática

Retorno      Largura da célula

**Hide()**

Desabilita a impressão da célula, porém calcula o posicionamento da célula.

**LineCount(IHeader)**

Quantidade de linhas a serem impressas para a célula.

IHeader      Se verdadeiro, verifica as linhas do cabeçalho

Retorno      Quantidade de linhas

**New(oParent,cName,cAlias,cTitle,cPicture,nSize,IPixel,bBlock,cAlign,ILineBreak,cHeaderAlign,ICellBreak,nColSpace,IAutoSize,nClrBack,nClrFore,IBold)**

Método construtor da classe TRCell.

Se o nome da célula informada for encontrada no Dicionário de Campos (SX3), as informações do campo serão carregadas para a célula, respeitando os parâmetros de título, picture e tamanho. Dessa forma o relatório sempre estará atualizado com as informações do Dicionário de Campos (SX3).

O nome da célula será utilizado junto ao alias informado pelo parâmetro como conteúdo para a impressão da célula, como por exemplo a impressão do campo A1\_COD da tabela SA1.

Se for informado o parâmetro com o bloco de código, o retorno deste será utilizado como conteúdo para impressão da célula, com a picture, tamanho e título definidos para a célula. Assim sendo, é possível criar células calculadas com formato de impressão baseada no Dicionário de Campos (SX3)

oParent	Objeto da classe TRSection que a célula pertence
cName	Nome da célula
cAlias	Tabela utilizada pela célula
cTitle	Título da célula
cPicture	Máscara da célula
nSize	Tamanho da célula
IPixel	Aponta se o tamanho foi informado em pixel
bBlock	Bloco de código com o retorno do campo
cAlign	Alinhamento da célula. "LEFT", "RIGHT" e "CENTER"
ILineBreak	Quebra linha se o conteúdo estourar o tamanho do campo
cHeaderAlign	Alinhamento do cabeçalho da célula. "LEFT", "RIGHT" e "CENTER"
ICellBreak	Compatibilidade – Não utilizado
nColSpace	Espaçamento entre as células
IAutoSize	Ajusta o tamanho da célula com base no tamanho da página e as Informações impressas
nClrBack	Cor de fundo da célula
nClrFore	Cor da fonte da célula
IBold	Imprime a fonte em negrito

#### **Picture()**

Retorna a máscara de impressão da célula.

Retorno      Máscara de impressão

#### **Print(ICanPrint,nXlsCol,IExcel)**

Imprime o conteúdo da célula.

ICanPrint	Valida impressão da célula
nXlsCol	Número da coluna na geração em planilha
IExcel	Geração em planilha

#### **PrintHeader(nSkipLine,IHeaderWidth,cXlsHStyle,IExcel)**

Imprime o cabeçalho da célula.

nSkipLine	Compatibilidade – Não utilizado
IHeaderWidth	Considera a largura do cabeçalho na impressão
cXlsHStyle	Estilo do cabeçalho na geração em planilha
IExcel	Geração em planilha

#### **ResetWidth()**

Inicializa o tamanho e largura da célula.

#### **ResetLineStart()**

Inicializa a linha que indica a primeira linha da célula a ser impressa no caso de quebra de linha.



**Say(cText,nAlign,nSayWidth,IPrintHeader)**

Imprime um texto na célula ou o conteúdo da célula.

cText            Texto que será impresso  
nAlign           Alinhamento do texto. 1-Esquerda, 2-Centro ou 3-Direita  
nSayWidth       Largura do texto  
IPrintHeader    Impressão do cabeçalho

**SetAlign(uAlign)**

Define o alinhamento da célula na impressão.

uAlign           Tipo Caracter: "LEFT" – esquerda, "RIGHT" – direita e "CENTER" - centro  
                    Tipo Numérico: 1 – esquerda, 2 – centro e 3 - direita

**SetAutoWidth(nAutoWidth)**

Apona a largura da célula gerada automaticamente pelo cálculo de posicionamento das células que atingirem a largura máxima da página.

nAutoWidth      Largura da célula

**SetBlock(bBlock)**

Define o bloco de código que retornará o conteúdo de impressão da célula.

Definindo o bloco de código para a célula, esta não utilizará mais o nome mais alias para retornar o conteúdo de impressão.

bBlock           Bloco de código que retorna o conteúdo

**SetBorder(uBorder,nWeight,nColor,IHeader)**

Define as bordas da célula.

uBorder           Tipo Caracter: "TOP", "BOTTOM", "LEFT", "RIGHT", "ALL"  
                    Tipo Numérico: 1-Superior, 2-Inferior, 3-Esquerda, 4-Direita, 5-Todas  
nWeight           Largura da borda  
nColor            Cor da borda  
IHeader           Aponta se é borda de cabeçalho

Retorno           Objeto do tipo TRBorder

**SetCanPrint(bCanPrint)**

Define o bloco de código que será utilizado para validar se a célula poderá ser impressa.

bCanPrint        Bloco de código com a validação da célula

**SetCBox(cBox)**

Define que a célula irá imprimir a descrição do item de um campo que tenha o formato ComboBox, utilizado no Dicionário de Campos (SX3).

cBox              O formato do parâmetro é o mesmo utilizado no Dicionário de Campos (SX3), como no exemplo abaixo:  
                    oCell:SetCBox('1=Sim;2=Não').  
                    Na impressão será utilizada a descrição do item. No exemplo acima se o conteúdo da célula for '1', será impresso 'Sim', sendo que o método já irá calcular o tamanho da célula baseado na maior descrição informada

**SetCellBreak()**

Compatibilidade – Não utilizado.

**SetHeaderAlign(uAlign)**

Define o alinhamento do cabeçalho da célula na impressão.

uAlign            Tipo Caracter: "LEFT" – esquerda, "RIGHT" – direita e "CENTER" - centro  
                    Tipo Numérico: 1 – esquerda, 2 – centro e 3 - direita

### **SetNegative(cNegative)**

Define o sinal de negativo utilizado na impressão.

cNegative        Tipo Caracter: "PARENTHESES" ( ) ou "SIGNAL" –

### **SetPicture(cPicture)**

Define a máscara de impressão da célula.

cPicture         Máscara da célula

### **SetPrintCell(IPrintCell)**

Define se a célula será impressa.

IPrintCell       Se verdadeiro, imprime a célula

### **SetRow(nRow)**

Quantidade de linhas para impressão das bordas.

nRow             Quantidade de linhas

### **SetRowDiff(nRowDiff)**

Quantidade de linhas a serem consideradas na impressão das bordas.

nRowDiff        Quantidade de linhas

### **SetSize(nSize,IPixel)**

Define o tamanho da célula.

nSize            Tamanho da célula

IPixel            Aponta se o tamanho é calculado em pixel

### **SetTitle(cTitle)**

Define o título da célula.

cTitle            Título da célula

### **SetType(cType)**

Define o tipo de dado da célula.

cType            Tipo de dado

### **SetValue(uValue)**

Define um valor constante para a célula.

uValue           Valor constante

Definindo um valor constante para a célula, esta não utilizará o nome mais o alias para impressão do conteúdo, ou seja, a célula passará a se comportar como um variável de um programa.

Se o parâmetro não for informado (Nulo), então a célula deixará de usar o valor constante, voltando a imprimir seu conteúdo utilizando nome mais alias.

### **XmlLoad(oXml)**

Carrega uma célula do layout personalizado pelo usuário.

oXml            Objeto do tipo XML com as personalizações da célula

### **XmlSave()**

Salva a célula no layout personalizado pelo usuário.

## TRPOSITION

Célula de posicionamento de uma seção de um relatório que utiliza a classe TReport.

As células de posicionamento são executadas em toda execução do método PrintLine da seção que elas pertencem, para posicionar as tabelas secundárias da seção que o usuário pode utilizar para incluir células baseadas em seus campos.

### Propriedades

cAlias	Tabela que será utilizada para posicionamento.
cClassName	Nome da classe. (TRPOSITION).
cNickName	Nickname da ordem do índice da tabela.
ISseek	Se verdadeiro, Executar o DbSeek com o conteúdo retornado pela fórmula.
nOrder	Ordem a ser utilizada na pesquisa.
uFormula	Fórmula de posicionamento.
uValue	Conteúdo retornado após a execução da fórmula.
oReport	Objeto TReport.
oParent	Objeto da classe TRSection que a célula pertence.

### MÉTODOS

#### ClassName()

Retorna o nome da classe. Exemplo: TRPOSITION.

#### Execute()

Executa a fórmula de posicionamento.

#### New(oParent,cAlias,uOrder,uFormula,ISseek)

Método construtor da classe TRPOSITION.

oParent	Objeto da classe TRSection que a célula pertence
cAlias	Tabela que será utilizada para posicionamento
uOrder	Tipo Caracter: Nickname da ordem do índice da tabela Tipo Numérico: Ordem do índice da tabela
uFormula	Fórmula de posicionamento Tipo Caracter: Expressão ADVPL para macro execução Tipo Bloco de Código: Bloco de Código com a expressão ADVPL para execução
ISseek	Se verdadeiro, Executar o DbSeek com o conteúdo retornado pela fórmula
Retorno	Objeto do tipo TRPosition

#### SetFormula(uFormula,ISseek)

Define a fórmula de posicionamento.

uFormula	Fórmula de posicionamento Tipo Caracter: Expressão ADVPL para macro execução Tipo Bloco de Código: Bloco de Código com a expressão ADVPL para execução
ISseek	Se verdadeiro, Executar o DbSeek com o conteúdo retornado pela fórmula

#### SetNickName(cNickName)

Define a ordem do índice pelo seu nickname.

cNickName	NickName da ordem de índice da tabela
-----------	---------------------------------------

#### SetOrder(nOrder)

Define a ordem do índice para o posicionamento.

nOrder            Ordem do índice

### **SetTable(cAlias)**

Define a tabela que será utilizada pela fórmula para o posicionamento.

cAlias            Tabela que será utilizada na fórmula

## TRFUNCTION

Totalizador de uma quebra, seção ou relatório que utiliza a classe TReport.

Um totalizador pode executar uma das seguintes funções abaixo, utilizando como referência uma célula da seção ou o retorno de uma fórmula definida para ele:

SUM	Somar
COUNT	Contar
MAX	Valor máximo
MIN	Valor mínimo
AVERAGE	Valor médio
ONPRINT	Valor atual
TIMESUM	Somar horas
TIMEAVERAGE	Valor medio de horas
TIMESUB	Subtrai horas

Na criação do totalizador sempre é informado uma seção (TRSection) a qual ele pertence, e automaticamente o totalizador será incluído no relatório (TReport) que a seção pertence.

Esta classe herda as propriedades e métodos da classe TRCELL.

### Propriedades

bCondition	Bloco de código com a condição de atualização dos valores do totalizador.
bOnPrint	Bloco de código para tratamentos antes da impressão do totalizador.
cFunction	Função que será utilizada pelo totalizador. Exemplo: SUM, COUNT, MAX, MIN.
ICollection	Se verdadeiro, aponta que o totalizador é do tipo Collection.
IEndPage	Se verdadeiro, aponta que o totalizador será impresso no final da página.
IEndReport	Se verdadeiro, aponta que o totalizador será impresso no final do relatório.
IEndSection	Se verdadeiro, aponta que o totalizador será impresso no final da seção.
IPageValue	Se verdadeiro, aponta que é impressão do total da página.
IPrintLayout	Se verdadeiro, aponta que é visualização do layout.
IReportValue	Se verdadeiro, aponta que é impressão do total geral.
ISectionValue	Se verdadeiro, aponta que é impressão do total da seção.
IPrintCollection	Se verdadeiro, aponta que é impressão de totalizador do tipo Collection.
nCount	Contador de registros impressos.
nCountPage	Contador de registros impressos para a página.
nCountReport	Contador geral de registros impressos.
nCountSection	Contador de registros impressos para a seção.
oCell	Objeto da classe TRCell que o totalizador se refere.
oTotal	Objeto da classe TRFunction ou TRCollection.
uFormula	Tipo Caracter: Expressão ADVPL para macro execução. Tipo Bloco de código: Bloco de Código com a expressão ADVPL para execução.
uLastValue	Último valor atualizado no totalizador.
uPage	Total acumulado por página.
uReport	Total acumulado geral do relatório.
uSection	Total acumulado por seção.

### MÉTODOS

#### ClrBack(IObject)

Retorna a cor de fundo do relatório.

IObject Aponta que deve ser retornado o objeto TBrush

**Retorno** Caso o parâmetro IObjeto seja verdadeiro o retorno será o objeto TBrush, caso contrário será o número da cor RGB.

### **ClrFore()**

Retorna a cor de fonte do relatório.

**IPrintHeader** Aponta impressão do cabeçalho da célula

**Retorno** Número da cor RGB

### **Col()**

Retorna a posição da coluna do totalizador.

**Retorno** Posição da coluna do totalizador

### **Enabled()**

Retorna se o totalizador está habilitado.

**Retorno** Se verdadeiro, aponta que o totalizador está habilitado

### **EndPage()**

Retorna se o totalizador será impresso no final de cada página.

**Retorno** Se verdadeiro, o totalizador será impresso

### **EndReport()**

Retorna se o totalizador será impresso no final do relatório.

**Retorno** Se verdadeiro, o totalizador será impresso

### **EndSection()**

Retorna se o totalizador será impresso na quebra de seção.

**Retorno** Se verdadeiro, o totalizador será impresso

### **EvalFunction()**

Utilizado pelo método Execute(), ele executa a função do totalizador, utilizando a fórmula definida pelo método New ou pelo método SetFormula.

### **Execute()**

Executa a função do totalizador, utilizando a fórmula definida pelo método New ou pelo método SetFormula.

O método Execute é executado para cada linha impressa pelo método PrintLine da seção que o totalizador pertence.

### **Formula()**

Retorna a fórmula do totalizador.

**Retorno** Tipo Caracter: Expressão ADVPL para macro execução  
Tipo Bloco de código: Bloco de Código com a expressão ADVPL para execução

### **FullID()**

Retorna o identificador do totalizador mais o identificador da célula.

Retorno Identificador do totalizador mais o identificador da célula

#### **GetFunction(uFunction)**

Retorna a função utilizada pelo totalizador. Exemplo: SUM, COUNT, MAX, MIN, AVERAGE.

uFunction Identificador do totalizador

Retorno Caso informado o identificador do totalizador o retorno será o objeto, caso contrário será a função

#### **GetValue()**

Retorna o valor do totalizador.

Retorno Valor do totalizador

#### **GetLastValue()**

Retorno o último valor atualizado no totalizador.

Retorno Último valor atualizado no totalizador

#### **GetWidth(IAutoWidth)**

Retorna a largura da célula.

IAutoWidth Aponta que a largura será calculada automaticamente

Retorno Largura da célula do totalizador

#### **GetPrintCollection()**

Retorna se é impressão de totalizador do tipo Collection.

Retorno Se verdadeiro, aponta impressão de totalizador do tipo Collection

#### **New(oCell,cName,cFunction,oBreak,cTitle,cPicture,uFormula,IEndSection,IEndReport,IEndPage,oParent,bCondition,IDisable,bCanPrint)**

Método construtor da classe TRFunction.

oCell	Objeto da classe TRCell que o totalizador se refere
cName	Identificação do totalizador
cFunction	Função que será utilizada pelo totalizador. Exemplo: SUM, COUNT, MAX, MIN
oBreak	Objeto da classe TRBreak que define em qual quebra o totalizador será impresso
cTitle	Título do totalizador. Se não informado será utilizado o título da célula que o totalizador se refere
cPicture	Máscara de impressão do totalizador. Se não informado será utilizado a máscara da célula que o totalizador se refere
uFormula	Tipo Caracter: Expressão ADVPL para macro execução Tipo Bloco de código: Bloco de Código com a expressão ADVPL para execução
IEndSection	Se verdadeiro. Indica se totalizador será impresso na quebra de seção
IEndReport	Se verdadeiro. Indica se totalizador será impresso no final do relatório
IEndPage	Se verdadeiro. Indica se totalizador será impresso no final de cada página
oParent	Objeto da classe TRSection que o totalizador se refere
bCondition	Bloco de código com a condição de atualização dos valores do totalizador
IDisable	Se verdadeiro. Define que não irá atualizar os valores do totalizador
bCanPrint	Bloco de código com a condição de impressão dos valores do totalizador

#### **PageValue()**

Retorna o valor atual do totalizador para a página.

Retorno          Valor atual da página

**Print(uValue)**

Imprime o total atual.

uValue          Se informado, define o valor a ser impresso e altera o valor do total, caso contrário imprime o valor atual

**PrintPage()**

Imprime o totalizador da página.

**PrintReport()**

Imprime o totalizador geral do relatório.

**PrintSection()**

Imprime o totalizador da seção.

**ReportValue()**

Retorna o valor geral do totalizador para o relatório.

Retorno          Valor geral do relatório

**Reset()**

Reinicia valor do totalizador.

**ResetPage()**

Reinicia valor do totalizador da página.

**ResetReport()**

Reinicia valor do totalizador geral do relatório.

**ResetSection()**

Reinicia valor do totalizador da seção.

**SectionValue()**

Retorna o valor atual do totalizador para a seção.

Retorno          Valor atual da seção

**SetBreak(oBreak)**

Define a quebra (TRBreak) que o totalizador será impresso.

oBreak          Objeto TRBreak

**SetCollection(ICollection)**

Define que o totalizador é do tipo collection.

ICollection      Se verdadeiro, aponta que o totalizador é do tipo collection

**SetCondition(bCondition)**

Define o bloco de código com a condição de atualização dos valores do totalizador.

bCondition      Bloco de código com a condição de atualização dos valores do totalizador

**SetEndPage(IEndPage)**

Define se o totalizador será impresso na quebra de cada página.

IEndPage        Se verdadeiro, aponta que imprime o totalizador



**SetEndReport(IEndReport)**

Define se o totalizador será impresso no final do relatório.

IEndReport Se verdadeiro, aponta que imprime o totalizador

**SetEndSection(IEndSection)**

Define se o totalizador será impresso na quebra de seção.

IEndSection Se verdadeiro, aponta que imprime o totalizador

**SetFormula(uFormula)**

Define a fórmula utilizada pelo totalizador no lugar de utilizar o conteúdo da célula que ele se refere.

uFormula Tipo Caracter: Expressão ADVPL para macro execução

Tipo Bloco de código: Bloco de código com a expressão ADVPL para execução

**SetFunction(cFunction)**

Define a função que será utilizada pelo totalizador. Exemplo: SUM, COUNT, MAX, MIN.

cFunction Função do totalizador

**SetTotalInLine(ITotalInLine)**

Define que a impressão dos totalizadores será em linha.

ITotalInLine Se verdadeiro, imprime os totalizadores em linha

**SetPrintCollection(IPrintCollection)**

Define que é impressão de collections.

IPrintCollection Se verdadeiro, aponta que é impressão de collections

## TRBREAK

Quebra de uma seção de um relatório que utiliza a classe TReport.

A classe TRBreak é utilizada quando há a necessidade de impressão dos totalizadores da seção antes do final da seção ou do relatório, dependendo da regra de quebra informada.

Esta classe herda as propriedades e métodos da classe TRSECTION.

### Propriedades

bOnBreak	Bloco de código com tratamentos a serem realizados antes da quebra.
bOnPrintTotal	Bloco de código com tratamentos a serem realizados após a quebra.
uBreak	Regra para quebra. Tipo Objeto: Objeto da classe TRCell. Tipo Caracter: Expressão ADVPL para macro execução. Tipo Bloco de Código: Bloco de código com expressão que será executada.
uContent	Último valor atualizado no controle da quebra.

### MÉTODOS

#### Execute(IForce)

Executa a regra e indica se houve a quebra.

O método Execute é executado para cada linha impressa pelo método PrintLine da seção que a quebra pertence.

IForce            Força a execução da quebra

Retorno           Se verdadeiro, aponta que houve quebra

#### GetLastValue()

Retorno o último valor atualizado no controle da quebra.

Retorna           Último valor atualizado

#### New(oParent,uBreak,uTitle,ITotalInLine,cName,IPageBreak)

Método construtor da classe TRBreak.

oParent	Objeto da classe TRSection que a quebra pertence
uBreak	Regra para quebra Tipo Objeto: Objeto da classe TRCell Tipo Caracter: Expressão ADVPL para macro execução Tipo Bloco de Código: Bloco de código com expressão que será executada
uTitle	Título da quebra
ITotalInLine	Se verdadeiro, aponta que os totalizadores serão impressos em linha
cName	Nome e identificador da quebra
IPageBreak	Se verdadeiro, aponta salta de página após a quebra

Retorno           Objeto da classe TRBreak

#### OnBreak(bOnBreak)

Define o bloco de código que será executado antes da impressão da quebra.

bOnBreak        Bloco de código com tratamentos a serem realizados antes na quebra

#### OnPrintTotal(bOnPrintTotal)

Define o bloco de código que será executado após a impressão da quebra.

**bOnPrintTotal** Bloco de código com tratamentos a serem realizados após a quebra

**Print()**

Executa a impressão da quebra.

**PrintTotal()**

Imprime a quebra e executa os tratamentos definidos no método OnPrintTotal.

**ResetBreak()**

Inicializa o valor utilizado para controlar a quebra.

**SetBreak(uBreak)**

Define a regra para a quebra.

uBreak           Regra para quebra  
                  Tipo Objeto: Objeto da classe TRCell  
                  Tipo Caracter: Expressão ADVPL para macro execução  
                  Tipo Bloco de Código: Bloco de código com expressão que será executada

**SetTitle(uTitle)**

Define o título que será impresso antes da impressão dos totalizadores.

uTitle           Texto a ser impresso no formato caracter ou bloco de código

**Title()**

Retorna o título que será impresso antes da impressão dos totalizadores.

Retorno        Título da quebra

## **TRORDER**

Ordena a seção de um relatório que utiliza a classe TReport.

Esta classe herda as propriedades e métodos da classe TRPOSITION.

### **Propriedades**

cTitle            Título da ordem

### **MÉTODOS**

#### **Execute()**

Ordena a tabela.

#### **GetOrder()**

Retorna a ordem da tabela.

Retorno            Posição da ordem no dicionário de índices - SIX

#### **New(oParent,uOrder,cTitle,cAlias)**

Método construtor da classe TROrder.

oParent            Objeto da classe TRSection que a ordem pertence  
uOrder            Tipo Caracter: NickName da ordem no dicionário de índices – SIX  
                      Tipo Numérico: Posição da ordem no dicionário de índices - SIX  
cTitle            Título da ordem  
cAlias            Tabela que será ordenada

Retorno            Objeto da classe TROrder

#### **SetTitle(cTitle)**

Define o título da ordem.

cTitle    Título da ordem

#### **Title()**

Retorna o título da ordem.

Retorno            Título da ordem

## TRRELATION

Relacionamento entre as seções que utiliza a classe TReport.

Esta classe herda as propriedades e métodos da classe TRPOSITION.

## MÉTODOS

### New(oParent)

Método construtor da classe TRRelation.

oParent                      Objeto da classe TRSection que o relacionamento pertence

### Execute()

Executa fórmula de relacionamento.

### SetFormula(uFormula,ISeek)

Define a fórmula de posicionamento.

uFormula              Fórmula de posicionamento  
                            Tipo Caracter: Expressão ADVPL para macro execução  
                            Tipo Bloco de Código: Bloco de Código com a expressão ADVPL para execução  
ISeek                    Se verdadeiro, Executar o DbSeek com o conteúdo retornado pela fórmula

### SetNickName(cNickName)

Define a ordem do índice pelo seu nickname.

cNickName      NickName da ordem de índice da tabela

### SetOrder(nOrder)

Define a ordem do índice para o posicionamento.

nOrder              Ordem do índice

### SetTable(cAlias)

Define a tabela que será utilizada pela fórmula para o posicionamento.

cAlias              Tabela que será utilizada na fórmula

## TRPAGE

Definição da página para relatório que utiliza a classe TReport.

Esta classe herda as propriedades e métodos da classe TREPORT.

### Propriedades

bPageFooter	Bloco de código utilizado para imprimir Informações customizadas, no qual, deverá ser utilizado métodos do TReport para impressão. Exemplo: Say, PrintText.
lEndPage	Aponta que deverá quebrar página.
lFirstPage	Aponta que é primeira página do relatório.
lFooterBefore	Aponta que o rodapé será impresso antes dos totalizadores.
lFooterDiff	Aponta que é considerado a altura da linha no rodapé para cálculo do altura da página.
lLandscape	Aponta a orientação de página do relatório como paisagem.
lPageDiff	Aponta que é considerado o tamanho da borda e rodapé para cálculo da altura da página.
lPageFooter	Aponta que está sendo impresso o rodapé da página.
lPageTotal	Aponta que está sendo impresso o total da página.
lPortrait	Aponta a orientação de página do relatório como retrato.
lTotalAfter	Aponta que o totalizador será impresso após a impressão do rodapé.
lTotalDiff	Aponta que será considerada a quantidade de linhas a serem impressos no rodapé no cálculo da altura da página.
nFooterDiff	Quantidade de linhas a serem impressas no rodapé considerando a altura da linha.
nHorzRes	Resolução horizontal da impressora configurada.
nLinesFooter	Quantidade de linhas reservadas para impressão do rodapé.
nPage	Página atual.
nPaperSize	Tamanho da página.
nTotalDiff	Quantidade de linhas a serem impressas no rodapé considerando o tamanho da linha e bordas.
nWidth	Largura da página.
nVertRes	Resolução vertical da impressora configurada.
oParent	Objeto da classe TRSection que a página pertence.
oReport	Objeto da classe TREPORT que a página pertence.

### MÉTODOS

#### EndPage(lFooter)

Imprime os totalizadores, rodapé e salta a página.

lFooter Aponta que o rodapé será impresso

#### FooterDiff()

Retorna a quantidade de linhas do rodapé considerando a altura da linha.

Retorno Quantidade de linhas

#### IsLandscape()

Aponta que a orientação do relatório é paisagem.

Retorno Se verdadeiro, é orientação paisagem

#### IsPortrait()

Aponta que a orientação do relatório é retrato.

Retorno Se verdadeiro, é orientação retrato

#### New(oParent,uTotalText,lTotalInLine)

Método construtor da classe TRPage.

oParent	Objeto da classe TRSection que a página pertence
uTotalText	Texto do totalizador da página
ITotalLnLine	Imprime os totalizadores em linhas

Retorno Objeto da classe TRPage

### **Page()**

Retorna a página atual.

Retorno Página atual

### **PageDiff()**

Retorna a quantidade de linhas utilizadas na impressão do rodapé.

Retorno Quantidade de linhas do rodapé

### **PageHeight()**

Retorna a altura da página.

Retorno Altura da página

### **PageWidth()**

Retorna a largura da página.

Retorno Largura da página

### **PaperSize()**

Retorna o tamanho do papel selecionado.

Retorno Tamanho do papel

### **PrintPageFooter(nDiff)**

Imprime o rodapé da página.

nDiff Quantidade de linhas reservadas para o rodapé

### **PrintTotal(nDiff,ISetPos)**

Imprime os totalizadores da página.

nDiff Quantidade de linhas reservadas para o rodapé

ISetPos Posiciona a linha de impressão na primeira linha reservada para o rodapé

### **ResetFooterDiff()**

Inicializa os valores referente a quantidade de linhas reservadas para o rodapé, utilizados no cálculo da altura da página.

### **ResetPage()**

Inicializa as variáveis da página, indicando como impressão da primeira página.

### **ResetTotalDiff()**

Inicializa os valores referente a quantidade de linhas reservadas para o rodapé.

### **SetLandscape()**

Define a orientação de página do relatório como paisagem.

**SetPageDiff(IPageDiff)**

Define que será considerada a quantidade de linhas a serem impressas no rodapé para o cálculo da altura da página .

**SetPageFooter(nLinesFooter,bPageFooter,IFooterBefore)**

Define a configuração para impressão do rodapé da página.

nLinesFooter Quantidade de linhas reservadas para impressão do rodapé

bPageFooter Bloco de código utilizado para imprimir Informações customizadas, no qual, deverá ser utilizado métodos do TReport para impressão. Exemplo: Say, PrintText.

IFooterBefore Imprime rodapé antes da impressão dos totalizadores

**SetPageNumber(nPage)**

Define o número da página atual.

nPage Número da página

**SetPaperSize(nPaperSize)**

Define o tamanho da página.

nPaperSize Tamanho da página

**SetPortrait()**

Define a orientação de página do relatório como retrato.

**SetTotalBefore(ITotalBefore)**

Define que o total da página será impressa antes do rodapé.

ITotalBefore Total da página antes do rodapé

**StartPage()**

Inicializa a impressão de uma nova página.

**TotalDiff()**

Retorna a quantidade total de linhas reservadas para impressão do rodapé.

Retorno Quantidade total de linhas reservadas



## **TRPARAMPAGE**

Imprime página de parâmetros para relatório que utiliza a classe TREPORT.

Este método herda as propriedades e métodos da classe TRSECTION.

### **Propriedades**

nSize            Tamanho da célula

### **MÉTODOS**

#### **New(oParent,cParam)**

Método construtor da classe TRParamPage.

oParent        Objeto da classe TREPORT

cParam        Grupo de perguntas

Retorno        Objeto da classe TRParamPage

#### **Print()**

Imprime a página de parâmetros.

#### **PrintFilter()**

Imprime as Informações de usuário e filtros.

## TRCOLLECTION

Acumulador de uma quebra, seção e relatório que utiliza a classe TReport.

Um Acumulador pode executar uma das seguintes funções abaixo, utilizando como referência uma célula da seção ou o retorno de uma fórmula definida para ele:

SUM	Somar
COUNT	Contar
MAX	Valor máximo
MIN	Valor mínimo
AVERAGE	Valor médio
ONPRINT	Valor atual
TIMESUM	Somar horas
TIMEAVERAGE	Valor medio de horas
TIMESUB	Subtrai horas

Na criação do acumulador sempre é informado uma seção (TRSection) a qual ele pertence, e automaticamente o acumulador será incluído no relatório (TReport) que a seção pertence.

Esta classe herda as propriedades e métodos da classe TRFUNCTION.

### Propriedades

bltemPrint	Compatibilidade – Não utilizado.
bltemReset	Compatibilidade – Não utilizado.
bltemValue	Compatibilidade – Não utilizado.
cGName	Título do gráfico personalizado.
cGSerie	Título da série do gráfico personalizado.
cltemPicture	Máscara de impressão do acumulador.
IGEndReport	Aponta impressão do gráfico no final do relatório.
IGraphic	Aponta a existência de gráficos no relatório.
IPrintLayout	Aponta que é impressão de visualização do layout.
IUserAccess	Se verdadeiro, permite o usuário utilizar acumuladores personalizados.
nGType	Tipo do gráfico personalizado. 1 – Linhas 2 – Area 3 – Pontos 4 – Barra 9 – Pizza 12 – Linhas Rápidas
uContent	Regra com o valor acumulado. Tipo Caracter: Expressão ADVPL para macro execução. Tipo Bloco de código: Bloco de Código com a expressão ADVPL para execução.

### MÉTODOS

#### Col()

Compatibilidade – Não utilizado.

#### EvalFunction()

Atualiza o conteúdo dos acumuladores.

## **New(cName,cFunction,oBreak,uTitle,cPicture,uFormula,lEndSection,lEndReport,oParent,bCondition,uContent)**

Método constructor da classe TRCollection.

cName	Identificação do acumulador
cFunction	Função que será utilizada pelo acumulador. Exemplo: SUM, COUNT, MAX, MIN
oBreak	Objeto da classe TRBreak que define em qual quebra o acumulador será impresso
uTitle	Título do acumulador. Texto a ser impresso no formato caracter ou bloco de código
cPicture	Máscara de impressão do acumulador
uFormula	Regra para acúmulo dos valores Tipo Caracter: Expressão ADVPL para macro execução Tipo Bloco de código: Bloco de Código com a expressão ADVPL para execução
lEndSection	Se verdadeiro. Indica se o acumulador será impresso na quebra de seção
lEndReport	Se verdadeiro. Indica se o acumulador será impresso no final do relatório
oParent	Objeto da classe TRSection que o acumulador se refere
bCondition	Bloco de código com a condição de atualização dos valores do acumulador
uContent	Regra com o valor acumulado Tipo Caracter: Expressão ADVPL para macro execução Tipo Bloco de código: Bloco de Código com a expressão ADVPL para execução

Retorno      Objeto da classe TRCollection

## **Reset()**

Inicializa o acumulador atual.

## **ResetPage()**

Inicializa o acumulador da página.

## **ResetReport()**

Inicializa o acumulador do relatório.

## **ResetSection()**

Inicializa o acumulador da seção.

## **ResetSize(nType)**

Inicializa o tamanho da célula de impressão do acumulador.

nType	Aponta o tipo de collection a ser verificado
	0 – Acumulador atual
	1 – Acumulador do relatório
	2 – Acumulador da seção
	3 – Acumulador da página

## **SetBreak(oBreak)**

Define a quebra (TRBreak) que o acumulador será impresso.

oBreak      Objeto da classe TRBreak

## **SetPicture(cPicture)**

Define a máscara de impressão do acumulador.

cPicture      Máscara do acumulador

## **SetTitle(uTitle)**

Define o título que será impresso antes da impressão dos acumuladores.

uTitle      Texto a ser impresso no formato caracter ou bloco de código

**SetTotalInLine(ITotalInLine)**

Define que o acumulador será impresso em linhas.

ITotalInLine     Se verdadeiro, aponta que o acumulador será impresso em linhas

**Title()**

Retorna o título do acumulador.

Retorno            Título do acumulador

**XmlLoad(oXml)**

Carrega um acumulador do tipo collection do layout personalizado pelo usuário.

oXml                Objeto do tipo XML com as personalizações do acumulador

**XmlSave()**

Salva o acumulador do tipo collection no layout personalizado pelo usuário.

## TRBORDER

Classe com as propriedades e métodos para impressão de bordas.

### Propriedades

nCol	Coluna da borda.
nColor	Numero da cor RGB.
nEdge	Tipo da borda. 1 – Topo, 2 – Fim, 3 – Esquerda, 4 – Direita e 5 – Tudo.
nLineStyle	Estilo da borda. 0 – Sem borda, 1 – Borda continua, 2 – Borda da classe superior, 3 – Borda do cabeçalho da classe superior, 4 – Borda da célula, 6 – Borda do totalizador e 7 – Borda da seção.
nRow	Linha inicial da borda.
nRowDiff	Quantidade de linhas a serem saltadas antes da impressão da borda.
nWeight	Largura/Altura da borda.
oParent	Objeto da classe que irá utilizar a borda.
oReport	Objeto da classe TREPORT que a borda pertence.
oPrint	Objeto TMSPrinter da classe TREPORT.
oRealBorder	Objeto da classe TRBorder.

### MÉTODOS

#### CheckBorder(nWeight,nColor,nLineStyle)

Verifica se a borda foi criada e caso contrário define com as Informações do parâmetro.

nWeight	Largura/Altura da borda
nColor	Numero da cor RGB
nLineStyle	Estilo da borda. 0 – Sem borda, 1 – Borda continua, 2 – Borda da classe superior, 3 – Borda do cabeçalho da classe superior, 4 – Borda da célula, 6 – Borda do totalizador e 7 – Borda da seção

#### Col()

Retorna a coluna em que a borda será impressa.

Retorno	Coluna da borda
---------	-----------------

#### Color()

Retorna a cor da borda.

Retorno	Número da cor RGB
---------	-------------------

#### LineStyle()

Retorna o estilo de borda utilizado.

Retorno	Estilo da borda 0 – Sem borda, 1 – Borda continua, 2 – Borda da classe superior, 3 – Borda do cabeçalho da classe superior, 4 – Borda da célula, 6 – Borda do totalizador e 7 – Borda da seção
---------	---

#### New(oParent,nEdge,nWeight,nLineStyle,nColor)

Método constructor da classe TRBorder.

oParent	Objeto da classe que irá utilizar a borda
nEdge	Tipo da borda. 1 – Topo, 2 – Fim, 3 – Esquerda, 4 – Direita e 5 – Tudo
nWeight	Largura/Altura da borda

nLineStyle      Estilo da borda. 0 – Sem borda,      1 – Borda continua, 2 – Borda da classe superior, 3 – Borda do cabeçalho da classe superior, 4 – Borda da célula, 6 – Borda do totalizador e 7 – Borda da seção

nColor          Número da cor RGB

Retorno          Objeto da classe TRBorder

### **Print()**

Imprime a borda.

### **RealBorder()**

Retorna a borda atual.

Retorno          Objeto da classe TRBorder

### **SetCol(nCol)**

Define a coluna da borda.

nCol              Coluna da borda

### **SetColor(nColor)**

Define a cor da borda.

nColor          Número da cor RGB

### **SetLineStyle(nLineStyle)**

Define o estilo da borda.

nLineStyle      0 – Sem borda  
                    1 – Borda continua  
                    2 – Borda da classe superior  
                    3 – Borda do cabeçalho da classe superior  
                    4 – Borda da célula  
                    6 – Borda do totalizador  
                    7 – Borda da seção

### **SetRow(nRow)**

Define a linha inicial a ser impressa a borda.

nRow              Linha inicial da borda

### **SetRowDiff(nRowDiff)**

Define a quantidade de linhas que deverão ser saltadas antes da impressão da borda.

nRowDiff          Quantidade de linhas a serem saltadas

### **SetWeight(nWeight)**

Define a largura/altura da borda.

nWeight          Largura/altura da borda

### **Weight()**

Retorna a largura/Altura da borda.

Retorno          Largura/Altura da borda

### **XmlLoad(oXml)**

Carrega as bordas do layout personalizado pelo usuário.

oXml              Objeto do tipo XML com as personalizações das bordas

**XmlSave()**

Salva as bordas no layout personalizado pelo usuário.

## Relatórios do tipo listagem com TREPORT

Gera um relatório de listagem simples utilizando a classe de impressão TReport.

A função MPReport substitui o uso da função ImpCadast para os relatórios de listagem simples.

### Sintaxe

MPReport ( < cReport > , < cAlias > , < cTitle > , [ cDescription ] , [ aOrder ] , [ ILoadOrder ] )

cReport        Nome do relatório (exemplo: MATR020)

cAlias         Tabela utilizada pelo relatório

cTitle         Título do relatório

cDescription   Descrição do relatório

aOrder         Vetor com as descrições das ordens utilizadas do pelo relatório

ILoadOrder    Se verdadeiro, carrega os índices do Dicionário de Índices (SIX)

### Exemplo

```
#include "protheus.ch"
```

```
//Informando o vetor com as ordens utilizadas pelo relatório
```

```
User Function MyReport1()
```

```
MPReport("MYREPORT1","SA1","Relacao de Clientes","Este relatório irá imprimir a relação de clientes",{ "Por  
Codigo","Alfabetica","Por "+RTrim(RetTitle("A1_CGC"))})
```

```
Return
```

```
//Informando para função carregar os índices do Dicionário de Índices (SIX) da tabela
```

```
User Function MyReport2()
```

```
MPReport("MYREPORT2","SA1","Relacao de Clientes","Este relatório irá imprimir a relacao de clientes",,.T.)
```

```
Return
```



## Exemplo de utilização do TREPORT

User Function testep()

```
Local oReport := TReport():New('TITULO','teste',/*cPerg*/,{|oReport| __PRPrint(oReport)},,,,,,,)
```

```
Local nl
```

```
Local oBreak
```

```
oReport:SetTotalInLine(.F.)
```

```
oReport:SetTitle('Protheus Report Utility')
```

```
oReport:SetLineHeight(30)
```

```
oReport:SetColSpace(1)
```

```
oReport:SetLeftMargin(0)
```

```
oReport:oPage:SetPageNumber(1)
```

```
oReport:cFontBody := 'Courier New'
```

```
oReport:nFontBody := 6
```

```
oReport:lBold := .F.
```

```
oReport:lUnderLine := .F.
```

```
oReport:lHeaderVisible := .T.
```

```
oReport:lFooterVisible := .T.
```

```
oReport:lParamPage := .F.
```

```
oTREPORT02:= TRSection():New(oReport,'Contas a Receber',,,,,,,)
```

```
oTREPORT02:SetTotalInLine(.F.)
```

```
oTREPORT02:SetTotalText('Contas a Receber')
```

```
oTREPORT02:lUserVisible := .T.
```

```
oTREPORT02:lHeaderVisible := .F.
```

```
oTREPORT02:SetLineStyle(.F.)
```

```
oTREPORT02:SetLineHeight(30)
```

```
oTREPORT02:SetColSpace(1)
```

```
oTREPORT02:SetLeftMargin(0)
```

```
oTREPORT02:SetLinesBefore(0)
```

```
oTREPORT02:SetCols(0)
```

```
oTREPORT02:SetHeaderSection(.T.)
```

```
oTREPORT02:SetHeaderPage(.F.)
```

```
oTREPORT02:SetHeaderBreak(.F.)
```

```
oTREPORT02:SetLineBreak(.F.)
```

```
oTREPORT02:SetAutoSize(.F.)
```

```
oTREPORT02:SetPageBreak(.F.)
```

```
oTREPORT02:SetClrBack(16777215)
```

```
oTREPORT02:SetClrFore(0)
```

```
oTREPORT02:SetBorder("")
```

```
oTREPORT02:SetBorder(",,,T.)
```

```
oTREPORT02:aTable := {}
```

```
oTREPORT02:AddTable('SE1')
```

```
oTREPORT02:AddTable('SA1')
```

```
oTREPORT02:OnPrintLine({| If(SE1->E1_FILIAL $ '01|02|', .T., .F.)})
```

```
TRCell():New(oTREPORT02,'__NEW__001','" ,,,,,,,')
```

```
oTREPORT02:Cell("__NEW__001"):SetName("A1_NOME")
```

```
oTREPORT02:Cell("A1_NOME"):cAlias := "SA1"
```

```
oTREPORT02:Cell("A1_NOME"):SetTitle("Nome")
```

```
oTREPORT02:Cell("A1_NOME"):SetSize(40)
```

```
oTREPORT02:Cell("A1_NOME"):SetPicture("@!")
```

```
oTREPORT02:Cell("A1_NOME"):SetAutoSize(.F.)
```

```
oTREPORT02:Cell("A1_NOME"):SetLineBreak(.F.)
```

```

oTREPORT02:Cell("A1_NOME"):SetHeaderSize(.F.)
oTREPORT02:Cell("A1_NOME"):nAlign := 1
oTREPORT02:Cell("A1_NOME"):nHeaderAlign := 1
oTREPORT02:Cell("A1_NOME"):SetClrBack(16777215)
oTREPORT02:Cell("A1_NOME"):SetClrFore(0)
oTREPORT02:Cell("A1_NOME"):cOrder := "A0"
oTREPORT02:Cell("A1_NOME"):nType := 1
oTREPORT02:Cell("A1_NOME"):cFormula := ""
oTREPORT02:Cell("A1_NOME"):cRealFormula := ""
oTREPORT02:Cell("A1_NOME"):cUserFunction := ""
oTREPORT02:Cell("A1_NOME"):lVisible := .T.
oTREPORT02:Cell("A1_NOME"):SetBorder("")
oTREPORT02:Cell("A1_NOME"):SetBorder("","",".T.)

TRCell():New(oTREPORT02,'__NEW__002','"','"')
oTREPORT02:Cell("__NEW__002"):SetName("E1_PREFIXO")
oTREPORT02:Cell("E1_PREFIXO"):cAlias := "SE1"
oTREPORT02:Cell("E1_PREFIXO"):SetTitle("Prefixo")
oTREPORT02:Cell("E1_PREFIXO"):SetSize(3)
oTREPORT02:Cell("E1_PREFIXO"):SetPicture("@!")
oTREPORT02:Cell("E1_PREFIXO"):SetAutoSize(.F.)
oTREPORT02:Cell("E1_PREFIXO"):SetLineBreak(.F.)
oTREPORT02:Cell("E1_PREFIXO"):SetHeaderSize(.F.)
oTREPORT02:Cell("E1_PREFIXO"):nAlign := 1
oTREPORT02:Cell("E1_PREFIXO"):nHeaderAlign := 1
oTREPORT02:Cell("E1_PREFIXO"):SetClrBack(16777215)
oTREPORT02:Cell("E1_PREFIXO"):SetClrFore(0)
oTREPORT02:Cell("E1_PREFIXO"):cOrder := "A1"
oTREPORT02:Cell("E1_PREFIXO"):nType := 1
oTREPORT02:Cell("E1_PREFIXO"):cFormula := ""
oTREPORT02:Cell("E1_PREFIXO"):cRealFormula := ""
oTREPORT02:Cell("E1_PREFIXO"):cUserFunction := ""
oTREPORT02:Cell("E1_PREFIXO"):lVisible := .T.
oTREPORT02:Cell("E1_PREFIXO"):SetBorder("")
oTREPORT02:Cell("E1_PREFIXO"):SetBorder("","",".T.)

TRCell():New(oTREPORT02,'__NEW__003','"','"')
oTREPORT02:Cell("__NEW__003"):SetName("E1_NUM")
oTREPORT02:Cell("E1_NUM"):cAlias := "SE1"
oTREPORT02:Cell("E1_NUM"):SetTitle("No. Titulo")
oTREPORT02:Cell("E1_NUM"):SetSize(9)
oTREPORT02:Cell("E1_NUM"):SetPicture("@!")
oTREPORT02:Cell("E1_NUM"):SetAutoSize(.F.)
oTREPORT02:Cell("E1_NUM"):SetLineBreak(.F.)
oTREPORT02:Cell("E1_NUM"):SetHeaderSize(.F.)
oTREPORT02:Cell("E1_NUM"):nAlign := 1
oTREPORT02:Cell("E1_NUM"):nHeaderAlign := 1
oTREPORT02:Cell("E1_NUM"):SetClrBack(16777215)
oTREPORT02:Cell("E1_NUM"):SetClrFore(0)
oTREPORT02:Cell("E1_NUM"):cOrder := "A2"
oTREPORT02:Cell("E1_NUM"):nType := 1
oTREPORT02:Cell("E1_NUM"):cFormula := ""
oTREPORT02:Cell("E1_NUM"):cRealFormula := ""
oTREPORT02:Cell("E1_NUM"):cUserFunction := ""

```

```

oTREPORT02:Cell("E1_NUM"):IVisible := .T.
oTREPORT02:Cell("E1_NUM"):SetBorder("")
oTREPORT02:Cell("E1_NUM"):SetBorder("","",.T.)

TRCell():New(oTREPORT02,'__NEW__004',' ',,,,,,,,,,,,,)
oTREPORT02:Cell("__NEW__004"):SetName("E1_PARCELA")
oTREPORT02:Cell("E1_PARCELA"):cAlias := "SE1"
oTREPORT02:Cell("E1_PARCELA"):SetTitle("Parcela")
oTREPORT02:Cell("E1_PARCELA"):SetSize(1)
oTREPORT02:Cell("E1_PARCELA"):SetPicture("@!")
oTREPORT02:Cell("E1_PARCELA"):SetAutoSize(.F.)
oTREPORT02:Cell("E1_PARCELA"):SetLineBreak(.F.)
oTREPORT02:Cell("E1_PARCELA"):SetHeaderSize(.F.)
oTREPORT02:Cell("E1_PARCELA"):nAlign := 1
oTREPORT02:Cell("E1_PARCELA"):nHeaderAlign := 1
oTREPORT02:Cell("E1_PARCELA"):SetClrBack(16777215)
oTREPORT02:Cell("E1_PARCELA"):SetClrFore(0)
oTREPORT02:Cell("E1_PARCELA"):cOrder := "A3"
oTREPORT02:Cell("E1_PARCELA"):nType := 1
oTREPORT02:Cell("E1_PARCELA"):cFormula := ""
oTREPORT02:Cell("E1_PARCELA"):cRealFormula := ""
oTREPORT02:Cell("E1_PARCELA"):cUserFunction := ""
oTREPORT02:Cell("E1_PARCELA"):IVisible := .T.
oTREPORT02:Cell("E1_PARCELA"):SetBorder("")
oTREPORT02:Cell("E1_PARCELA"):SetBorder("","",.T.)

TRCell():New(oTREPORT02,'__NEW__005',' ',,,,,,,,,,,,,)
oTREPORT02:Cell("__NEW__005"):SetName("E1_CLIENTE")
oTREPORT02:Cell("E1_CLIENTE"):cAlias := "SE1"
oTREPORT02:Cell("E1_CLIENTE"):SetTitle("Cliente")
oTREPORT02:Cell("E1_CLIENTE"):SetSize(6)
oTREPORT02:Cell("E1_CLIENTE"):SetPicture("@!")
oTREPORT02:Cell("E1_CLIENTE"):SetAutoSize(.F.)
oTREPORT02:Cell("E1_CLIENTE"):SetLineBreak(.F.)
oTREPORT02:Cell("E1_CLIENTE"):SetHeaderSize(.F.)
oTREPORT02:Cell("E1_CLIENTE"):nAlign := 1
oTREPORT02:Cell("E1_CLIENTE"):nHeaderAlign := 1
oTREPORT02:Cell("E1_CLIENTE"):SetClrBack(16777215)
oTREPORT02:Cell("E1_CLIENTE"):SetClrFore(0)
oTREPORT02:Cell("E1_CLIENTE"):cOrder := "A4"
oTREPORT02:Cell("E1_CLIENTE"):nType := 1
oTREPORT02:Cell("E1_CLIENTE"):cFormula := ""
oTREPORT02:Cell("E1_CLIENTE"):cRealFormula := ""
oTREPORT02:Cell("E1_CLIENTE"):cUserFunction := ""
oTREPORT02:Cell("E1_CLIENTE"):IVisible := .T.
oTREPORT02:Cell("E1_CLIENTE"):SetBorder("")
oTREPORT02:Cell("E1_CLIENTE"):SetBorder("","",.T.)

TRCell():New(oTREPORT02,'__NEW__006',' ',,,,,,,,,,,,,)
oTREPORT02:Cell("__NEW__006"):SetName("E1_LOJA")
oTREPORT02:Cell("E1_LOJA"):cAlias := "SE1"
oTREPORT02:Cell("E1_LOJA"):SetTitle("Loja")
oTREPORT02:Cell("E1_LOJA"):SetSize(1)
oTREPORT02:Cell("E1_LOJA"):SetPicture("@!")

```

```

oTREPORT02:Cell("E1_LOJA"):SetAutoSize(.F.)
oTREPORT02:Cell("E1_LOJA"):SetLineBreak(.F.)
oTREPORT02:Cell("E1_LOJA"):SetHeaderSize(.F.)
oTREPORT02:Cell("E1_LOJA"):nAlign := 1
oTREPORT02:Cell("E1_LOJA"):nHeaderAlign := 1
oTREPORT02:Cell("E1_LOJA"):SetClrBack(16777215)
oTREPORT02:Cell("E1_LOJA"):SetClrFore(0)
oTREPORT02:Cell("E1_LOJA"):cOrder := "A5"
oTREPORT02:Cell("E1_LOJA"):nType := 1
oTREPORT02:Cell("E1_LOJA"):cFormula := ""
oTREPORT02:Cell("E1_LOJA"):cRealFormula := ""
oTREPORT02:Cell("E1_LOJA"):cUserFunction := ""
oTREPORT02:Cell("E1_LOJA"):IVisible := .T.
oTREPORT02:Cell("E1_LOJA"):SetBorder("")
oTREPORT02:Cell("E1_LOJA"):SetBorder("",,,.T.)

```

```
TRPosition():New(oTREPORT02,'SA1',1,{ || xFilial()+SE1->(E1_CLIENTE+E1_LOJA) } )
```

```

oBreak := TRBreak():New(oTREPORT02,{
oTREPORT02:Cell('E1_CLIENTE'):uPrint+oTREPORT02:Cell('E1_LOJA'):uPrint },'Sub-Total',.F.)
TRFunction():New(oTREPORT02:Cell('E1_CLIENTE'),, 'COUNT',oBreak ,,,,F.,.F.,.F., oTREPORT02)

```

```

oTREPORT02:LoadOrder()
oReport:PrintDialog()

```

Return

## Informações Técnicas

Tabelas Utilizadas	
Rotinas Envolvidas	REPORT01-TReport
Sistemas Operacionais	Windows/Linux
Número do Plano	00000026372/2009