

FlexCellImport

FlexCellImport allows you to read/write an Excel file by code. It's targeted at reading files (hey, that's why it's called "import"), but you can write data too.

As you can write an Excel file, this gives an alternative to FlexCelReport, but to create files I strongly recommend using FlexCelReport

Note that all the cell/sheet references begin in 1, that is, cell A1 is CelValue[1,1], and to make the activesheet the first one, you set ActiveSheet:=1

INDEX

FlexCellImport.....	1
INDEX.....	1
Properties.....	2
property Adapter: TExcelAdapter;.....	2
Methods.....	2
procedure OpenFile(const FileName: TFileName).....	2
procedure Save(const FileName: TFileName).....	2
procedure CloseFile.....	2
property SheetCount: byte.....	2
property ActiveSheet: byte.....	2
property ActiveSheetName: widestring.....	2
property MaxRow: integer.....	2
function IsEmptyRow(const aRow: integer): boolean;.....	2
property CellValue[Row, Col: integer]: variant.....	2
procedure InsertAndCopyRows(const FirstRow, LastRow, DestRow, aCount: integer);.....	2
procedure DeleteRows(const aRow, aCount: word);.....	2
property CommentsCount[Row: integer]: integer.....	2
property Comments[Row, Col: integer]: variant.....	2
function IsLoaded: boolean.....	2

Properties

property Adapter: TExcelAdapter;

This property is connected with a TOLEAdapter or a TXLSAdapter you must drop into the Datamodule. If you connect it with a TOLEAdapter, all the operations will be made using Excel OLE Automation. If you connect it to a TXLSAdapter, everything will be made natively, without calling Excel.

Methods

procedure OpenFile(const FileName: TFileName)

This opens a file for reading/writing. If you are using a TemplateStore, the file will be read from there.

procedure Save(const FileName: TFileName)

This method saves the file with the specified filename

procedure CloseFile

Use this when you are done editing the file.

property SheetCount: byte

The number of sheets of the file.

property ActiveSheet: byte

Read this to get the active sheet, write to it to set it.

property ActiveSheetName: widestring

This is the name of the active sheet, you can read or change it.

property MaxRow: integer

This is the row count in the active sheet.

function IsEmptyRow(const aRow: integer): boolean;

True if the row contains no cells. You can not set a cell value in an empty row.

property CellValue[Row, Col: integer]: variant

By accessing this property, you can read or change a cell contents. Be aware that you cannot write to an empty row (property IsEmptyRow=true). To do it, first InsertAndCopyRows.

procedure InsertAndCopyRows(const FirstRow, LastRow, DestRow, aCount: integer);

By calling this method you copy a range including both FirstRow and LastRow, inserted on DestRow and copied aCount times.

procedure DeleteRows(const aRow, aCount: word);

This method deletes aCount rows starting with aRow

property CommentsCount[Row: integer]: integer

This is the number of comments on the worksheet.

property Comments[Row, Col: integer]: variant

With this property you can read or change a specific comment.

function IsLoaded: boolean

True if a spreadsheet has been loaded