WDS-JniPMML-XLL Documentation

General ReadMe

The primary objective of WDS-JniPMML-XLL is to provide model evaluators to Excel. In particular, access to the standard PMML evaluator is a starting point, both for use or for comparison. Later versions will be include other model specs and implement other evaluators.

See documentation articles for a brief introduction.

Through this version, WDS-JniPMML-XLL provides:

- A pair of Excel AddIns (XLLs) and VBA support for:
  - Evaluating PMML models
    - As an Excel function call
    - Using the *de facto* standard implementations
    - Using input data from an in-worksheet table
      Uses XmlMap'd exportable ListObjects, but provides tools to facilitate
    - Can evaluate one or multiple observations (rows) per call
    - Results returned as normal function outputs
    - With cachable models for efficiency
  - Additional data wrangling tools for
    - Importing/Exporting HDF5 compound datasets
    - Importing/Exporting flat files
  - Additional VBA module handling
- A Java wrapper of jpmml.evaluator
  - Callable from the XLL via jni
  - Testable as a standalone from the command line
    But, can be called through the Excel AddIn using the JVM.
  - Input and output data can be:
    - HDF5 compound datasets
    - Flat files
    - In memory (as when called through jni)
- Examples are included
  - A test workbook and launch .bat to run the AddIns without installing
  - A test set of the usual PMML cases

Prerequisites

- 64 bit Excel
  Although, if compiling, 32 bit could possibly be added.
- Access to the VBA project object model (if using the VBA module handlers
- HDF5 and HDFView
  - The HDF5 and HDFView libs are required if compiling, but the functionality could be removed.
  - The provided jars require at least HDFView be on the path or the path passed in as a command line option when starting Excel
- Java jdk-12
  Required when using the latest HDFView install.
- Compiling environment
  The github configurations are for Visual Studio Community Edition and Intellij Community edition.
- DocFx

DocFx is use for the documentation build, including the DocFxDoclet for on the JavaDoc side.

License Note

TODOs, version 0.5.0

Outstanding items and items for the next version:

- Additional documentation and expanded test suite.
- Date and DateTime datatypes are not fully implemented. They are preliminarily setup to pass as doubles, but wrangling of string values, detection via cell information, and testing needs to be performed.
- The object cache and Handle/Tag handlers could be written better.
- Additional HDF5 utilities, such as returning a layout and querying an element when an HDF5 is used like a memory mapped file.
- Expanded VBA library.

An interesting thought or wish list:

- External memory mapped files, using the DBB wrangler.
- A parallel assembly for calling from MSSQLServer.

Related projects to be published on WDataSci's github site:

- WDataSci XML Model Specification, documentation and tools, including xsl implementation transformations.
- WDataSci Systems Model, documentation and tools.

MIT License

Articles

Brief Introduction

Additional Usage Notes

Notes on JniPMML

Notes on Java x Cs

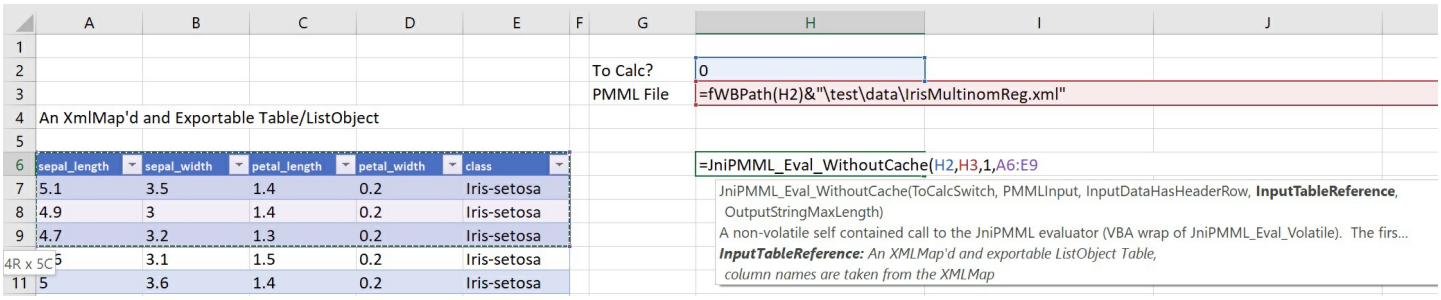
Brief Intro

The primary objective of WDS-JniPMML-XLL is to provide model evaluators to Excel. In particular, access to the standard PMML evaluators is a starting point, both for use or for comparison. Later versions will be include other model specs and implement other evaluators.

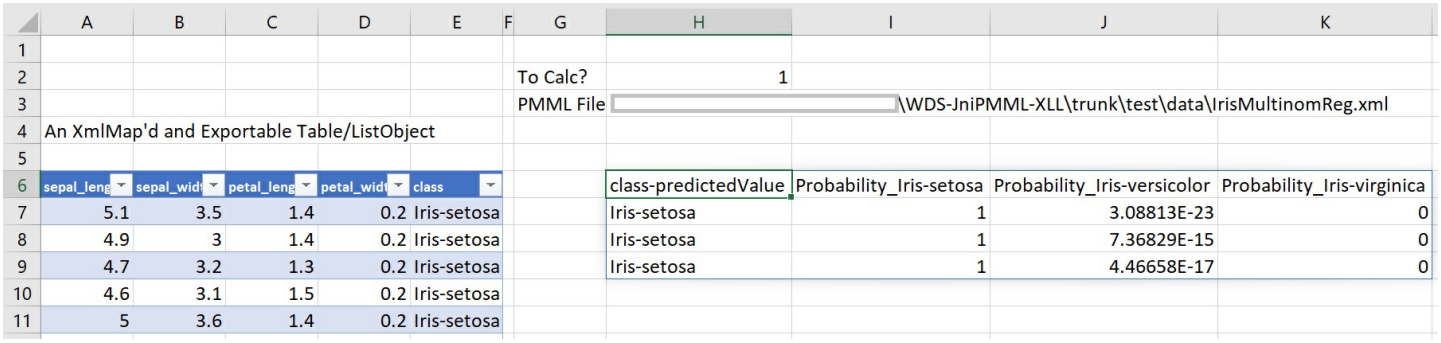
A quick easy way to evaluate PMML should be available to anyone, even those without access to the latest data science tools. In the finance industry, Excel is ubiquitous. To academic-type data scientists who might scoff at using Excel for anything, it is still a tool which can be used smartly or extremely poorly (which can certainly be said of Python, R, Java, C#, or anything else).

Simple Example

For one-time evaluations (less efficient, but simple), data is arranged in a table object. A macro is provided to assist in providing a technical requirement on the table. The JniPMML_Eval_WithoutCache function takes just a few arguments as below:



In older versions of Excel, results could be returned as an array-valued function, but in Excel 2016, the *spill* feature allows the function to return a dynamic number of rows and columns:



One thing that might not be obvious from the images above is these are function results. The inputs may even be randomized and (relatively) instantaneous evaluations returned.

Slightly More Complex

More efficient model evaluation involves caching the model and then repeated calls to the evaluator without having to do all of the parsing process of the model implementation spec for every calculation. There are several steps involved, but are simplified in the workbook:

- Pick a *Tag* for model
  If you are thinking this should be called a *Handle*, that would seem correct. Except, the Handles are provided and controlled by the Java side and Tags are used on the Excel/C# side.

  A Handle actually has two parts, *HandleMajor.HandleMinor*. The HandleMajor is unique to the Tag and the cached model on the Java side. The HandleMinor increments with subsequant configuraton changes, such as caching the input schema from the ListObject and the output schema. Why do this? It is an Excel-trick. When other ranges depend on the HandleMajor.HandleMinor value of some cell, and that cell recalculates, the correct cascade of recalculations occurs.

- Provide a PMML Model

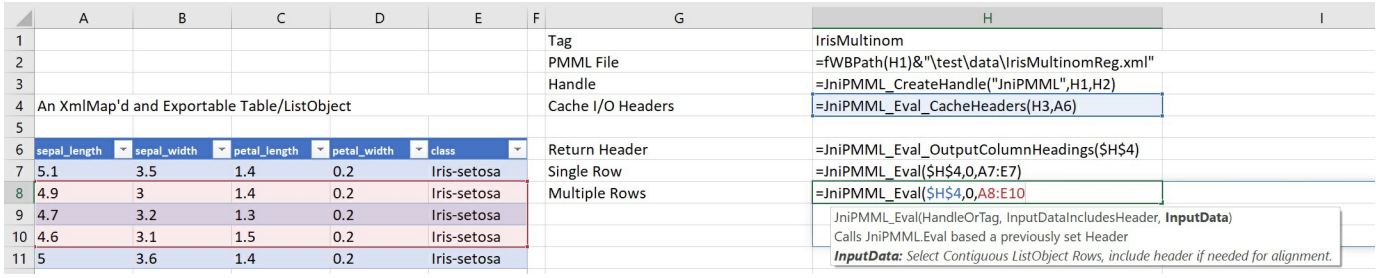
Here the model can be either a path to a PMML file or (more interestingly) a full PMML file as a string. Why take the string? One could build the file in the workbook. Perhaps one is testing some transformation structure or just wants to see what happens.

- Create a handle
  Cache the PMML model on the Java side and return the new handle to Excel.

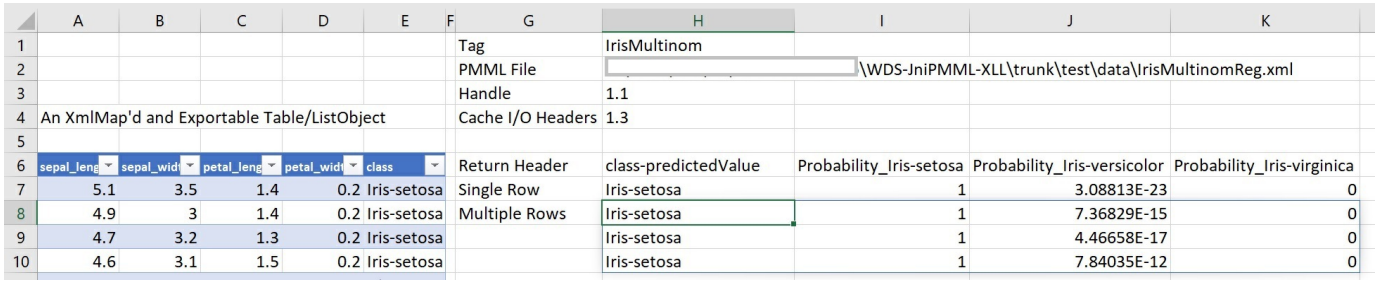
- Cache the input and output headers
  Based on the input XmlMap'd exportable table, update the HandleMinor. This also internally caches the model outputs which can be queried for column headings.
  Note: Some PMML models are harder to fully determine the output structures and results are returned as a dictionary-like structure. In the current version and for this case, there is a function, JniPMML_Expand_ComplexValue that can be used to return an expansion. See the WDS-JniPMML-XLL-Test.xlsm workbook for an example.

- Point to input and return the evaluation
  The input is an XmlMap'd exportable table. Before that raises any concerns, there is a macro available through the ribbon, "Add XmlMap to Selected ListObject", which will assign one to it through the following steps:
  - Select a cell in a table or the entire table
  - Hit the macro and you will be queried for one of the following
    - Point to an external XSD file
    - Point to an XSD as a string in a cell
    - Infer one from the table
    - Use a cached PMML dictionary, matching by column name, and infer where a column is not in the dictionary

  The slightly more complex example:



With result:



See the provided Excel test workbook for additional examples.

Additional Usage Notes

- Accessing Java via JNICode creates a COM AddIn
  Efforts have been made to make sure COM objects are clean up. However, should the process break for whatever reason, there may be an Excel process hanging around. In that case, look in the taskmgr's details. Or, use something like the powershell snippets in the scripts folder to find and stop.

Notes on JniPMML

Author: Christian Wypasek

Simple Motivation

My daughter, a college student, asked me to explain this project in one sentence and this was as close as I could get: Scientists build models. For even something as simple as linear regression, there is a formula that needs to evaluated. It might be for my own purposes, or it might be for a company I work for, but model implementation needs to be easily accessible. Even though data scientists might use special tools, everyone in financial services at least has Excel.

Slightly More Technical Motivation

Regardless of whether or not Excel might be highly regarded as a computational framework among academicians, it is ubiquitous in financial services (even if it might not be used well). Therefore, it makes sense that invoking an XML based evaluator from within Excel would be worthwhile. In particular, since Excel can enable rapid visualization, one should also be able to compare evaluator implementations and view model response to variable changes and/or model structure in a live manner.

XML/PMML

For someone like myself who works across the spectrum of big data projects (project management and business interface, data science, and data engineer) and works across multiple programming languages, consistency of treatments is a fundamental key to efficiency. After years of engineering databases, building complex statistical models for financial instruments, and incorporating these models into asset backed cash flow valuations, the greatest risks in this data science process are often operational. There is the most obvious question, "Is the data being used for forecasting sufficiently like the data the model was fit on?", but one also has to ask "Is the model being calculated correctly?".

From personal experience, hand coding something like a scoring model requires significant quality checks and carries the persistent risk that something was overlooked. It does not take too many hand coding events to make one believe there has got to be a better way, both for efficiency of process and the reduction of mistakes that come from mind numbing exercises. Starting back in 1998/1999, I started using markup styles to facilitate both the modeling process and facilitating the implementation for scoring and other types of regresson and non-parametric models. Since then, PMML (predictive modeling markup language) as become an industry standard.

The PMML standard has evolved and early versions were not sophisticated enough for my needs. For example, the Scorecard implementation was not added until the end of 2011, and transformations were not added until 2014. For all that it is, PMML is still a communication standard for model implementation and is often generated after a model has been fit. Continued diligence is required so the communicated model truly represents the intended relationship between the input data and the output results. A process oriented view of statistical model building starts with data preparation and can be exploited at every step of the process through to final implementation.

There may be more than one way to skin a cat, but very few which leave you with anything looks like a cat. My personal work has included using mathematical and statistical model specifications in XML with implementations in SAS, C++, Python, R, in-database (Vertica, MSSQL) UDTFs in C++/Java/R, and VBA (in Excel). After drilling into PMML implementation details, there is still much to be desired. An updated XML specification used by WDataSci for model fitting and alternate implementations will be released on the its github as a later project, but transformation (such as through XSLT) into PMML for delivery is reasonable given the industry standardization that PMML offers. Other model implementation specifications, such as pfa, will emerge, and Excel will remain a platform for either a model delivery or easy comparison.

WDS-JniPMML as a multi-language project

The JniPMML project combines several APIs, each for a specific purpose:

- Java
  The *de facto* implementation of PMML is jpmml.evaluator. JniPMML-Java wraps the implementation in a manner that creates a standalone jar that can also be called from C# via jni.
- C#
  Using the ExcelDna project to facilitate Excel functionality, the JniPMML-Cs assembly wraps the jni calls.
- VB
  Some odds and ends which I have traditionally done in VBA, but using ExcelDna .Net. In particular, wrangling of the VBA modules is done in VB.
- VBA
  Certain Excel functions created with ExcelDna through either C# or VB become *volatile* in that they recalculate at every calculation event (which can be a bad thing). However, good old fashioned VBA can do the same thing in a non-volatile manner.

Working in different languages for different aspects of a larger project is not unusual. For example, database work might be done in SQL, with processing either in database or written-out-processed-read-back-in, and final summaries might have an entirely different framework. When sub-projects have many parallel functions, the tendency of programmers to have a project on one side and then start from scratch on the other side, can lead to unexpected differences which the programmer then might struggle to balance. Complete one side, move to the other, discover some new or useful treatment, go back to the first side, restart loop. This project also started in that manner.

Passing data back and forth in-memory between Java and C# involves packing memory in a particular way, which also turned out to be the HDF5.PInvoke bulk writes a HDF5 compound dataset (such as R can export). Development of the project included consideration of in-memory HDF5s, which despite HDF5 docs, is not ready for prime-time. For testing purposes, HDF5 CompoundDS and flat file functionality is included in the JniPMML-Java project and the Excel AddIn.

Finally, the Excel AddIn also includes other tools representative of some extended functionality I have come to expect over the years, such as VBA component wrangling and other examples. Even if this project is not used extensively outside of WDataSci, this project also become an in-house reference for C#/Java differences and quirks, DocFx, Excel AddIns (quirks across C#, ExcelDna, VB, VBA, COM, non-COM), PMML (and jpmml quirks), HDF5 (and quirks across HDF.PInvoke, HDF-Java, HDF-Object), etc.

Notes on Java x Cs

Author: Christian Wypasek

The mirroring of the C# and Java code is meant not to be slick or cute. It is simply because both implementations are reading and writing the same formats. When handing off data in a ByteBuffer between C# and Java, in both directions, the formats must be *exactly* the same. (Note, not going down the AST route. It seems like if you are going to go that route, you should be all in.)

Some syntax differences are too big to bridge, such as in how enums are more flexible in Java than C#. With enums, just the values and methods (extensions in C#) are in common. The source codes will still be organized similarly, but this is also why enums are not otherwise in the files with their naturally associated classes.

Some syntax differences are not marked but obvious:

- Non-method properties, such as length/Length or boolean/Boolean, which are easy enough to fix in IDEs.
- To break String object references in Java where C# does not require it, a simple new_String() function in C# is a pass through and differs only with the "_".
- Method *throws* required in Java but not C# are on separate lines and commented out in C#.
- In switch-case statements on enum values where Java case statements do not require qualified names, there will be two lines one uncommented for Java, the other commented for C#, and visa-versa.

The syntax differences for many common methods amount only to the case of the leading letter, such as with Java's String.toString() vs C#'s String.ToString(). When this leading case issue is on a class method, they can be minimized through C#'s static extension methods, included in a static class, JavaLikeExtensions. Why not just let one letter differences ride, like in length/Length above? One line in one file and one less thing to highlight a difference in vimdiff. Other differences can be eliminated through specially named classes, mimicking names and methods used on the Java side, such as Map, PrintWriter, and ArrayList. Even though broken out in the documentation, on C#, they can all be included in the WDSXJava.cs, along with JavaLikeExtensions.

Syntax differences over lines or blocks are handled in two ways: First, when a one line change is required, a comment leading with //Java or //C# precedes the line. On the Java side, the //C# and subsequent line are collapsed, commenting out the C# syntax. The reverse treatment is used on the C# side.

For example, in C# version:

```
//C#
if ( !base.Equals(arg) ) return false;
//Java if ( !super.Equals(arg) ) return false;
```

And in the Java version:

```
//C# if ( !base.Equals(arg) ) return false;
//Java
if ( !super.Equals(arg) ) return false;
```

For larger blocks, we can exploit the behavior that an open-comment /* jumps over other open comments until the first closing */. Therefore, in the C# version (Note that the Java >>> comment is open):

```
/* C# >>> */
if ( !base.Equals(arg) ) return false;
/* <<< C# */
/* Java >>> *
if ( !super.Equals(arg) ) return false;
/* <<< Java */
```

And in the Java version (Note that the C# >>> comment is open):

```
/* C# >>> *
if ( !base.Equals(arg) ) return false;
/* <<< C# */
/* Java >>> */
if ( !super.Equals(arg) ) return false;
/* <<< Java */
```

There are multiple programming languages used in this project:

Note: This documentation bundle was created using DocFx, which was confusing documentation across APIs. Therefore, a separate PDF has been generated for each.

Java APIs

JniPMML-Java

The initial design of JniPMML-Java is to wrap jpmml into a single jar which can be called from Excel/C# via JNICode. However, it is a standalone that can be used with command line calls. It therefore has wranglers for text and HDF5 files in addition to the ByteBuffers for interacting with C#.

WDS-Java

General utilitities that independent of JniPMML code. To simplify assemblies and jars, this code is included in the larger projects, but is also compiled as a stand alone.

On the java side, there is a separate WDS-00.00.00.jar generated but it is pulled into a shaded jar so that only one WDS-JniPMML-00.00.00.jar needs to be used in practice.

C# APIs

JniPMML-Cs

The Java style com.WDataSci namespaces are specifically for C# code which mirrors the Java modules.

The JNI namespace originated externally, but with a few local completion and extensions.

WDS-Cs

General utilitities that independent of JniPMML code. To simplify assemblies and jars, this code is included in the larger projects, but is also compiled as a stand alone.

The Java style com.WDataSci namespaces are specifically for C# code which mirrors the Java modules.

The C# style namespaces are not specifically mirrored in the Java code.

VB APIs

JniPMML-VB

The JniPMML-VB code is primarily for some additional Excel manipulation functionality. In particular, the wrangling the Excel VBE components. The ExcelDna and Microsoft.Office.Interop.Excel libraries are generally mirrored in both C# and VB, however, ExcelDna UDF functions which take references as objects so that information about the caller can be determined at run-time become automatically volatile. For this reason, there are several function wrappers implemented in VBA which must be either in an another addin, or as a VBA module in the workbook. The JniPMML-VB (and supporting WDS-VB code which is pulled into the assembly) addin facilitates these wrapped functions by providing a wrangler for a WDSJniPMML.bas module.

WDS-VB

General utilitities that independent of JniPMML code. To simplify assemblies and jars, this code is included in the larger projects, but is also compiled as a stand alone.

VBA APIs

WDS-VBA

The WDS-VBA code is a collection of VBA macros that can be included in Excel workbooks and there are C#/VB macros accessibly through the ribbon to wrangle them in and out of workbooks as needed.

There is a necessity for at least the WDSCore macro for adding ExcelDna Intellisense capabilities.

C# APIs

JniPMML-Cs

The Java style com.WDataSci namespaces are specifically for C# code which mirrors the Java modules.

The JNI namespace originated externally, but with a few local completion and extensions.

WDS-Cs

General utilitities that independent of JniPMML code. To simplify assemblies and jars, this code is included in the larger projects, but is also compiled as a stand alone.

The Java style com.WDataSci namespaces are specifically for C# code which mirrors the Java modules.

The C# style namespaces are not specifically mirrored in the Java code.

JniPMML-Cs

The Java style com.WDataSci namespaces are specifically for C# code which mirrors the Java modules.

The JNI namespace originated externally, but with a few local completion and extensions.

Namespace com.WDataSci.JniPMML

The com.WDataSci.JniPMML namespace mirrors the Java com.WDataSci.JniPMML package.

See Java x Cs Notes for more cross language details.

Classes

DBB

DBB is short for Direct ByteBuffer, which on the Java side, is used as the internal data memory map which is being used to hand off data between C# and Java.

Despite the overhead on the both the C# and Java sides, the effective use is as a contiguous array of byte data. The main block can be thought of as a rectangular region for a table of data where each column has the same width in bytes (variable length fields are a long's width pointing to variable space). This is essentially the way HDF.PInvoke packs memory before bulk writing a compound dataset. Here, we add leading bytes for the essential layout and a block of space after the rectangular region with enough space for the variable length fields.

There are also three layouts:

- WDSH
  a header layout which includes a record for each column/member/field
- WDSD
  a record set layout associated with a header
- WDSC
  [TODO] a combined layout, with leading bytes, space for the header, and space for the record set

The layout for the leading bytes is common across all three and takes 8*8=64 bytes:

START	LENGTH	PURPOSE
0	8	The layout style, 8 bytes for 4 UTF-16 characters for WDSH/WDSD/WDSC
8	8	The total number of bytes required for this layout
16	8	The number of leading bytes before the fixed length region begins
24	8	The total number of bytes for the rectangular or fixed length region
32	8	The total number of bytes for the variable length region
40	8	The number of records (or rows) of data communicated
48	8	The fixed number of bytes per record
56	8	The maximum variable number of bytes each record may require

WDSH, the header layout needs to communicate just enough information to describe the core meta data for each column. This includes its name, a possible second name (generally used to point to a pre-mapped FieldName used in the PMML dictionary), and several additional meta data ints and longs. 40 fixed length bytes per column are for:

START	LENGTH	PURPOSE
0	8	Name variable length string pointer (long)
8	8	Optional second name variable length string pointer (long), 0 if not used/mapped
16	4	integer for data type code (see enum eDTyp)
20	8	long for number of bytes column data requires in the contiguous portion (variable length fields will have no more than 8 for a pointer to the data)
28	8	long for maximum byte length of data (fixed length strings are packed directly, null padded, variable length strings represent a long (IntPtr) to other memory)
36	4	(extra integer space, not used)

For variable length strings, the rectangular space will hold the offset from the start of the leader bytes of the space holding the string. For allocation purposes, each string could be expected to occupy 4+MaxByteLength. When a string is greater than MaxByteLength, it will be trimmed to MaxByteLength-2 to null terminate, this may change, but any 2 (UTF-16) consecutive 0 bytes will be taken as a null termination.

For further illustration, the WDSD layout including leading bytes is:

BLOCK	BLOCK START	BLOCK LENGTH	START	LENGTH	PURPOSE
Header	0	LeadTotal=64	0	8	UTF-16 bytes for WDSD
			8	8	LayoutTotal=LeadTotal+LayoutFLenTotal+LayoutVLenTotal
			16	8	LeadTotal
			24	8	LayoutFLenTotal=NRecords*RecordFLen*
			32	8	LayoutVLenTotal=NRecordsRecordVLen
			40	8	NRecords
			48	8	RecordFLen

BLOCK	BLOCK START	BLOCK LENGTH	START	LENGTH	PURPOSE					
			56	8	RecordVLen					
FLen Region	LeadTotal	LayoutFLenTotal	0	LayoutFLenTotal	The densely populated rectangular region, RecordFLen bytes for each of the NRecords	VLen Region	LeadTotal+LayoutVLenTotal	LayoutVLenTotal	0	LayoutVLenTotal

[TODO] For a WDSC layout, there is only one record, it's fixed length region is the space for the WDSH and the variable length region holds the corresponding WDSD.

DBB.Default

Default

FieldBaseMD

FieldBaseMD.Default

FieldMD

FieldMDEnums

FieldMDExt

FieldName

JniPMML

JniPMMLItem

The JniPMMLItem class is mirrored on the C# and Java sides, but with a few specific differences: On the C# side, JniPMMLItem also works as an IExcelObservable object handle, while also holding the information needed to pass data to Java. The handle storage is effectively handled on the Java side.

JniPMMLItem.__ConfigMatter

JniPMMLItem.__InputMatter

JniPMMLItem.__OutputMatter

JniPMMLItem.__PMMLMatter

RecordSet

RecordSetMD

RecordSetMD.__ModeMatter

RecordSetMD.__SchemaMatter

RecordSetMDEnums

RecordSetMDExt

Util

WranglerDBB

WranglerDBB.__DBBMatter

WranglerFlatFile

WranglerHDF5

WranglerHDF5.HDF5DataType

WranglerXSD

Interfaces

FieldMDIKey<T>

Enums

FieldMDEnums.eDTyp

FieldMDEnums.eRTyp

RecordSetMDEnums.eMode

RecordSetMDEnums.eSchemaType

RecordSetMDEnums.eType

Class DBB

DBB is short for Direct ByteBuffer, which on the Java side, is used as the internal data memory map which is being used to hand off data between C# and Java.

Despite the overhead on the both the C# and Java sides, the effective use is as a contiguous array of byte data. The main block can be thought of as a rectangular region for a table of data where each column has the same width in bytes (variable length fields are a long's width pointing to variable space). This is essentially the way HDF.PInvoke packs memory before bulk writing a compound dataset. Here, we add leading bytes for the essential layout and a block of space after the rectangular region with enough space for the variable length fields.

There are also three layouts:

- WDSH
  a header layout which includes a record for each column/member/field
- WDSD
  a record set layout associated with a header
- WDSC
  [TODO] a combined layout, with leading bytes, space for the header, and space for the record set

The layout for the leading bytes is common across all three and takes 8*8=64 bytes:

START	LENGTH	PURPOSE
0	8	The layout style, 8 bytes for 4 UTF-16 characters for WDSH/WDSD/WDSC
8	8	The total number of bytes required for this layout
16	8	The number of leading bytes before the fixed length region begins
24	8	The total number of bytes for the rectangular or fixed length region
32	8	The total number of bytes for the variable length region
40	8	The number of records (or rows) of data communicated
48	8	The fixed number of bytes per record
56	8	The maximum variable number of bytes each record may require

WDSH, the header layout needs to communicate just enough information to describe the core meta data for each column. This includes its name, a possible second name (generally used to point to a pre-mapped FieldName used in the PMML dictionary), and several additional meta data ints and longs. 40 fixed length bytes per column are for:

START	LENGTH	PURPOSE
0	8	Name variable length string pointer (long)
8	8	Optional second name variable length string pointer (long), 0 if not used/mapped
16	4	integer for data type code (see enum eDTyp)
20	8	long for number of bytes column data requires in the contiguous portion (variable length fields will have no more than 8 for a pointer to the data)
28	8	long for maximum byte length of data (fixed length strings are packed directly, null padded, variable length strings represent a long (IntPtr) to other memory)
36	4	(extra integer space, not used)

For variable length strings, the rectangular space will hold the offset from the start of the leader bytes of the space holding the string. For allocation purposes, each string could be expected to occupy 4+MaxByteLength. When a string is greater than MaxByteLength, it will be trimmed to MaxByteLength-2 to null terminate, this may change, but any 2 (UTF-16) consecutive 0 bytes will be taken as a null termination.

For further illustration, the WDSD layout including leading bytes is:

BLOCK	BLOCK START	BLOCK LENGTH	START	LENGTH	PURPOSE
Header	0	LeadTotal=64	0	8	UTF-16 bytes for WDSD
			8	8	LayoutTotal=LeadTotal+LayoutFLenTotal+LayoutVLenTotal
			16	8	LeadTotal
			24	8	LayoutFLenTotal=NRecords*RecordFLen*
			32	8	LayoutVLenTotal=NRecordsRecordVLen
			40	8	NRecords
			48	8	RecordFLen
			56	8	RecordVLen

BLOCK	BLOCK START	BLOCK LENGTH	START	LENGTH	PURPOSE					
FLen Region	LeadTotal	LayoutFLenTotal	0	LayoutFLenTotal	The densely populated rectangular region, RecordFLen bytes for each of the NRecords	VLen Region	LeadTotal+LayoutVLenTotal	LayoutVLenTotal	0	LayoutVLenTotal

[TODO] For a WDSC layout, there is only one record, it's fixed length region is the space for the WDSH and the variable length region holds the corresponding WDSD.

Inheritance

System.Object
DBB

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class DBB
```

Constructors

DBB()

Declaration

```
public DBB()
```

DBB(DBB, Boolean)

Declaration

```
public DBB(DBB arg, bool bJustData)
```

Parameters

TYPE	NAME	DESCRIPTION
DBB	arg	
System.Boolean	bJustData	

DBB(ref Byte[])

Declaration

```
public DBB(ref byte[] arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Byte[]	arg	

DBB(ref Byte[], Int32, Boolean)

Declaration

```
public DBB(ref byte[] arg, int offset, bool bIsBigEndian)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Byte[]	arg	
System.Int32	offset	
System.Boolean	bIsBigEndian	

DBB(ref Byte[], Int32, Int32, Boolean)

Declaration

```
public DBB(ref byte[] arg, int offset, int length, bool bIsBigEndian)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Byte[]	arg	

TYPE	NAME	DESCRIPTION
System.Int32	offset	
System.Int32	length	
System.Boolean	bIsBigEndian	

Fields

bHasFLenVLenSplit

Declaration

```
public bool bHasFLenVLenSplit
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

bHasLeaders

Declaration

```
public bool bHasLeaders
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

bIsBigEndian

Declaration

```
public bool bIsBigEndian
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

bIsReadOnly

Declaration

```
public bool bIsReadOnly
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

data

Declaration

```
public byte[] data
```

Field Value

TYPE	DESCRIPTION
System.Byte[]	

datawrap

Declaration

```
public byte[] datawrap
```

Field Value

TYPE	DESCRIPTION
System.Byte[]	

flenlength

Declaration

```
public long flenlength
```

Field Value

TYPE	DESCRIPTION
System.Int64	

flenoffset

Declaration

```
public long flenoffset
```

Field Value

TYPE	DESCRIPTION
System.Int64	

flenptr

Declaration

```
public long flenptr
```

Field Value

TYPE	DESCRIPTION
System.Int64	

LayoutStyle

Declaration

```
public string LayoutStyle
```

Field Value

TYPE	DESCRIPTION
System.String	

Length

Declaration

```
public long Length
```

Field Value

TYPE	DESCRIPTION
System.Int64	

nDBBFLenBytes

Declaration

```
public long nDBBFLenBytes
```

Field Value

TYPE	DESCRIPTION
System.Int64	

nDBBLeadingBytes

Declaration

```
public long nDBBLeadingBytes
```

Field Value

TYPE	DESCRIPTION
System.Int64	

nDBBRequiredBytes

Declaration

```
public long nDBBRequiredBytes
```

Field Value

TYPE	DESCRIPTION
System.Int64	

nDBBVLenBytes

Declaration

```
public long nDBBVLenBytes
```

Field Value

TYPE	DESCRIPTION
System.Int64	

nRecordFLenBytes

Declaration

```
public long nRecordFLenBytes
```

Field Value

TYPE	DESCRIPTION
System.Int64	

nRecords

Declaration

```
public long nRecords
```

Field Value

TYPE	DESCRIPTION
System.Int64	

nRecordVLenBytes

Declaration

```
public long nRecordVLenBytes
```

Field Value

TYPE	DESCRIPTION
System.Int64	

offset

Declaration

```
public long offset
```

Field Value

TYPE	DESCRIPTION
System.Int64	

ptr

Declaration

```
public long ptr
```

Field Value

TYPE	DESCRIPTION
System.Int64	

vlenlength

Declaration

```
public long vlenlength
```

Field Value

TYPE	DESCRIPTION
System.Int64	

vlenoffset

Declaration

```
public long vlenoffset
```

Field Value

TYPE	DESCRIPTION
System.Int64	

vlenptr

Declaration

```
public long vlenptr
```

Field Value

TYPE	DESCRIPTION
System.Int64	

Methods

cAsBigEndian()

Declaration

```
public DBB cAsBigEndian()
```

Returns

TYPE	DESCRIPTION
DBB	

cAsHDF5BulkCompoundDSWriteLayout(Int64, Int64)

Declaration

```
public DBB cAsHDF5BulkCompoundDSWriteLayout(long nRecords, long nRecordFLenBytes)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int64	nRecords	
System.Int64	nRecordFLenBytes	

Returns

TYPE	DESCRIPTION
DBB	

cAsNotBigEndian()

Declaration

```
public DBB cAsNotBigEndian()
```

Returns

TYPE	DESCRIPTION
DBB	

cAsReadOnly()

Declaration

```
public DBB cAsReadOnly()
```

Returns

TYPE	DESCRIPTION
DBB	

cAsSimple()

Declaration

```
public DBB cAsSimple()
```

Returns

TYPE	DESCRIPTION
DBB	

cAsUsualLayout(String, Int64, Int64, Int64)

Declaration

```
public DBB cAsUsualLayout(string LayoutStyle, long nRecords, long nRecordFLenBytes, long nRecordVLenBytes)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	LayoutStyle	
System.Int64	nRecords	
System.Int64	nRecordFLenBytes	
System.Int64	nRecordVLenBytes	

Returns

TYPE	DESCRIPTION
DBB	

cAsUsualLayout(String, Int64, Int64, Int64, Int64)

Declaration

```
public DBB cAsUsualLayout(string LayoutStyle, long nLeadingBytes, long nRecords, long nRecordFLenBytes, long nRecordVLenBytes)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	LayoutStyle	

TYPE	NAME	DESCRIPTION
System.Int64	nLeadingBytes	
System.Int64	nRecords	
System.Int64	nRecordFLenBytes	
System.Int64	nRecordVLenBytes	

Returns

TYPE	DESCRIPTION
DBB	

cReadExistingLayout()

Declaration

```
public DBB cReadExistingLayout()
```

Returns

TYPE	DESCRIPTION
DBB	

cWithLength(Int32)

Declaration

```
public DBB cWithLength(int length)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	length	

Returns

TYPE	DESCRIPTION
DBB	

cWithOffset(Int32)

Declaration

```
public DBB cWithOffset(int offset)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	offset	

Returns

TYPE	DESCRIPTION
DBB	

cWrap(ref Byte[], Int32, Boolean)

Declaration

```
public DBB cWrap(ref byte[] arg, int offset, bool bIsBigEndian)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Byte[]	arg	
System.Int32	offset	
System.Boolean	bIsBigEndian	

Returns

TYPE	DESCRIPTION
DBB	

cWrap(ref Byte[], Int32, Int32, Boolean)

Declaration

```
public DBB cWrap(ref byte[] arg, int offset, int length, bool bIsBigEndian)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Byte[]	arg	
System.Int32	offset	
System.Int32	length	
System.Boolean	bIsBigEndian	

Returns

TYPE	DESCRIPTION
DBB	

Dispose()

Declaration

```
public void Dispose()
```

Finalize()

Declaration

```
protected void Finalize()
```

GetLayerByte(Int32)

Declaration

```
public byte GetLayerByte(int layer)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	

Returns

TYPE	DESCRIPTION
System.Byte	

GetLayerByteAt(Int32, Int64)

Declaration

```
public byte GetLayerByteAt(int layer, long arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int64	arg	

Returns

TYPE	DESCRIPTION
System.Byte	

GetLayerDouble(Int32)

Declaration

```
public double? GetLayerDouble(int layer)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	

Returns

TYPE	DESCRIPTION
System.Nullable<System.Double>	

GetLayerDouble(Int32, Int64)

Declaration

```
public double? GetLayerDouble(int layer, long atarg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	

TYPE	NAME	DESCRIPTION
System.Int64	atarg	

Returns

TYPE	DESCRIPTION
System.Nullable<System.Double>	

GetLayerFLenString(Int32, Int64)

Declaration

```
public string GetLayerFLenString(int layer, long nByteMaxLength)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int64	nByteMaxLength	

Returns

TYPE	DESCRIPTION
System.String	

GetLayerFLenString(Int32, Int64, Int64)

Declaration

```
public string GetLayerFLenString(int layer, long atarg, long nByteMaxLength)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int64	atarg	
System.Int64	nByteMaxLength	

Returns

TYPE	DESCRIPTION
System.String	

GetLayerInt(Int32)

Declaration

```
public int? GetLayerInt(int layer)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	

Returns

TYPE	DESCRIPTION
System.Nullable<System.Int32>	

GetLayerInt(Int32, Int64)

Declaration

```
public int? GetLayerInt(int layer, long atarg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int64	atarg	

Returns

TYPE	DESCRIPTION
System.Nullable<System.Int32>	

GetLayerLong(Int32)

Declaration

```
public long? GetLayerLong(int layer)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	

Returns

TYPE	DESCRIPTION
System.Nullable<System.Int64>	

GetLayerLong(Int32, Int64)

Declaration

```
public long? GetLayerLong(int layer, long atarg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int64	atarg	

Returns

TYPE	DESCRIPTION
System.Nullable<System.Int64>	

GetLayerVLenString(Int32, Int64)

Declaration

```
public string GetLayerVLenString(int layer, long nByteMaxLength)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int64	nByteMaxLength	

Returns

TYPE	DESCRIPTION
System.String	

GetLayerVLenString(Int32, Int64, Int64)

Declaration

```
public string GetLayerVLenString(int layer, long atarg, long nByteMaxLength)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int64	atarg	
System.Int64	nByteMaxLength	

Returns

TYPE	DESCRIPTION
System.String	

isDirect()

Declaration

```
public bool isDirect()
```

Returns

TYPE	DESCRIPTION
System.Boolean	

isValid()

Declaration

```
public bool isValid()
```

Returns

TYPE	DESCRIPTION
System.Boolean	

position(Int64, Int64, Int64)

Declaration

```
public void position(long ptr, long flenptr, long vlenptr)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int64	ptr	
System.Int64	flenptr	
System.Int64	vlenptr	

PutLayerBytes(Int32, Byte[])

Declaration

```
public void PutLayerBytes(int layer, byte[] value)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Byte[]	value	

PutLayerBytes(Int32, Int64, Byte[])

Declaration

```
public void PutLayerBytes(int layer, long atarg, byte[] value)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int64	atarg	
System.Byte[]	value	

PutLayerDouble(Int32, Double)

Declaration

```
public void PutLayerDouble(int layer, double value)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Double	value	

PutLayerDouble(Int32, Int64, Double)

Declaration

```
public void PutLayerDouble(int layer, long atarg, double value)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int64	atarg	
System.Double	value	

PutLayerDouble(Int32, Int64, Object)

Declaration

```
public void PutLayerDouble(int layer, long atarg, object obj)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int64	atarg	

TYPE	NAME	DESCRIPTION
System.Object	obj	

PutLayerDouble(Int32, Object)

Declaration

```
public void PutLayerDouble(int layer, object obj)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Object	obj	

PutLayerFLenString(Int32, Int64, String, Int32, Int32)

Declaration

```
public void PutLayerFLenString(int layer, long atarg, string value, int nByteMaxLength, int nZeroBytes)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int64	atarg	
System.String	value	
System.Int32	nByteMaxLength	
System.Int32	nZeroBytes	

PutLayerFLenString(Int32, String, Int32, Int32)

Declaration

```
public void PutLayerFLenString(int layer, string value, int nByteMaxLength, int nZeroBytes)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.String	value	
System.Int32	nByteMaxLength	
System.Int32	nZeroBytes	

PutLayerInt(Int32, Int32)

Declaration

```
public void PutLayerInt(int layer, int value)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int32	value	

PutLayerInt(Int32, Int64, Int32)

Declaration

```
public void PutLayerInt(int layer, long atarg, int value)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int64	atarg	
System.Int32	value	

PutLayerInt(Int32, Int64, Object)

Declaration

```
public void PutLayerInt(int layer, long atarg, object obj)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int64	atarg	
System.Object	obj	

PutLayerInt(Int32, Object)

Declaration

```
public void PutLayerInt(int layer, object obj)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Object	obj	

PutLayerLong(Int32, Int64)

Declaration

```
public void PutLayerLong(int layer, long value)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int64	value	

PutLayerLong(Int32, Int64, Int64)

Declaration

```
public void PutLayerLong(int layer, long atarg, long value)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int64	atarg	
System.Int64	value	

PutLayerLong(Int32, Int64, Object)

Declaration

```
public void PutLayerLong(int layer, long atarg, object obj)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int64	atarg	
System.Object	obj	

PutLayerLong(Int32, Object)

Declaration

```
public void PutLayerLong(int layer, object obj)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Object	obj	

PutLayerVLenString(Int32, Int64, String, Int32, Int32)

Declaration

```
public void PutLayerVLenString(int layer, long atarg, string value, int nByteMaxLength, int nZeroBytes)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	

TYPE	NAME	DESCRIPTION
System.Int64	atarg	
System.String	value	
System.Int32	nByteMaxLength	
System.Int32	nZeroBytes	

PutLayerVLenString(Int32, String, Int32, Int32)

Declaration

```
public void PutLayerVLenString(int layer, string value, int nByteMaxLength, int nZeroBytes)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.String	value	
System.Int32	nByteMaxLength	
System.Int32	nZeroBytes	

PutLayerZeros(Int32, Int32)

Declaration

```
public void PutLayerZeros(int layer, int value)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int32	value	

PutLayerZeros(Int32, Int64, Int32)

Declaration

```
public void PutLayerZeros(int layer, long atarg, int value)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	layer	
System.Int64	atarg	
System.Int32	value	

Reset()

Declaration

```
public void Reset()
```

Wrap(ref Byte[])

Declaration

```
public DBB Wrap(ref byte[] arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Byte[]	arg	

Returns

TYPE	DESCRIPTION
DBB	

Extension Methods

JavaLikeExtensions.toString(Object)

Class DBB.Default

Inheritance

System.Object
DBB.Default

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public static class Default
```

Fields

nLeadingBytes

Declaration

```
public static long nLeadingBytes
```

Field Value

TYPE	DESCRIPTION
System.Int64	

Class Default

Inheritance

System.Object
Default

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class Default
```

Fields

anyVLenRead

Declaration

```
public static bool anyVLenRead
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

anyVLenWrite

Declaration

```
public static bool anyVLenWrite
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

HeaderStringMaxLength

Declaration

```
public static int HeaderStringMaxLength
```

Field Value

TYPE	DESCRIPTION
System.Int32	

Declaration

```
public static bool ISCSHARP
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

ISJAVA

Declaration

```
public static bool ISJAVA
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

StringMaxLength

Declaration

```
public static int StringMaxLength
```

Field Value

TYPE	DESCRIPTION
System.Int32	

Extension Methods

JavaLikeExtensions.toString(Object)

Class FieldBaseMD

Inheritance

System.Object
FieldBaseMD
FieldMD

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class FieldBaseMD
```

Constructors

FieldBaseMD()

Declaration

```
public FieldBaseMD()
```

FieldBaseMD(FieldBaseMD)

Declaration

```
public FieldBaseMD(FieldBaseMD arg)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldBaseMD	arg	

FieldBaseMD(String, FieldMDEnums.eDTyp)

Declaration

```
public FieldBaseMD(string _Name, FieldMDEnums.eDTyp _DTyp)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	_Name	
FieldMDEnums.eDTyp	_DTyp	

FieldBaseMD(String, FieldMDEnums.eDTyp, Int32)

Declaration

```
public FieldBaseMD(string Name, FieldMDEnums.eDTyp DTyp, int StringMaxLength)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	Name	
FieldMDEnums.eDTyp	DTyp	
System.Int32	StringMaxLength	

FieldBaseMD(String, Int32, Int32, Int32, Int32)

Declaration

```
public FieldBaseMD(string Name, int hclass, int hlength, int horder, int hsign)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	Name	
System.Int32	hclass	
System.Int32	hlength	
System.Int32	horder	
System.Int32	hsign	

Fields

ByteMaxLength

Declaration

```
public long ByteMaxLength
```

Field Value

TYPE	DESCRIPTION
System.Int64	

ByteMemLength

Declaration

```
public long ByteMemLength
```

Field Value

TYPE	DESCRIPTION
System.Int64	

ByteMemOffset

Declaration

```
public long ByteMemOffset
```

Field Value

TYPE	DESCRIPTION
System.Int64	

DTyp

Declaration

```
public FieldMDEnums.eDTyp DTyp
```

Field Value

TYPE	DESCRIPTION
FieldMDEnums.eDTyp	

ExternalDTyp

Declaration

```
public FieldMDEnums.eDTyp ExternalDTyp
```

Field Value

TYPE	DESCRIPTION
FieldMDEnums.eDTyp	

Format

Declaration

```
public string Format
```

Field Value

TYPE	DESCRIPTION
System.String	

HDF5DataType

Declaration

```
public WranglerHDF5.HDF5DataType HDF5DataType
```

Field Value

TYPE	DESCRIPTION
WranglerHDF5.HDF5DataType	

Name

Declaration

```
public string Name
```

Field Value

TYPE	DESCRIPTION
System.String	

RTyp

Declaration

```
public FieldMDEnums.eRTyp RTyp
```

Field Value

TYPE	DESCRIPTION
FieldMDEnums.eRTyp	

StringMaxLength

Declaration

```
public int StringMaxLength
```

Field Value

TYPE	DESCRIPTION
System.Int32	

Methods

Consistency()

Declaration

```
public void Consistency()
```

Copy(FieldBaseMD)

Declaration

```
public void Copy(FieldBaseMD arg)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldBaseMD	arg	

Equals(FieldBaseMD)

Declaration

```
public bool Equals(FieldBaseMD arg)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldBaseMD	arg	

Returns

TYPE	DESCRIPTION
System.Boolean	

FLenByteLength()

Declaration

```
public long FLenByteLength()
```

Returns

TYPE	DESCRIPTION
System.Int64	

isMappedToHDF5DataType()

Declaration

```
public bool isMappedToHDF5DataType()
```

Returns

TYPE	DESCRIPTION
System.Boolean	

isVLen()

Declaration

```
public bool isVLen()
```

Returns

TYPE	DESCRIPTION
System.Boolean	

MapToHDF5DataType(FieldMDEnums.eDTyp)

Declaration

```
public FieldBaseMD MapToHDF5DataType(FieldMDEnums.eDTyp DTyp)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eDTyp	DTyp	

Returns

TYPE	DESCRIPTION
FieldBaseMD	

MapToHDF5DataType(FieldMDEnums.eDTyp, Int32, Boolean)

Declaration

```
public FieldBaseMD MapToHDF5DataType(FieldMDEnums.eDTyp DTyp, int nStringMaxLength, bool anyVLen)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eDTyp	DTyp	
System.Int32	nStringMaxLength	
System.Boolean	anyVLen	

Returns

TYPE	DESCRIPTION
FieldBaseMD	

MapToHDF5DataType(Int32, Int32, Int32, Int32)

Declaration

```
public FieldBaseMD MapToHDF5DataType(int hclass, int hlength, int horder, int hsign)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	hclass	
System.Int32	hlength	
System.Int32	horder	
System.Int32	hsign	

Returns

TYPE	DESCRIPTION
FieldBaseMD	

MapToHDF5DataType(Int64)

Declaration

```
public FieldBaseMD MapToHDF5DataType(long arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int64	arg	

Returns

TYPE	DESCRIPTION
FieldBaseMD	

Extension Methods

JavaLikeExtensions.toString(Object)

Class FieldBaseMD.Default

Inheritance

System.Object
FieldBaseMD.Default

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public static class Default
```

Fields

HeaderStringMaxLength

Declaration

```
public static int HeaderStringMaxLength
```

Field Value

TYPE	DESCRIPTION
System.Int32	

StringMaxLength

Declaration

```
public static int StringMaxLength
```

Field Value

TYPE	DESCRIPTION
System.Int32	

Class FieldMD

Inheritance

System.Object
FieldBaseMD
FieldMD

Implements

FieldMDIKey<FieldName>

Inherited Members

FieldBaseMD.Name
FieldBaseMD.DTyp
FieldBaseMD.RTyp
FieldBaseMD.StringMaxLength
FieldBaseMD.ByteMaxLength
FieldBaseMD.ByteMemLength
FieldBaseMD.ExternalDTyp
FieldBaseMD.ByteMemOffset
FieldBaseMD.Format
FieldBaseMD.HDF5DataType
FieldBaseMD.Equals(FieldBaseMD)
FieldBaseMD.Copy(FieldBaseMD)
FieldBaseMD.FLenByteLength()
FieldBaseMD.isVLen()
FieldBaseMD.Consistency()
FieldBaseMD.isMappedToHDF5DataType()
FieldBaseMD.MapToHDF5DataType(FieldMDEnums.eDTyp)
FieldBaseMD.MapToHDF5DataType(FieldMDEnums.eDTyp, Int32, Boolean)
FieldBaseMD.MapToHDF5DataType(Int32, Int32, Int32, Int32)
FieldBaseMD.MapToHDF5DataType(Int64)
System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class FieldMD : FieldBaseMD, FieldMDIKey<FieldName>
```

Constructors

FieldMD()

Declaration

```
public FieldMD()
```

FieldMD(FieldMD)

Declaration

```
public FieldMD(FieldMD arg)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMD	arg	

Fields

MapKey

Declaration

```
public FieldName MapKey
```

Field Value

TYPE	DESCRIPTION
FieldName	

Methods

Copy(FieldMD)

Declaration

```
public void Copy(FieldMD arg)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMD	arg	

Equals(FieldMD)

Declaration

```
public bool Equals(FieldMD arg)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMD	arg	

Returns

TYPE	DESCRIPTION
System.Boolean	

hasMapKey()

Declaration

```
public bool hasMapKey()
```

Returns

TYPE	DESCRIPTION
System.Boolean	

MappedKey()

Declaration

```
public FieldName MappedKey()
```

Returns

TYPE	DESCRIPTION
FieldName	

MappedKeyValue()

Declaration

```
public string MappedKeyValue()
```

Returns

TYPE	DESCRIPTION
System.String	

MapToMapKey(FieldName)

Declaration

```
public void MapToMapKey(FieldName aFieldName)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldName	aFieldName	

MapToMapKey(String)

Declaration

```
public FieldMD MapToMapKey(string aFieldStringName)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aFieldStringName	

Returns

TYPE	DESCRIPTION
FieldMD	

Implements

FieldMDIKey<T>

Extension Methods

JavaLikeExtensions.toString(Object)

Class FieldMDEnums

Inheritance

System.Object

FieldMDEnums

Inherited Members

System.Object.ToString()

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class FieldMDEnums
```

Extension Methods

JavaLikeExtensions.toString(Object)

Enum FieldMDEnums.eDTyp

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public enum eDTyp
```

Fields

NAME	DESCRIPTION
Bln	
Byt	
Dbl	
Dte	
DTm	
Int	
Lng	
Str	
Unk	
VLS	

Extension Methods

JavaLikeExtensions.toString()
FieldMDExt.FromInt(Int32)
FieldMDExt.AsInt()
FieldMDExt.eDTyp_AsInt()
FieldMDExt.toString()
FieldMDExt.equals(FieldMDEnums.eDTyp)
FieldMDExt.eDTyp_bIn(FieldMDEnums.eDTyp[])
FieldMDExt.bIn(FieldMDEnums.eDTyp[])
FieldMDExt.isString()
FieldMDExt.isNumeric()

Enum FieldMDEnums.eRTyp

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public enum eRTyp
```

Fields

NAME	DESCRIPTION
Cluster	
Feature	
Output	
Target	
Unknown	

Extension Methods

JavaLikeExtensions.toString()
FieldMDExt.FromInt(Int32)
FieldMDExt.AsInt()
FieldMDExt.eRTyp_AsInt()
FieldMDExt.toString()
FieldMDExt.equals(FieldMDEnums.eRTyp)
FieldMDExt.eRTyp_bIn(FieldMDEnums.eRTyp[])
FieldMDExt.bIn(FieldMDEnums.eRTyp[])

Class FieldMDExt

Inheritance

System.Object
FieldMDExt

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public static class FieldMDExt
```

Methods

AsInt(FieldMDEnums.eDTyp)

Declaration

```
public static int AsInt(this FieldMDEnums.eDTyp self)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eDTyp	self	

Returns

TYPE	DESCRIPTION
System.Int32	

AsInt(FieldMDEnums.eRTyp)

Declaration

```
public static int AsInt(this FieldMDEnums.eRTyp self)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eRTyp	self	

Returns

TYPE	DESCRIPTION
System.Int32	

bIn(FieldMDEnums.eDTyp, FieldMDEnums.eDTyp[])

Declaration

```
public static bool bIn(this FieldMDEnums.eDTyp self, params FieldMDEnums.eDTyp[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eDTyp	self	
FieldMDEnums.eDTyp[]	args	

Returns

TYPE	DESCRIPTION
System.Boolean	

bIn(FieldMDEnums.eRTyp, FieldMDEnums.eRTyp[])

Declaration

```
public static bool bIn(this FieldMDEnums.eRTyp self, params FieldMDEnums.eRTyp[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eRTyp	self	
FieldMDEnums.eRTyp[]	args	

Returns

TYPE	DESCRIPTION
System.Boolean	

eDTyp_AsInt(FieldMDEnums.eDTyp)

Declaration

```
public static int eDTyp_AsInt(this FieldMDEnums.eDTyp self)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eDTyp	self	

Returns

TYPE	DESCRIPTION
System.Int32	

eDTyp_bIn(FieldMDEnums.eDTyp, FieldMDEnums.eDTyp[])

Declaration

```
public static bool eDTyp_bIn(this FieldMDEnums.eDTyp self, params FieldMDEnums.eDTyp[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eDTyp	self	
FieldMDEnums.eDTyp[]	args	

Returns

TYPE	DESCRIPTION
System.Boolean	

eDTyp_FromAlias(String, ref Int32[])

Declaration

```
public static FieldMDEnums.eDTyp eDTyp_FromAlias(string arg, ref int[] typl)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	
System.Int32[]	typl	

Returns

TYPE	DESCRIPTION
FieldMDEnums.eDTyp	

eDTyp_FromInt(Int32)

Declaration

```
public static FieldMDEnums.eDTyp eDTyp_FromInt(int arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	arg	

Returns

TYPE	DESCRIPTION
FieldMDEnums.eDTyp	

equals(FieldMDEnums.eDTyp, FieldMDEnums.eDTyp)

Declaration

```
public static bool equals(this FieldMDEnums.eDTyp self, FieldMDEnums.eDTyp arg)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eDTyp	self	
FieldMDEnums.eDTyp	arg	

Returns

TYPE	DESCRIPTION
System.Boolean	

equals(FieldMDEnums.eRTyp, FieldMDEnums.eRTyp)

Declaration

```
public static bool equals(this FieldMDEnums.eRTyp self, FieldMDEnums.eRTyp arg)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eRTyp	self	
FieldMDEnums.eRTyp	arg	

Returns

TYPE	DESCRIPTION
System.Boolean	

eRTyp_AsInt(FieldMDEnums.eRTyp)

Declaration

```
public static int eRTyp_AsInt(this FieldMDEnums.eRTyp self)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eRTyp	self	

Returns

TYPE	DESCRIPTION
System.Int32	

eRTyp_bIn(FieldMDEnums.eRTyp, FieldMDEnums.eRTyp[])

Declaration

```
public static bool eRTyp_bIn(this FieldMDEnums.eRTyp self, params FieldMDEnums.eRTyp[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eRTyp	self	
FieldMDEnums.eRTyp[]	args	

Returns

TYPE	DESCRIPTION
System.Boolean	

eRTyp_FromAlias(String)

Declaration

```
public static FieldMDEnums.eRTyp eRTyp_FromAlias(string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Returns

TYPE	DESCRIPTION
FieldMDEnums.eRTyp	

eRTyp_FromInt(Int32)

Declaration

```
public static FieldMDEnums.eRTyp eRTyp_FromInt(int arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	arg	

Returns

TYPE	DESCRIPTION
FieldMDEnums.eRTyp	

FromInt(FieldMDEnums.eDTyp, Int32)

Declaration

```
public static FieldMDEnums.eDTyp FromInt(this FieldMDEnums.eDTyp self, int arg)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eDTyp	self	

TYPE	NAME	DESCRIPTION
System.Int32	arg	

Returns

TYPE	DESCRIPTION
FieldMDEnums.eDTyp	

FromInt(FieldMDEnums.eRTyp, Int32)

Declaration

```
public static FieldMDEnums.eRTyp FromInt(this FieldMDEnums.eRTyp self, int arg)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eRTyp	self	
System.Int32	arg	

Returns

TYPE	DESCRIPTION
FieldMDEnums.eRTyp	

isNumeric(FieldMDEnums.eDTyp)

Declaration

```
public static bool isNumeric(this FieldMDEnums.eDTyp self)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eDTyp	self	

Returns

TYPE	DESCRIPTION
System.Boolean	

isString(FieldMDEnums.eDTyp)

Declaration

```
public static bool isString(this FieldMDEnums.eDTyp self)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eDTyp	self	

Returns

TYPE	DESCRIPTION
System.Boolean	

toString(FieldMDEnums.eDTyp)

Declaration

```
public static string toString(this FieldMDEnums.eDTyp self)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eDTyp	self	

Returns

TYPE	DESCRIPTION
System.String	

toString(FieldMDEnums.eRTyp)

Declaration

```
public static string toString(this FieldMDEnums.eRTyp self)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eRTyp	self	

Returns

TYPE	DESCRIPTION
System.String	

Interface FieldMDIKey<T>

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public interface FieldMDIKey<T>
```

Type Parameters

NAME	DESCRIPTION
T	

Methods

hasMapKey()

Declaration

```
bool hasMapKey()
```

Returns

TYPE	DESCRIPTION
System.Boolean	

MappedKey()

Declaration

```
T MappedKey()
```

Returns

TYPE	DESCRIPTION
T	

MapToMapKey(T)

Declaration

```
void MapToMapKey(T arg)
```

Parameters

TYPE	NAME	DESCRIPTION
T	arg	

Extension Methods

JavaLikeExtensions.toString(Object)

Class FieldName

Inheritance

System.Object
FieldName

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class FieldName
```

Constructors

FieldName()

Declaration

```
public FieldName()
```

FieldName(String)

Declaration

```
public FieldName(string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Methods

getValue()

Declaration

```
public string getValue()
```

Returns

TYPE	DESCRIPTION
System.String	

Extension Methods

JavaLikeExtensions.toString(Object)

Class JniPMML

Inheritance

System.Object
JniPMML

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class JniPMML
```

Constructors

JniPMML(JavaNativeInterface, IntPtr)

Declaration

```
public JniPMML(JavaNativeInterface Java, IntPtr java_init_classid)
```

Parameters

TYPE	NAME	DESCRIPTION
JavaNativeInterface	Java	
System.IntPtr	java_init_classid	

Fields

__Java

Declaration

```
public readonly JavaNativeInterface __Java
```

Field Value

TYPE	DESCRIPTION
JavaNativeInterface	

__java_init_classid

Declaration

```
public readonly IntPtr __java_init_classid
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

Item

Declaration

```
public Dictionary<int, JniPMMLItem> Item
```

Field Value

TYPE	DESCRIPTION
System.Collections.Generic.Dictionary<System.Int32, JniPMMLItem>	

Methods

Add(JniPMMLItem)

Declaration

```
public JniPMMLItem Add(JniPMMLItem aJniPMMLItem)
```

Parameters

TYPE	NAME	DESCRIPTION
JniPMMLItem	aJniPMMLItem	

Returns

TYPE	DESCRIPTION
JniPMMLItem	

CreateHandle(String, Object[], Func<String, Object[], JniPMMLItem>)

Declaration

```
public object CreateHandle(string aEvaluatorType, object[] args, Func<string, object[], JniPMMLItem> lFunc)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aEvaluatorType	
System.Object[]	args	
System.Func<System.String, System.Object[], JniPMMLItem>	lFunc	

Returns

TYPE	DESCRIPTION
System.Object	

CreateHandle(String, String, Object, String)

Declaration

```
public JniPMMLItem CreateHandle(string aEvaluatorType, string aTag, object bFileOrString, string src)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aEvaluatorType	
System.String	aTag	
System.Object	bFileOrString	
System.String	src	

Returns

TYPE	DESCRIPTION
JniPMMLItem	

Dispose()

Declaration

```
public void Dispose()
```

Finalize()

Declaration

```
protected void Finalize()
```

GetOrAddItemWithHandle(Int32)

Declaration

```
public JniPMMLItem GetOrAddItemWithHandle(int aHandle)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	aHandle	

Returns

TYPE	DESCRIPTION
JniPMMLItem	

GetOrAddItemWithTag(String)

Declaration

```
public JniPMMLItem GetOrAddItemWithTag(string aTag)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aTag	

Returns

TYPE	DESCRIPTION
JniPMMLItem	

Handle(String)

Declaration

```
public int Handle(string aTag)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aTag	

Returns

TYPE	DESCRIPTION
System.Int32	

isHandle(Int32)

Declaration

```
public bool isHandle(int aHandle)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	aHandle	

Returns

TYPE	DESCRIPTION
System.Boolean	

isTag(String)

Declaration

```
public bool isTag(string aTag)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aTag	

Returns

TYPE	DESCRIPTION
System.Boolean	

Remove(JniPMMLItem)

Declaration

```
public void Remove(JniPMMLItem arg)
```

Parameters

TYPE	NAME	DESCRIPTION
JniPMMLItem	arg	

Remove(Int32)

Declaration

```
public void Remove(int aHandle)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	aHandle	

Remove(String)

Declaration

```
public void Remove(string aTag)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aTag	

Tag(Int32)

Declaration

```
public string Tag(int aHandle)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	aHandle	

Returns

TYPE	DESCRIPTION
System.String	

TryGetObject(Int32, out JniPMMLItem)

Declaration

```
public bool TryGetObject(int aHandle, out JniPMMLItem value)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	aHandle	
JniPMMLItem	value	

Returns

TYPE	DESCRIPTION
System.Boolean	

TryGetObject(String, out JniPMMLItem)

Declaration

```
public bool TryGetObject(string aTag, out JniPMMLItem value)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aTag	
JniPMMLItem	value	

Returns

TYPE	DESCRIPTION
System.Boolean	

Extension Methods

JavaLikeExtensions.toString(Object)

Class JniPMMLItem

The JniPMMLItem class is mirrored on the C# and Java sides, but with a few specific differences: On the C# side, JniPMMLItem also works as an IExcelObservable object handle, while also holding the information needed to pass data to Java. The handle storage is effectively handled on the Java side.

Inheritance

System.Object
JniPMMLItem

Implements

ExcelDna.Integration.IExcelObservable
System.IDisposable

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class JniPMMLItem : IExcelObservable, IDisposable
```

Constructors

JniPMMLItem(String, JniPMML)

Declaration

```
public JniPMMLItem(string aTag, JniPMML aJniPMML)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aTag	
JniPMML	aJniPMML	

Fields

__ExcelObserver

Declaration

```
public IExcelObserver __ExcelObserver
```

Field Value

TYPE	DESCRIPTION
ExcelDna.Integration.IExcelObserver	

__Handle

Declaration

```
public readonly int __Handle
```

Field Value

TYPE	DESCRIPTION
System.Int32	

__JniPMML

Declaration

```
public JniPMML __JniPMML
```

Field Value

TYPE	DESCRIPTION
JniPMML	

__Tag

Declaration

```
public readonly string __Tag
```

Field Value

TYPE	DESCRIPTION
System.String	

ConfigMatter

Declaration

```
public JniPMMLItem.__ConfigMatter ConfigMatter
```

Field Value

TYPE	DESCRIPTION
JniPMMLItem.__ConfigMatter	

HandleMajorMinor

Declaration

```
public string HandleMajorMinor
```

Field Value

TYPE	DESCRIPTION
System.String	

InputMatter

Declaration

```
public JniPMMLItem.__InputMatter InputMatter
```

Field Value

TYPE	DESCRIPTION
JniPMMLItem.__InputMatter	

OutputMatter

Declaration

```
public JniPMMLItem.__OutputMatter OutputMatter
```

Field Value

TYPE	DESCRIPTION
JniPMMLItem.__OutputMatter	

PMMLMatter

Declaration

```
public JniPMMLItem.__PMMLMatter PMMLMatter
```

Field Value

TYPE	DESCRIPTION
JniPMMLItem.__PMMLMatter	

Methods

Dispose()

Declaration

```
public void Dispose()
```

Finalize()

Declaration

```
protected void Finalize()
```

Handle()

Declaration

```
public int Handle()
```

Returns

TYPE	DESCRIPTION
System.Int32	

PMMLDataFieldStringNames()

Declaration

```
public string[] PMMLDataFieldStringNames()
```

Returns

TYPE	DESCRIPTION
System.String[]	

PreDispose()

Declaration

```
public void PreDispose()
```

Reset()

Declaration

```
public void Reset()
```

Subscribe(IExcelObserver)

Declaration

```
public IDisposable Subscribe(IExcelObserver arg)
```

Parameters

TYPE	NAME	DESCRIPTION
ExcelDna.Integration.IExcelObserver	arg	

Returns

TYPE	DESCRIPTION
System.IDisposable	

Tag()

Declaration

```
public string Tag()
```

Returns

TYPE	DESCRIPTION
System.String	

UpdateHandleMajorMinor()

Declaration

```
public void UpdateHandleMajorMinor()
```

Implements

ExcelDna.Integration.IExcelObservable
System.IDisposable

Extension Methods

JavaLikeExtensions.toString(Object)

Class JniPMMLItem.__ConfigMatter

Inheritance

System.Object
JniPMMLItem.__ConfigMatter

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class __ConfigMatter
```

Fields

InternalString

Declaration

```
public string InternalString
```

Field Value

TYPE	DESCRIPTION
System.String	

Extension Methods

JavaLikeExtensions.toString(Object)

Class JniPMMLItem.__InputMatter

Inheritance

System.Object
JniPMMLItem.__InputMatter

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class __InputMatter
```

Fields

_XSDDoc

Declaration

```
public XmlDocument _XSDDoc
```

Field Value

TYPE	DESCRIPTION
System.Xml.XmlDocument	

_XSDFileName

Declaration

```
public string _XSDFileName
```

Field Value

TYPE	DESCRIPTION
System.String	

_XSDString

Declaration

```
public string _XSDString
```

Field Value

TYPE	DESCRIPTION
System.String	

RecordSet

Declaration

```
public RecordSet RecordSet
```

Field Value

TYPE	DESCRIPTION
RecordSet	

RecordSetMD

Declaration

```
public RecordSetMD RecordSetMD
```

Field Value

TYPE	DESCRIPTION
RecordSetMD	

Methods

Dispose()

Declaration

```
public void Dispose()
```

Finalize()

Declaration

```
protected void Finalize()
```

Extension Methods

JavaLikeExtensions.toString(Object)

Class JniPMMLItem.__OutputMatter

Inheritance

System.Object
JniPMMLItem.__OutputMatter

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class __OutputMatter
```

Fields

RecordSet

Declaration

```
public RecordSet RecordSet
```

Field Value

TYPE	DESCRIPTION
RecordSet	

RecordSetMD

Declaration

```
public RecordSetMD RecordSetMD
```

Field Value

TYPE	DESCRIPTION
RecordSetMD	

Methods

Dispose()

Declaration

```
public void Dispose()
```

Finalize()

Declaration

```
protected void Finalize()
```

Extension Methods

JavaLikeExtensions.toString(Object)

Class JniPMMLItem.__PMMLMatter

Inheritance

System.Object
JniPMMLItem.__PMMLMatter

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class __PMMLMatter
```

Fields

_XMLFileName

Declaration

```
public string _XMLFileName
```

Field Value

TYPE	DESCRIPTION
System.String	

_XMLString

Declaration

```
public string _XMLString
```

Field Value

TYPE	DESCRIPTION
System.String	

Doc

Declaration

```
public XmlDocument Doc
```

Field Value

TYPE	DESCRIPTION
System.Xml.XmlDocument	

Evaluator

Declaration

```
public XmlDocument Evaluator
```

Field Value

TYPE	DESCRIPTION
System.Xml.XmlDocument	

Methods

Dispose()

Declaration

```
public void Dispose()
```

Finalize()

Declaration

```
protected void Finalize()
```

Extension Methods

JavaLikeExtensions.toString(Object)

Class RecordSet

Inheritance

System.Object
RecordSet

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class RecordSet
```

Constructors

RecordSet()

Declaration

```
public RecordSet()
```

RecordSet(List<Map<FieldName, Object>>)

Declaration

```
public RecordSet(List<Map<FieldName, object>> _Records)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.List<Map<FieldName, System.Object>>	_Records	

Fields

isInput

Declaration

```
public bool isInput
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

Records

Declaration

```
public List<Map<FieldName, object>> Records
```

Field Value

TYPE	DESCRIPTION
System.Collections.Generic.List<Map<FieldName, System.Object>>	

Records_Orig

Declaration

```
public List<object[]> Records_Orig
```

Field Value

TYPE	DESCRIPTION
System.Collections.Generic.List<System.Object[]>	

Methods

cAsInput()

Declaration

```
public RecordSet cAsInput()
```

Returns

TYPE	DESCRIPTION
RecordSet	

cAsOutput()

Declaration

```
public RecordSet cAsOutput()
```

Returns

TYPE	DESCRIPTION
RecordSet	

cAsOutput(List<Map<FieldName, Object>>)

Declaration

```
public RecordSet cAsOutput(List<Map<FieldName, object>> _Records)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.List<Map<FieldName, System.Object>>	_Records	

Returns

TYPE	DESCRIPTION
RecordSet	

Dispose()

Declaration

```
public void Dispose()
```

Finalize()

Declaration

```
protected void Finalize()
```

isEmpty()

Declaration

```
public bool isEmpty()
```

Returns

TYPE	DESCRIPTION
System.Boolean	

mReadRecordSet(RecordSetMD)

Declaration

```
public RecordSet mReadRecordSet(RecordSetMD aRecordSetMD)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aRecordSetMD	

Returns

TYPE	DESCRIPTION
RecordSet	

mWriteRecordSet(RecordSetMD, RecordSetMD, RecordSet)

Declaration

```
public RecordSet mWriteRecordSet(RecordSetMD aOutputRecordSetMD, RecordSetMD aInputRecordSetMD, RecordSet
aInputRecordSet)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aOutputRecordSetMD	
RecordSetMD	aInputRecordSetMD	
RecordSet	aInputRecordSet	

Returns

TYPE	DESCRIPTION
RecordSet	

Extension Methods

JavaLikeExtensions.toString(Object)

Class RecordSetMD

Inheritance

System.Object
RecordSetMD

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class RecordSetMD
```

Constructors

RecordSetMD(RecordSetMDEnums.eMode)

Declaration

```
public RecordSetMD(RecordSetMDEnums.eMode arg)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eMode	arg	

Fields

Column

Declaration

```
public FieldMD[] Column
```

Field Value

TYPE	DESCRIPTION
FieldMD[]	

DBBMatter

Declaration

```
public WranglerDBB DBBMatter
```

Field Value

TYPE	DESCRIPTION
WranglerDBB	

DefaultHeaderMaxStringByteLength

Declaration

```
public static long DefaultHeaderMaxStringByteLength
```

Field Value

TYPE	DESCRIPTION
System.Int64	

DefaultHeaderMaxStringLength

Declaration

```
public static long DefaultHeaderMaxStringLength
```

Field Value

TYPE	DESCRIPTION
System.Int64	

DefaultMaxStringByteLength

Declaration

```
public static long DefaultMaxStringByteLength
```

Field Value

TYPE	DESCRIPTION
System.Int64	

DefaultMaxStringLength

Declaration

```
public static long DefaultMaxStringLength
```

Field Value

TYPE	DESCRIPTION
System.Int64	

FileMatter

Declaration

```
public WranglerFlatFile FileMatter
```

Field Value

TYPE	DESCRIPTION
WranglerFlatFile	

HDF5Matter

Declaration

```
public WranglerHDF5 HDF5Matter
```

Field Value

TYPE	DESCRIPTION
WranglerHDF5	

Mode

Declaration

```
public RecordSetMDEnums.eMode Mode
```

Field Value

TYPE	DESCRIPTION
RecordSetMDEnums.eMode	

ModeMatter

Declaration

```
public RecordSetMD.__ModeMatter ModeMatter
```

Field Value

TYPE	DESCRIPTION
RecordSetMD.__ModeMatter	

SchemaMatter

Declaration

```
public RecordSetMD.__SchemaMatter SchemaMatter
```

Field Value

TYPE	DESCRIPTION
RecordSetMD.__SchemaMatter	

SchemaType

Declaration

```
public RecordSetMDEnums.eSchemaType SchemaType
```

Field Value

TYPE	DESCRIPTION
RecordSetMDEnums.eSchemaType	

Type

Declaration

```
public RecordSetMDEnums.eType Type
```

Field Value

TYPE	DESCRIPTION
RecordSetMDEnums.eType	

Methods

cAs(RecordSetMDEnums.eType)

Declaration

```
public RecordSetMD cAs(RecordSetMDEnums.eType arg)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eType	arg	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cAs(RecordSetMDEnums.eType, RecordSetMDEnums.eSchemaType)

Declaration

```
public RecordSetMD cAs(RecordSetMDEnums.eType arg, RecordSetMDEnums.eSchemaType schema)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eType	arg	
RecordSetMDEnums.eSchemaType	schema	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cAs(RecordSetMDEnums.eType, RecordSetMDEnums.eSchemaType, Boolean, String)

Declaration

```
public RecordSetMD cAs(RecordSetMDEnums.eType arg, RecordSetMDEnums.eSchemaType schema, bool isFileName,
string schemadetails)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eType	arg	

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eSchemaType	schema	
System.Boolean	isFileName	
System.String	schemadetails	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cAsDlmFile(String)

Declaration

```
public RecordSetMD cAsDlmFile(string aFileName)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aFileName	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cAsDlmFile(String, String)

Declaration

```
public RecordSetMD cAsDlmFile(string aFileName, string dlm)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aFileName	
System.String	dlm	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cFromFile(String)

Declaration

```
public RecordSetMD cFromFile(string aFileName)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aFileName	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cRepeatInputSet()

Declaration

```
public RecordSetMD cRepeatInputSet()
```

Returns

TYPE	DESCRIPTION
RecordSetMD	

cRepeatInputSetWithSuffix(String)

Declaration

```
public RecordSetMD cRepeatInputSetWithSuffix(string aInputFieldSuffix)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aInputFieldSuffix	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cRepeatInputSetWithSuffix(String, String)

Declaration

```
public RecordSetMD cRepeatInputSetWithSuffix(string aInputFieldSuffix, string aCompositeFieldNameDlm)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aInputFieldSuffix	
System.String	aCompositeFieldNameDlm	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cSetHeaderBufferAs(DBB, Int32, Int32, Int32)

Declaration

```
public RecordSetMD cSetHeaderBufferAs(DBB arg, int nRecords, int nRecordCoreLength, int nRecordVariableLength)
```

Parameters

TYPE	NAME	DESCRIPTION
DBB	arg	
System.Int32	nRecords	
System.Int32	nRecordCoreLength	
System.Int32	nRecordVariableLength	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cSetHeaderBufferFrom(DBB)

Declaration

```
public RecordSetMD cSetHeaderBufferFrom(DBB arg)
```

Parameters

TYPE	NAME	DESCRIPTION
DBB	arg	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cSetRecordSetBufferAs(DBB)

Declaration

```
public RecordSetMD cSetRecordSetBufferAs(DBB arg)
```

Parameters

TYPE	NAME	DESCRIPTION
DBB	arg	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cSetRecordSetBufferAs(DBB, Int64, Int64, Int64, Int64, Int64)

Declaration

```
public RecordSetMD cSetRecordSetBufferAs(DBB arg, long nRecords, long nRecordCoreLength, long
nRecordVariableLength, long nCoreLength, long nTotalLength)
```

Parameters

TYPE	NAME	DESCRIPTION
DBB	arg	
System.Int64	nRecords	
System.Int64	nRecordCoreLength	
System.Int64	nRecordVariableLength	
System.Int64	nCoreLength	
System.Int64	nTotalLength	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cSetRecordSetBufferFrom(DBB)

Declaration

```
public RecordSetMD cSetRecordSetBufferFrom(DBB arg)
```

Parameters

TYPE	NAME	DESCRIPTION
DBB	arg	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cToFile(String)

Declaration

```
public RecordSetMD cToFile(string aFileName)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aFileName	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cWithCompositeFieldNameDlm(String)

Declaration

```
public RecordSetMD cWithCompositeFieldNameDlm(string aCompositeFieldNameDlm)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aCompositeFieldNameDlm	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cWithDataSetName(String)

Declaration

```
public RecordSetMD cWithDataSetName(string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cWithDelimiter(String)

Declaration

```
public RecordSetMD cWithDelimiter(string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cWithDlm(String)

Declaration

```
public RecordSetMD cWithDlm(string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cWithHeaderRow()

Declaration

```
public RecordSetMD cWithHeaderRow()
```

Returns

TYPE	DESCRIPTION
RecordSetMD	

cWithOutRepeatInputSet()

Declaration

```
public RecordSetMD cWithOutRepeatInputSet()
```

Returns

TYPE	DESCRIPTION
RecordSetMD	

cWithRecordElementName(String)

Declaration

```
public RecordSetMD cWithRecordElementName(string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cWithRecordSetAndRecordElementNames(String, String)

Declaration

```
public RecordSetMD cWithRecordSetAndRecordElementNames(string arg, string arg1)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	
System.String	arg1	

Returns

TYPE	DESCRIPTION
RecordSetMD	

cWithRecordSetElementName(String)

Declaration

```
public RecordSetMD cWithRecordSetElementName(string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Returns

TYPE	DESCRIPTION
RecordSetMD	

Dispose()

Declaration

```
public void Dispose()
```

Equals(RecordSetMD, Boolean)

Declaration

```
public bool Equals(RecordSetMD arg, bool bIgnoreMode)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	arg	

TYPE	NAME	DESCRIPTION
System.Boolean	bIgnoreMode	

Returns

TYPE	DESCRIPTION
System.Boolean	

Finalize()

Declaration

```
protected void Finalize()
```

isModeValid()

Declaration

```
public bool isModeValid()
```

Returns

TYPE	DESCRIPTION
System.Boolean	

mBytesRequired(Int64, out Int64, out Int64, out Int64)

Declaration

```
public void mBytesRequired(long nRecords, out long rsize, out long rflensize, out long rvlensize)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int64	nRecords	
System.Int64	rsize	
System.Int64	rflensize	
System.Int64	rvlensize	

mBytesRequired(Int64, out Int64, out Int64, out Int64, out Int64, out Int64, out Int64, out Int64, out Int64, out Int64, out Int64)

Declaration

```
public void mBytesRequired(long nRecords, out long csize, out long hsize, out long rsize, out long cleadsize,
out long hleadsize, out long hflensize, out long hvlensize, out long rleadsize, out long rflensize, out long
rvlensize)
```

Parameters

TYPE	NAME	DESCRIPTION

TYPE	NAME	DESCRIPTION
System.Int64	nRecords	
System.Int64	csize	
System.Int64	hsize	
System.Int64	rsize	
System.Int64	cleadsize	
System.Int64	hleadsize	
System.Int64	hflensize	
System.Int64	hvlensize	
System.Int64	rleadsize	
System.Int64	rflensize	
System.Int64	rvlensize	

mColumnConsistency()

Declaration

```
public RecordSetMD mColumnConsistency()
```

Returns

TYPE	DESCRIPTION
RecordSetMD	

mCopyColumnsFrom(RecordSetMD)

Declaration

```
public RecordSetMD mCopyColumnsFrom(RecordSetMD arg)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	arg	

Returns

TYPE	DESCRIPTION
RecordSetMD	

mPrepForOutput<T>(RecordSetMD, JniPMMLItem, List<Map<T, Object>>)

Declaration

```
public RecordSetMD mPrepForOutput<T>(RecordSetMD aInputRecordSetMD, JniPMMLItem aJniPMML, List<Map<T, object>>
Results)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aInputRecordSetMD	
JniPMMLItem	aJniPMML	
System.Collections.Generic.List<Map<T, System.Object>>	Results	

Returns

TYPE	DESCRIPTION
RecordSetMD	

Type Parameters

NAME	DESCRIPTION
T	

mReadMapFor(JniPMMLItem, PrintWriter, Boolean)

Declaration

```
public RecordSetMD mReadMapFor(JniPMMLItem aJniPMMLItem, PrintWriter pw, bool bFillDictionaryNames)
```

Parameters

TYPE	NAME	DESCRIPTION
JniPMMLItem	aJniPMMLItem	
PrintWriter	pw	
System.Boolean	bFillDictionaryNames	

Returns

TYPE	DESCRIPTION
RecordSetMD	

mUpdateWithPMMLSchema(JniPMMLItem)

Declaration

```
public void mUpdateWithPMMLSchema(JniPMMLItem aJniPMML)
```

Parameters

TYPE	NAME	DESCRIPTION
JniPMMLItem	aJniPMML	

mWriteMapToBuffer()

Declaration

```
public int mWriteMapToBuffer()
```

Returns

TYPE	DESCRIPTION
System.Int32	

mWritePrepFor(Int32)

Declaration

```
public RecordSetMD mWritePrepFor(int nRows)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	nRows	

Returns

TYPE	DESCRIPTION
RecordSetMD	

nColumns()

Declaration

```
public int nColumns()
```

Returns

TYPE	DESCRIPTION
System.Int32	

nColumns(FieldMDEnums.eRTyp)

Declaration

```
public int nColumns(FieldMDEnums.eRTyp arg)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eRTyp	arg	

Returns

TYPE	DESCRIPTION
System.Int32	

nHeaderByteMaxLength()

Declaration

```
public long nHeaderByteMaxLength()
```

Returns

TYPE	DESCRIPTION
System.Int64	

nHeaderStringMaxLength()

Declaration

```
public long nHeaderStringMaxLength()
```

Returns

TYPE	DESCRIPTION
System.Int64	

Extension Methods

JavaLikeExtensions.toString(Object)

Class RecordSetMD.__ModeMatter

Inheritance

System.Object
RecordSetMD.__ModeMatter

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class __ModeMatter
```

Fields

bRepeatInputFields

Declaration

```
public bool bRepeatInputFields
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

CompositeInputNameSuffix

Declaration

```
public string CompositeInputNameSuffix
```

Field Value

TYPE	DESCRIPTION
System.String	

CompositeNameDlm

Declaration

```
public string CompositeNameDlm
```

Field Value

TYPE	DESCRIPTION
System.String	

nInputFields

Declaration

```
public int nInputFields
```

Field Value

TYPE	DESCRIPTION
System.Int32	

OutputMaxStringLength

Declaration

```
public int OutputMaxStringLength
```

Field Value

TYPE	DESCRIPTION
System.Int32	

Methods

Equals(RecordSetMD.__ModeMatter)

Declaration

```
public bool Equals(RecordSetMD.__ModeMatter arg)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD.__ModeMatter	arg	

Returns

TYPE	DESCRIPTION
System.Boolean	

Extension Methods

JavaLikeExtensions.toString(Object)

Class RecordSetMD.__SchemaMatter

Inheritance

System.Object

WranglerXSD

RecordSetMD.__SchemaMatter

Inherited Members

WranglerXSD.mReadMapFor(RecordSetMD, JniPMMLItem, PrintWriter, Boolean)

WranglerXSD.mReadMapFor(XmlDocument, RecordSetMD, JniPMMLItem, PrintWriter, Boolean)

WranglerXSD.XSDHeader()

WranglerXSD.XSDRecordSet_Open(String, String)

WranglerXSD.XSDRecordSet_Close()

WranglerXSD.XSDFooter()

WranglerXSD.XSDColumn(String, String)

WranglerXSD.XSDTypes()

System.Object.ToString()

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class __SchemaMatter : WranglerXSD
```

Fields

InputSchema

Declaration

```
public XmlDocument InputSchema
```

Field Value

TYPE	DESCRIPTION
System.Xml.XmlDocument	

InputSchemaFileName

Declaration

```
public string InputSchemaFileName
```

Field Value

TYPE	DESCRIPTION
System.String	

InputSchemaString

Declaration

```
public string InputSchemaString
```

Field Value

TYPE	DESCRIPTION
System.String	

RecordElementName

Declaration

```
public string RecordElementName
```

Field Value

TYPE	DESCRIPTION
System.String	

RecordSetElementName

Declaration

```
public string RecordSetElementName
```

Field Value

TYPE	DESCRIPTION
System.String	

Methods

Dispose()

Declaration

```
public void Dispose()
```

Equals(RecordSetMD.__SchemaMatter)

Declaration

```
public bool Equals(RecordSetMD.__SchemaMatter arg)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD.__SchemaMatter	arg	

Returns

TYPE	DESCRIPTION
System.Boolean	

Finalize()

Declaration

```
protected void Finalize()
```

Extension Methods

JavaLikeExtensions.toString(Object)

Class RecordSetMDEnums

Inheritance

System.Object

RecordSetMDEnums

Inherited Members

System.Object.ToString()

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class RecordSetMDEnums
```

Extension Methods

JavaLikeExtensions.toString(Object)

Enum RecordSetMDEnums.eMode

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public enum eMode
```

Fields

NAME	DESCRIPTION
Input	
Internal	
Output	
Unknown	

Extension Methods

JavaLikeExtensions.toString()
RecordSetMDExt.AsInt()
RecordSetMDExt.equals(RecordSetMDEnums.eMode)
RecordSetMDExt.bIn(RecordSetMDEnums.eMode[])
RecordSetMDExt.bIn(String[])

Enum RecordSetMDEnums.eSchemaType

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public enum eSchemaType
```

Fields

NAME	DESCRIPTION
DBB	
HDF5	
JSON	
NamingConvention	
SQL	
Unknown	
XDataMap	
XML	
XSD	

Extension Methods

JavaLikeExtensions.toString()
RecordSetMDExt.AsInt()
RecordSetMDExt.equals(RecordSetMDEnums.eSchemaType)
RecordSetMDExt.toString()
RecordSetMDExt.bIn(RecordSetMDEnums.eSchemaType[])
RecordSetMDExt.bIn(String[])

Enum RecordSetMDEnums.eType

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public enum eType
```

Fields

NAME	DESCRIPTION
CSV	
DBB	
Dlm	
HDF5	
JSON	
SQL	
TXT	
Unknown	
XML	

Extension Methods

JavaLikeExtensions.toString()
RecordSetMDExt.isFlatFile()
RecordSetMDExt.AsInt()
RecordSetMDExt.equals(RecordSetMDEnums.eType)
RecordSetMDExt.bIn(RecordSetMDEnums.eType[])
RecordSetMDExt.bIn(String[])
RecordSetMDExt.toString()

Class RecordSetMDExt

Inheritance

System.Object
RecordSetMDExt

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public static class RecordSetMDExt
```

Methods

AsInt(RecordSetMDEnums.eMode)

Declaration

```
public static int AsInt(this RecordSetMDEnums.eMode self)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eMode	self	

Returns

TYPE	DESCRIPTION
System.Int32	

AsInt(RecordSetMDEnums.eSchemaType)

Declaration

```
public static int AsInt(this RecordSetMDEnums.eSchemaType self)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eSchemaType	self	

Returns

TYPE	DESCRIPTION
System.Int32	

AsInt(RecordSetMDEnums.eType)

Declaration

```
public static int AsInt(this RecordSetMDEnums.eType self)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eType	self	

Returns

TYPE	DESCRIPTION
System.Int32	

bIn(RecordSetMDEnums.eMode, RecordSetMDEnums.eMode[])

Declaration

```
public static bool bIn(this RecordSetMDEnums.eMode self, params RecordSetMDEnums.eMode[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eMode	self	
RecordSetMDEnums.eMode[]	args	

Returns

TYPE	DESCRIPTION
System.Boolean	

bIn(RecordSetMDEnums.eMode, String[])

Declaration

```
public static bool bIn(this RecordSetMDEnums.eMode self, params string[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eMode	self	
System.String[]	args	

Returns

TYPE	DESCRIPTION
System.Boolean	

bIn(RecordSetMDEnums.eSchemaType, RecordSetMDEnums.eSchemaType[])

Declaration

```
public static bool bIn(this RecordSetMDEnums.eSchemaType self, params RecordSetMDEnums.eSchemaType[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eSchemaType	self	
RecordSetMDEnums.eSchemaType[]	args	

Returns

TYPE	DESCRIPTION
System.Boolean	

bIn(RecordSetMDEnums.eSchemaType, String[])

Declaration

```
public static bool bIn(this RecordSetMDEnums.eSchemaType self, params string[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eSchemaType	self	
System.String[]	args	

Returns

TYPE	DESCRIPTION
System.Boolean	

bIn(RecordSetMDEnums.eType, RecordSetMDEnums.eType[])

Declaration

```
public static bool bIn(this RecordSetMDEnums.eType self, params RecordSetMDEnums.eType[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eType	self	
RecordSetMDEnums.eType[]	args	

Returns

TYPE	DESCRIPTION
System.Boolean	

bIn(RecordSetMDEnums.eType, String[])

Declaration

```
public static bool bIn(this RecordSetMDEnums.eType self, params string[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eType	self	
System.String[]	args	

Returns

TYPE	DESCRIPTION
System.Boolean	

eMode_FromAlias(String)

Declaration

```
public static RecordSetMDEnums.eMode eMode_FromAlias(string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Returns

TYPE	DESCRIPTION
RecordSetMDEnums.eMode	

eMode_FromInt(Int32)

Declaration

```
public static RecordSetMDEnums.eMode eMode_FromInt(int arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	arg	

Returns

TYPE	DESCRIPTION
RecordSetMDEnums.eMode	

equals(RecordSetMDEnums.eMode, RecordSetMDEnums.eMode)

Declaration

```
public static bool equals(this RecordSetMDEnums.eMode self, RecordSetMDEnums.eMode arg)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eMode	self	
RecordSetMDEnums.eMode	arg	

Returns

TYPE	DESCRIPTION
System.Boolean	

equals(RecordSetMDEnums.eSchemaType, RecordSetMDEnums.eSchemaType)

Declaration

```
public static bool equals(this RecordSetMDEnums.eSchemaType self, RecordSetMDEnums.eSchemaType arg)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eSchemaType	self	
RecordSetMDEnums.eSchemaType	arg	

Returns

TYPE	DESCRIPTION
System.Boolean	

equals(RecordSetMDEnums.eType, RecordSetMDEnums.eType)

Declaration

```
public static bool equals(this RecordSetMDEnums.eType self, RecordSetMDEnums.eType arg)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eType	self	
RecordSetMDEnums.eType	arg	

Returns

TYPE	DESCRIPTION
System.Boolean	

eSchemaType_FromAlias(String)

Declaration

```
public static RecordSetMDEnums.eSchemaType eSchemaType_FromAlias(string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Returns

TYPE	DESCRIPTION
RecordSetMDEnums.eSchemaType	

eSchemaType_FromInt(Int32)

Declaration

```
public static RecordSetMDEnums.eSchemaType eSchemaType_FromInt(int arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	arg	

Returns

TYPE	DESCRIPTION
RecordSetMDEnums.eSchemaType	

eType_FromAlias(String)

Declaration

```
public static RecordSetMDEnums.eType eType_FromAlias(string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Returns

TYPE	DESCRIPTION
RecordSetMDEnums.eType	

eType_FromInt(Int32)

Declaration

```
public static RecordSetMDEnums.eType eType_FromInt(int arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	arg	

Returns

TYPE	DESCRIPTION
RecordSetMDEnums.eType	

isFlatFile(RecordSetMDEnums.eType)

Declaration

```
public static bool isFlatFile(this RecordSetMDEnums.eType self)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eType	self	

Returns

TYPE	DESCRIPTION
System.Boolean	

toString(RecordSetMDEnums.eSchemaType)

Declaration

```
public static string toString(this RecordSetMDEnums.eSchemaType self)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eSchemaType	self	

Returns

TYPE	DESCRIPTION
System.String	

toString(RecordSetMDEnums.eType)

Declaration

```
public static string toString(this RecordSetMDEnums.eType self)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMDEnums.eType	self	

Returns

TYPE	DESCRIPTION
System.String	

Class Util

Inheritance

System.Object
Util

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class Util
```

Methods

RecordSetElementName(XmlDocument)

Declaration

```
public static string RecordSetElementName(XmlDocument xInputSchema)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Xml.XmlDocument	xInputSchema	

Returns

TYPE	DESCRIPTION
System.String	

RecordSingleName(String)

Declaration

```
public static string RecordSingleName(string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Returns

TYPE	DESCRIPTION
System.String	

Extension Methods

JavaLikeExtensions.toString(Object)

Class WranglerDBB

Inheritance

System.Object
WranglerDBB

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class WranglerDBB
```

Constructors

WranglerDBB()

Declaration

```
public WranglerDBB()
```

Fields

Header

Declaration

```
public WranglerDBB.__DBBMatter Header
```

Field Value

TYPE	DESCRIPTION
WranglerDBB.__DBBMatter	

RecordSet

Declaration

```
public WranglerDBB.__DBBMatter RecordSet
```

Field Value

TYPE	DESCRIPTION
WranglerDBB.__DBBMatter	

Methods

cSetHeaderBufferAs(DBB, Int32, Int32, Int32)

Declaration

```
public WranglerDBB cSetHeaderBufferAs(DBB arg, int nRecords, int nRecordCoreLength, int nRecordVariableLength)
```

Parameters

TYPE	NAME	DESCRIPTION
DBB	arg	
System.Int32	nRecords	
System.Int32	nRecordCoreLength	
System.Int32	nRecordVariableLength	

Returns

TYPE	DESCRIPTION
WranglerDBB	

cSetHeaderBufferFrom(DBB)

Declaration

```
public WranglerDBB cSetHeaderBufferFrom(DBB arg)
```

Parameters

TYPE	NAME	DESCRIPTION
DBB	arg	

Returns

TYPE	DESCRIPTION
WranglerDBB	

cSetRecordSetBufferAs(DBB)

Declaration

```
public WranglerDBB cSetRecordSetBufferAs(DBB arg)
```

Parameters

TYPE	NAME	DESCRIPTION
DBB	arg	

Returns

TYPE	DESCRIPTION
WranglerDBB	

cSetRecordSetBufferAs(DBB, Int64, Int64, Int64, Int64, Int64)

Declaration

```
public WranglerDBB cSetRecordSetBufferAs(DBB arg, long nRecords, long nRecordCoreLength, long
nRecordVariableLength, long nCoreLength, long nTotalLength)
```

Parameters

TYPE	NAME	DESCRIPTION
DBB	arg	
System.Int64	nRecords	
System.Int64	nRecordCoreLength	
System.Int64	nRecordVariableLength	
System.Int64	nCoreLength	
System.Int64	nTotalLength	

Returns

TYPE	DESCRIPTION
WranglerDBB	

cSetRecordSetBufferFrom(DBB)

Declaration

```
public WranglerDBB cSetRecordSetBufferFrom(DBB arg)
```

Parameters

TYPE	NAME	DESCRIPTION
DBB	arg	

Returns

TYPE	DESCRIPTION
WranglerDBB	

Dispose()

Declaration

```
public void Dispose()
```

Finalize()

Declaration

```
protected void Finalize()
```

isValid()

Declaration

```
public bool isValid()
```

Returns

TYPE	DESCRIPTION
System.Boolean	

mBytesRequired(RecordSetMD, Int64, out Int64, out Int64, out Int64)

Declaration

```
public void mBytesRequired(RecordSetMD aRecordSetMD, long nRecords, out long rsize, out long rflensize, out long rvlensize)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aRecordSetMD	
System.Int64	nRecords	
System.Int64	rsize	
System.Int64	rflensize	
System.Int64	rvlensize	

mBytesRequired(RecordSetMD, Int64, out Int64, out Int64, out Int64, out Int64, out Int64, out Int64, out Int64, out Int64, out Int64)

Declaration

```
public void mBytesRequired(RecordSetMD aRecordSetMD, long nRecords, out long csize, out long hsize, out long rsize, out long cleadsize, out long hleadsize, out long hflensize, out long hvlensize, out long rleadsize, out long rflensize, out long rvlensize)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aRecordSetMD	
System.Int64	nRecords	
System.Int64	csize	
System.Int64	hsize	
System.Int64	rsize	
System.Int64	cleadsize	
System.Int64	hleadsize	
System.Int64	hflensize	

TYPE	NAME	DESCRIPTION
System.Int64	hvlensize	
System.Int64	rleadsize	
System.Int64	rflensize	
System.Int64	rvlensize	

mReadMap(RecordSetMD, JniPMMLItem, PrintWriter, Boolean)

Declaration

```
public void mReadMap(RecordSetMD aRecordSetMD, JniPMMLItem aJniPMML, PrintWriter pw, bool
bFillDictionaryNames)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aRecordSetMD	
JniPMMLItem	aJniPMML	
PrintWriter	pw	
System.Boolean	bFillDictionaryNames	

mReadRecordSet(RecordSetMD, RecordSet, PrintWriter)

Declaration

```
public void mReadRecordSet(RecordSetMD aInputRecordSetMD, RecordSet aInputRecordSet, PrintWriter pw)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aInputRecordSetMD	
RecordSet	aInputRecordSet	
PrintWriter	pw	

mWriteMap(RecordSetMD)

Declaration

```
public int mWriteMap(RecordSetMD aRecordSetMD)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aRecordSetMD	

Returns

TYPE	DESCRIPTION
System.Int32	

mWritePrepFor(RecordSetMD, Int64)

Declaration

```
public void mWritePrepFor(RecordSetMD aRecordSetMD, long nRecords)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aRecordSetMD	
System.Int64	nRecords	

mWriteRecordSet(RecordSetMD, RecordSet, RecordSetMD, RecordSet)

Declaration

```
public int mWriteRecordSet(RecordSetMD outRecordSetMD, RecordSet aOutputRecordSet, RecordSetMD inRecordSetMD,
RecordSet aInputRecordSet)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	outRecordSetMD	
RecordSet	aOutputRecordSet	
RecordSetMD	inRecordSetMD	
RecordSet	aInputRecordSet	

Returns

TYPE	DESCRIPTION
System.Int32	

mWriteRecordSet(RecordSetMD, ListObject)

Declaration

```
public int mWriteRecordSet(RecordSetMD inRecordSetMD, ListObject aListObject)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	inRecordSetMD	
Microsoft.Office.Interop.Excel.ListObject	aListObject	

Returns

TYPE	DESCRIPTION
System.Int32	

mWriteRecordSet(RecordSetMD, Object[,], Boolean)

Declaration

```
public int mWriteRecordSet(RecordSetMD inRecordSetMD, object[, ] r, bool bIncludesHeaderRow)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	inRecordSetMD	
System.Object[,]	r	
System.Boolean	bIncludesHeaderRow	

Returns

TYPE	DESCRIPTION
System.Int32	

Extension Methods

JavaLikeExtensions.toString(Object)

Class WranglerDBB.__DBBMatter

Inheritance

System.Object
WranglerDBB.__DBBMatter

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class __DBBMatter
```

Fields

bIsManagedInJava

Declaration

```
public bool bIsManagedInJava
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

Buffer

Declaration

```
public DBB Buffer
```

Field Value

TYPE	DESCRIPTION
DBB	

MaxStringByteLength

Declaration

```
public long MaxStringByteLength
```

Field Value

TYPE	DESCRIPTION
System.Int64	

MaxStringLength

Declaration

```
public long MaxStringLength
```

Field Value

TYPE	DESCRIPTION
System.Int64	

Methods

Dispose()

Declaration

```
public void Dispose()
```

Finalize()

Declaration

```
protected void Finalize()
```

Extension Methods

JavaLikeExtensions.toString(Object)

Class WranglerFlatFile

Inheritance

System.Object
WranglerFlatFile

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class WranglerFlatFile
```

Fields

__CSV

Declaration

```
public CsvParser __CSV
```

Field Value

TYPE	DESCRIPTION
CsvHelper.CsvParser	

__StreamReader

Declaration

```
public StreamReader __StreamReader
```

Field Value

TYPE	DESCRIPTION
System.IO.StreamReader	

Dlm

Declaration

```
public string Dlm
```

Field Value

TYPE	DESCRIPTION
System.String	

FileName

Declaration

```
public string FileName
```

Field Value

TYPE	DESCRIPTION
System.String	

hasHeaderRow

Declaration

```
public bool hasHeaderRow
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

Path

Declaration

```
public string Path
```

Field Value

TYPE	DESCRIPTION
System.String	

Methods

cPointToFile(RecordSetMD, String, Boolean, String)

Declaration

```
public void cPointToFile(RecordSetMD aInputRecordSetMD, string aFileName, bool hasHeaderRow, string dlm)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aInputRecordSetMD	
System.String	aFileName	
System.Boolean	hasHeaderRow	
System.String	dlm	

cPointToFile(RecordSetMD, String, String, Boolean, String)

Declaration

```
public void cPointToFile(RecordSetMD aInputRecordSetMD, string aPath, string aFileName, bool hasHeaderRow,
string dlm)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aInputRecordSetMD	
System.String	aPath	
System.String	aFileName	
System.Boolean	hasHeaderRow	
System.String	dlm	

Dispose()

Declaration

```
public void Dispose()
```

Equals(WranglerFlatFile)

Declaration

```
public bool Equals(WranglerFlatFile arg)
```

Parameters

TYPE	NAME	DESCRIPTION
WranglerFlatFile	arg	

Returns

TYPE	DESCRIPTION
System.Boolean	

Finalize()

Declaration

```
protected void Finalize()
```

mReadMapFor(RecordSetMD, JniPMMLItem, PrintWriter, Boolean)

Declaration

```
public void mReadMapFor(RecordSetMD aRecordSetMD, JniPMMLItem aJniPMML, PrintWriter pw, bool
bFillDictionaryNames)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aRecordSetMD	
JniPMMLItem	aJniPMML	
PrintWriter	pw	

TYPE	NAME	DESCRIPTION
System.Boolean	bFillDictionaryNames	

mReadRecordSet(RecordSetMD, RecordSet, PrintWriter)

Declaration

```
public void mReadRecordSet(RecordSetMD aInputRecordSetMD, RecordSet aInputRecordSet, PrintWriter pw)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aInputRecordSetMD	
RecordSet	aInputRecordSet	
PrintWriter	pw	

mReadRecordSet(RecordSetMD, RecordSet, String, String, Boolean, String, PrintWriter)

Declaration

```
public void mReadRecordSet(RecordSetMD aInputRecordSetMD, RecordSet aInputRecordSet, string aPath, string
aFileName, bool hasHeaderRow, string dlm, PrintWriter pw)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aInputRecordSetMD	
RecordSet	aInputRecordSet	
System.String	aPath	
System.String	aFileName	
System.Boolean	hasHeaderRow	
System.String	dlm	
PrintWriter	pw	

mWriteRecordSet(RecordSetMD, RecordSet)

Declaration

```
public void mWriteRecordSet(RecordSetMD aOutputRecordSetMD, RecordSet aOutputRecordSet)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aOutputRecordSetMD	

TYPE	NAME	DESCRIPTION
RecordSet	aOutputRecordSet	

mWriteRecordSet(RecordSetMD, RecordSet, RecordSetMD, RecordSet)

Declaration

```
public void mWriteRecordSet(RecordSetMD aOutputRecordSetMD, RecordSet aOutputRecordSet, RecordSetMD
aInputRecordSetMD, RecordSet aInputRecordSet)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aOutputRecordSetMD	
RecordSet	aOutputRecordSet	
RecordSetMD	aInputRecordSetMD	
RecordSet	aInputRecordSet	

mWriteRecordSet(RecordSetMD, ListObject)

Declaration

```
public void mWriteRecordSet(RecordSetMD aOutputRecordSetMD, ListObject aListObject)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aOutputRecordSetMD	
Microsoft.Office.Interop.Excel.ListObject	aListObject	

Extension Methods

JavaLikeExtensions.toString(Object)

Class WranglerHDF5

Inheritance

System.Object
WranglerHDF5

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class WranglerHDF5
```

Constructors

WranglerHDF5()

Declaration

```
public WranglerHDF5()
```

Fields

bIsInMemory

Declaration

```
public bool bIsInMemory
```

Field Value

TYPE	DESCRIPTION
System.Boolean	

CompoundDS

Declaration

```
public long CompoundDS
```

Field Value

TYPE	DESCRIPTION
System.Int64	

DSName

Declaration

```
public string DSName
```

Field Value

TYPE	DESCRIPTION
System.String	

File

Declaration

```
public long File
```

Field Value

TYPE	DESCRIPTION
System.Int64	

Methods

Dispose()

Declaration

```
public void Dispose()
```

Finalize()

Declaration

```
protected void Finalize()
```

mReadMapFor(RecordSetMD, JniPMMLItem, PrintWriter, Boolean)

Declaration

```
public void mReadMapFor(RecordSetMD aRecordSetMD, JniPMMLItem aJniPMML, PrintWriter pw, bool
bFillDictionaryNames)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aRecordSetMD	
JniPMMLItem	aJniPMML	
PrintWriter	pw	
System.Boolean	bFillDictionaryNames	

mReadPrepFor(RecordSetMD, PrintWriter)

Declaration

```
public long mReadPrepFor(RecordSetMD aRecordSetMD, PrintWriter pw)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aRecordSetMD	
PrintWriter	pw	

Returns

TYPE	DESCRIPTION
System.Int64	

mReadRecordSet(RecordSetMD, RecordSet, PrintWriter)

Declaration

```
public void mReadRecordSet(RecordSetMD aInputRecordSetMD, RecordSet aInputRecordSet, PrintWriter pw)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aInputRecordSetMD	
RecordSet	aInputRecordSet	
PrintWriter	pw	

mWriteMap(RecordSetMD)

Declaration

```
public int mWriteMap(RecordSetMD aRecordSetMD)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aRecordSetMD	

Returns

TYPE	DESCRIPTION
System.Int32	

mWritePrepFor(RecordSetMD, PrintWriter)

Declaration

```
public long mWritePrepFor(RecordSetMD aRecordSetMD, PrintWriter pw)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aRecordSetMD	

TYPE	NAME	DESCRIPTION
PrintWriter	pw	

Returns

TYPE	DESCRIPTION
System.Int64	

mWriteRecordSet(RecordSetMD, RecordSet, RecordSetMD, RecordSet)

Declaration

```
public void mWriteRecordSet(RecordSetMD aOutputRecordSetMD, RecordSet aOutputRecordSet, RecordSetMD
aInputRecordSetMD, RecordSet aInputRecordSet)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aOutputRecordSetMD	
RecordSet	aOutputRecordSet	
RecordSetMD	aInputRecordSetMD	
RecordSet	aInputRecordSet	

mWriteRecordSet(RecordSetMD, ListObject)

Declaration

```
public int mWriteRecordSet(RecordSetMD aRecordSetMD, ListObject aListObject)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aRecordSetMD	
Microsoft.Office.Interop.Excel.ListObject	aListObject	

Returns

TYPE	DESCRIPTION
System.Int32	

new_HDF5DataType(FieldMDEnums.eDTyp)

Declaration

```
public WranglerHDF5.HDF5DataType new_HDF5DataType(FieldMDEnums.eDTyp DTyp)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eDTyp	DTyp	

Returns

TYPE	DESCRIPTION
WranglerHDF5.HDF5DataType	

new_HDF5DataType(FieldMDEnums.eDTyp, Int64, Boolean)

Declaration

```
public WranglerHDF5.HDF5DataType new_HDF5DataType(FieldMDEnums.eDTyp DTyp, long nStringMaxLength, bool
anyVLen)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eDTyp	DTyp	
System.Int64	nStringMaxLength	
System.Boolean	anyVLen	

Returns

TYPE	DESCRIPTION
WranglerHDF5.HDF5DataType	

new_HDF5DataType(WranglerHDF5.HDF5DataType)

Declaration

```
public WranglerHDF5.HDF5DataType new_HDF5DataType(WranglerHDF5.HDF5DataType arg)
```

Parameters

TYPE	NAME	DESCRIPTION
WranglerHDF5.HDF5DataType	arg	

Returns

TYPE	DESCRIPTION
WranglerHDF5.HDF5DataType	

new_HDF5DataType(Int32, Int32, Int32, Int32)

Declaration

```
public WranglerHDF5.HDF5DataType new_HDF5DataType(int hclass, int hlength, int horder, int hsign)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	hclass	
System.Int32	hlength	
System.Int32	horder	
System.Int32	hsign	

Returns

TYPE	DESCRIPTION
WranglerHDF5.HDF5DataType	

new_HDF5DataType(Int64)

Declaration

```
public WranglerHDF5.HDF5DataType new_HDF5DataType(long arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int64	arg	

Returns

TYPE	DESCRIPTION
WranglerHDF5.HDF5DataType	

UpdateOutputMapForHDF5(RecordSetMD, PrintWriter)

Declaration

```
public void UpdateOutputMapForHDF5(RecordSetMD aRecordSetMD, PrintWriter pw)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aRecordSetMD	
PrintWriter	pw	

Extension Methods

JavaLikeExtensions.toString(Object)

Class WranglerHDF5.HDF5DataType

Inheritance

System.Object
WranglerHDF5.HDF5DataType

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class HDF5DataType
```

Constructors

HDF5DataType(FieldMDEnums.eDTyp, Int64, Boolean)

Declaration

```
public HDF5DataType(FieldMDEnums.eDTyp DTyp, long nStringMaxLength, bool anyVLen)
```

Parameters

TYPE	NAME	DESCRIPTION
FieldMDEnums.eDTyp	DTyp	
System.Int64	nStringMaxLength	
System.Boolean	anyVLen	

HDF5DataType(Int64)

Declaration

```
public HDF5DataType(long arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int64	arg	

HDF5DataType(Int64, Int32, Int32, Int32)

Declaration

```
public HDF5DataType(long hclass, int hlength, int horder, int hsign)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int64	hclass	
System.Int32	hlength	
System.Int32	horder	
System.Int32	hsign	

Fields

data

Declaration

```
public long data
```

Field Value

TYPE	DESCRIPTION
System.Int64	

Methods

eDTyp()

Declaration

```
public FieldMDEnums.eDTyp eDTyp()
```

Returns

TYPE	DESCRIPTION
FieldMDEnums.eDTyp	

Equals(WranglerHDF5.HDF5DataType)

Declaration

```
public bool Equals(WranglerHDF5.HDF5DataType arg)
```

Parameters

TYPE	NAME	DESCRIPTION
WranglerHDF5.HDF5DataType	arg	

Returns

TYPE	DESCRIPTION
System.Boolean	

Extension Methods

JavaLikeExtensions.toString(Object)

Class WranglerXSD

Inheritance

System.Object

WranglerXSD

RecordSetMD.__SchemaMatter

Inherited Members

System.Object.ToString()

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

Namespace: com.WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class WranglerXSD
```

Methods

mReadMapFor(RecordSetMD, JniPMMLItem, PrintWriter, Boolean)

Declaration

```
public void mReadMapFor(RecordSetMD aRecordSetMD, JniPMMLItem aJniPMMLItem, PrintWriter pw, bool
bFillDictionaryNames)
```

Parameters

TYPE	NAME	DESCRIPTION
RecordSetMD	aRecordSetMD	
JniPMMLItem	aJniPMMLItem	
PrintWriter	pw	
System.Boolean	bFillDictionaryNames	

mReadMapFor(XmlDocument, RecordSetMD, JniPMMLItem, PrintWriter, Boolean)

Declaration

```
public void mReadMapFor(XmlDocument aDoc, RecordSetMD aRecordSetMD, JniPMMLItem aJniPMMLItem, PrintWriter pw,
bool bFillDictionaryNames)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Xml.XmlDocument	aDoc	
RecordSetMD	aRecordSetMD	

TYPE	NAME	DESCRIPTION
JniPMMLItem	aJniPMMLItem	
PrintWriter	pw	
System.Boolean	bFillDictionaryNames	

XSDColumn(String, String)

Declaration

```
public static string XSDColumn(string name, string dtyp)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	name	
System.String	dtyp	

Returns

TYPE	DESCRIPTION
System.String	

XSDFooter()

Declaration

```
public static string XSDFooter()
```

Returns

TYPE	DESCRIPTION
System.String	

XSDHeader()

Declaration

```
public static string XSDHeader()
```

Returns

TYPE	DESCRIPTION
System.String	

XSDRecordSet_Close()

Declaration

```
public static string XSDRecordSet_Close()
```

Returns

TYPE	DESCRIPTION
System.String	

XSDRecordSet_Open(String, String)

Declaration

```
public static string XSDRecordSet_Open(string rns, string rn)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	rns	
System.String	rn	

Returns

TYPE	DESCRIPTION
System.String	

XSDTypes()

Declaration

```
public static string XSDTypes()
```

Returns

TYPE	DESCRIPTION
System.String	

Extension Methods

JavaLikeExtensions.toString(Object)

Namespace com.WDataSci.WDS

The com.WDataSci.WDS namespace mirrors the Java com.WDataSci.WDS package, but also contains C# extensions and classes to minimize the code C#/Java code differences.

See Java x Cs Notes for more cross language details.

Classes

ArrayList<A>

JavaLikeExtensions

Map<A, B>

PrintWriter

Util

WDSCommon

WDSException

Class ArrayList<A>

Inheritance

System.Object
System.Collections.Generic.List<A>
ArrayList<A>

Implements

System.Collections.Generic.IList<A>
System.Collections.Generic.ICollection<A>
System.Collections.IList
System.Collections.ICollection
System.Collections.Generic.IReadOnlyList<A>
System.Collections.Generic.IReadOnlyCollection<A>
System.Collections.Generic.IEnumerable<A>
System.Collections.IEnumerable

Inherited Members

System.Collections.Generic.List<A>.System.Collections.IList.get_Item(System.Int32)
System.Collections.Generic.List<A>.System.Collections.IList.set_Item(System.Int32, System.Object)
System.Collections.Generic.List<A>.Add(A)
System.Collections.Generic.List<A>.System.Collections.IList.Add(System.Object)
System.Collections.Generic.List<A>.AddRange(System.Collections.Generic.IEnumerable<A>)
System.Collections.Generic.List<A>.AsReadOnly()
System.Collections.Generic.List<A>.BinarySearch(System.Int32, System.Int32, A, System.Collections.Generic.IComparer<A>)
System.Collections.Generic.List<A>.BinarySearch(A)
System.Collections.Generic.List<A>.BinarySearch(A, System.Collections.Generic.IComparer<A>)
System.Collections.Generic.List<A>.Clear()
System.Collections.Generic.List<A>.Contains(A)
System.Collections.Generic.List<A>.System.Collections.IList.Contains(System.Object)
System.Collections.Generic.List<A>.ConvertAll<TOutput>(System.Converter<A, TOutput>)
System.Collections.Generic.List<A>.CopyTo(A[])
System.Collections.Generic.List<A>.System.Collections.ICollection.CopyTo(System.Array, System.Int32)
System.Collections.Generic.List<A>.CopyTo(System.Int32, A[], System.Int32, System.Int32)
System.Collections.Generic.List<A>.CopyTo(A[], System.Int32)
System.Collections.Generic.List<A>.Exists(System.Predicate<A>)
System.Collections.Generic.List<A>.Find(System.Predicate<A>)
System.Collections.Generic.List<A>.FindAll(System.Predicate<A>)
System.Collections.Generic.List<A>.FindIndex(System.Predicate<A>)
System.Collections.Generic.List<A>.FindIndex(System.Int32, System.Predicate<A>)
System.Collections.Generic.List<A>.FindIndex(System.Int32, System.Int32, System.Predicate<A>)
System.Collections.Generic.List<A>.FindLast(System.Predicate<A>)
System.Collections.Generic.List<A>.FindLastIndex(System.Predicate<A>)
System.Collections.Generic.List<A>.FindLastIndex(System.Int32, System.Predicate<A>)
System.Collections.Generic.List<A>.FindLastIndex(System.Int32, System.Int32, System.Predicate<A>)
System.Collections.Generic.List<A>.ForEach(System.Action<A>)
System.Collections.Generic.List<A>.GetEnumerator()
System.Collections.Generic.List<A>.System.Collections.Generic.IEnumerable<A>.GetEnumerator()
System.Collections.Generic.List<A>.System.Collections.IEnumerable.GetEnumerator()
System.Collections.Generic.List<A>.GetRange(System.Int32, System.Int32)
System.Collections.Generic.List<A>.IndexOf(A)
System.Collections.Generic.List<A>.System.Collections.IList.IndexOf(System.Object)
System.Collections.Generic.List<A>.IndexOf(A, System.Int32)

System.Collections.Generic.List<A>.IndexOf(A, System.Int32, System.Int32)
System.Collections.Generic.List<A>.Insert(System.Int32, A)
System.Collections.Generic.List<A>.System.Collections.IList.Insert(System.Int32, System.Object)
System.Collections.Generic.List<A>.InsertRange(System.Int32, System.Collections.Generic.IEnumerable<A>)
System.Collections.Generic.List<A>.LastIndexOf(A)
System.Collections.Generic.List<A>.LastIndexOf(A, System.Int32)
System.Collections.Generic.List<A>.LastIndexOf(A, System.Int32, System.Int32)
System.Collections.Generic.List<A>.Remove(A)
System.Collections.Generic.List<A>.System.Collections.IList.Remove(System.Object)
System.Collections.Generic.List<A>.RemoveAll(System.Predicate<A>)
System.Collections.Generic.List<A>.RemoveAt(System.Int32)
System.Collections.Generic.List<A>.RemoveRange(System.Int32, System.Int32)
System.Collections.Generic.List<A>.Reverse()
System.Collections.Generic.List<A>.Reverse(System.Int32, System.Int32)
System.Collections.Generic.List<A>.Sort()
System.Collections.Generic.List<A>.Sort(System.Collections.Generic.IComparer<A>)
System.Collections.Generic.List<A>.Sort(System.Int32, System.Int32, System.Collections.Generic.IComparer<A>)
System.Collections.Generic.List<A>.Sort(System.Comparison<A>)
System.Collections.Generic.List<A>.ToArray()
System.Collections.Generic.List<A>.TrimExcess()
System.Collections.Generic.List<A>.TrueForAll(System.Predicate<A>)
System.Collections.Generic.List<A>.Capacity
System.Collections.Generic.List<A>.Count
System.Collections.Generic.List<A>.System.Collections.IList.IsFixedSize
System.Collections.Generic.List<A>.System.Collections.Generic.ICollection<A>.IsReadOnly
System.Collections.Generic.List<A>.System.Collections.IList.IsReadOnly
System.Collections.Generic.List<A>.System.Collections.ICollection.IsSynchronized
System.Collections.Generic.List<A>.System.Collections.ICollection.SyncRoot
System.Collections.Generic.List<A>.Item[System.Int32]
System.Collections.Generic.List<A>.System.Collections.IList.Item[System.Int32]
System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.WDS

Assembly: WDS-Cs.dll

Syntax

```
public class ArrayList<A> : List<A>, IList<A>, ICollection<A>, IList, ICollection, IReadOnlyList<A>,
IReadOnlyCollection<A>, IEnumerable<A>, IEnumerable
```

Type Parameters

NAME	DESCRIPTION
A	

Implements

System.Collections.Generic.IList<T>
System.Collections.Generic.ICollection<T>

System.Collections.IList
System.Collections.ICollection
System.Collections.Generic.IReadOnlyList<T>
System.Collections.Generic.IReadOnlyCollection<T>
System.Collections.Generic.IEnumerable<T>
System.Collections.IEnumerable

Extension Methods

JavaLikeExtensions.toString(Object)
JavaLikeExtensions.add<T>(List<T>, T)
JavaLikeExtensions.clear<T>(List<T>)
JavaLikeExtensions.get<T>(List<T>, Int32)
JavaLikeExtensions.size<T>(List<T>)

Class JavaLikeExtensions

Inheritance

System.Object
JavaLikeExtensions

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.WDS

Assembly: WDS-Cs.dll

Syntax

```
public static class JavaLikeExtensions
```

Methods

add<T>(List<T>, T)

Declaration

```
public static void add<T>(this List<T> self, T v)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.List<T>	self	
T	v	

Type Parameters

NAME	DESCRIPTION
T	

add<A, B>(List<Map<A, B>>, Map<A, B>)

Declaration

```
public static void add<A, B>(this List<Map<A, B>> self, Map<A, B> arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.List<Map<A, B>>	self	
Map<A, B>	arg	

Type Parameters

NAME	DESCRIPTION
A	
B	

add<A, B>(List<Dictionary<A, B>>, Dictionary<A, B>)

Declaration

```
public static void add<A, B>(this List<Dictionary<A, B>> self, Dictionary<A, B> arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.List<System.Collections.Generic.Dictionary<A, B>>	self	
System.Collections.Generic.Dictionary<A, B>	arg	

Type Parameters

NAME	DESCRIPTION
A	
B	

bIn(Int64, Int64[])

Declaration

```
public static bool bIn(long arg, params long[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int64	arg	
System.Int64[]	args	

Returns

TYPE	DESCRIPTION
System.Boolean	

clear<T>(List<T>)

Declaration

```
public static void clear<T>(this List<T> self)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.List<T>	self	

Type Parameters

NAME	DESCRIPTION
T	

close(CsvWriter)

Declaration

```
public static void close(this CsvWriter self)
```

Parameters

TYPE	NAME	DESCRIPTION
CsvHelper.CsvWriter	self	

close(StreamWriter)

Declaration

```
public static void close(this StreamWriter self)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IO.StreamWriter	self	

endsWith(String, String)

Declaration

```
public static bool endsWith(this string self, string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	self	
System.String	arg	

Returns

TYPE	DESCRIPTION
System.Boolean	

equals(String, String)

Declaration

```
public static bool equals(this string self, string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	self	
System.String	arg	

Returns

TYPE	DESCRIPTION
System.Boolean	

flush(CsvWriter)

Declaration

```
public static void flush(this CsvWriter self)
```

Parameters

TYPE	NAME	DESCRIPTION
CsvHelper.CsvWriter	self	

get(String[], Int32)

Declaration

```
public static string get(this string[] self, int i)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String[]	self	
System.Int32	i	

Returns

TYPE	DESCRIPTION
System.String	

get<T>(List<T>, Int32)

Declaration

```
public static T get<T>(this List<T> self, int i)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.List<T>	self	
System.Int32	i	

Returns

TYPE	DESCRIPTION
T	

Type Parameters

NAME	DESCRIPTION
T	

get<A, B>(Dictionary<A, B>, A)

Declaration

```
public static B get<A, B>(this Dictionary<A, B> self, A arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.Dictionary<A, B>	self	
A	arg	

Returns

TYPE	DESCRIPTION
B	

Type Parameters

NAME	DESCRIPTION
A	
B	

get<A, B>(List<Dictionary<A, B>>, Int32)

Declaration

```
public static Dictionary<A, B> get<A, B>(this List<Dictionary<A, B>> self, int i)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.List<System.Collections.Generic.Dictionary<A, B>>	self	
System.Int32	i	

Returns

TYPE	DESCRIPTION
System.Collections.Generic.Dictionary<A, B>	

Type Parameters

NAME	DESCRIPTION
A	
B	

getBytes(String)

Declaration

```
public static byte[] getBytes(this string self)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	self	

Returns

TYPE	DESCRIPTION
System.Byte[]	

getLength(XmlNodeList)

Declaration

```
public static int getLength(this XmlNodeList self)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Xml.XmlNodeList	self	

Returns

TYPE	DESCRIPTION
System.Int32	

indexOf(String, String)

Declaration

```
public static int indexOf(this string self, string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	self	
System.String	arg	

Returns

TYPE	DESCRIPTION
System.Int32	

isEmpty(String)

Declaration

```
public static bool isEmpty(this string self)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	self	

Returns

TYPE	DESCRIPTION
System.Boolean	

lastIndexOf(String, String)

Declaration

```
public static int lastIndexOf(this string self, string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	self	
System.String	arg	

Returns

TYPE	DESCRIPTION
System.Int32	

length(String)

Declaration

```
public static int length(this string self)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	self	

Returns

TYPE	DESCRIPTION
System.Int32	

new_String(String)

Declaration

```
public static string new_String(string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Returns

TYPE	DESCRIPTION
System.String	

printRecord(CsvWriter, List<String>)

Declaration

```
public static void printRecord(this CsvWriter self, List<string> arg)
```

Parameters

TYPE	NAME	DESCRIPTION
CsvHelper.CsvWriter	self	
System.Collections.Generic.List<System.String>	arg	

printRecord(CsvWriter, String[])

Declaration

```
public static void printRecord(this CsvWriter self, string[] arg)
```

Parameters

TYPE	NAME	DESCRIPTION
CsvHelper.CsvWriter	self	
System.String[]	arg	

replaceAll(String, String, String)

Declaration

```
public static string replaceAll(this string self, string RegexToFind, string RegexToReplaceWith)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	self	
System.String	RegexToFind	

TYPE	NAME	DESCRIPTION
System.String	RegexToReplaceWith	

Returns

TYPE	DESCRIPTION
System.String	

ReplaceAll(String, String, String)

Declaration

```
public static string ReplaceAll(this string self, string RegexToFind, string RegexToReplaceWith)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	self	
System.String	RegexToFind	
System.String	RegexToReplaceWith	

Returns

TYPE	DESCRIPTION
System.String	

size(String[])

Declaration

```
public static int size(this string[] self)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String[]	self	

Returns

TYPE	DESCRIPTION
System.Int32	

size<T>(List<T>)

Declaration

```
public static int size<T>(this List<T> self)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.List<T>	self	

Returns

TYPE	DESCRIPTION
System.Int32	

Type Parameters

NAME	DESCRIPTION
T	

startsWith(String, String)

Declaration

```
public static bool startsWith(this string self, string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	self	
System.String	arg	

Returns

TYPE	DESCRIPTION
System.Boolean	

substring(String, Int32)

Declaration

```
public static string substring(this string self, int start)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	self	
System.Int32	start	

Returns

TYPE	DESCRIPTION
System.String	

substring(String, Int32, Int32)

Declaration

```
public static string substring(this string self, int start, int len)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	self	
System.Int32	start	
System.Int32	len	

Returns

TYPE	DESCRIPTION
System.String	

toLowerCase(String)

Declaration

```
public static string toLowerCase(this string self)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	self	

Returns

TYPE	DESCRIPTION
System.String	

toString(Object)

Declaration

```
public static string toString(this object self)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	self	

Returns

TYPE	DESCRIPTION
System.String	

toString(String)

Declaration

```
public static string toString(this string self)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	self	

Returns

TYPE	DESCRIPTION
System.String	

Class Map<A, B>

Inheritance

System.Object
System.Collections.Generic.Dictionary<A, B>
Map<A, B>

Implements

System.Collections.Generic.IDictionary<A, B>
System.Collections.Generic.ICollection<System.Collections.Generic.KeyValuePair<A, B>>
System.Collections.IDictionary
System.Collections.ICollection
System.Collections.Generic.IReadOnlyDictionary<A, B>
System.Collections.Generic.IReadOnlyCollection<System.Collections.Generic.KeyValuePair<A, B>>
System.Collections.Generic.IEnumerable<System.Collections.Generic.KeyValuePair<A, B>>
System.Collections.IEnumerable
System.Runtime.Serialization.ISerializable
System.Runtime.Serialization.IDeserializationCallback

Inherited Members

System.Collections.Generic.Dictionary<A, B>.Add(A, B)
System.Collections.Generic.Dictionary<A, B>.System.Collections.Generic.ICollection<System.Collections.Generic.KeyValuePair<A, B>>.Add(System.Collections.Generic.KeyValuePair<A, B>)
System.Collections.Generic.Dictionary<A, B>.System.Collections.Generic.ICollection<System.Collections.Generic.KeyValuePair<A, B>>.Contains(System.Collections.Generic.KeyValuePair<A, B>)
System.Collections.Generic.Dictionary<A, B>.System.Collections.Generic.ICollection<System.Collections.Generic.KeyValuePair<A, B>>.Remove(System.Collections.Generic.KeyValuePair<A, B>)
System.Collections.Generic.Dictionary<A, B>.Clear()
System.Collections.Generic.Dictionary<A, B>.ContainsKey(A)
System.Collections.Generic.Dictionary<A, B>.ContainsValue(B)
System.Collections.Generic.Dictionary<A, B>.GetEnumerator()
System.Collections.Generic.Dictionary<A, B>.System.Collections.Generic.IEnumerable<System.Collections.Generic.KeyValuePair<A, B>>.GetEnumerator()
System.Collections.Generic.Dictionary<A, B>.GetObjectData(System.Runtime.Serialization.SerializationInfo, System.Runtime.Serialization.StreamingContext)
System.Collections.Generic.Dictionary<A, B>.OnDeserialization(System.Object)
System.Collections.Generic.Dictionary<A, B>.Remove(A)
System.Collections.Generic.Dictionary<A, B>.TryGetValue(A, B)
System.Collections.Generic.Dictionary<A, B>.System.Collections.Generic.ICollection<System.Collections.Generic.KeyValuePair<A, B>>.CopyTo(System.Collections.Generic.KeyValuePair<A, B>[], System.Int32)
System.Collections.Generic.Dictionary<A, B>.System.Collections.ICollection.CopyTo(System.Array, System.Int32)
System.Collections.Generic.Dictionary<A, B>.System.Collections.IEnumerable.GetEnumerator()
System.Collections.Generic.Dictionary<A, B>.System.Collections.IDictionary.get_Item(System.Object)
System.Collections.Generic.Dictionary<A, B>.System.Collections.IDictionary.set_Item(System.Object, System.Object)
System.Collections.Generic.Dictionary<A, B>.System.Collections.IDictionary.Add(System.Object, System.Object)
System.Collections.Generic.Dictionary<A, B>.System.Collections.IDictionary.Contains(System.Object)
System.Collections.Generic.Dictionary<A, B>.System.Collections.IDictionary.GetEnumerator()
System.Collections.Generic.Dictionary<A, B>.System.Collections.IDictionary.Remove(System.Object)
System.Collections.Generic.Dictionary<A, B>.Comparer
System.Collections.Generic.Dictionary<A, B>.Count
System.Collections.Generic.Dictionary<A, B>.Keys
System.Collections.Generic.Dictionary<A, B>.System.Collections.Generic.IDictionary<A, B>.Keys
System.Collections.Generic.Dictionary<A, B>.System.Collections.Generic.IReadOnlyDictionary<A, B>.Keys

System.Collections.Generic.Dictionary<A, B>.Values
System.Collections.Generic.Dictionary<A, B>.System.Collections.Generic.IDictionary<A, B>.Values
System.Collections.Generic.Dictionary<A, B>.System.Collections.Generic.IReadOnlyDictionary<A, B>.Values
System.Collections.Generic.Dictionary<A, B>.Item[A]
System.Collections.Generic.Dictionary<A, B>.System.Collections.Generic.ICollection<System.Collections.Generic.KeyValuePair<A, B>>.IsReadOnly
System.Collections.Generic.Dictionary<A, B>.System.Collections.ICollection.IsSynchronized
System.Collections.Generic.Dictionary<A, B>.System.Collections.ICollection.SyncRoot
System.Collections.Generic.Dictionary<A, B>.System.Collections.IDictionary.IsFixedSize
System.Collections.Generic.Dictionary<A, B>.System.Collections.IDictionary.IsReadOnly
System.Collections.Generic.Dictionary<A, B>.System.Collections.IDictionary.Keys
System.Collections.Generic.Dictionary<A, B>.System.Collections.IDictionary.Values
System.Collections.Generic.Dictionary<A, B>.System.Collections.IDictionary.Item[System.Object]
System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.WDS

Assembly: WDS-Cs.dll

Syntax

```
public class Map<A, B> : Dictionary<A, B>, IDictionary<A, B>, ICollection<KeyValuePair<A, B>>, IDictionary, ICollection, IReadOnlyDictionary<A, B>, IReadOnlyCollection<KeyValuePair<A, B>>, IEnumerable<KeyValuePair<A, B>>, IEnumerable, ISerializable, IDeserializationCallback
```

Type Parameters

NAME	DESCRIPTION
A	
B	

Methods

get(A)

Declaration

```
public B get(A arg)
```

Parameters

TYPE	NAME	DESCRIPTION
A	arg	

Returns

TYPE	DESCRIPTION
B	

keyArray()

Declaration

```
public A[] keyArray()
```

Returns

TYPE	DESCRIPTION
A[]	

keySet()

Declaration

```
public HashSet<A> keySet()
```

Returns

TYPE	DESCRIPTION
System.Collections.Generic.HashSet<A>	

put(A, B)

Declaration

```
public void put(A k, B v)
```

Parameters

TYPE	NAME	DESCRIPTION
A	k	
B	v	

Implements

System.Collections.Generic.IDictionary<TKey, TValue>
System.Collections.Generic.ICollection<T>
System.Collections.IDictionary
System.Collections.ICollection
System.Collections.Generic.IReadOnlyDictionary<TKey, TValue>
System.Collections.Generic.IReadOnlyCollection<T>
System.Collections.Generic.IEnumerable<T>
System.Collections.IEnumerable
System.Runtime.Serialization.ISerializable
System.Runtime.Serialization.IDeserializationCallback

Extension Methods

JavaLikeExtensions.toString(Object)
JavaLikeExtensions.get<A, B>(Dictionary<A, B>, A)

Class PrintWriter

Inheritance

System.Object
System.MarshalByRefObject
System.IO.TextWriter
System.IO.StringWriter
PrintWriter

Implements

System.IDisposable

Inherited Members

System.IO.StringWriter.Close()
System.IO.StringWriter.Dispose(System.Boolean)
System.IO.StringWriter.GetStringBuilder()
System.IO.StringWriter.Write(System.Char)
System.IO.StringWriter.Write(System.Char[], System.Int32, System.Int32)
System.IO.StringWriter.Write(System.String)
System.IO.StringWriter.WriteAsync(System.Char)
System.IO.StringWriter.WriteAsync(System.String)
System.IO.StringWriter.WriteAsync(System.Char[], System.Int32, System.Int32)
System.IO.StringWriter.WriteLineAsync(System.Char)
System.IO.StringWriter.WriteLineAsync(System.String)
System.IO.StringWriter.WriteLineAsync(System.Char[], System.Int32, System.Int32)
System.IO.StringWriter.FlushAsync()
System.IO.StringWriter.ToString()
System.IO.StringWriter.Encoding
System.IO.TextWriter.Null
System.IO.TextWriter.CoreNewLine
System.IO.TextWriter.Dispose()
System.IO.TextWriter.Flush()
System.IO.TextWriter.Synchronized(System.IO.TextWriter)
System.IO.TextWriter.Write(System.Char[])
System.IO.TextWriter.Write(System.Boolean)
System.IO.TextWriter.Write(System.Int32)
System.IO.TextWriter.Write(System.UInt32)
System.IO.TextWriter.Write(System.Int64)
System.IO.TextWriter.Write(System.UInt64)
System.IO.TextWriter.Write(System.Single)
System.IO.TextWriter.Write(System.Double)
System.IO.TextWriter.Write(System.Decimal)
System.IO.TextWriter.Write(System.Object)
System.IO.TextWriter.Write(System.String, System.Object)
System.IO.TextWriter.Write(System.String, System.Object, System.Object)
System.IO.TextWriter.Write(System.String, System.Object, System.Object, System.Object)
System.IO.TextWriter.Write(System.String, System.Object[])
System.IO.TextWriter.WriteLine()
System.IO.TextWriter.WriteLine(System.Char)
System.IO.TextWriter.WriteLine(System.Char[])
System.IO.TextWriter.WriteLine(System.Char[], System.Int32, System.Int32)
System.IO.TextWriter.WriteLine(System.Boolean)
System.IO.TextWriter.WriteLine(System.Int32)

System.IO.TextWriter.WriteLine(System.UInt32)
System.IO.TextWriter.WriteLine(System.Int64)
System.IO.TextWriter.WriteLine(System.UInt64)
System.IO.TextWriter.WriteLine(System.Single)
System.IO.TextWriter.WriteLine(System.Double)
System.IO.TextWriter.WriteLine(System.Decimal)
System.IO.TextWriter.WriteLine(System.String)
System.IO.TextWriter.WriteLine(System.Object)
System.IO.TextWriter.WriteLine(System.String, System.Object)
System.IO.TextWriter.WriteLine(System.String, System.Object, System.Object)
System.IO.TextWriter.WriteLine(System.String, System.Object, System.Object, System.Object)
System.IO.TextWriter.WriteLine(System.String, System.Object[])
System.IO.TextWriter.WriteAsync(System.Char[])
System.IO.TextWriter.WriteLineAsync(System.Char[])
System.IO.TextWriter.WriteLineAsync()
System.IO.TextWriter.FormatProvider
System.IO.TextWriter.NewLine
System.MarshalByRefObject.MemberwiseClone(System.Boolean)
System.MarshalByRefObject.GetLifetimeService()
System.MarshalByRefObject.InitializeLifetimeService()
System.MarshalByRefObject.CreateObjRef(System.Type)
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.WDS

Assembly: WDS-Cs.dll

Syntax

```
public class PrintWriter : StringWriter, IDisposable
```

Methods

flush()

Declaration

```
public void flush()
```

printf(String, Object[])

Declaration

```
public void printf(string fmt, params object[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	fmt	
System.Object[]	args	

println(Object)

Declaration

```
public void println(object arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	arg	

println(String)

Declaration

```
public void println(string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Implements

System.IDisposable

Extension Methods

JavaLikeExtensions.toString(Object)

Class Util

Inheritance

System.Object
Util

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.WDS

Assembly: WDS-Cs.dll

Syntax

```
public class Util
```

Methods

bIn(String, String[])

Declaration

```
public static bool bIn(string arg0, params string[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg0	
System.String[]	args	

Returns

TYPE	DESCRIPTION
System.Boolean	

CleanAsNMToken(String)

Declaration

```
[ExcelFunction(Name = "CleanAsNMToken", Category = "WDS", Description = "Returns a clean and valid \\i\\c* NMToken (name token) string for a given input, following XML 1.1, through \\uFFFF. Note use CleanAsNMTokenXSD where the first character is not treated differently.", IsVolatile = false, ExplicitRegistration = true)]
public static string CleanAsNMToken([ExcelArgument(Name = "aInputString", Description = "A general string")]
string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Returns

TYPE	DESCRIPTION
System.String	

CleanAsNMTokenXSD(String)

Declaration

```
[ExcelFunction(Name = "CleanAsNMTokenXSD", Category = "WDS", Description = "Returns a clean and valid \\c* NMToken (name token) string for a given input, through \\uFFFF. Note use CleanAsNMTokenXSD where the first character is not treated differently.", IsVolatile = false, ExplicitRegistration = true)]
public static string CleanAsNMTokenXSD([ExcelArgument(Name = "aInputString", Description = "A general string")] string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Returns

TYPE	DESCRIPTION
System.String	

CleanAsToken(String)

Declaration

```
[ExcelFunction(Name = "CleanAsToken", Category = "WDS", Description = "Returns a TOKEN of the input string where white-space is normalized.  Additionally, ascii non-printables are removed.", IsVolatile = false, ExplicitRegistration = true)]
public static string CleanAsToken([ExcelArgument(Name = "aInputString", Description = "A general string")] string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Returns

TYPE	DESCRIPTION
System.String	

CleanDeadWhiteSpaceInXML(String)

Declaration

```
[ExcelFunction(Name = "CleanDeadWhiteSpaceInXML", Category = "WDS", Description = "Removes inter-element space and non-printables in XML", IsVolatile = false, ExplicitRegistration = true)]
public static string CleanDeadWhiteSpaceInXML([ExcelArgument(Name = "aInputString", Description = "A general string")] string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Returns

TYPE	DESCRIPTION
System.String	

CleanQuotes(String)

Declaration

```
[ExcelFunction(Name = "CleanQuotes", Category = "WDS", Description = "Removes double or single quotes.",
IsVolatile = false, ExplicitRegistration = true)]
public static string CleanQuotes(string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Returns

TYPE	DESCRIPTION
System.String	

CleanStringWithRegex(String, String, String)

Declaration

```
[ExcelFunction(Name = "CleanStringWithRegex", Category = "WDS", Description = "Performs a C# eval of
Regex.Replace(InputString,RegexToFind,RegexToReplaceWith)", IsVolatile = false, ExplicitRegistration = true)]
public static string CleanStringWithRegex([ExcelArgument(Name = "InputString", Description = "A general
string")] string arg0, [ExcelArgument(Name = "RegexToFind", Description = "A Regex Expression")] string arg1,
[ExcelArgument(Name = "RegexToReplaceWith", Description = "Replacement String")] string arg2)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg0	
System.String	arg1	
System.String	arg2	

Returns

TYPE	DESCRIPTION
System.String	

FetchFileAsString(String)

Declaration

```
[ExcelFunction(Name = "FetchFileAsString", Category = "WDS", Description = "Pulls the contents of a file and returns as one string.", ExplicitRegistration = true)]
public static string FetchFileAsString([ExcelArgument(Name = "FileName")] string arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg	

Returns

TYPE	DESCRIPTION
System.String	

MatchingNullity(Object, Object)

Declaration

```
public static bool MatchingNullity(object A, object B)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	A	
System.Object	B	

Returns

TYPE	DESCRIPTION
System.Boolean	

MatchingNullity(String, String)

Declaration

```
public static bool MatchingNullity(string A, string B)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	A	
System.String	B	

Returns

TYPE	DESCRIPTION
System.Boolean	

MatchingNullityAndValueEquals(String, String)

Declaration

```
public static bool MatchingNullityAndValueEquals(string A, string B)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	A	
System.String	B	

Returns

TYPE	DESCRIPTION
System.Boolean	

PathAndName(String, String)

Declaration

```
public static string PathAndName(string aPath, string aFileName)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aPath	
System.String	aFileName	

Returns

TYPE	DESCRIPTION
System.String	

Extension Methods

JavaLikeExtensions.toString(Object)

Class WDSCommon

Inheritance

System.Object

WDSCommon

Inherited Members

System.Object.ToString()

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

Namespace: com.WDataSci.WDS

Assembly: WDS-Cs.dll

Syntax

```
public class WDSCommon
```

Methods

CrossProductEnumeration(Double, Object[])

Declaration

```
[ExcelFunction(Name = "CrossProductEnumeration", Category = "WDS", Description = "Returns an array of all combinations of inputs", IsThreadSafe = true, IsVolatile = false, ExplicitRegistration = true)]
public static object[, ] CrossProductEnumeration([ExcelArgument(Name = "Directive", Description = "0 for Count-Row-Values-Indices, 1 for Count")] double _pDirective, [ExcelArgument(Name = "Input", Description = "Set of values for an enumeration dimension, add as many as needed")] params object[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Double	_pDirective	
System.Object[]	args	

Returns

TYPE	DESCRIPTION
System.Object[,]	

Extension Methods

JavaLikeExtensions.toString(Object)

Class WDSException

Inheritance

System.Object
System.Exception
WDSException

Implements

System.Runtime.Serialization.ISerializable
System.Runtime.InteropServices._Exception

Inherited Members

System.Exception.GetBaseException()
System.Exception.ToString()
System.Exception.GetObjectData(System.Runtime.Serialization.SerializationInfo, System.Runtime.Serialization.StreamingContext)
System.Exception.GetType()
System.Exception.Message
System.Exception.Data
System.Exception.InnerException
System.Exception.TargetSite
System.Exception.StackTrace
System.Exception.HelpLink
System.Exception.Source
System.Exception.HResult
System.Exception.SerializeObjectState
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.WDS

Assembly: WDS-Cs.dll

Syntax

```
public class WDSException : Exception, ISerializable, _Exception
```

Constructors

WDSException(String)

Declaration

```
public WDSException(string msg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	msg	

WDSException(String, WDSException)

Declaration

```
public WDSException(string msg, WDSException e)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	msg	
WDSException	e	

WDSException(String, Exception)

Declaration

```
public WDSException(string msg, Exception e)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	msg	
System.Exception	e	

Fields

__Message

Declaration

```
public string __Message
```

Field Value

TYPE	DESCRIPTION
System.String	

Methods

getMessage()

Declaration

```
public string getMessage()
```

Returns

TYPE	DESCRIPTION
System.String	

toString()

Declaration

```
public string toString()
```

Returns

TYPE	DESCRIPTION
System.String	

Implements

System.Runtime.Serialization.ISerializable
System.Runtime.InteropServices._Exception

Extension Methods

JavaLikeExtensions.toString(Object)

The JNI namespace (and the files in the JNICode subdirectory of the project) come from an excellent article by Simon Agholor, Using the Java Native Interface in C#, with a link to source code.

Here there are some minor changes. Most notably, a slight adjustment on thrown exceptions and some additional signature handling. In particular, the passing of a direct ByteBuffer has been added for passing large block of data from C# to Java/jni more efficiently.

Class JavaNativeInterface

Inheritance

System.Object
JavaNativeInterface

Implements

System.IDisposable

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: JNI

Assembly: JniPMML-Cs.dll

Syntax

```
public class JavaNativeInterface : IDisposable
```

Properties

AttachToCurrentJVMThread

Declaration

```
public bool AttachToCurrentJVMThread { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Boolean	

Methods

CallMethod<T>(IntPtr, Boolean, String, List<Object>)

Declaration

```
public T CallMethod<T>(IntPtr methodId, bool bIsStaticMethod, string sig, List<object> param)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	methodId	
System.Boolean	bIsStaticMethod	
System.String	sig	
System.Collections.Generic.List<System.Object>	param	

Returns

TYPE	DESCRIPTION
T	

Type Parameters

NAME	DESCRIPTION
T	

CallMethod<T>(String, String, List<Object>)

Declaration

```
public T CallMethod<T>(string methodName, string sig, List<object> param)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	methodName	
System.String	sig	
System.Collections.Generic.List<System.Object>	param	

Returns

TYPE	DESCRIPTION
T	

Type Parameters

NAME	DESCRIPTION
T	

CallVoidMethod(String, String, List<Object>)

Declaration

```
public void CallVoidMethod(string methodName, string sig, List<object> param)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	methodName	
System.String	sig	
System.Collections.Generic.List<System.Object>	param	

Dispose()

Declaration

```
public void Dispose()
```

Dispose(Boolean)

Declaration

```
protected virtual void Dispose(bool disposing)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Boolean	disposing	

Env()

Declaration

```
public JNIEnv Env()
```

Returns

TYPE	DESCRIPTION
JNIEnv	

Finalize()

Declaration

```
protected void Finalize()
```

FindClassID(String)

Declaration

```
public IntPtr FindClassID(string ClassName)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	ClassName	

Returns

TYPE	DESCRIPTION
System.IntPtr	

FindClassObjectID(IntPtr)

Declaration

```
public IntPtr FindClassObjectID(IntPtr ClassID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	ClassID	

Returns

TYPE	DESCRIPTION
System.IntPtr	

FindMethodID(IntPtr, String, String)

Declaration

```
public IntPtr FindMethodID(IntPtr ClassObjID, string MethodName, string Signature)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	ClassObjID	
System.String	MethodName	
System.String	Signature	

Returns

TYPE	DESCRIPTION
System.IntPtr	

FindStaticMethodID(IntPtr, String, String)

Declaration

```
public IntPtr FindStaticMethodID(IntPtr ClassObjID, string MethodName, string Signature)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	ClassObjID	
System.String	MethodName	
System.String	Signature	

Returns

TYPE	DESCRIPTION
System.IntPtr	

InstantiateJavaObject(String)

Declaration

```
public void InstantiateJavaObject(string ClassName)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	ClassName	

JavaVersion()

Declaration

```
public string JavaVersion()
```

Returns

TYPE	DESCRIPTION
System.String	

LoadVM(Dictionary<String, String>, Boolean)

Declaration

```
public void LoadVM(Dictionary<string, string> options, bool AddToExistingJVM)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Collections.Generic.Dictionary<System.String, System.String>	options	
System.Boolean	AddToExistingJVM	

Implements

System.IDisposable

Extension Methods

JavaLikeExtensions.toString(Object)

Class JavaNativeMethod

Inheritance

System.Object
JavaNativeMethod

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: JNI

Assembly: JniPMML-Cs.dll

Syntax

```
public class JavaNativeMethod
```

Methods

CreateNativeMethod(Type, String, String, String)

Declaration

```
public JNINativeMethod CreateNativeMethod(Type type, string javaName, string clrName, string javaSignature)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Type	type	
System.String	javaName	
System.String	clrName	
System.String	javaSignature	

Returns

TYPE	DESCRIPTION
JNINativeMethod	

CreateType(Type)

Declaration

```
public void CreateType(Type type)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Type	type	

Extension Methods

JavaLikeExtensions.toString(Object)

Class JavaVM

Inheritance

System.Object
JavaVM

Implements

System.IDisposable

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: JNI

Assembly: JniPMML-Cs.dll

Syntax

```
public class JavaVM : IDisposable
```

Constructors

JavaVM(IntPtr)

Declaration

```
public JavaVM(IntPtr pointer)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	pointer	

Fields

CC

Declaration

```
public const CallingConvention CC = CallingConvention.Winapi
```

Field Value

TYPE	DESCRIPTION
System.Runtime.InteropServices.CallingConvention	

Methods

AttachCurrentThreadAsDaemon(out JNIEnv, Nullable<JavaVMInitArgs>)

Declaration

```
public int AttachCurrentThreadAsDaemon(out JNIEnv penv, JavaVMInitArgs? args)
```

Parameters

TYPE	NAME	DESCRIPTION
JNIEnv	penv	
System.Nullable<JavaVMInitArgs>	args	

Returns

TYPE	DESCRIPTION
System.Int32	

BooleanToByte(Boolean)

Declaration

```
public static byte BooleanToByte(bool value)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Boolean	value	

Returns

TYPE	DESCRIPTION
System.Byte	

ByteToBoolean(Byte)

Declaration

```
public static bool ByteToBoolean(byte b)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Byte	b	

Returns

TYPE	DESCRIPTION
System.Boolean	

DestroyJavaVM()

Declaration

```
public int DestroyJavaVM()
```

Returns

TYPE	DESCRIPTION
System.Int32	

DetachCurrentThread()

Declaration

```
public int DetachCurrentThread()
```

Returns

TYPE	DESCRIPTION
System.Int32	

Dispose()

Declaration

```
public void Dispose()
```

Dispose(Boolean)

Declaration

```
protected virtual void Dispose(bool disposing)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Boolean	disposing	

Finalize()

Declaration

```
protected void Finalize()
```

GetDelegateForFunctionPointer<T>(IntPtr, ref T)

Declaration

```
public static void GetDelegateForFunctionPointer<T>(IntPtr ptr, ref T res)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	ptr	
T	res	

Type Parameters

NAME	DESCRIPTION
T	

GetEnv(out JNIEnv, Int32)

Declaration

```
public int GetEnv(out JNIEnv penv, int version)
```

Parameters

TYPE	NAME	DESCRIPTION
JNIEnv	penv	
System.Int32	version	

Returns

TYPE	DESCRIPTION
System.Int32	

Implements

System.IDisposable

Extension Methods

JavaLikeExtensions.toString(Object)

Struct JavaVM.JNIInvokeInterface

Inherited Members

System.ValueType.Equals(System.Object)
System.ValueType.GetHashCode()
System.ValueType.ToString()
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetType()

Namespace: JNI

Assembly: JniPMML-Cs.dll

Syntax

```
public struct JNIInvokeInterface
```

Fields

AttachCurrentThread

Declaration

```
public IntPtr AttachCurrentThread
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

AttachCurrentThreadAsDaemon

Declaration

```
public IntPtr AttachCurrentThreadAsDaemon
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

DestroyJavaVM

Declaration

```
public IntPtr DestroyJavaVM
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

DetachCurrentThread

Declaration

```
public IntPtr DetachCurrentThread
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetEnv

Declaration

```
public IntPtr GetEnv
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

reserved0

Declaration

```
public IntPtr reserved0
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

reserved1

Declaration

```
public IntPtr reserved1
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

reserved2

Declaration

```
public IntPtr reserved2
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

Extension Methods

JavaLikeExtensions.toString(Object)

Struct JavaVM.JNIInvokeInterface_

Inherited Members

System.ValueType.Equals(System.Object)
System.ValueType.GetHashCode()
System.ValueType.ToString()
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetType()

Namespace: JNI

Assembly: JniPMML-Cs.dll

Syntax

```
public struct JNIInvokeInterface_
```

Extension Methods

JavaLikeExtensions.toString(Object)

Struct JavaVMInitArgs

Inherited Members

System.ValueType.Equals(System.Object)
System.ValueType.GetHashCode()
System.ValueType.ToString()
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetType()

Namespace: JNI

Assembly: JniPMML-Cs.dll

Syntax

```
public struct JavaVMInitArgs
```

Fields

ignoreUnrecognized

Declaration

```
public byte ignoreUnrecognized
```

Field Value

TYPE	DESCRIPTION
System.Byte	

nOptions

Declaration

```
public int nOptions
```

Field Value

TYPE	DESCRIPTION
System.Int32	

options

Declaration

```
public JavaVMOption*options
```

Field Value

TYPE	DESCRIPTION
JavaVMOption*	

version

Declaration

```
public int version
```

Field Value

TYPE	DESCRIPTION
System.Int32	

Extension Methods

JavaLikeExtensions.toString(Object)

Struct JavaVMOption

Inherited Members

System.ValueType.Equals(System.Object)
System.ValueType.GetHashCode()
System.ValueType.ToString()
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetType()

Namespace: JNI

Assembly: JniPMML-Cs.dll

Syntax

```
public struct JavaVMOption
```

Fields

extraInfo

Declaration

```
public IntPtr extraInfo
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

optionString

Declaration

```
public IntPtr optionString
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

Extension Methods

JavaLikeExtensions.toString(Object)

Struct JNIBooleanValue

Inherited Members

System.ValueType.Equals(System.Object)
System.ValueType.GetHashCode()
System.ValueType.ToString()
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetType()

Namespace: JNI

Assembly: JniPMML-Cs.dll

Syntax

```
public struct JNIBooleanValue
```

Fields

JNI_FALSE

Declaration

```
public const byte JNI_FALSE = 0
```

Field Value

TYPE	DESCRIPTION
System.Byte	

JNI_TRUE

Declaration

```
public const byte JNI_TRUE = 1
```

Field Value

TYPE	DESCRIPTION
System.Byte	

Extension Methods

JavaLikeExtensions.toString(Object)

Class JNIEnv

Inheritance

System.Object
JNIEnv

Implements

System.IDisposable

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: JNI

Assembly: JniPMML-Cs.dll

Syntax

```
public class JNIEnv : IDisposable
```

Methods

CallBooleanMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public bool CallBooleanMethod(IntPtr obj, IntPtr methodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	methodId	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.Boolean	

CallByteMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public byte CallByteMethod(IntPtr obj, IntPtr methodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	

TYPE	NAME	DESCRIPTION
System.IntPtr	methodId	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.Byte	

CallCharMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public char CallCharMethod(IntPtr obj, IntPtr methodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	methodId	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.Char	

CallDoubleMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public double CallDoubleMethod(IntPtr obj, IntPtr methodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	methodId	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.Double	

CallFloatMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public float CallFloatMethod(IntPtr obj, IntPtr methodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	methodId	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.Single	

CallIntMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public int CallIntMethod(IntPtr obj, IntPtr methodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	methodId	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.Int32	

CallLongMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public long CallLongMethod(IntPtr obj, IntPtr methodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	methodId	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.Int64	

CallObjectMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public IntPtr CallObjectMethod(IntPtr obj, IntPtr methodID, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	methodID	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.IntPtr	

CallShortMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public short CallShortMethod(IntPtr obj, IntPtr methodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	methodId	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.Int16	

CallStaticBooleanMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public bool CallStaticBooleanMethod(IntPtr jniClass, IntPtr MethodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	jniClass	
System.IntPtr	MethodId	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.Boolean	

CallStaticByteMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public byte CallStaticByteMethod(IntPtr jniClass, IntPtr MethodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	jniClass	
System.IntPtr	MethodId	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.Byte	

CallStaticCharMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public char CallStaticCharMethod(IntPtr jniClass, IntPtr MethodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	jniClass	
System.IntPtr	MethodId	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.Char	

CallStaticDoubleMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public double CallStaticDoubleMethod(IntPtr jniClass, IntPtr MethodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	jniClass	
System.IntPtr	MethodId	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.Double	

CallStaticFloatMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public float CallStaticFloatMethod(IntPtr jniClass, IntPtr MethodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	jniClass	
System.IntPtr	MethodId	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.Single	

CallStaticIntMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public int CallStaticIntMethod(IntPtr jniClass, IntPtr MethodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	jniClass	
System.IntPtr	MethodId	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.Int32	

CallStaticLongMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public long CallStaticLongMethod(IntPtr jniClass, IntPtr MethodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	jniClass	
System.IntPtr	MethodId	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.Int64	

CallStaticObjectMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public IntPtr CallStaticObjectMethod(IntPtr obj, IntPtr methodID, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	methodID	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.IntPtr	

CallStaticShortMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public short CallStaticShortMethod(IntPtr jniClass, IntPtr MethodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	jniClass	
System.IntPtr	MethodId	
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.Int16	

CallStaticVoidMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public void CallStaticVoidMethod(IntPtr jniClass, IntPtr methodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	jniClass	
System.IntPtr	methodId	
JValue[]	args	

CallVoidMethod(IntPtr, IntPtr, JValue[])

Declaration

```
public void CallVoidMethod(IntPtr obj, IntPtr methodId, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	methodId	
JValue[]	args	

CatchJavaException()

Declaration

```
public string CatchJavaException()
```

Returns

TYPE	DESCRIPTION
System.String	

CheckJavaExceptionAndThrow()

Declaration

```
public bool CheckJavaExceptionAndThrow()
```

Returns

TYPE	DESCRIPTION
System.Boolean	

Dispose()

Declaration

```
public void Dispose()
```

Dispose(Boolean)

Declaration

```
protected virtual void Dispose(bool disposing)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Boolean	disposing	

ExceptionClear()

Declaration

```
public void ExceptionClear()
```

ExceptionDescribe()

Declaration

```
public void ExceptionDescribe()
```

ExceptionOccurred()

Declaration

```
public IntPtr ExceptionOccurred()
```

Returns

TYPE	DESCRIPTION
System.IntPtr	

FatalError(String)

Declaration

```
public void FatalError(string message)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	message	

Finalize()

Declaration

```
protected void Finalize()
```

FindClass(String)

Declaration

```
public IntPtr FindClass(string name)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	name	

Returns

TYPE	DESCRIPTION
System.IntPtr	

FromReflectedField(IntPtr)

Declaration

```
public IntPtr FromReflectedField(IntPtr FieldId)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	FieldId	

Returns

TYPE	DESCRIPTION
System.IntPtr	

FromReflectedMethod(IntPtr)

Declaration

```
public IntPtr FromReflectedMethod(IntPtr methodId)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	methodId	

Returns

TYPE	DESCRIPTION
System.IntPtr	

GetArrayLength(IntPtr)

Declaration

```
public int GetArrayLength(IntPtr obj)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	

Returns

TYPE	DESCRIPTION
System.Int32	

GetBooleanField(IntPtr, IntPtr)

Declaration

```
public bool GetBooleanField(IntPtr obj, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.Boolean	

GetByteField(IntPtr, IntPtr)

Declaration

```
public byte GetByteField(IntPtr obj, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.Byte	

GetCharField(IntPtr, IntPtr)

Declaration

```
public char GetCharField(IntPtr obj, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.Char	

GetDirectBufferAddress(IntPtr)

Declaration

```
public IntPtr GetDirectBufferAddress(IntPtr buf)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	buf	

Returns

TYPE	DESCRIPTION
System.IntPtr	

GetDirectBufferCapacity(IntPtr)

Declaration

```
public long GetDirectBufferCapacity(IntPtr buf)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	buf	

Returns

TYPE	DESCRIPTION
System.Int64	

GetDoubleField(IntPtr, IntPtr)

Declaration

```
public double GetDoubleField(IntPtr obj, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.Double	

GetFieldID(IntPtr, String, String)

Declaration

```
public IntPtr GetFieldID(IntPtr jniClass, string name, string sig)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	jniClass	
System.String	name	
System.String	sig	

Returns

TYPE	DESCRIPTION
System.IntPtr	

GetFloatField(IntPtr, IntPtr)

Declaration

```
public float GetFloatField(IntPtr obj, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.Single	

GetIntArray(IntPtr)

Declaration

```
public int[] GetIntArray(IntPtr obj)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	

Returns

TYPE	DESCRIPTION
System.Int32[]	

GetIntField(IntPtr, IntPtr)

Declaration

```
public int GetIntField(IntPtr obj, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.Int32	

GetJavaVM()

Declaration

```
public JavaVM GetJavaVM()
```

Returns

TYPE	DESCRIPTION
JavaVM	

GetLongField(IntPtr, IntPtr)

Declaration

```
public long GetLongField(IntPtr obj, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.Int64	

GetMajorVersion()

Declaration

```
public int GetMajorVersion()
```

Returns

TYPE	DESCRIPTION
System.Int32	

GetMethodId(IntPtr, String, String)

Declaration

```
public IntPtr GetMethodId(IntPtr jniClass, string name, string sig)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	jniClass	
System.String	name	
System.String	sig	

Returns

TYPE	DESCRIPTION
System.IntPtr	

GetMinorVersion()

Declaration

```
public int GetMinorVersion()
```

Returns

TYPE	DESCRIPTION
System.Int32	

GetObjectField(IntPtr, IntPtr)

Declaration

```
public IntPtr GetObjectField(IntPtr obj, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.IntPtr	

GetShortField(IntPtr, IntPtr)

Declaration

```
public short GetShortField(IntPtr obj, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	obj	
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.Int16	

GetStaticBooleanField(IntPtr, IntPtr)

Declaration

```
public bool GetStaticBooleanField(IntPtr clazz, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	clazz	
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.Boolean	

GetStaticByteField(IntPtr, IntPtr)

Declaration

```
public byte GetStaticByteField(IntPtr classHandle, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	classHandle	
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.Byte	

GetStaticCharField(IntPtr, IntPtr)

Declaration

```
public char GetStaticCharField(IntPtr classHandle, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	classHandle	
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.Char	

GetStaticDoubleField(IntPtr, IntPtr)

Declaration

```
public double GetStaticDoubleField(IntPtr classHandle, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	classHandle	
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.Double	

GetStaticFieldID(IntPtr, String, String)

Declaration

```
public IntPtr GetStaticFieldID(IntPtr classHandle, string name, string sig)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	classHandle	
System.String	name	
System.String	sig	

Returns

TYPE	DESCRIPTION
System.IntPtr	

GetStaticFloatField(IntPtr, IntPtr)

Declaration

```
public float GetStaticFloatField(IntPtr classHandle, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	classHandle	
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.Single	

GetStaticIntField(IntPtr, IntPtr)

Declaration

```
public int GetStaticIntField(IntPtr classHandle, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	classHandle	
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.Int32	

GetStaticLongField(IntPtr, IntPtr)

Declaration

```
public long GetStaticLongField(IntPtr classHandle, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	classHandle	
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.Int64	

GetStaticMethodID(IntPtr, String, String)

Declaration

```
public IntPtr GetStaticMethodID(IntPtr jniClass, string name, string sig)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	jniClass	
System.String	name	
System.String	sig	

Returns

TYPE	DESCRIPTION
System.IntPtr	

GetStaticObjectField(IntPtr, IntPtr)

Declaration

```
public IntPtr GetStaticObjectField(IntPtr clazz, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	clazz	

TYPE	NAME	DESCRIPTION
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.IntPtr	

GetStaticShortField(IntPtr, IntPtr)

Declaration

```
public short GetStaticShortField(IntPtr classHandle, IntPtr fieldID)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	classHandle	
System.IntPtr	fieldID	

Returns

TYPE	DESCRIPTION
System.Int16	

GetVersion()

Declaration

```
public int GetVersion()
```

Returns

TYPE	DESCRIPTION
System.Int32	

IsAssignableFrom(IntPtr, IntPtr)

Declaration

```
public byte IsAssignableFrom(IntPtr subclassHandle, IntPtr superclassHandle)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	subclassHandle	
System.IntPtr	superclassHandle	

Returns

TYPE	DESCRIPTION
System.Byte	

NewByteArray(Int32, IntPtr)

Declaration

```
public IntPtr NewByteArray(int len, IntPtr classHandle)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	len	
System.IntPtr	classHandle	

Returns

TYPE	DESCRIPTION
System.IntPtr	

NewCharArray(Int32, IntPtr)

Declaration

```
public IntPtr NewCharArray(int len, IntPtr classHandle)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	len	
System.IntPtr	classHandle	

Returns

TYPE	DESCRIPTION
System.IntPtr	

NewDirectByteBuffer(IntPtr, Int64)

Declaration

```
public IntPtr NewDirectByteBuffer(IntPtr address, long capacity)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	address	
System.Int64	capacity	

Returns

TYPE	DESCRIPTION
System.IntPtr	

NewDoubleArray(Int32, IntPtr)

Declaration

```
public IntPtr NewDoubleArray(int len, IntPtr classHandle)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	len	
System.IntPtr	classHandle	

Returns

TYPE	DESCRIPTION
System.IntPtr	

NewFloatArray(Int32, IntPtr)

Declaration

```
public IntPtr NewFloatArray(int len, IntPtr classHandle)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	len	
System.IntPtr	classHandle	

Returns

TYPE	DESCRIPTION
System.IntPtr	

NewGlobalRef(IntPtr)

Declaration

```
public IntPtr NewGlobalRef(IntPtr objectHandle)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	objectHandle	

Returns

TYPE	DESCRIPTION
System.IntPtr	

NewIntArray(Int32, IntPtr)

Declaration

```
public IntPtr NewIntArray(int len, IntPtr classHandle)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	len	
System.IntPtr	classHandle	

Returns

TYPE	DESCRIPTION
System.IntPtr	

NewLongArray(Int32, IntPtr)

Declaration

```
public IntPtr NewLongArray(int len, IntPtr classHandle)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	len	
System.IntPtr	classHandle	

Returns

TYPE	DESCRIPTION
System.IntPtr	

NewObject(IntPtr, IntPtr, JValue[])

Declaration

```
public IntPtr NewObject(IntPtr classHandle, IntPtr methodID, params JValue[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	classHandle	
System.IntPtr	methodID	

TYPE	NAME	DESCRIPTION
JValue[]	args	

Returns

TYPE	DESCRIPTION
System.IntPtr	

NewObjectArray(Int32, IntPtr, IntPtr)

Declaration

```
public IntPtr NewObjectArray(int len, IntPtr classHandle, IntPtr init)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	len	
System.IntPtr	classHandle	
System.IntPtr	init	

Returns

TYPE	DESCRIPTION
System.IntPtr	

NewShortArray(Int32, IntPtr)

Declaration

```
public IntPtr NewShortArray(int len, IntPtr classHandle)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	len	
System.IntPtr	classHandle	

Returns

TYPE	DESCRIPTION
System.IntPtr	

NewString(String, Int32)

Declaration

```
public IntPtr NewString(string unicode, int len)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	unicode	
System.Int32	len	

Returns

TYPE	DESCRIPTION
System.IntPtr	

NewStringUFT(IntPtr)

Declaration

```
public IntPtr NewStringUFT(IntPtr UFT)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	UFT	

Returns

TYPE	DESCRIPTION
System.IntPtr	

RegisterNatives(IntPtr, JNINativeMethod*, Int32)

Declaration

```
public int RegisterNatives(IntPtr classHandle, JNINativeMethod*methods, int nMethods)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	classHandle	
JNINativeMethod*	methods	
System.Int32	nMethods	

Returns

TYPE	DESCRIPTION
System.Int32	

SetObjectArrayElement(IntPtr, Int32, IntPtr)

Declaration

```
public void SetObjectArrayElement(IntPtr array, int index, IntPtr val)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	array	
System.Int32	index	
System.IntPtr	val	

ThrowNew(IntPtr, String)

Declaration

```
public void ThrowNew(IntPtr classHandle, string message)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	classHandle	
System.String	message	

ToReflectedField(IntPtr, IntPtr, Boolean)

Declaration

```
public IntPtr ToReflectedField(IntPtr classHandle, IntPtr fieldID, bool isStatic)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	classHandle	
System.IntPtr	fieldID	
System.Boolean	isStatic	

Returns

TYPE	DESCRIPTION
System.IntPtr	

ToReflectedMethod(IntPtr, IntPtr, Boolean)

Declaration

```
public IntPtr ToReflectedMethod(IntPtr classHandle, IntPtr methodId, bool isStatic)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	classHandle	
System.IntPtr	methodId	

TYPE	NAME	DESCRIPTION
System.Boolean	isStatic	

Returns

TYPE	DESCRIPTION
System.IntPtr	

UnregisterNatives(IntPtr)

Declaration

```
public int UnregisterNatives(IntPtr classHandle)
```

Parameters

TYPE	NAME	DESCRIPTION
System.IntPtr	classHandle	

Returns

TYPE	DESCRIPTION
System.Int32	

Implements

System.IDisposable

Extension Methods

JavaLikeExtensions.toString(Object)

Struct JNINativeInterface

Inherited Members

System.ValueType.Equals(System.Object)
System.ValueType.GetHashCode()
System.ValueType.ToString()
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetType()

Namespace: JNI

Assembly: JniPMML-Cs.dll

Syntax

```
public struct JNINativeInterface
```

Fields

AllocObject

Declaration

```
public IntPtr AllocObject
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallBooleanMethod

Declaration

```
public IntPtr CallBooleanMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallBooleanMethodA

Declaration

```
public IntPtr CallBooleanMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallBooleanMethodV

Declaration

```
public IntPtr CallBooleanMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallByteMethod

Declaration

```
public IntPtr CallByteMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallByteMethodA

Declaration

```
public IntPtr CallByteMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallByteMethodV

Declaration

```
public IntPtr CallByteMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallCharMethod

Declaration

```
public IntPtr CallCharMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallCharMethodA

Declaration

```
public IntPtr CallCharMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallCharMethodV

Declaration

```
public IntPtr CallCharMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallDoubleMethod

Declaration

```
public IntPtr CallDoubleMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallDoubleMethodA

Declaration

```
public IntPtr CallDoubleMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallDoubleMethodV

Declaration

```
public IntPtr CallDoubleMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallFloatMethod

Declaration

```
public IntPtr CallFloatMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallFloatMethodA

Declaration

```
public IntPtr CallFloatMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallFloatMethodV

Declaration

```
public IntPtr CallFloatMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallIntMethod

Declaration

```
public IntPtr CallIntMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallIntMethodA

Declaration

```
public IntPtr CallIntMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallIntMethodV

Declaration

```
public IntPtr CallIntMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallLongMethod

Declaration

```
public IntPtr CallLongMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallLongMethodA

Declaration

```
public IntPtr CallLongMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallLongMethodV

Declaration

```
public IntPtr CallLongMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualBooleanMethod

Declaration

```
public IntPtr CallNonvirtualBooleanMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualBooleanMethodA

Declaration

```
public IntPtr CallNonvirtualBooleanMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualBooleanMethodV

Declaration

```
public IntPtr CallNonvirtualBooleanMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualByteMethod

Declaration

```
public IntPtr CallNonvirtualByteMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualByteMethodA

Declaration

```
public IntPtr CallNonvirtualByteMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualByteMethodV

Declaration

```
public IntPtr CallNonvirtualByteMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualCharMethod

Declaration

```
public IntPtr CallNonvirtualCharMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualCharMethodA

Declaration

```
public IntPtr CallNonvirtualCharMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualCharMethodV

Declaration

```
public IntPtr CallNonvirtualCharMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualDoubleMethod

Declaration

```
public IntPtr CallNonvirtualDoubleMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualDoubleMethodA

Declaration

```
public IntPtr CallNonvirtualDoubleMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualDoubleMethodV

Declaration

```
public IntPtr CallNonvirtualDoubleMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualFloatMethod

Declaration

```
public IntPtr CallNonvirtualFloatMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualFloatMethodA

Declaration

```
public IntPtr CallNonvirtualFloatMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualFloatMethodV

Declaration

```
public IntPtr CallNonvirtualFloatMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualIntMethod

Declaration

```
public IntPtr CallNonvirtualIntMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualIntMethodA

Declaration

```
public IntPtr CallNonvirtualIntMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualIntMethodV

Declaration

```
public IntPtr CallNonvirtualIntMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualLongMethod

Declaration

```
public IntPtr CallNonvirtualLongMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualLongMethodA

Declaration

```
public IntPtr CallNonvirtualLongMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualLongMethodV

Declaration

```
public IntPtr CallNonvirtualLongMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualObjectMethod

Declaration

```
public IntPtr CallNonvirtualObjectMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualObjectMethodA

Declaration

```
public IntPtr CallNonvirtualObjectMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualObjectMethodV

Declaration

```
public IntPtr CallNonvirtualObjectMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualShortMethod

Declaration

```
public IntPtr CallNonvirtualShortMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualShortMethodA

Declaration

```
public IntPtr CallNonvirtualShortMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualShortMethodV

Declaration

```
public IntPtr CallNonvirtualShortMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualVoidMethod

Declaration

```
public IntPtr CallNonvirtualVoidMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualVoidMethodA

Declaration

```
public IntPtr CallNonvirtualVoidMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallNonvirtualVoidMethodV

Declaration

```
public IntPtr CallNonvirtualVoidMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallObjectMethod

Declaration

```
public IntPtr CallObjectMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallObjectMethodA

Declaration

```
public IntPtr CallObjectMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallObjectMethodV

Declaration

```
public IntPtr CallObjectMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallShortMethod

Declaration

```
public IntPtr CallShortMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallShortMethodA

Declaration

```
public IntPtr CallShortMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallShortMethodV

Declaration

```
public IntPtr CallShortMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticBooleanMethod

Declaration

```
public IntPtr CallStaticBooleanMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticBooleanMethodA

Declaration

```
public IntPtr CallStaticBooleanMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticBooleanMethodV

Declaration

```
public IntPtr CallStaticBooleanMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticByteMethod

Declaration

```
public IntPtr CallStaticByteMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticByteMethodA

Declaration

```
public IntPtr CallStaticByteMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticByteMethodV

Declaration

```
public IntPtr CallStaticByteMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticCharMethod

Declaration

```
public IntPtr CallStaticCharMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticCharMethodA

Declaration

```
public IntPtr CallStaticCharMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticCharMethodV

Declaration

```
public IntPtr CallStaticCharMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticDoubleMethod

Declaration

```
public IntPtr CallStaticDoubleMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticDoubleMethodA

Declaration

```
public IntPtr CallStaticDoubleMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticDoubleMethodV

Declaration

```
public IntPtr CallStaticDoubleMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticFloatMethod

Declaration

```
public IntPtr CallStaticFloatMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticFloatMethodA

Declaration

```
public IntPtr CallStaticFloatMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticFloatMethodV

Declaration

```
public IntPtr CallStaticFloatMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticIntMethod

Declaration

```
public IntPtr CallStaticIntMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticIntMethodA

Declaration

```
public IntPtr CallStaticIntMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticIntMethodV

Declaration

```
public IntPtr CallStaticIntMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticLongMethod

Declaration

```
public IntPtr CallStaticLongMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticLongMethodA

Declaration

```
public IntPtr CallStaticLongMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticLongMethodV

Declaration

```
public IntPtr CallStaticLongMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticObjectMethod

Declaration

```
public IntPtr CallStaticObjectMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticObjectMethodA

Declaration

```
public IntPtr CallStaticObjectMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticObjectMethodV

Declaration

```
public IntPtr CallStaticObjectMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticShortMethod

Declaration

```
public IntPtr CallStaticShortMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticShortMethodA

Declaration

```
public IntPtr CallStaticShortMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticShortMethodV

Declaration

```
public IntPtr CallStaticShortMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticVoidMethod

Declaration

```
public IntPtr CallStaticVoidMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticVoidMethodA

Declaration

```
public IntPtr CallStaticVoidMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallStaticVoidMethodV

Declaration

```
public IntPtr CallStaticVoidMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallVoidMethod

Declaration

```
public IntPtr CallVoidMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallVoidMethodA

Declaration

```
public IntPtr CallVoidMethodA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

CallVoidMethodV

Declaration

```
public IntPtr CallVoidMethodV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

DefineClass

Declaration

```
public IntPtr DefineClass
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

DeleteGlobalRef

Declaration

```
public IntPtr DeleteGlobalRef
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

DeleteLocalRef

Declaration

```
public IntPtr DeleteLocalRef
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

DeleteWeakGlobalRef

Declaration

```
public IntPtr DeleteWeakGlobalRef
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

EnsureLocalCapacity

Declaration

```
public IntPtr EnsureLocalCapacity
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ExceptionCheck

Declaration

```
public IntPtr ExceptionCheck
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ExceptionClear

Declaration

```
public IntPtr ExceptionClear
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ExceptionDescribe

Declaration

```
public IntPtr ExceptionDescribe
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ExceptionOccurred

Declaration

```
public IntPtr ExceptionOccurred
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

FatalError

Declaration

```
public IntPtr FatalError
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

FindClass

Declaration

```
public IntPtr FindClass
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

FromReflectedField

Declaration

```
public IntPtr FromReflectedField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

FromReflectedMethod

Declaration

```
public IntPtr FromReflectedMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetArrayLength

Declaration

```
public IntPtr GetArrayLength
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetBooleanArrayElements

Declaration

```
public IntPtr GetBooleanArrayElements
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetBooleanArrayRegion

Declaration

```
public IntPtr GetBooleanArrayRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetBooleanField

Declaration

```
public IntPtr GetBooleanField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetByteArrayElements

Declaration

```
public IntPtr GetByteArrayElements
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetByteArrayRegion

Declaration

```
public IntPtr GetByteArrayRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetByteField

Declaration

```
public IntPtr GetByteField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetCharArrayElements

Declaration

```
public IntPtr GetCharArrayElements
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetCharArrayRegion

Declaration

```
public IntPtr GetCharArrayRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetCharField

Declaration

```
public IntPtr GetCharField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetDirectBufferAddress

Declaration

```
public IntPtr GetDirectBufferAddress
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetDirectBufferCapacity

Declaration

```
public IntPtr GetDirectBufferCapacity
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetDoubleArrayElements

Declaration

```
public IntPtr GetDoubleArrayElements
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetDoubleArrayRegion

Declaration

```
public IntPtr GetDoubleArrayRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetDoubleField

Declaration

```
public IntPtr GetDoubleField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetFieldID

Declaration

```
public IntPtr GetFieldID
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetFloatArrayElements

Declaration

```
public IntPtr GetFloatArrayElements
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetFloatArrayRegion

Declaration

```
public IntPtr GetFloatArrayRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetFloatField

Declaration

```
public IntPtr GetFloatField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetIntArrayElements

Declaration

```
public IntPtr GetIntArrayElements
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetIntArrayRegion

Declaration

```
public IntPtr GetIntArrayRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetIntField

Declaration

```
public IntPtr GetIntField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetJavaVM

Declaration

```
public IntPtr GetJavaVM
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetLongArrayElements

Declaration

```
public IntPtr GetLongArrayElements
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetLongArrayRegion

Declaration

```
public IntPtr GetLongArrayRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetLongField

Declaration

```
public IntPtr GetLongField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetMethodID

Declaration

```
public IntPtr GetMethodID
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetObjectArrayElement

Declaration

```
public IntPtr GetObjectArrayElement
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetObjectClass

Declaration

```
public IntPtr GetObjectClass
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetObjectField

Declaration

```
public IntPtr GetObjectField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetPrimitiveArrayCritical

Declaration

```
public IntPtr GetPrimitiveArrayCritical
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetShortArrayElements

Declaration

```
public IntPtr GetShortArrayElements
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetShortArrayRegion

Declaration

```
public IntPtr GetShortArrayRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetShortField

Declaration

```
public IntPtr GetShortField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStaticBooleanField

Declaration

```
public IntPtr GetStaticBooleanField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStaticByteField

Declaration

```
public IntPtr GetStaticByteField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStaticCharField

Declaration

```
public IntPtr GetStaticCharField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStaticDoubleField

Declaration

```
public IntPtr GetStaticDoubleField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStaticFieldID

Declaration

```
public IntPtr GetStaticFieldID
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStaticFloatField

Declaration

```
public IntPtr GetStaticFloatField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStaticIntField

Declaration

```
public IntPtr GetStaticIntField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStaticLongField

Declaration

```
public IntPtr GetStaticLongField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStaticMethodID

Declaration

```
public IntPtr GetStaticMethodID
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStaticObjectField

Declaration

```
public IntPtr GetStaticObjectField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStaticShortField

Declaration

```
public IntPtr GetStaticShortField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStringChars

Declaration

```
public IntPtr GetStringChars
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStringCritical

Declaration

```
public IntPtr GetStringCritical
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStringLength

Declaration

```
public IntPtr GetStringLength
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStringRegion

Declaration

```
public IntPtr GetStringRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStringUTFChars

Declaration

```
public IntPtr GetStringUTFChars
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStringUTFLength

Declaration

```
public IntPtr GetStringUTFLength
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetStringUTFRegion

Declaration

```
public IntPtr GetStringUTFRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetSuperclass

Declaration

```
public IntPtr GetSuperclass
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

GetVersion

Declaration

```
public IntPtr GetVersion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

IsAssignableFrom

Declaration

```
public IntPtr IsAssignableFrom
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

IsInstanceOf

Declaration

```
public IntPtr IsInstanceOf
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

IsSameObject

Declaration

```
public IntPtr IsSameObject
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

MonitorEnter

Declaration

```
public IntPtr MonitorEnter
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

MonitorExit

Declaration

```
public IntPtr MonitorExit
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewBooleanArray

Declaration

```
public IntPtr NewBooleanArray
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewByteArray

Declaration

```
public IntPtr NewByteArray
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewCharArray

Declaration

```
public IntPtr NewCharArray
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewDirectByteBuffer

Declaration

```
public IntPtr NewDirectByteBuffer
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewDoubleArray

Declaration

```
public IntPtr NewDoubleArray
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewFloatArray

Declaration

```
public IntPtr NewFloatArray
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewGlobalRef

Declaration

```
public IntPtr NewGlobalRef
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewIntArray

Declaration

```
public IntPtr NewIntArray
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewLocalRef

Declaration

```
public IntPtr NewLocalRef
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewLongArray

Declaration

```
public IntPtr NewLongArray
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewObject

Declaration

```
public IntPtr NewObject
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewObjectA

Declaration

```
public IntPtr NewObjectA
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewObjectArray

Declaration

```
public IntPtr NewObjectArray
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewObjectV

Declaration

```
public IntPtr NewObjectV
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewShortArray

Declaration

```
public IntPtr NewShortArray
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewString

Declaration

```
public IntPtr NewString
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewStringUTF

Declaration

```
public IntPtr NewStringUTF
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

NewWeakGlobalRef

Declaration

```
public IntPtr NewWeakGlobalRef
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

PopLocalFrame

Declaration

```
public IntPtr PopLocalFrame
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

PushLocalFrame

Declaration

```
public IntPtr PushLocalFrame
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

RegisterNatives

Declaration

```
public IntPtr RegisterNatives
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ReleaseBooleanArrayElements

Declaration

```
public IntPtr ReleaseBooleanArrayElements
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ReleaseByteArrayElements

Declaration

```
public IntPtr ReleaseByteArrayElements
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ReleaseCharArrayElements

Declaration

```
public IntPtr ReleaseCharArrayElements
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ReleaseDoubleArrayElements

Declaration

```
public IntPtr ReleaseDoubleArrayElements
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ReleaseFloatArrayElements

Declaration

```
public IntPtr ReleaseFloatArrayElements
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ReleaseIntArrayElements

Declaration

```
public IntPtr ReleaseIntArrayElements
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ReleaseLongArrayElements

Declaration

```
public IntPtr ReleaseLongArrayElements
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ReleasePrimitiveArrayCritical

Declaration

```
public IntPtr ReleasePrimitiveArrayCritical
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ReleaseShortArrayElements

Declaration

```
public IntPtr ReleaseShortArrayElements
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ReleaseStringChars

Declaration

```
public IntPtr ReleaseStringChars
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ReleaseStringCritical

Declaration

```
public IntPtr ReleaseStringCritical
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ReleaseStringUTFChars

Declaration

```
public IntPtr ReleaseStringUTFChars
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

reserved0

Declaration

```
public IntPtr reserved0
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

reserved1

Declaration

```
public IntPtr reserved1
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

reserved2

Declaration

```
public IntPtr reserved2
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

reserved3

Declaration

```
public IntPtr reserved3
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetBooleanArrayRegion

Declaration

```
public IntPtr SetBooleanArrayRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetBooleanField

Declaration

```
public IntPtr SetBooleanField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetByteArrayRegion

Declaration

```
public IntPtr SetByteArrayRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetByteField

Declaration

```
public IntPtr SetByteField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetCharArrayRegion

Declaration

```
public IntPtr SetCharArrayRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetCharField

Declaration

```
public IntPtr SetCharField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetDoubleArrayRegion

Declaration

```
public IntPtr SetDoubleArrayRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetDoubleField

Declaration

```
public IntPtr SetDoubleField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetFloatArrayRegion

Declaration

```
public IntPtr SetFloatArrayRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetFloatField

Declaration

```
public IntPtr SetFloatField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetIntArrayRegion

Declaration

```
public IntPtr SetIntArrayRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetIntField

Declaration

```
public IntPtr SetIntField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetLongArrayRegion

Declaration

```
public IntPtr SetLongArrayRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetLongField

Declaration

```
public IntPtr SetLongField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetObjectArrayElement

Declaration

```
public IntPtr SetObjectArrayElement
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetObjectField

Declaration

```
public IntPtr SetObjectField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetShortArrayRegion

Declaration

```
public IntPtr SetShortArrayRegion
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetShortField

Declaration

```
public IntPtr SetShortField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetStaticBooleanField

Declaration

```
public IntPtr SetStaticBooleanField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetStaticByteField

Declaration

```
public IntPtr SetStaticByteField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetStaticCharField

Declaration

```
public IntPtr SetStaticCharField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetStaticDoubleField

Declaration

```
public IntPtr SetStaticDoubleField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetStaticFloatField

Declaration

```
public IntPtr SetStaticFloatField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetStaticIntField

Declaration

```
public IntPtr SetStaticIntField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetStaticLongField

Declaration

```
public IntPtr SetStaticLongField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetStaticObjectField

Declaration

```
public IntPtr SetStaticObjectField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

SetStaticShortField

Declaration

```
public IntPtr SetStaticShortField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

Throw

Declaration

```
public IntPtr Throw
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ThrowNew

Declaration

```
public IntPtr ThrowNew
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ToReflectedField

Declaration

```
public IntPtr ToReflectedField
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

ToReflectedMethod

Declaration

```
public IntPtr ToReflectedMethod
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

UnregisterNatives

Declaration

```
public IntPtr UnregisterNatives
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

Extension Methods

JavaLikeExtensions.toString(Object)

Struct JNINativeMethod

Implements

System.IDisposable

Inherited Members

System.ValueType.Equals(System.Object)
System.ValueType.GetHashCode()
System.ValueType.ToString()
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetType()

Namespace: JNI

Assembly: JniPMML-Cs.dll

Syntax

```
public struct JNINativeMethod : IDisposable
```

Fields

fnPtr

Declaration

```
public IntPtr fnPtr
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

name

Declaration

```
public IntPtr name
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

signature

Declaration

```
public IntPtr signature
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

Methods

CreateNativeMethod(String, String, IntPtr)

Declaration

```
public static JNINativeMethod CreateNativeMethod(string javaName, string javaSignature, IntPtr funcPointer)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	javaName	
System.String	javaSignature	
System.IntPtr	funcPointer	

Returns

TYPE	DESCRIPTION
JNINativeMethod	

Dispose()

Declaration

```
public void Dispose()
```

Implements

System.IDisposable

Extension Methods

JavaLikeExtensions.toString(Object)

Struct JNIReturnValue

Inherited Members

System.ValueType.Equals(System.Object)
System.ValueType.GetHashCode()
System.ValueType.ToString()
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetType()

Namespace: JNI

Assembly: JniPMML-Cs.dll

Syntax

```
public struct JNIReturnValue
```

Fields

JNI_ABORT

Declaration

```
public const int JNI_ABORT = 2
```

Field Value

TYPE	DESCRIPTION
System.Int32	

JNI_COMMIT

Declaration

```
public const int JNI_COMMIT = 1
```

Field Value

TYPE	DESCRIPTION
System.Int32	

JNI_EDETACHED

Declaration

```
public const int JNI_EDETACHED = -2
```

Field Value

TYPE	DESCRIPTION
System.Int32	

JNI_EEXIST

Declaration

```
public const int JNI_EEXIST = -5
```

Field Value

TYPE	DESCRIPTION
System.Int32	

JNI_EINVAL

Declaration

```
public const int JNI_EINVAL = -6
```

Field Value

TYPE	DESCRIPTION
System.Int32	

JNI_ENOJava

Declaration

```
public const int JNI_ENOJava = 101
```

Field Value

TYPE	DESCRIPTION
System.Int32	

JNI_ENOMEM

Declaration

```
public const int JNI_ENOMEM = -4
```

Field Value

TYPE	DESCRIPTION
System.Int32	

JNI_ERR

Declaration

```
public const int JNI_ERR = -1
```

Field Value

TYPE	DESCRIPTION
System.Int32	

JNI_EVERSION

Declaration

```
public const int JNI_EVERSION = -3
```

Field Value

TYPE	DESCRIPTION
System.Int32	

JNI_OK

Declaration

```
public const int JNI_OK = 0
```

Field Value

TYPE	DESCRIPTION
System.Int32	

Extension Methods

JavaLikeExtensions.toString(Object)

Struct JNIVersion

Inherited Members

System.ValueType.Equals(System.Object)
System.ValueType.GetHashCode()
System.ValueType.ToString()
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetType()

Namespace: JNI

Assembly: JniPMML-Cs.dll

Syntax

```
public struct JNIVersion
```

Fields

JNI_VERSION_1_2

Declaration

```
public const int JNI_VERSION_1_2 = 65538
```

Field Value

TYPE	DESCRIPTION
System.Int32	

JNI_VERSION_1_4

Declaration

```
public const int JNI_VERSION_1_4 = 65540
```

Field Value

TYPE	DESCRIPTION
System.Int32	

JNI_VERSION_1_6

Declaration

```
public const int JNI_VERSION_1_6 = 65542
```

Field Value

TYPE	DESCRIPTION
System.Int32	

JNI_VERSION_1_8

Declaration

```
public const int JNI_VERSION_1_8 = 65544
```

Field Value

TYPE	DESCRIPTION
System.Int32	

JNI_VERSION_10

Declaration

```
public const int JNI_VERSION_10 = 655360
```

Field Value

TYPE	DESCRIPTION
System.Int32	

JNI_VERSION_9

Declaration

```
public const int JNI_VERSION_9 = 589824
```

Field Value

TYPE	DESCRIPTION
System.Int32	

Extension Methods

JavaLikeExtensions.toString(Object)

Struct JValue

Inherited Members

System.ValueType.Equals(System.Object)
System.ValueType.GetHashCode()
System.ValueType.ToString()
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetType()

Namespace: JNI

Assembly: JniPMML-Cs.dll

Syntax

```
public struct JValue
```

Fields

b

Declaration

```
public byte b
```

Field Value

TYPE	DESCRIPTION
System.Byte	

c

Declaration

```
public char c
```

Field Value

TYPE	DESCRIPTION
System.Char	

d

Declaration

```
public double d
```

Field Value

TYPE	DESCRIPTION
System.Double	

f

Declaration

```
public float f
```

Field Value

TYPE	DESCRIPTION
System.Single	

i

Declaration

```
public int i
```

Field Value

TYPE	DESCRIPTION
System.Int32	

j

Declaration

```
public long j
```

Field Value

TYPE	DESCRIPTION
System.Int64	

l

Declaration

```
public IntPtr l
```

Field Value

TYPE	DESCRIPTION
System.IntPtr	

s

Declaration

```
public short s
```

Field Value

TYPE	DESCRIPTION
System.Int16	

z

Declaration

```
public byte z
```

Field Value

TYPE	DESCRIPTION
System.Byte	

Extension Methods

JavaLikeExtensions.toString(Object)

Namespace WDataSci.FlatFile

The WDataSci.FlatFile namespace encompasses classes and methods which the addin uses to create and read flat files, such as CSV or delimited files.

Classes

Cmds

Class Cmds

Inheritance

System.Object

Cmds

Inherited Members

System.Object.ToString()

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

Namespace: WDataSci.FlatFile

Assembly: JniPMML-Cs.dll

Syntax

```
public class Cmds
```

Methods

ExportXmlMappedListToCSV()

Declaration

```
[ExcelCommand(Description = "Export XMLMapped List Object to CSV", ExplicitRegistration = true)]
public static void ExportXmlMappedListToCSV()
```

ImportCSVToXMLMappedList()

Declaration

```
[ExcelCommand(Description = "Import CSV to XMLMapped List", ExplicitRegistration = true)]
public static void ImportCSVToXMLMappedList()
```

XSDUserInput()

Declaration

```
public static string XSDUserInput()
```

Returns

TYPE	DESCRIPTION
System.String	

Extension Methods

JavaLikeExtensions.toString(Object)

Namespace WDataSci.HDF5

The WDataSci.HDF5 namespace encompasses classes and methods which the addin uses to create and read HDF5 files.

Classes

Cmds

Class Cmds

Inheritance

System.Object

Cmds

Inherited Members

System.Object.ToString()

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

Namespace: WDataSci.HDF5

Assembly: JniPMML-Cs.dll

Syntax

```
public class Cmds
```

Methods

ExportXmlMappedListToHDF5()

Declaration

```
[ExcelCommand(Description = "Export XmlMapped List Object to HDF5", ExplicitRegistration = true)]
public static void ExportXmlMappedListToHDF5()
```

ImportHDF5CompoundDS()

Declaration

```
[ExcelCommand(Description = "Import Compound DataSet from HDf5", ExplicitRegistration = true)]
public static void ImportHDF5CompoundDS()
```

Extension Methods

JavaLikeExtensions.toString(Object)

Namespace WDataSci.JniPMML

The WDataSci.JniPMML namespace encompasses the C# classes which wrap the calls to the JniPMML jar.

Classes

AddIn

The AddIn:IExcelAddIn is partialled in AddInCmds.JniPMML.cs, to add calls to the JniPMML wrapper.

RibbonController

WDataSci Excel ribbon creation.

Class AddIn

The AddIn:IExcelAddIn is partialled in AddInCmds.JniPMML.cs, to add calls to the JniPMML wrapper.

Inheritance

System.Object
AddIn

Implements

ExcelDna.Integration.IExcelAddIn

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
public class AddIn : IExcelAddIn
```

Fields

__JniPMML

Declaration

```
public static JniPMML __JniPMML
```

Field Value

TYPE	DESCRIPTION
JniPMML	

Methods

AutoClose()

Declaration

```
public void AutoClose()
```

AutoOpen()

Declaration

```
public void AutoOpen()
```

CleanAsNMToken_ViaJni(String)

Declaration

```
[ExcelFunction(Name = "CleanAsNMToken_ViaJni", Category = "WDS.JniPMML", Description = "Returns a NMToken of
the input string with invalid characters removed", IsVolatile = false, ExplicitRegistration = true)]
public static string CleanAsNMToken_ViaJni([ExcelArgument(Name = "InputString", Description = "A general
string")] string aInputString)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aInputString	

Returns

TYPE	DESCRIPTION
System.String	

CleanStringWithRegex_ViaJni(String, String, String, Object, Object, Object)

Declaration

```
[ExcelFunction(Name = "CleanStringWithRegex_ViaJni", Category = "WDS.JniPMML", Description = "Returns a
NMToken of the input string with invalid characters removed", IsVolatile = false, ExplicitRegistration =
true)]
public static string CleanStringWithRegex_ViaJni([ExcelArgument(Name = "InputString")] string aInputString,
[ExcelArgument(Name = "RegexToFind")] string aRegexString, [ExcelArgument(Name = "RegexToReplaceWith")] string
aReplaceWithString, [ExcelArgument(Name = "JVM_class")] object _aClassName, [ExcelArgument(Name =
"JVM_method")] object _aMethodName, [ExcelArgument(Name = "JVM_signature")] object _aSignatureString)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aInputString	
System.String	aRegexString	
System.String	aReplaceWithString	
System.Object	_aClassName	
System.Object	_aMethodName	
System.Object	_aSignatureString	

Returns

TYPE	DESCRIPTION
System.String	

Dispose()

Declaration

```
public void Dispose()
```

FetchFileAsString_ViaJni(String)

Declaration

```
[ExcelFunction(Name = "FetchFileAsString_ViaJni", Category = "WDS.JniPMML", Description = "Pulls the contents
of a file and returns as one string.", ExplicitRegistration = true)]
public static string FetchFileAsString_ViaJni([ExcelArgument(Name = "FileName")] string arg1)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg1	

Returns

TYPE	DESCRIPTION
System.String	

Finalize()

Declaration

```
protected void Finalize()
```

HandleMajorFrom(Object)

Declaration

```
[ExcelFunction(Name = "JniPMML_HandleMajorFrom", Category = "WDS.JniPMML", Description = "Returns HandleMajor from \"Major.Minor\", or the string that has been parsed and is cached in the JVM.  Bad input (such as ExcelError) will return 0 which is the default first handle, returns -2 if not resolvable.", IsVolatile = false, ExplicitRegistration = true)]
public static int HandleMajorFrom([ExcelArgument(Name = "HandleOrTag", Description = "Use a \"Major.Minor\" Handle output from CreateHandle to chain calcuation dependency")] object arg0)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	arg0	

Returns

TYPE	DESCRIPTION
System.Int32	

JniPMML_Cmd()

Declaration

```
[ExcelCommand(Description = "Runs the JniPMML command based on the arguments in a cell.", ExplicitRegistration = true)]
public static void JniPMML_Cmd()
```

JniPMML_Cmd_Prep()

Declaration

```
[ExcelCommand(Description = "Create Worksheet to demo or call JniPMML java cmd line", ExplicitRegistration = true)]
public static void JniPMML_Cmd_Prep()
```

JniPMML_CreateHandle(String, String, String)

Declaration

```
[ExcelFunction(Name = "JniPMML_CreateHandle", Category = "WDS.JniPMML", Description = "Creates a handle and initializes with a PMML File, input as either a FileName or a String.", IsVolatile = false, ExplicitRegistration = true)]
public static object JniPMML_CreateHandle([ExcelArgument(Name = "HandleType", Description = "[JniPMML|others....]")] string arg0, [ExcelArgument(Name = "Tag", Description = "String Handle")] string arg1, [ExcelArgument(Name = "FileNameOrString", Description = "Path to external PMML filename or entire file as a string (an entire string is determined by the usual XML starting characters)")] string arg3)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	arg0	
System.String	arg1	
System.String	arg3	

Returns

TYPE	DESCRIPTION
System.Object	

JniPMML_Eval(Object, Int32, Object[,])

Declaration

```
[ExcelFunction(Name = "JniPMML_Eval", Category = "WDS.JniPMML", Description = "Calls JniPMML.Eval based a previously set Header", IsThreadSafe = true, IsMacroType = true, IsVolatile = false, ExplicitRegistration = true)]
public static object[, ] JniPMML_Eval([ExcelArgument(Name = "HandleOrTag", Description = "Use the \"Major.Minor\" Handle output for the matching Model to maintain dependency")] object arg0, [ExcelArgument(Name = "InputDataIncludesHeader", Description = "0/1 indicates both to skip eval on first row and whether to include output header")] int bInputDataHasHeaderRow, [ExcelArgument(Name = "InputData", Description = "Select Contiguous ListObject Rows, include header if needed for alignment")] object[, ] data)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	arg0	
System.Int32	bInputDataHasHeaderRow	
System.Object[,]	data	

Returns

TYPE	DESCRIPTION
System.Object[,]	

JniPMML_Eval_CacheHeaders_guts(Int32, XmlMap, String, Int32)

Declaration

```
protected static object JniPMML_Eval_CacheHeaders_guts(int h, XmlMap xm, string xmschema, int nOutputStringMaxLength)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	h	
Microsoft.Office.Interop.Excel.XmlMap	xm	
System.String	xmschema	
System.Int32	nOutputStringMaxLength	

Returns

TYPE	DESCRIPTION
System.Object	

JniPMML_Eval_CacheHeaders_Volatile(Object, Object, Object)

Declaration

```
[ExcelFunction(Name = "JniPMML_Eval_CacheHeaders_Volatile", Category = "WDS.JniPMML", Description = "Caches just the input and output headers for Eval, on both the C# and Java sides, subsequent calls to WDS.JniPMML__Headerless can follow.  Set the headerless calls to depend on this \"Major.Minor\" output.", IsThreadSafe = true, IsMacroType = true, IsVolatile = false, ExplicitRegistration = true)]
public static object JniPMML_Eval_CacheHeaders_Volatile([ExcelArgument(Name = "HandleOrTag", Description = "Use a \"Major.Minor\" Handle output from CreateHandle to chain calcuation dependency")] object arg0,
[ExcelArgument(Name = "XmlMappedList", Description = "Point to a reference cell or range of the XmlMap'd list (one that does not change with data, such as the header)", AllowReference = true)] object arg,
[ExcelArgument(Name = "nOutputStringMaxLength", Description = "An alternate maximum string length for output fields, defaults to 64", AllowReference = false)] object _nOutputStringMaxLength)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	arg0	
System.Object	arg	
System.Object	_nOutputStringMaxLength	

Returns

TYPE	DESCRIPTION
System.Object	

JniPMML_Eval_OutputColumnHeadings(Object)

Declaration

```
[ExcelFunction(Name = "JniPMML_Eval_OutputColumnHeadings", Category = "WDS.JniPMML", Description = "Returns just the field names associated with a cached model and headers", IsThreadSafe = true, IsMacroType = true, IsVolatile = false, ExplicitRegistration = true)]
public static object[, ] JniPMML_Eval_OutputColumnHeadings([ExcelArgument(Name = "HandleOrTag", Description = "Use the \"Major.Minor\" Handle output for the matching Model to maintain dependency")] object arg0)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	arg0	

Returns

TYPE	DESCRIPTION
System.Object[,]	

JniPMML_Eval_Volatile(Int32, String, Int32, Object, Object)

Declaration

```
[ExcelFunction(Name = "JniPMML_Eval_Volatile", Category = "WDS.JniPMML", Description = "A volatile self contained call to the JniPMML evaluator.  The first argument is just to turn it off/on to kill the drag on calculation time.", IsThreadSafe = true, IsMacroType = true, IsVolatile = false, ExplicitRegistration = true)]
public static object[, ] JniPMML_Eval_Volatile([ExcelArgument(Name = "bToCalcSwitch", Description = "0/1")]
int arg0, [ExcelArgument(Name = "PMMLInput", Description = "Path to external PMML filename or entire file as a string (an entire string is determined by the usual XML starting characters)")] string PMMLFile,
[ExcelArgument(Name = "bInputDataHasHeaderRow", Description = "0/1, If input includes header row, output will include header row.")] int bInputDataHasHeaderRow, [ExcelArgument(Name = "InputTableReference", Description = "An XMLMap'd Table, column names are taken from the XMLMap", AllowReference = true)] object arg,
[ExcelArgument(Name = "nOutputStringMaxLength", Description = "An alternate maximum string length for output fields, defaults to 64", AllowReference = false)] object _nOutputStringMaxLength)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	arg0	
System.String	PMMLFile	
System.Int32	bInputDataHasHeaderRow	
System.Object	arg	
System.Object	_nOutputStringMaxLength	

Returns

TYPE	DESCRIPTION
System.Object[,]	

JniPMML_Eval_XmlMappedList()

Declaration

```
[ExcelCommand(Description = "Evaluate cached PMML on XmlMapped List", ExplicitRegistration = true)]
public static void JniPMML_Eval_XmlMappedList()
```

JniPMML_Handle(Object, Object)

Declaration

```
[ExcelFunction(Name = "JniPMML_Handle", Category = "WDS.JniPMML", ExplicitRegistration = true, HelpTopic =
"Help Topic to be filled in", IsVolatile = false)]
public static string JniPMML_Handle([ExcelArgument(Name = "HandleOrTag", Description = "Use a \"Major.Minor\"
Handle output from CreateHandle to chain calcuation dependency")] object arg0, [ExcelArgument(Name =
"AdditionalRangeDependencies")] object arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	arg0	
System.Object	arg	

Returns

TYPE	DESCRIPTION
System.String	

JniPMML_Handle_LastUsed(Object)

Declaration

```
[ExcelFunction(Name = "JniPMML_Handle_LastUsed", Description = "Last handle used in Java JniPMML", Category =
"WDS.JniPMML", ExplicitRegistration = true, HelpTopic = "Help Topic to be filled in", IsVolatile = false)]
public static string JniPMML_Handle_LastUsed([ExcelArgument(Name = "AdditionalRangeDependencies")] object arg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	arg	

Returns

TYPE	DESCRIPTION
System.String	

JniPMML_HandleMajor(Object, Object)

Declaration

```
[ExcelFunction(Name = "JniPMML_HandleMajor", Description = "Either returns the HandleMajor of the last or
working index, or returns a new handle for the value (filling in all empty values lower)", Category =
"WDS.JniPMML", ExplicitRegistration = true, HelpTopic = "Help Topic to be filled in")]
public static int JniPMML_HandleMajor([ExcelArgument(Name = "HandleOrTag", Description = "Use a
\"Major.Minor\" Handle output from CreateHandle to chain calcuation dependency")] object arg0,
[ExcelArgument(Name = "AdditionalRangeDependencies", Description = "Provides an Excel calculation dependency,
otherwise not used")] object deps)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	arg0	
System.Object	deps	

Returns

TYPE	DESCRIPTION
System.Int32	

JniPMML_isValidHandle(Object)

Declaration

```
[ExcelFunction(Name = "JniPMML_isValidHandle", Category = "WDS.JniPMML", ExplicitRegistration = true,
HelpTopic = "Help Topic to be filled in")]
public static bool JniPMML_isValidHandle([ExcelArgument(Name = "HandleOrTag", Description = "Use a
\"Major.Minor\" Handle output from CreateHandle to chain calcuation dependency")] object arg0)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	arg0	

Returns

TYPE	DESCRIPTION
System.Boolean	

JniPMML_LoadedFileName(Object)

Declaration

```
[ExcelFunction(Name = "JniPMML_LoadedFileName", Category = "WDS.JniPMML", Description = "Returns the string
that has been parsed and is cached in the JVM", IsVolatile = false, ExplicitRegistration = true)]
public static string JniPMML_LoadedFileName([ExcelArgument(Name = "HandleOrTag", Description = "Use a
\"Major.Minor\" Handle output from CreateHandle to chain calcuation dependency")] object arg0)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	arg0	

Returns

TYPE	DESCRIPTION
System.String	

JniPMML_LoadedString(Object)

Declaration

```
[ExcelFunction(Name = "JniPMML_LoadedString", Category = "WDS.JniPMML", Description = "Returns the string that
has been parsed and is cached in the JVM", IsVolatile = false, ExplicitRegistration = true)]
public static string JniPMML_LoadedString([ExcelArgument(Name = "HandleOrTag", Description = "Use a
\"Major.Minor\" Handle output from CreateHandle to chain calcuation dependency")] object arg0)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	arg0	

Returns

TYPE	DESCRIPTION
System.String	

JniPMML_LoadFromFile(Object, String)

Declaration

```
[ExcelFunction(Name = "JniPMML_LoadFromFile", Category = "WDS.JniPMML", Description = "Creates a handle and
parses PMML model from an external file, caching in JVM", IsVolatile = false, ExplicitRegistration = true)]
public static string JniPMML_LoadFromFile([ExcelArgument(Name = "HandleOrTag", Description = "Use a
\"Major.Minor\" Handle output from CreateHandle to chain calcuation dependency")] object arg0,
[ExcelArgument(Name = "FileName")] string arg1)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	arg0	
System.String	arg1	

Returns

TYPE	DESCRIPTION
System.String	

JniPMML_LoadFromString(Object, String)

Declaration

```
[ExcelFunction(Name = "JniPMML_LoadFromString", Category = "WDS.JniPMML", Description = "Creates a handle and
parses PMML model from string input, caching in JVM", IsVolatile = false, ExplicitRegistration = true)]
public static string JniPMML_LoadFromString([ExcelArgument(Name = "HandleOrTag", Description = "Use a
\"Major.Minor\" Handle output from CreateHandle to chain calcuation dependency")] object arg0,
[ExcelArgument(Name = "FileName")] string arg1)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	arg0	
System.String	arg1	

Returns

TYPE	DESCRIPTION
System.String	

JniPMML_XmlMap_Helper()

Declaration

```
[ExcelCommand(Description = "Add XmlMap to List Object", ExplicitRegistration = true)]
public static void JniPMML_XmlMap_Helper()
```

PathElementOf(String, String)

Declaration

```
public string PathElementOf(string path, string loc)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	path	
System.String	loc	

Returns

TYPE	DESCRIPTION
System.String	

RegisterFunctions()

Declaration

```
public void RegisterFunctions()
```

UpdateHelpTopic(ExcelFunctionRegistration)

Declaration

```
public ExcelFunctionRegistration UpdateHelpTopic(ExcelFunctionRegistration funcReg)
```

Parameters

TYPE	NAME	DESCRIPTION
ExcelDna.Registration.ExcelFunctionRegistration	funcReg	

Returns

TYPE	DESCRIPTION
ExcelDna.Registration.ExcelFunctionRegistration	

UpdateMajorMinor(String, Object[])

Declaration

```
public static JniPMMLItem UpdateMajorMinor(string aEvaluatorType, object[] args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	aEvaluatorType	

TYPE	NAME	DESCRIPTION
System.Object[]	args	

Returns

TYPE	DESCRIPTION
JniPMMLItem	

Implements

ExcelDna.Integration.IExcelAddIn

Extension Methods

JavaLikeExtensions.toString(Object)

Class RibbonController

WDataSci Excel ribbon creation.

Inheritance

System.Object
ExcelDna.Integration.ExcelComAddIn
ExcelDna.Integration.CustomUI.ExcelRibbon
RibbonController

Inherited Members

ExcelDna.Integration.CustomUI.ExcelRibbon.NamespaceCustomUI2010
ExcelDna.Integration.CustomUI.ExcelRibbon.NamespaceCustomUI2007
ExcelDna.Integration.CustomUI.ExcelRibbon.LoadImage(System.String)
ExcelDna.Integration.CustomUI.ExcelRibbon.RunTagMacro(ExcelDna.Integration.CustomUI.IRibbonControl)
ExcelDna.Integration.ExcelComAddIn.OnConnection(System.Object, ExcelDna.Integration.Extensibility.ext_ConnectMode, System.Object, System.Array)
ExcelDna.Integration.ExcelComAddIn.OnDisconnection(ExcelDna.Integration.Extensibility.ext_DisconnectMode, System.Array)
ExcelDna.Integration.ExcelComAddIn.OnAddInsUpdate(System.Array)
ExcelDna.Integration.ExcelComAddIn.OnStartupComplete(System.Array)
ExcelDna.Integration.ExcelComAddIn.OnBeginShutdown(System.Array)
ExcelDna.Integration.ExcelComAddIn.ProgId
System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: WDataSci.JniPMML

Assembly: JniPMML-Cs.dll

Syntax

```
[ComVisible(true)]
public class RibbonController : ExcelRibbon, IDTExtensibility2, IRibbonExtensibility
```

Methods

GetCustomUI(String)

Declaration

```
public override string GetCustomUI(string uiName)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	uiName	

Returns

TYPE	DESCRIPTION
System.String	

Overrides

ExcelDna.Integration.CustomUI.ExcelRibbon.GetCustomUI(System.String)

ShowAboutForm()

Declaration

```
[ExcelCommand(Description = "About", ExplicitRegistration = true)]
public static void ShowAboutForm()
```

VBACheck()

Declaration

```
[ExcelCommand(Description = "VBA Module Check", ExplicitRegistration = true)]
public static void VBACheck()
```

VBACheckRefresh()

Declaration

```
[ExcelCommand(Description = "VBA Module Check Refresh", ExplicitRegistration = true)]
public static void VBACheckRefresh()
```

VBACheckRemove()

Declaration

```
[ExcelCommand(Description = "Remove VBA Module Check", ExplicitRegistration = true)]
public static void VBACheckRemove()
```

VBAComponentAdd_WDSCore()

Declaration

```
[ExcelCommand(Description = "Add VBA Module: WDSCore", ExplicitRegistration = true)]
public static void VBAComponentAdd_WDSCore()
```

VBAComponentAdd_WDSJniPMML()

Declaration

```
[ExcelCommand(Description = "Add VBA Module: WDSJniPMML", ExplicitRegistration = true)]
public static void VBAComponentAdd_WDSJniPMML()
```

VBAComponentRemove_WDSCore()

Declaration

```
[ExcelCommand(Description = "Remove VBA Module: WDSCore", ExplicitRegistration = true)]
public static void VBAComponentRemove_WDSCore()
```

VBAComponentRemove_WDSJniPMML()

Declaration

```
[ExcelCommand(Description = "Remove VBA Module: WDSJniPMML", ExplicitRegistration = true)]
public static void VBAComponentRemove_WDSJniPMML()
```

Extension Methods

JavaLikeExtensions.toString(Object)

WDS-Cs

General utilitities that independent of JniPMML code. To simplify assemblies and jars, this code is included in the larger projects, but is also compiled as a stand alone.

The Java style com.WDataSci namespaces are specifically for C# code which mirrors the Java modules.

The C# style namespaces are not specifically mirrored in the Java code.

Class WDSAddIn

Inheritance

System.Object
WDSAddIn

Implements

ExcelDna.Integration.IExcelAddIn

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: com.WDataSci.WDS

Assembly: WDS-Cs.dll

Syntax

```
public class WDSAddIn : IExcelAddIn
```

Methods

AutoClose()

Declaration

```
public void AutoClose()
```

AutoOpen()

Declaration

```
public void AutoOpen()
```

RegisterFunctions()

Declaration

```
public void RegisterFunctions()
```

UpdateHelpTopic(ExcelFunctionRegistration)

Declaration

```
public ExcelFunctionRegistration UpdateHelpTopic(ExcelFunctionRegistration funcReg)
```

Parameters

TYPE	NAME	DESCRIPTION
ExcelDna.Registration.ExcelFunctionRegistration	funcReg	

Returns

TYPE	DESCRIPTION
ExcelDna.Registration.ExcelFunctionRegistration	

Implements

ExcelDna.Integration.IExcelAddIn

Extension Methods

JavaLikeExtensions.toString(Object)