table.ok - Function

Package	**utility**
Module	**table**
Tool	**table**

Is the table tool ok?

Synopsis
ok()

Description
Perform a number of sanity checks and return T if ok. Failure (returning F) is a sign of a bug.

Returns
Bool

1.7.2 tablecolumn - Tool

Package	**utility**
Module	**table**

Allows access to a column in a table

Description
This is a convenience tool for access to a table column. It offers the same functionality as the column access functions in the table tool (e.g. getcell). The advantage is that the column name does not need to be specified each time.

Example

```
include "table.g"
vis:=table("3C273XC1.MS", readonly=T);
vistimc:=tablecolumn(vis,"TIME");
vistimc.getcol();
vistimc.done();
```

Constructors

tablecolumn	Construct table column tool

Functions

arraytype	return the column array type
close	close the tablecolumn tool
datatype	return the column data type
done	end the tablecolumn tool
fieldnames	get the names of fields in a keyword in a column
getcell	get a specific cell
getcellslice	get a slice from a specific cell
getcol	get a specific column
getcolslice	get a slice from a specific column
getdesc	get the description of a specific column
getkeyword	get value of specific column keyword
getkeywords	get values of all keywords for a column
getshapestring	get shape of arrays in a specific column
getvarcol	get a specific column (for variable arrays)
iscelldefined	test if a specific cell contains a value
isscalar	is the specified column scalar?
isvar	tell if column contains variable shaped arrays
keywordnames	get the names of all keywords in a column
makeindex	make a table index tool
makeiter	make a table iterator tool
name	return the column name
nrows	return number of rows
putcell	put a specific cell
putcellslice	put a slice into a specific cell
putcol	put a specific column
putcolslice	put a slice into a specific column
putkeyword	put a specific keyword for a column
putkeywords	put multiple keywords for a column
putvarcol	put a specific column (for variable arrays)
removekeyword	remove a specific keyword for a column
table	return the table tool

tablecolumn - Constructor

Package	**utility**
Module	**table**
Tool	**tablecolumn**

Construct table column tool

Synopsis
tablecolumn(tab, column)

Arguments

tab	in	table to be accessed Allowed:table tool
column	in	column name Allowed:String

---

tablecolumn.table - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

return the table tool

Synopsis
table()

Description
Return the table tool used with this tablecolumn tool.

Arguments

Returns
table tool

---

tablecolumn.name - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

return the column name

Synopsis
name()

Description
Return the name of the column used with this tablecolumn tool.

Arguments

Returns
String

---

tablecolumn.isscalar - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

is the specified column scalar?

Synopsis
isscalar()

Arguments

Returns
Bool

---

tablecolumn.isvar - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

tell if column contains variable shaped arrays

Synopsis
isvar()

Arguments

Returns
Bool

---

tablecolumn.datatype - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

return the column data type

Synopsis
datatype()

Arguments

Returns
String

---

tablecolumn.arraytype - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

return the column array type

Synopsis
arraytype()

Arguments

Returns
String

---

tablecolumn.nrows - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

return number of rows

Synopsis
nrows()

Returns
Int

---

tablecolumn.getdesc - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

get the description of a specific column

Synopsis
getdesc()

Arguments

Returns
Column description record

---

tablecolumn.iscelldefined - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

test if a specific cell contains a value

Synopsis
iscelldefined(rownr)

Arguments

rownr	in	Row number, starting at 1 Allowed:Int

Returns
Bool

---

tablecolumn.getcell - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

get a specific cell

Synopsis
getcell(rownr)

Arguments

rownr	in	Row number, starting at 1 Allowed:Int

Returns
Value

---

tablecolumn.getcellslice - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

get a slice from a specific cell

Synopsis
getcellslice(rownr, blc, trc, incr)

Arguments

rownr	in	Row number, starting at 1 Allowed:Int
blc	in	Bottom left corner (e.g. [1,1,1] is start of 3D array) Allowed:Vector of Ints
trc	in	Top right corner Allowed:Vector of Ints
incr	in	Stride (defaults to 1 for all axes) Allowed:Vector of Ints

Returns
Array

---

tablecolumn.getcol - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

get a specific column

Synopsis
getcol(startrow, nrow, rowincr)

Arguments

startrow	in	First row to read (default 1) Allowed:Int
nrow	in	Number of rows to read (default -1 means till the end) Allowed:Int
rowincr	in	Increment in rows to read (default 1) Allowed:Int

Returns
Array

---

tablecolumn.getvarcol - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

get a specific column (for variable arrays)

Synopsis
getvarcol(startrow, nrow, rowincr)

Arguments

startrow	in	First row to read (default 1) Allowed:Int
nrow	in	Number of rows to read (default -1 means till the end) Allowed:Int
rowincr	in	Increment in rows to read (default 1) Allowed:Int

Returns
Record

---

tablecolumn.getcolslice - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

get a slice from a specific column

Synopsis
getcolslice(blc, trc, incr, startrow, nrow, rowincr)

Arguments

blc	in	Bottom left corner (e.g. [1,1,1] is start of 3D array) Allowed:Vector of Ints
trc	in	Top right corner Allowed:Vector of Ints
incr	in	Stride (defaults to 1 for all axes) Allowed:Vector of Ints
startrow	in	First row to read (default 1) Allowed:Int
nrow	in	Number of rows to read (default -1 means till the end) Allowed:Int
rowincr	in	Increment in rows to read (default 1) Allowed:Int

Returns
Array

---

tablecolumn.putcell - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

put a specific cell

Synopsis
putcell(rownr, value)

Arguments

rownr	in	Row number(s) (1-relative) Allowed:Vector of Ints
value	in	Value

Returns
Bool

---

tablecolumn.putcellslice - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

put a slice into a specific cell

Synopsis
putcellslice(rownr, value, blc, trc, incr)

Arguments

rownr	in	Row number, starting at 1 Allowed:Int
value	in	Value
blc	in	Bottom left corner (e.g. [1,1,1] is start of 3D array) Allowed:Vector of Ints
trc	in	Top right corner Allowed:Vector of Ints
incr	in	Stride (defaults to 1 for all axes) Allowed:Vector of Ints

Returns
Bool

---

tablecolumn.putcol - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

put a specific column

Synopsis
putcol(value, startrow, nrow, rowincr)

Arguments

value	in	Array
startrow	in	First row to put (default 1) Allowed:Int
nrow	in	Number of rows to put (default -1 means till the end) Allowed:Int
rowincr	in	Increment in rows to put (default 1) Allowed:Int

Returns
Bool

---

tablecolumn.putvarcol - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

put a specific column (for variable arrays)

Synopsis
putvarcol(value, startrow, nrow, rowincr)

Arguments

value	in	Record with values
startrow	in	First row to put (default 1) Allowed:Int
nrow	in	Number of rows to put (default -1 means till the end) Allowed:Int
rowincr	in	Increment in rows to put (default 1) Allowed:Int

Returns

Bool

---

tablecolumn.putcolslice - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

put a slice into a specific column

Synopsis
putcolslice(value, blc, trc, incr, startrow, nrow, rowincr)

Arguments

value	in	Array
blc	in	Bottom left corner (e.g. [1,1,1] is start of 3D array) Allowed:Vector of Ints
trc	in	Top right corner Allowed:Vector of Ints
incr	in	Stride (defaults to 1 for all axes) Allowed:Vector of Ints
startrow	in	First row to put (default 1) Allowed:Int
nrow	in	Number of rows to put (default -1 means till the end) Allowed:Int
rowincr	in	Increment in rows to put (default 1) Allowed:Int

Returns
Bool

---

tablecolumn.getshapestring - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

get shape of arrays in a specific column

Synopsis
getshapestring(startrow, nrow, rowincr)

Arguments

startrow	in	First row to read (default 1) Allowed:Int
nrow	in	Number of rows to read (default -1 means till the end) Allowed:Int
rowincr	in	Increment in rows to read (default 1) Allowed:Int

Returns
Array of Strings

---

tablecolumn.getkeyword - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

get value of specific column keyword

Synopsis
getkeyword(keyword)

Arguments

keyword	in	Name or seqnr of keyword Allowed:String—Int

Returns
Value

---

tablecolumn.getkeywords - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

get values of all keywords for a column

Synopsis
getkeywords()

Arguments

Returns
Record

---

tablecolumn.putkeyword - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

put a specific keyword for a column

Synopsis
putkeyword(keyword, value)

Arguments

keyword	in	Name or seqnr of keyword Allowed:String—Int
value	in	Value of keyword

Returns
Bool

---

tablecolumn.putkeywords - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

put multiple keywords for a column

Synopsis
putkeywords(value)

Arguments

value	in	Record of keyword=value pairs

Returns
Bool

---

tablecolumn.removekeyword - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

remove a specific keyword for a column

Synopsis
removekeyword(keyword)

Arguments

keyword	in	Name or seqnr of keyword Allowed:String—Int

Returns
Bool

---

tablecolumn.keywordnames - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

get the names of all keywords in a column

Synopsis
keywordnames()

Arguments

Returns
Vector of Strings

---

tablecolumn.fieldnames - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

get the names of fields in a keyword in a column

Synopsis
fieldnames(keyword)

Arguments

keyword	keyword name Allowed:String Default: ”

Returns
Vector of Strings

---

tablecolumn.makeiter - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

make a table iterator tool

Synopsis
makeiter(order, sort)

Description
Create a tableiterator tool based on this column.

Arguments

order	in	Order when sorting the table Allowed:String Default: ”
sort	in	Sort the table first? Allowed:Bool,String Default: T

Returns
tableiterator tool

---

tablecolumn.makeindex - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

make a table index tool

Synopsis
makeindex(sort)

Description
Create a tableindex tool based on this column.

Arguments

sort	in	Sort the table first? Allowed:Bool,String Default: T

Returns
tableindex tool

---

tablecolumn.close - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

close the tablecolumn tool

Synopsis
close()

Returns
Bool

---

tablecolumn.done - Function

Package	**utility**
Module	**table**
Tool	**tablecolumn**

end the tablecolumn tool

Synopsis
done()

Description
End the tablecolumn tool and free up all memory. It cannot be reused anymore.

Returns
Bool
