Module **table**
Tool **tablerow**

End the tablerow tool

Synopsis
done()

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

1.7.4 tableiterator - Tool

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

Allows iteration through a table tool

Description
A tableiterator allows iteration through a table based on the contents of one or more columns. Each step in the iteration process forms a subset of the table for which the specified columns have the same value.

Example

```
vis:=table("3C273XC1.MS");
vis.summary();
visbyants:=tableiterator(vis, "ANTENNA1 ANTENNA2");
visbyants.reset();
while(visbyants.next()) {
  nrows:=visbyants.table().nrows();
  print "Antenna1 ", visbyants.table().getcell("ANTENNA1", 1),
        " Antenna2 ", visbyants.table().getcell("ANTENNA2", 1),
        ": ", nrows, "samples";
}
```

This example iterates through the table on ANTENNA1 and ANTENNA2. Thus each call to `next` forms a (reference) table containing the rows of a single

baseline in the main table. The `table` function can be used to get access to the current subset.
Note that a `next` has to be done to get the first subset.

Constructors

tableiterator	Construct a tableiterator

Functions

close	close the tableiterator tool
done	end the tableiterator tool
next	Advance to the next table subset
reset	Reset iteration to the beginning
table	Return the current table subset
terminate	Terminate the iteration and clean up memory

tableiterator - Constructor

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

Construct a tableiterator

Synopsis
tableiterator(tab, columns, order, sort)

Description
Construct an iterator for the given table. Each step in the iteration returns a subtable for which the values of the given columns are equal. By default the table will be sorted before the iteration is started to achieve that all equal values are grouped together.
The optional order argument can be given to define how the table has to be sorted. 'a' is ascending, 'd' is descending, other is don't care (which comes to ascending). Instead of a single letter any word can be used starting with a or d (e.g. asc).
The optional sort argument can be given to define if the table has to be sorted. F means no sorting. The default T means sorting. The default sort algorithm used is heapsort. By giving the sort argument a string value, it can also be used to specify another sort algorithm. 'n' is no sort, 'q' is quicksort, 'i' is insertion sort, other is heapsort. A full word can also be used.

Arguments

tab	in	table to be iterated Allowed:table tool
columns	in	Array of column names Allowed:Vector of Strings
order	in	Order when sorting the table Allowed:String Default: ”
sort	in	Sort the table first? Allowed:Bool,String Default: T

---

tableiterator.table - Function

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

Return the current table subset

Synopsis
table()

Description
Return a `table` tool for the current subset.
Note that `next` has to be called to form the first subset.

Returns
table tool—Bool

---

tableiterator.reset - Function

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

Reset iteration to the beginning

Synopsis
reset()

Description
It resets the iteration to the beginning of the table. Note that `next` has to be called to get the first subset.

Returns
Bool

---

tableiterator.next - Function

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

Advance to the next table subset

Synopsis
next()

Description
Form the next subset of the table. It returns a false status if no more subsets are available The subset can be accessed using the `table` function.
Note that `next` has to be called to form the first subset.

Returns
Bool

---

tableiterator.terminate - Function

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

Terminate the iteration and clean up memory

Synopsis
terminate()

Description
This function has to be called if the iteration is ended prematurely. Otherwise tables are left open. If the iteration is done until the end, `terminate` does not need to be called, but it does not harm to call it.
Hereafter it is still possible to restart the iteration using the `reset` function.

Returns
Bool

---

tableiterator.close - Function

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

close the tableiterator tool

Synopsis
close()

Description
A synonym for terminate.

Returns
Bool

---

tableiterator.done - Function

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

end the tableiterator tool