

Sorter tool

\$sorter is used to sort a collection of report templates.

\$sorter.sort(Collection, fieldName)

The sort function for report templates. The context name of this class is "sorter". Use \$sorter to access public functions of this class through templates.

	Name	Type	Description
Parameter(s)	collection	java.util.Collection	A collection to be sorted
	fieldname	java.lang.String	A fieldName to be sorted and the sort direction.
Return	-	java.util.Collection	A sorted collection.

For example:

```
#foreach ($rel in $sorter.sort($package, "name"))
$rel.name
#end
```

- \$package is the collection to be sorted.
- "name:desc" is separated by ":" in two parts:
 - The first part is to identify fieldName to be sorted.
 - The second part is the option to identify the sorting direction. Sometimes, the direction is not identified. It is ascending by default.

\$sorter.sort(Collection)

This is the sort function for report templates. The context name of this class is "sorter". Use \$sorter to access public functions of this class through the templates.

	Name	Type	Description
Parameter(s)	collection	java.util.Collection	A collection to be sorted.
Return	-	java.util.Collection	A sorted collection.

For example:

```
#foreach ($rel in $sorter.sort($package))
$rel.name
#end
```

- \$package is a collection to be sorted.

 The collection will be sorted by name, in ascending order, by default.

\$sorter.sortByFirstNumber(Collection, fieldName)

The sortByFirstNumber function is for report templates. The context name of this class is "sorter". Use \$sorter to access public functions of this class through templates.

	Name	Type	Description
Parameter(s)	collection	java.util.Collection	A collection to be sorted.

	fieldName	java.lang.String	A fieldName to be sorted and the sorting direction.
Return	-	java.util.Collection	A sorted collection.

For example:

```
#foreach ($rel in $sorter.sortByFirstNumber($package, "name:desc"))
$rel.name
#end
```

- \$package is a collection to be sorted by **FirstNumber**.
- "name:desc" is separated by ":" in two parts:
 - The first part is to identify fieldName to be sorted.
 - The second part is the option to identify the sorting direction. Sometimes, the direction is not identified. It is ascending by default.

\$sorter.sortByFirstNumber(Collection)

The sortByFirstNumber function for report templates. The context name of this class is "sorter". Use \$sorter to access public functions of this class through templates.

	Name	Type	Description
Parameter(s)	collection	java.util.Collection	A collection to be sorted.
Return	-	java.util.Collection	A sorted collection.

For example:

```
#foreach ($rel in $sorter.sortByFirstNumber($package))
$rel.name
#end
```

- \$package is a collection to be sorted by FirstNumber.

\$sorter.sortByLocale(Collection, String)

This is a function to sort for report templates. The context name of this class is "sorter". Use \$sorter to access public functions of this class through templates. To sort a given collection by a particular language, identify the country code to specify the language.

	Name	Type	Description
Parameter(s)	collection	java.util.Collection	A collection to be sorted.
	countryCode	java.lang.String	The country code to specify a language to perform sorting.
Return	-	java.util.Collection	A sorted collection.

For example:

```
#foreach ($p in $sorter.sortByLocale($package, "DE"))
$p.name
#end
```

- \$package is a collection to be sorted by country code.
- "DE" is the country code for GERMANY (ISO country code).



This method sorts a collection by "name" attribute of each element by default.

\$sorter.sortByLocale(Collection, String, String)

This is a function to sort for report templates. The context name of this class is "sorter". Use \$sorter to access public functions of this class through templates. To sort a given collection by a particular language, specify the language by identifying the country code and field name.

	Name	Type	Description
Parameter(s)	collection	java.util.Collection	A collection to be sorted.
	fieldName	java.lang.String	A fieldName to be sorted.
	countryCode	java.lang.String	The country code to specify a language to perform sorting.
Return	-	java.util.Collection	A sorted collection.

For example:

```
#foreach ($p in $sorter.sortByLocale($package, "name", "DE"))
  $p.name
#end
```

- \$package is the collection to be sorted.
- "name" is the field name to be sorted.
- "DE" is the country code for GERMANY (ISO country code).

\$sorter.humanSort(collection, fieldName)

This is a special function to sort text in a human-like order. It splits text into numeric and non-numeric chunks and sorts them in numerical order. For example, "foo10" is ordered after "foo2".

	Name	Type	Description
Parameter(s)	collection	java.util.Collection	A collection to be sorted.
	fieldName	java.lang.String	A fieldName to be sorted and the sorting direction.
Return	-	java.util.Collection	A sorted collection.

For example:

```
#foreach ($rel in $sorter.humanSort($package, "name:desc"))
  $rel.name
#end
```

- \$package is the collection to be sorted.
- "name:desc" is separated by ":" in two parts:
 - The first part is to identify fieldName to be sorted.
 - The second part is the option to identify the sorting scheme. Sometimes, the order is not identified. It is ascending by default.

\$sorter.humanSort(collection)

This is a special function to sort text in a human-like order. It splits text into numeric and non-numeric chunks and sorts them in numerical order. For example, "foo10" is ordered after "foo2".

	Name	Type	Description
Parameter(s)	Collection	java.util.Collection	A collection to be sorted.
Return	-	java.util.Collection	A sorted collection.

For example:

```
#foreach ($rel in $sorter.humanSort($package))
$rel.name
#end
```

- \$package is the collection to be sorted.