

Package ‘Seqnames’

February 12, 2014

Type Package

Title Utilities for manipulating chromosome and other 'seqname' identifiers

Version 0.99.10

Author Sonali Arora, H. Pages, Marc Carlson, Martin Morgan

Maintainer Sonali Arora <sarora@fhcrc.org>

Depends R (>= 3.1)

Suggests RUnit, BiocGenerics, BiocStyle, knitr, BSgenome.Scerevisiae.UCSC.sacCer2

Description The Seqnames package contains data and functions that define and allow translation between different chromosome sequence naming conventions (e.g., ``chr1" versus ``1"), including a function that attempts to place sequence names in their natural, rather than lexicographic, order.

License Artistic-2.0

biocViews Genetics, DataRepresentation, Annotation

Collate seqnames.R seqnames-helpers.R makeSeqnameIds.R

VignetteBuilder knitr

R topics documented:

makeSeqnameIds	2
seqname-helpers	3
Index	6

makeSeqnameIds	<i>Assign sequence IDs to sequence names</i>
----------------	--

Description

makeSeqnameIds assigns a unique ID to each unique sequence name in the input vector. The returned IDs span 1:N where N is the number of unique sequence names in the input vector.

Usage

```
makeSeqnameIds(seqnames, X.is.sexchrom=NA)
```

Arguments

seqnames	A character vector or factor containing sequence names.
X.is.sexchrom	A logical indicating whether X refers to the sexual chromosome or to chromosome with Roman Numeral X. If NA, makeSeqnameIds does its best to "guess".

Value

An integer vector of the same length as seqnames. The values in the vector span 1:N where N is the number of unique sequence names in the input vector.

Author(s)

H. Pages

See Also

- [sortSeqlevels](#) for sorting the sequence levels of an object in "natural" order.

Examples

```
library(BSgenome.Scerevisiae.UCSC.sacCer2)
makeSeqnameIds(seqnames(Scerevisiae))
makeSeqnameIds(seqnames(Scerevisiae)[c(1:5,5:1)])
```

Description

List the supported seqname styles for all supported organisms

Usage

```
supportedStyles(species)
extractSeqnameSet(species, style)
extractSeqnameSetByGroup(species, style, group)
seqnamesOrder(seqnames, X.is.sexchrom = NA)
findSequenceRenamingMaps(seqnames, style, best.only=TRUE, drop=TRUE)
seqnamesInGroup(seqnames, group, species, style)
seqnameStyle(x)
seqnameStyle(x) <- value
```

Arguments

species	The genus and species of the organism in question separated by a single space. Don't forget to capitalize the genus.
style	a character vector with a single element to specify the style.
group	Group can be 'auto' for autosomes, 'sex' for sex chromosomes/allosomes, 'linear' for linear chromosomes. The default is 'all' which returns all the chromosomes.
best.only	if TRUE (the default), then only the "best" sequence renaming maps (i.e. the rows with less NAs) are returned.
drop	if TRUE (the default), then a vector is returned instead of a matrix when the matrix has only 1 row.
seqnames	a character vector containing the labels attached to the chromosomes in a given genome for a given style. For example : For <i>Homo sapiens</i> , NCBI style - they are "1","2","3",...,"X","Y","MT"
X.is.sexchrom	A logical indicating whether X refers to the sexual chromosome or to chromosome with Roman Numeral X. If NA, makeSeqnameIds does its best to "guess".
x	The object from/on which to get/set the sequence information.
value	Typically a Seqinfo object for the seqinfo setter. Either a named or unnamed character vector for the seqlevels setter. A vector containing the sequence information to store for the other setters.

Details

`supportedStyles`: Different organizations have different naming conventions for how they name the biologically defined sequence elements (usually chromosomes) for each organism they support. The Seqnames package contains a database that defines these different conventions.

`supportedStyles()` returns the list of all supported seqname mappings, one per supported organism. Each mapping is represented as a data frame with 1 column per seqname style and 1 row per chromosome name (not all chromosomes of a given organism necessarily belong to the mapping).

`supportedStyles(species)` returns a data.frame only for the given organism with all its supported seqname mappings.

`extractSeqnameSet`: Extracts the seqnames for a single style and species.

`extractSeqnameSetByGroup`: Extracts the seqnames for a single style and species by group. Group can be 'auto' for autosomes, 'sex' for sex chromosomes/allosomes, 'linear' for linear chromosomes. The default is 'all' which returns all the chromosomes.

`seqnamesOrder`: Attempt to provide the "natural" order of seqnames, e.g., chr1, chr2, chr3, ...

`findSequenceRenamingMaps`: Returns a matrix with 1 column per supplied sequence name and 1 row per sequence renaming map compatible with the specified style. If `best.only` is TRUE (the default), only the "best" renaming maps (i.e. the rows with less NAs) are returned.

`seqnamesInGroup`: It takes a character vector along with a group and optional style and species. If group is not specified, it returns "all" or standard/top level seqnames. Returns a character vector of seqnames after subsetting for the group specified by the user. See examples for more details.

`seqnameStyle(x)`: finds the seqnameStyle for a given character vector.

Value

A character list containing character vectors with the names of all seqname styles for each supported species. The names of the list are the species names for each supported organism.

Author(s)

H. Pages, Marc Carlson, Sonali Arora <sarora@fhcrc.org>

Examples

```
supportedStyles()
supportedStyles("Homo sapiens")

## List the supported seqname style for the given species and the given style
extractSeqnameSet(species="Drosophila melanogaster", style="Ensembl")

## List all sex chromosomes for Homo sapiens using style UCSC
## 3 groups are supported: auto for autosomes, sex for allosomes
## and linear for linear chromosomes
extractSeqnameSetByGroup(species="Homo sapiens", style="UCSC", group="sex")

## find whether the seqnames belong to a given group
newchr <- paste0("chr",c(1:22,"X","Y","M","1_g1000192_random","4_ctg9_hap1"))
```

```
seqnamesInGroup(newchr, group="sex")

newchr <- as.character(c(1:22,"X","Y","MT"))
seqnamesInGroup(newchr, group="all","Homo sapiens","NCBI")

## find the seqname Style for a given character vector
seqnameStyle(paste0("chr",c(1:30)))

## order a character vector of seqnames
seqnames <- c("chr1","chr9", "chr2", "chr3", "chr10")
seqnames[seqnamesOrder(seqnames)]

## find mapped seqnamestyles for existing seqnames
findSequenceRenamingMaps(c("chrII", "chrIII", "chrM"), "NCBI")
findSequenceRenamingMaps(c("chrII", "chrIII", "chrM"), "Ensembl")
```

Index

*Topic **manip**

- `makeSeqnameIds`, [2](#)
- `extractSeqnameSet (seqname-helpers)`, [3](#)
- `extractSeqnameSetByGroup`
`(seqname-helpers)`, [3](#)
- `findSequenceRenamingMaps`
`(seqname-helpers)`, [3](#)
- `makeSeqnameIds`, [2](#)
- `seqname-helpers`, [3](#)
- `seqnamesInGroup (seqname-helpers)`, [3](#)
- `seqnamesOrder (seqname-helpers)`, [3](#)
- `seqnameStyle (seqname-helpers)`, [3](#)
- `seqnameStyle, character-method`
`(seqname-helpers)`, [3](#)
- `seqnameStyle<- (seqname-helpers)`, [3](#)
- `seqnameStyle<- , character-method`
`(seqname-helpers)`, [3](#)
- `sortSeqlevels`, [2](#)
- `supportedStyles (seqname-helpers)`, [3](#)