

# Package ‘pisar’

July 8, 2021

**Type** Package

**Title** pisar: pISA-tree Support Functions

**Version** 0.1.0.9000

**Description** The package provides several functions for support and use of pISA-tree.

**License** GPL-3

**URL** <https://github.com/NIB-SI/pisar>

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.1

**Imports** knitr,  
rio,  
tools,  
RCurl,  
httr,  
jsonlite

**VignetteBuilder** knitr

**Suggests** knitr,  
rmarkdown

## R topics documented:

fileName . . . . .	2
fileType . . . . .	2
fsummary . . . . .	3
getLayer . . . . .	4
getMeta . . . . .	4
getRoot . . . . .	5
out.path . . . . .	6
pasteMeta . . . . .	7
pisa . . . . .	8
pisar . . . . .	9
print.pISAMeta . . . . .	9
readMeta . . . . .	10
<b>Index</b>	<b>12</b>

---

fileName	<i>Extract file name</i>
----------	--------------------------

---

**Description**

Extract file name from a file path.

**Usage**

```
fileName(x, ...)
```

**Arguments**

x	Complete file path or file name.
...	Any other arguments.

**Value**

File name (character string).

**Note**

Parameter ... is ignored at this time.

**Author(s)**

Andrej Blejec <andrej.blejec@nib.si>

**See Also**

[fileType](#)

**Examples**

```
fileName(".\\validation.Rnw")
fileName("./bla/validation.Rnw")
fileName("./validation.")
fileName("./validation")
```

---

fileType	<i>Extract file type</i>
----------	--------------------------

---

**Description**

Extract file type from a file path.

**Usage**

```
fileType(x, ...)
```

**Arguments**

x	Complete file path or file name.
...	Any other arguments.

**Value**

File type (character string).

**Note**

Parameter ... is ignored at this time.

**Author(s)**

Andrej Blejec <andrej.blejec@nib.si>

**Examples**

```
fileName(".\\validation.Rnw")
fileName("./bla/validation.Rnw")
fileName("./validation.")
fileName("./validation")
```

---

fsummary

*As factor summary of a data frame*

---

**Description**

As factor summary of a data frame

**Usage**

```
fsummary(x, ...)
```

**Arguments**

x	Data frame.
...	Any other arguments.

**Value**

Summary object.

**Note**

Argument ... not used

**Author(s)**

Andrej Blejec <andrej.blejec@nib.si>

**Examples**

```
fsummary(data.frame(x=rnorm(20),txt=sample(letters,20,rep=TRUE)))
```

---

getLayer	<i>Get pISA layer name</i>
----------	----------------------------

---

**Description**

Get pISA layer name

**Usage**

```
getLayer(x, path = getwd())
```

**Arguments**

x	Layer character (one of "p", "I", "S", or "A").
path	Directory path, defaults to working directory.

**Value**

Character string with layer name.

**Author(s)**

Andrej Blejec <andrej.blejec@nib.si>

**Examples**

```
## Not run:
astring <- "_p_Demo/_I_Test/_S_Show/_A_Work-R/other"
oldwd <- setwd(system.file("extdata",astring,package="pisar"))
oldwd
.pname <- getLayer("p")
.pname
getLayer("I")
getLayer("S")
getLayer("A")
if(interactive()) setwd(oldwd)
getwd()

## End(Not run)
```

---

getMeta	<i>Get metadata value</i>
---------	---------------------------

---

**Description**

Get metadata value

**Usage**

```
getMeta(x, key, nl = TRUE)
```

**Arguments**

x	Two column character data frame with Key / Value pairs.
key	String, key name. Trailing colon can be omitted.
nl	Logical, expand backslash character for new lines.
...	Any other arguments (not used at the moment).

**Value**

Key value (character string or number).

**Note**

Argument key is matched exactly to the key names.

**Author(s)**

Andrej Blejec <andrej.blejec@nib.si>

**Examples**

```
## Not run:
astring <- "_p_Demo/_I_Test/_S_Show/_A_Work-R/other"
oldwd <- setwd(system.file("extdata",astring,package="pisar"))
oldwd
.iroot <- getRoot("I")
.imeta <- readMeta(.iroot)
getMeta(.imeta, "Description")
# list
listmeta <- list(Title = "My title"
  , Description = "A longer description")
getMeta( listmeta, "Title")
getMeta( listmeta, "Description")
setwd(oldwd)

## End(Not run)
```

---

getRoot

*Get root directory for pISA layer*


---

**Description**

Get root directory for pISA layer

**Usage**

```
getRoot(x = "p", path = getwd(), ...)
```

**Arguments**

x	Character characteristic for pISA layer (one of "p", "I", "S", or "A").
path	Path within the pISA-tree.
...	Any other arguments.

**Value**

Relative path to the layer directory (from working directory).

**Note**

The path should be compliant with the pISA-tree structure. Path defaults to the working directory, which is usually in or below an assay. Argument ... is not used.

**Author(s)**

Andrej Blejec <andrej.blejec@nib.si>

**Examples**

```
getRoot("p", path="d:/_p_prj/_I_inv/_S_st/_A_asy/other/doc")
getRoot("A", path="d:/_p_prj/_I_inv/_S_st/_A_asy")
```

---

out.path	<i>Create output directory</i>
----------	--------------------------------

---

**Description**

Create output directory, name it with appended arguments.

**Usage**

```
out.path(out.dir = "../out", args = "", which = 1:length(args))
```

**Arguments**

out.dir	Character string, base output directory.
args	Character vector, arguments used for sub-analysis. or from a batch call
which	Numeric vector, which arguments to use.

**Value**

Directory name.

**Note**

Directory is created

**Author(s)**

Andrej Blejec <andrej.blejec@nib.si>

**Examples**

```

out.path()
out.path(args="")
dir.exists("../out/xx-ena-dva")
out.path(args=c("xx.txt", "ena", "dva"))
dir.exists("../out/xx-ena-dva")
unlink("../out/xx-ena-dva", recursive=TRUE)
dir.exists("../out/xx-ena-dva")

```

pasteMeta

*Paste metadata values***Description**

Paste metadata values into a string.

**Usage**

```
pasteMeta(x, kvsep = "\t", nlsep = "\n")
```

**Arguments**

x	Two column character data frame with Key / Value pairs or a list of Vales named by Keys.
kvsep	Key/Value separator (default is a tab character).
nlsep	line separator (default is new line).

**Value**

Character string with concatenated key values.

**Author(s)**

Andrej Blejec <andrej.blejec@nib.si>

**Examples**

```

## Not run:
astring <- "_p_Demo/_I_Test/_S_Show/_A_Work-R/other"
oldwd <- setwd(system.file("extdata", astring, package="pisar"))
oldwd
.iroot <- getRoot("I")
.imeta <- readMeta(.iroot)
x <- pasteMeta(.imeta)
x
cat(x)
# list
listmeta <- list(Title = "My title"
  , Description = "A longer description")
x <- pasteMeta( listmeta )
x
cat(x)
setwd(oldwd)

```

```
## End(Not run)
```

---

pisa	<i>Extract pISA-tree details</i>
------	----------------------------------

---

## Description

Extract pISA-tree details: name, root and metadata for all layers above the current directory (below or in the Assay).

## Usage

```
pisa(path = getwd(), addArgs = NULL, global = TRUE)
```

## Arguments

path	Path to part of pISA-tree, defaults to working directory.
addArgs	Character vector, additional arguments.
global	If TRUE (default) auxiliary objects will be created in the global environment (see note).

## Value

A list with layer information components, possibly changing objects in the global environment (See Note).

## Note

If argument global is TRUE (default), auxiliary objects with pISA related information will be created in the global environment. Sometimes it is more convenient to use such object instead of the elements of the (invisibly) returned pisa list. The object are hidden (names start with dot); use `ls(pattern="^\\.", all.names=TRUE)` to get a full list of hiddent objects. The created objects are

- .`[pisa name]`** layer name
- .`[pisa root]`** layer path (relative to the working directory)
- .`[pisa meta]`** data frame with corresponding layer metadata
- .`oroot`** output directory path
- .`inroot`** input (data) directory path
- .`reproot`** report directory path
- .`reproot`** report directory path
- .`outputFile`** output file path and name
- .`args`** a vector of additional arguments (possibly from a batch call)
- .`pfn`** phenodata file path and name (relative to Investigation)
- .`ffn`** featuredata fiel path and name
- .`outfn`** output file basename (no type)
- .`rnwfn`** knitr source file name (\*.Rnw or \*.Rmd)



**Author(s)**

Andrej Blejec <andrej.blejec@nib.si>

**Examples**

```
## Not run:
astring <- "_p_Demo/_I_Test/_S_Show/_A_Work-R/other"
oldwd <- setwd(system.file("extdata", astring, package="pisar"))
oldwd
pisa <- pisa(global=FALSE)
str(pisa)
names(pisa)
dir(pisa$p$root)
# Set pisa options
options(pisa=pisa(global=FALSE))
# Access details with $
options()$pisa$p$root
setwd(oldwd)

## End(Not run)
```

---

pisar

---

*pisar: pISA-tree support functions*


---

**Description**

The package provides several functions for support and use of pISA-tree.

**Usage**

pISA-tree is a standardized directory tree for storing project information under ISA paradigm. This set of functions enables use of metadata for reproducible documents.

---

print.pISAMeta

---

*Print metadata object as Dlist*


---

**Description**

Print metadata object as Dlist

**Usage**

```
## S3 method for class 'pISAMeta'
print(x, width = max(nchar(x[, 1])) * 3.5, ...)
```

**Arguments**

x	Metadata object, data.frame with two columns.
width	Estimated text width.
...	Any other arguments.

**Note**

Metadata table is printed in convenient Dlist form.

**Author(s)**

Andrej Blejec <andrej.blejec@nib.si>

**Examples**

```
## Not run:
.pISALoc <- system.file("extdata", "_p_Demo", package="pisar")
readMeta(.pISALoc)

## End(Not run)
```

---

readMeta	<i>Read metadata file from the given directory</i>
----------	--

---

**Description**

Read metadata file from the given directory

**Usage**

```
readMeta(x = ".", ...)
```

**Arguments**

x	File path to the pISA layer.
...	Any other arguments.

**Value**

Data frame with Key/Value pairs with class 'pISAMeta'.

**Note**

Metadata table gets the class 'Dlist' to inherit a convenient print.

**Author(s)**

Andrej Blejec <andrej.blejec@nib.si>

**Examples**

```
## Not run:  
astring <- "_p_Demo/_I_Test/_S_Show/_A_Work-R/other"  
oldwd <- setwd(system.file("extdata",astring,package="pisar"))  
oldwd  
.aroot <- getRoot("A")  
.ameta <- readMeta(.aroot)  
.ameta  
setwd(oldwd)  
  
## End(Not run)
```

# Index

- \* **file**
  - fileName, [2](#)
  - fileType, [2](#)
- \* **pISA**
  - getRoot, [5](#)
  - readMeta, [10](#)
- \* **package**
  - print.pISAMeta, [9](#)
- \* **pisa**
  - getLayer, [4](#)
  - getMeta, [4](#)
  - out.path, [6](#)
  - pasteMeta, [7](#)
  - pisa, [8](#)
- \* **summary**
  - fsummary, [3](#)

fileName, [2](#)  
fileType, [2](#), [2](#)  
fsummary, [3](#)

getLayer, [4](#)  
getMeta, [4](#)  
getRoot, [5](#)

out.path, [6](#)

pasteMeta, [7](#)  
pisa, [8](#)  
pisar, [9](#)  
print.pISAMeta, [9](#)

readMeta, [10](#)