

Package ‘MagneticDB’

July 22, 2015

Title MagneticDB R Package

Description MagneticDB R Package furnishes some tools to process the data exported from the MagneticDB Android Application.

Version 1.0

License MIT

Author V. Sasyan <valentin.sasyan@gmail.com>

Maintainer LAGGISS, uOttawa <laggiss@gmail.com>

R topics documented:

classification	2
convertJsonExport	2
generate_qgs	3
getASCinfo	3
getCategory	4
getContrastEnhancement	4
getEditType	5
getGoogle	5
getIdType	6
getLayerTreeLayer	6
getLegendlayer	7
getMapcanvas	8
getMapLayerAsc	8
getMapLayerGoogle	9
getMapLayerShp	9
getNoData	10
getPipe	10
getPointList	11
getProperties	11
getRendererv2	12
getSHPinfo	12
getSpatialrefsys	13
getSymbol	13
interpolation	14
isWritable	14
listTypeBuilding	15
processData	15
qualifyClassif	16

readExport	17
sizeData	17

Index	19
--------------	-----------

classification	<i>Classify the given SpatialPointsDataFrame</i>
----------------	--

Description

Classify the given SpatialPointsDataFrame

Usage

```
classification(proj.df, useX = FALSE, lessType = FALSE, debug = FALSE)
```

Arguments

proj.df	SpatialPointsDataFrame, the points to classify
useX	boolean, use axe X for the classification
lessType	boolean, merge similar types
debug	boolean, to use the debug mode

Value

SpatialPointsDataFrame, the points classified

Author(s)

Valentin SASYAN, v. 2.2.0, 07/17/2015

convertJsonExport	<i>Convert the raw JSON export in a usable listPointXYZ</i>
-------------------	---

Description

Convert the raw JSON export in a usable listPointXYZ

Usage

```
convertJsonExport(x, folder)
```

Arguments

x	list, the raw JSON export
---	---------------------------

Value

list, the usable listPointXYZ

Author(s)

Valentin SASYAN, v. 1.2.0, 06/18/2015

generate_qgs	Create a QGIS project (.qgs file) with all the .asc and all the .shp files found in the folder
--------------	--

Description

Create a QGIS project (.qgs file) with all the .asc and all the .shp files found in the folder

Usage

```
generate_qgs(folder, file, EPSG, Google = TRUE)
```

Arguments

folder	string, folder to scan
file	string, name of the QGIS project file
EPSG	list, EPSG description of the reference system used as destination
Google	boolean, true to add a Google Streets Layer at the QGIS Project

Value

nothing

Author(s)

Valentin SASYAN, v. 1.2.0, 06/22/2015

Examples

```
generate_qgs('data/generated/', 'qgis_project.qgs', EPSG, FALSE)
```

getASCinfo	Get information for all the .asc files of the folder given for parameter
------------	--

Description

Get information for all the .asc files of the folder given for parameter

Usage

```
getASCinfo(folder)
```

Arguments

folder	string, folder to scan
--------	------------------------

Value

list, a list with all the information

Author(s)

Valentin SASYAN, v. 1.0.1, 06/12/2015

Examples

```
getASCinfo('data/generated/')
```

getCategory	<i>Generate the 'category' XML element of a .shp 'categories' XML element</i>
-------------	---

Description

Generate the 'category' XML element of a .shp 'categories' XML element

Usage

```
getCategory(i)
```

Arguments

i	integer, indice of the symbol in the symbols list
---	---

Value

XMLNode, the XML element generated

Author(s)

Valentin SASYAN, v. 1.0.0, 06/22/2015

getContrastEnhancement	<i>Generate the 'ContrastEnhancement' XML element of a pipe XML element</i>
------------------------	---

Description

Generate the 'ContrastEnhancement' XML element of a pipe XML element

Usage

```
getContrastEnhancement(color, liste)
```

Arguments

color	string, name of the color
liste	list, min and max for this color

Value

XMLNode, the XML element generated

Author(s)

Valentin SASYAN, v. 1.0.0, 06/12/2015

Examples

```
getContrastEnhancement('green', c(min=-0.84, max=0.42))
```

getEditType	<i>Generate the 'edittype' XML element of a .shp 'maplayer' XML element</i>
-------------	---

Description

Generate the 'edittype' XML element of a .shp 'maplayer' XML element

Usage

```
getEditType(editTypes)
```

Arguments

editTypes list, list of the name of the file's band

Value

XMLNode, the XML element generated

Author(s)

Valentin SASYAN, v. 1.0.0, 06/12/2015

getGoogle	<i>Create information for a Google Streets Layer</i>
-----------	--

Description

Create information for a Google Streets Layer

Usage

```
getGoogle(Google)
```

Arguments

Google boolean, must create the information or not?

Value

list, a list with all the information

Author(s)

Valentin SASYAN, v. 1.0.0, 06/15/2015

Examples

```
google <- getGoogle(TRUE)
google <- getGoogle(FALSE)
```

getIdType

Convert the String type of a building in a integer type

Description

Convert the String type of a building in a integer type

Usage

```
getIdType(typeString)
```

Arguments

typeString String, the String type of the building

Value

integer, the integer type of the building

Author(s)

Valentin SASYAN, v. 1.0.0, 06/18/2015

getLayerTreeLayer

Generate the 'layer-tree-layer' XML element of a 'layer-tree-group' XML element

Description

Generate the 'layer-tree-layer' XML element of a 'layer-tree-group' XML element

Usage

```
getLayerTreeLayer(Name, ID)
```

Arguments

Name string, name of the layer
ID string, id of the layer

Value

XMLNode, the XML element generated

Author(s)

Valentin SASYAN, v. 1.0.0, 06/12/2015

Examples

```
getLayerTreeLayer('exportedData', 'exportedData20150612104242')
```

getLegendlayer

Generate the 'noData' XML element of a 'legend' XML element

Description

Generate the 'noData' XML element of a 'legend' XML element

Usage

```
getLegendlayer(Name, ID)
```

Arguments

Name string, name of the layer

ID string, id of the layer

Value

XMLNode, the XML element generated

Author(s)

Valentin SASYAN, v. 1.0.0, 06/12/2015

Examples

```
getLegendlayer('exportedData', 'exportedData20150612104242')
```

getMapcanvas	<i>Generate the 'mapcanvas' XML element of a 'qgis' XML element</i>
--------------	---

Description

Generate the 'mapcanvas' XML element of a 'qgis' XML element

Usage

```
getMapcanvas(xmin, ymin, xmax, ymax, EPSG)
```

Arguments

xmin	double, xmin limit of the map canevas
ymin	double, ymin limit of the map canevas
xmax	double, xmax limit of the map canevas
ymax	double, ymax limit of the map canevas
EPSG	list, EPSG description of the reference systeme used as destination

Value

XMLNode, the XML element generated

Author(s)

Valentin SASYAN, v. 1.1.0, 06/15/2015

Examples

```
getMapcanvas(-42,-42,42,42)
```

getMapLayerAsc	<i>Generate the 'maplayer' XML element of a .asc</i>
----------------	--

Description

Generate the 'maplayer' XML element of a .asc

Usage

```
getMapLayerAsc(id, datasource, layername, spatialrefsys, noData, pipe)
```

Arguments

id	string, id of the layer
datasource	string, datasource of the layer
layername	string, layername of the layer
spatialrefsys	XML element, specifications of the spatial reference systeme
noData	XML element, number of the data
pipe	XML element, specifications of the colorisation of the layer

Value

XMLNode, the XML element generated

Author(s)

Valentin SASYAN, v. 1.0.0, 06/12/2015

getMapLayerGoogle	<i>Generate the 'maplayer' XML element of a Google Streets Layer</i>
-------------------	--

Description

Generate the 'maplayer' XML element of a Google Streets Layer

Usage

```
getMapLayerGoogle(id, layername, spatialrefsys)
```

Arguments

id	string, id of the layer
layername	string, layername of the layer
spatialrefsys	XML element, specifications of the spatial reference systeme

Value

XMLNode, the XML element generated

Author(s)

Valentin SASYAN, v. 1.0.0, 06/15/2015

getMapLayerShp	<i>Generate the 'maplayer' XML element of a .shp</i>
----------------	--

Description

Generate the 'maplayer' XML element of a .shp

Usage

```
getMapLayerShp(id, datasource, layername, spatialrefsys, editTypes)
```

Arguments

id	string, id of the layer
datasource	string, datasource of the layer
layername	string, layername of the layer
spatialrefsys	XML element, specifications of the spatial reference systeme
editTypes	list, list of the name of the file's band

Value

XMLNode, the XML element generated

Author(s)

Valentin SASYAN, v. 1.0.0, 06/12/2015

getNoData	<i>Generate the 'noData' XML element of a .asc 'maplayer' XML element</i>
-----------	---

Description

Generate the 'noData' XML element of a .asc 'maplayer' XML element

Usage

```
getNoData(min, max)
```

Arguments

min	int, first number of the data
max	int, last number of the data

Value

XMLNode, the XML element generated

Author(s)

Valentin SASYAN, v. 1.0.0, 06/12/2015

Examples

```
getNoData(1,3)
```

getPipe	<i>Generate the 'pipe' XML element of a .asc 'maplayer' XML element</i>
---------	---

Description

Generate the 'pipe' XML element of a .asc 'maplayer' XML element

Usage

```
getPipe(liste, opacity = 1)
```

Arguments

liste	list, specification of the color used for each band
opacity	double, opacity of the layer

Value

XMLNode, the XML element generated

Author(s)

Valentin SASYAN, v. 1.1.0, 06/15/2015

Examples

```
getPipe(list(red=c(min=-1,max=1),green=c(min=-0.84, max=0.42),blue=c(min=-0.42,max=0.84))
```

getPointList	<i>Generate a list of SpatialPoint</i>
--------------	--

Description

Generate a list of SpatialPoint

Usage

```
getPointList(extent, resolution)
```

Arguments

extent	extent, area to cover
resolution	double, resolution of the grid of interpolation (size of 1 px in real)

Value

list, a list of SpatialPoint

Author(s)

Valentin SASYAN, v. 1.1.0, 06/12/2015

getProperties	<i>Generate the 'properties' XML element of a .asc 'maplayer' XML element</i>
---------------	---

Description

Generate the 'properties' XML element of a .asc 'maplayer' XML element

Usage

```
getProperties(EPSPG)
```

Arguments

EPSPG	list, EPSG description of the reference system used as destination
-------	--

Value

XMLNode, the XML element generated

Author(s)

Valentin SASYAN, v. 1.1.0, 06/15/2015

getRendererv2

Generate the 'renderer-v2' XML element of a .shp 'maplayer' XML element

Description

Generate the 'renderer-v2' XML element of a .shp 'maplayer' XML element

Usage

getRendererv2()

Value

XMLNode, the XML element generated

Author(s)

Valentin SASYAN, v. 1.1.0, 06/22/2015

getSHPinfo

Get information for all the .shp files of the folder given for parameter

Description

Get information for all the .shp files of the folder given for parameter

Usage

getSHPinfo(folder)

Arguments

folder string, folder to scan

Value

list, a list with all the information

Author(s)

Valentin SASYAN, v. 1.0.1, 06/12/2015

Examples

getSHPinfo('data/generated/')

getSpatialrefsys	<i>Generate the 'spatialrefsys' XML element of a 'maplayer' XML element</i>
------------------	---

Description

Generate the 'spatialrefsys' XML element of a 'maplayer' XML element

Usage

```
getSpatialrefsys(EPSG)
```

Arguments

EPSG	list, EPSG description of the reference system used as destination
------	--

Value

XMLNode, the XML element generated

Author(s)

Valentin SASYAN, v. 1.1.0, 06/15/2015

getSymbol	<i>Generate the 'category' XML element of a .shp 'categories' XML element</i>
-----------	---

Description

Generate the 'category' XML element of a .shp 'categories' XML element

Usage

```
getSymbol(i)
```

Arguments

i	integer, indice of the symbol in the symbols list
---	---

Value

XMLNode, the XML element generated

Author(s)

Valentin SASYAN, v. 1.1.0, 06/23/2015

interpolation	<i>Interpolate the date in the SpatialPointsDataFrame given for parameter</i>
---------------	---

Description

Interpolate the date in the SpatialPointsDataFrame given for parameter

Usage

```
interpolation(proj.df, resolution = 100, EPSG, interpolation = "idw",
  param = 0.5)
```

Arguments

proj.df	SpatialPointsDataFrame, the data to interpolate
resolution	double, resolution of the grid of interpolation (size of 1 px in real)
EPSG	list, EPSG description of the reference system used as destination
interpolation	string, the name of the interpolation to use
param	variant, parameter for some interpolation

Value

SpatialPointsDataFrame, the data interpolated

Author(s)

Valentin SASYAN, v. 1.3.1, 06/17/2015

isWritable	<i>Check if the script can write the specified file (erasing or not an eventual existing file)</i>
------------	--

Description

Check if the script can write the specified file (erasing or not an eventual existing file)

Usage

```
isWritable(erase, file)
```

Arguments

erase	boolean, the script can erase an eventual existing file
file	string, path of the file to test

Value

boolean, true if the file can be written

Author(s)

Valentin SASYAN, v1.0.0, 06/12/2015

listTypeBuilding	<i>List of the possible building type</i>
------------------	---

Description

List of the possible building type

Usage

listTypeBuilding

Format

chr [1:16] "(W1) Wood, Light Frame (<= 5,000 sq. ft.)" "(W2) Wood, Commercial and Industrial (>5,000 sq. ft.)"

Author(s)

LAGGISS, uOttawa v. 1.1.1, 06/19/2015

processData	<i>Process (interpolation + classification) the data generated by the Android App and exported in the JSON file</i>
-------------	---

Description

Process (interpolation + classification) the data generated by the Android App and exported in the JSON file

Usage

```
processData(filter = "uOttawa", resolution = 0, export = TRUE,
            erase = TRUE, EPSG = EPSG_, interpolation = "idw", param = 0.5,
            classif = TRUE, qualify = FALSE, useX = FALSE, lessType = FALSE)
```

Arguments

filter	RegExpr, filter for the name of the folders to use
resolution	double, resolution of the grid of interpolation (size of 1 px in real), 0 for no interpolation
export	bool, true if the function have to export the used data
erase	bool, true if the function can erase existing used data
EPSG	list, EPSG description of the reference system used as destination
interpolation	string, the name of the interpolation to use
param	variant, parameter for some interpolation

classif	boolean, do or not the the classification
qualify	boolean, show or not the classification statistics (needs classif=TRUE)
useX	boolean, use axe X for the classification
lessType	boolean, merge similar types

Value

nothing

Author(s)

Valentin SASYAN, v. 0.7.0, 07/22/2015

Examples

```
generate_qgs(filter='u0ttawa.*',resolution=1,erase=FALSE)
```

qualifyClassif	<i>Qualify the given classification</i>
----------------	---

Description

Qualify the given classification

Usage

```
qualifyClassif(proj.classif)
```

Arguments

proj.classif SpatialPointsDataFrame, the points classification

Value

nothing

Author(s)

Valentin SASYAN, v. 1.0.1, 07/22/2015

readExport	<i>Read the date in the json file</i>
------------	---------------------------------------

Description

Read the date in the json file

Usage

```
readExport(filter, EPSG)
```

Arguments

filter	RegExpr, filter for the name of the folders to use
EPSG	list, EPSG description of the reference system used as destination

Value

SpatialPointsDataFrame, the data of the JSON in a SPDF ready for the interpolation

Author(s)

Valentin SASYAN, v. 1.1.0, 06/16/2015

sizeData	<i>Calculate the size of the interpolation mesh for the given data and the given interpolation</i>
----------	--

Description

Calculate the size of the interpolation mesh for the given data and the given interpolation

Usage

```
sizeData(file = "generated", resolution = 1000, EPSG = EPSG_)
```

Arguments

file	string, name of the JSON file in the data folder (without the .json extension)
resolution	double, resolution of the grid of interpolation (size of 1 px in real)
EPSG	list, EPSG description of the reference system used as destination

Value

list, the X size and the Y size

Author(s)

Valentin SASYAN, v. 0.2.0, 06/15/2015

Examples

```
sizeData(file'generated',resolution=1)
```

Index

*Topic **datasets**

listTypeBuilding, [15](#)

classification, [2](#)

convertJsonExport, [2](#)

generate_qgs, [3](#)

getASCinfo, [3](#)

getCategory, [4](#)

getContrastEnhancement, [4](#)

getEditType, [5](#)

getGoogle, [5](#)

getIdType, [6](#)

getLayerTreeLayer, [6](#)

getLegendlayer, [7](#)

getMapcanvas, [8](#)

getMapLayerAsc, [8](#)

getMapLayerGoogle, [9](#)

getMapLayerShp, [9](#)

getNoData, [10](#)

getPipe, [10](#)

getPointList, [11](#)

getProperties, [11](#)

getRendererv2, [12](#)

getSHPinfo, [12](#)

getSpatialrefsys, [13](#)

getSymbol, [13](#)

interpolation, [14](#)

isWritable, [14](#)

listTypeBuilding, [15](#)

processData, [15](#)

qualifyClassif, [16](#)

readExport, [17](#)

sizeData, [17](#)