Package ‘ows4R’

May 27, 2020

Version 0.1-5
Date 2020-05-27
Title Interface to OGC Web-Services (OWS)
Maintainer Emmanuel Blondel <emmanuel.blondel1@gmail.com>
Depends R (>= 3.3.0), methods, geometa
Imports R6, httr, openssl, XML (>= 3.96-1.1), sf, rgdal
Suggests testthat
Description Provides an Interface to Web-Services defined as standards by the Open Geospatial Consortium (OGC), including Web Feature Service (WFS) for vector data, Catalogue Service (CSW) for ISO/OGC metadata and associated standards such as the common web-service specification (OWS) and OGC Filter Encoding. The long-term purpose is to add support for additional OGC service standards such as Web Coverage Service (WCS) and Web Processing Service (WPS).
License MIT + file LICENSE
URL https://github.com/eblondel/ows4R,
  http://www.opengeospatial.org/standards
BugReports https://github.com/eblondel/ows4R/issues
LazyLoad yes
RoxygenNote 6.1.1
NeedsCompilation no
Author Emmanuel Blondel [aut, cre] (<https://orcid.org/0000-0002-5870-5762>),
Norbert Billet [ctb]
Repository CRAN
Date/Publication 2020-05-27 15:20:02 UTC
R topics documented:

And ................................................................. 3
BBOX ............................................................... 3
BinaryComparisonOpType ........................................ 4
BinaryLogicOpType ................................................. 5
CSWCapabilities .................................................... 6
CSWClient .......................................................... 7
CSWConstraint ...................................................... 8
CSWDescribeRecord ............................................... 9
CSWGetRecordById ............................................... 10
CSWGetRecords ................................................... 10
CSWHarvest ......................................................... 11
CSWQuery .......................................................... 12
CSWRecordProperty ............................................... 13
CSWTTransaction .................................................. 14
Not ................................................................. 15
OGCAbstractObject ................................................. 15
OGCExpression .................................................... 16
OGCFilter .......................................................... 17
Or ................................................................. 17
ows4R ............................................................... 18
OWSCapabilities ................................................... 19
OWSClient ........................................................ 20
OWSGetCapabilities .............................................. 21
OWSOperation ...................................................... 21
OWSOperationsMetadata ......................................... 22
OWSRequest ....................................................... 23
OWSServiceIdentification ....................................... 24
OWSServiceProvider ............................................. 25
OWSUtils .......................................................... 26
PropertyIsBetween ................................................. 27
PropertyIsEqualTo ................................................ 27
PropertyIsGreaterThan .......................................... 28
PropertyIsGreaterThanOrEqualTo ............................... 28
PropertyIsGreaterThanOrEqualTo ............................... 29
PropertyIsLessThan .............................................. 29
PropertyIsLessThanOrEqualTo .................................. 30
PropertyIsLike .................................................... 31
PropertyIsNotEqualTo .......................................... 32
PropertyIsNull ..................................................... 32
UnaryLogicOpType ............................................... 33
WFSCapabilities .................................................. 34
WFSClient ........................................................ 35
WFSDescribeFeatureType ....................................... 36
WFSFeature ....................................................... 37
WFSFeatureType .................................................. 38
WFSFeatureTypeElement ......................................... 39

Index 40
And

Description
And

Usage
And

Format
And

R6Class object.

Value
Object of R6Class for modelling an And operator

Methods
new(...) This method is used to instantiate an And operator

Author(s)
Emmanuel Blondel <emmanuel.blondel1@gmail.com>

Examples
expr1 <- PropertyIsEqualTo$new(PropertyName = "property1", Literal = "value1")
expr2 <- PropertyIsEqualTo$new(PropertyName = "property2", Literal = "value2")
and <- And$new(expr1,expr2)
and_xml <- and$encode() #see how it looks like in XML

BBOX

Description
BBOX

Usage
BBOX
BinaryComparisonOpType

Format

*R6Class* object.

Value

Object of *R6Class* for modelling an BBOX

Methods

`new(bbox, srsName)` This method is used to instantiate an BBOX

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

Examples

```r
bbox <- OWSUtils$toBBOX(-180,-90,180,90)
expr <- BBOX$new(bbox)
expr_xml <- expr$encode() # see how it looks like in XML
```

BinaryComparisonOpType

---

**Description**

BinaryComparisonOpType

**Usage**

BinaryComparisonOpType

**Format**

*R6Class* object.

**Value**

Object of *R6Class* for modelling an BinaryComparisonOpType

**Methods**

`new(PropertyName, Literal, matchCase)` This method is used to instantiate an BinaryComparisonOpType
BinaryLogicOpType

Note
abstract super class of all the property operation classes

Author(s)
Emmanuel Blondel <emmanuel.blondel1@gmail.com>

---

BinaryLogicOpType  BinaryLogicOpType

Description
BinaryLogicOpType

Usage
BinaryLogicOpType

Format
R6Class object.

Value
Object of R6Class for modelling an BinaryLogicOpType

Methods
new(...) This method is used to instantiate an BinaryLogicOpType

Note
abstract super class of all the binary logical operation classes
Description

CSWCapabilities

Usage

CSWCapabilities

Format

R6Class object.

Value

Object of R6Class with methods for interfacing an OGC Catalogue Service for the Web (CSW) Get Capabilities document.

Methods

new(url, version, logger) This method is used to instantiate a WFSGetCapabilities object

Note

Class used to read a CSWCapabilities document. The use of CSWClient is recommended instead to benefit from the full set of capabilities associated to a CSW server.

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

Examples

#example based on CSW endpoint responding at http://localhost:8000/csw
caps <- CSWCapabilities$new("http://localhost:8000/geonetwork/csw", version = "2.0.2")
Description

CSWClient

Usage

CSWClient

Format

- **R6Class** object.

Value

Object of **R6Class** with methods for interfacing an OGC Catalogue Service for the Web.

Methods

- **new(url, serviceVersion, user, pwd, token, logger)** This method is used to instantiate a CSWClient with the `url` of the OGC service. Authentication is supported either with a basic (`user/pwd`) authentication or a token-based authentication. By default, the `logger` argument will be set to NULL (no logger). This argument accepts two possible values: INFO: to print only **ows4R** logs, DEBUG: to print more verbose logs.

- **getCapabilities()** Get service capabilities. Inherited from OWS Client

- **reloadCapabilities()** Reload service capabilities

- **describeRecord(namespace, ...)** Describe records. Retrieves the XML schema for CSW records. By default, returns the XML schema for the CSW records (http://www.opengis.net/cat/csw/2.0.2). For other schemas, specify the `outputSchema` required, e.g. http://www.isotc211.org/2005/gmd for ISO 19115/19139 schema.

- **getRecordById(id, elementSetName, ...)** Get a record by ID. By default, the record will be returned following the CSW schema (http://www.opengis.net/cat/csw/2.0.2). For other schemas, specify the `outputSchema` required, e.g. http://www.isotc211.org/2005/gmd for ISO 19115/19139 records. The parameter `elementSetName` should among values "full", "brief", "summary". The default “full” corresponds to the full metadata sheet returned. "brief" and "summary" will contain only a subset of the metadata content.

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>
Examples

```r
# example based on CSW endpoint responding at http://localhost:8000/csw
csw <- CSWClient$new("http://localhost:8000/csw", serviceVersion = "2.0.2")

# get capabilities
caps <- csw$getCapabilities()

# get records
records <- csw$getRecords()

# get record by id
record <- csw$getRecordById("my-metadata-id")

# Advanced examples at https://github.com/eblondel/ows4R/wiki#csw
```

---

<table>
<thead>
<tr>
<th>CSWConstraint</th>
<th>CSWConstraint</th>
</tr>
</thead>
</table>

**Description**

CSWConstraint

**Usage**

CSWConstraint

**Format**

- R6Class object.

**Value**

Object of R6Class for modelling an CSW Constraint

**Methods**

new(cqlText, filter, serviceVersion) This method is used to instantiate an CSWConstraint object.

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>
Examples

```r
filter <- OGCFilter$new(PropertyIsEqualTo$new("apiso:Identifier", "12345") )
cons <- CSWConstraint$new(filter = filter)
cons_xml <- cons$encode() # how it looks like in XML
```

Description

CSWDescribeRecord

Usage

CSWDescribeRecord

Format

- `R6Class` object.

Value

Object of `R6Class` for modelling a CSW DescribeRecord request

Methods

- `new(url, version, namespace, ...)` This method is used to instantiate a CSWDescribeRecord object

Note

Class used internally by `ows4R` to trigger a CSW DescribeRecord request

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>
CSWGetRecordById

Description
CSWGetRecordById

Usage
CSWGetRecordById

Format
R6Class object.

Value
Object of R6Class for modelling a CSW GetRecordById request

Methods
new(op, url, serviceVersion, user, pwd, token, id, elementSetName, logger, ...) This method is used to instantiate a CSWGetRecordById object

Note
Class used internally by ows4R to trigger a CSW GetRecordById request

Author(s)
Emmanuel Blondel <emmanuel.blondel1@gmail.com>

CSWGetRecords

Description
CSWGetRecords

Usage
CSWGetRecords

Format
R6Class object.
Value

Object of \texttt{R6Class} for modelling a CSW GetRecords request

Methods

\texttt{new(op, url, serviceVersion, user, pwd, token, query, logger, \ldots\text{)}} This method is used to instantiate a CSWGetRecords object

Note

Class used internally by \texttt{ows4R} to trigger a CSW GetRecords request

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

---

Description

CSWHarvest

Usage

CSWHarvest

Format

\texttt{R6Class} object.

Value

Object of \texttt{R6Class} for modelling a CSW Harvest request

Methods

\texttt{new(op, url, serviceVersion, user, pwd, source, resourceType, resourceFormat, logger, \ldots\text{)}}

This method is used to instantiate a CSWHarvest object

Note

Class used internally by \texttt{ows4R} to trigger a CSW Harvest request

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>
CSWQuery

Description

CSWQuery

Usage

CSWQuery

Format

R6Class object.

Value

Object of R6Class for modelling an CSW Query

Methods

new(filter, serviceVersion) This method is used to instantiate an CSWQuery object.

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

Examples

#CSWQuery - elementSetName
query_full <- CSWQuery$new()
query_brief <- CSWQuery$new(elementSetName = "brief")
query_summary <- CSWQuery$new(elementSetName = "summary")

#CSWQuery - cqlText with title
cons <- CSWConstraint$new(cqlText = "dc:title like '%ips%'")
query <- CSWQuery$new(constraint = cons)

#CSW 2.0.2 - Query - Filter / AnyText
filter <- OGCFilter$new(PropertyIsLike$new("csw:AnyText", "%Physio%"))
cons <- CSWConstraint$new(filter = filter)
query <- CSWQuery$new(constraint = cons)

#CSW 2.0.2 - Query - Filter / AnyText Equal
filter <- OGCFilter$new(PropertyIsEqualTo$new("csw:AnyText", "species"))
cons <- CSWConstraint$new(filter = filter)
query <- CSWQuery$new(constraint = cons)

#CSW 2.0.2 - Query - Filter / AnyText And Not
CSWRecordProperty

Description
CSWRecordProperty

Usage
CSWRecordProperty

Format
R6Class object.

Value
Object of R6Class for modelling an CSW RecordProperty
Methods

new(name, value)  This method is used to instantiate an CSWRecordProperty object.

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

Examples

rp <- CSWRecordProperty$new(name = "NAME", value = "VALUE")
rp_xml <- rp$encode()  #see how it looks in XML

CSWTransaction

Description

CSWTransaction

Usage

CSWTransaction

Format

**R6Class** object.

Value

Object of **R6Class** for modelling a CSW Transaction request

Methods

new(url, version, id)  This method is used to instantiate a CSWTransaction object

Note

Class used internally by **ows4R** to trigger a CSW Transaction request

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>
Description

Not

Usage

Not

Format

R6Class object.

Value

Object of R6Class for modelling an Not

Methods

new(...) This method is used to instantiate an Not operator

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

Examples

expr <- PropertyIsEqualTo$new(PropertyName = "property", Literal = "value")
not <- Not$new(expr)
not_xml <- not$encode() # see how it looks like in XML
Format

\textbf{R6Class} object.

Value

Object of \textbf{R6Class} for modelling an OGC\texttt{AbstractObject}

Methods

new() This method is used to instantiate an OGC\texttt{AbstractObject}

Note

abstract class used by \texttt{ows4R}

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

\begin{tabular}{ll}
\textbf{OGCExpression} & \textit{OGCExpression} \\
\end{tabular}

\begin{tabular}{ll}
Description & OGCExpression \\
Usage & OGCExpression \\
Format & \textbf{R6Class} object. \\
Value & Object of \textbf{R6Class} for modelling an OGC Expression \\
Methods & \texttt{new()} This method is used to instantiate an OGCExpression object \\
Note & abstract class \\
Author(s) & Emmanuel Blondel <emmanuel.blondel1@gmail.com>
\end{tabular}
OGCFilter

Description

OGCFilter

Usage

OGCFilter

Format

R6Class object.

Value

Object of R6Class for modelling an OGC Filter

Methods

new(expr) This method is used to instantiate an OGCFilter object. The unique argument should be an object of class OGCExpression

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

Examples

expr <- PropertyIsEqualTo$new(PropertyName = "property", Literal = "value")
not <- Not$new(expr)
not_xml <- not$encode() #see how it looks like in XML

Or

Or

Description

Or

Usage

Or
Format

R6Class object.

Value

Object of R6Class for modelling an Or operator

Methods

new(...) This method is used to instantiate an Or operator

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

Examples

expr1 <- PropertyIsEqualTo$new(PropertyName = "property1", Literal = "value1")
expr2 <- PropertyIsEqualTo$new(PropertyName = "property2", Literal = "value2")
or <- Or$new(expr1, expr2)
or_xml <- or$encode() #see how it looks like in XML

ows4R

Interface to OGC Web-Services

Description

Provides an Interface to Web-Services defined as standards by the Open Geospatial Consortium (OGC), including Web Feature Service (WFS) for vector data, Catalogue Service (CSW) for ISO/OGC metadata and associated standards such as the common web-service specification (OWS) and OGC Filter Encoding. The long-term purpose is to add support for additional OGC service standards such as Web Coverage Service (WCS) and Web Processing Service (WPS).

Details

<table>
<thead>
<tr>
<th>Package</th>
<th>ows4R</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Package</td>
</tr>
<tr>
<td>Version</td>
<td>0.1-5</td>
</tr>
<tr>
<td>Date</td>
<td>2020-05-27</td>
</tr>
<tr>
<td>License</td>
<td>MIT</td>
</tr>
<tr>
<td>LazyLoad</td>
<td>yes</td>
</tr>
</tbody>
</table>
OWSCapabilities

Author(s)
Emmanuel Blondel <emmanuel.blondel1@gmail.com>

---------------------------
OWSCapabilities
OWSGetCapabilities
---------------------------

Description

OWSGetCapabilities

Usage

OWSCapabilities

Format

R6Class object.

Value

Object of R6Class with methods for interfacing an abstract OWS Get Capabilities document.

Methods

new(url, service, serviceVersion, owsVersion, logger) This method is used to instantiate a OWSGetCapabilities object
getUrl() Get URL
getVersion() Get version
getRequest() Get request
getServiceIdentification() Get the service identification
getServiceProvider() Get the service provider
getOperationsMetadata() Get the service operations metadata

Note

abstract class used by ows4R

Author(s)
Emmanuel Blondel <emmanuel.blondel1@gmail.com>
**OWSClient**

**Description**

OWSClient

**Usage**

OWSClient

**Format**

- **R6Class** object.

**Value**

Object of **R6Class** with methods for interfacing a Common OGC web-service.

**Fields**

- **url** the Base url of OWS service
- **version** the version of OWS service

**Methods**

- **new(url, service, serviceVersion, user, pwd, logger)** This method is used to instantiate an OWSClient with the url of the OGC service. Authentication (user/pwd) is not yet supported and will be added with the support of service transactional modes. By default, the logger argument will be set to NULL (no logger). This argument accepts two possible values: INFO: to print only **ows4R** logs, DEBUG: to print more verbose logs
- **getUrl()** Get the service URL
- **getVersion()** Get the service version
- **getCapabilities()** Get the service capabilities

**Note**

Abstract class used internally by **ows4R**

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>
**Description**

OWSGetCapabilities

**Usage**

OWSGetCapabilities

**Format**

R6Class object.

**Value**

Object of R6Class for modelling a GetCapabilities request

**Methods**

`new(op, url, service, version)` This method is used to instantiate a OWSGetCapabilities object

**Note**

Abstract class used internally by ows4R

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**Description**

OWSOperation

**Usage**

OWSOperation

**Format**

R6Class object.
Value

Object of R6Class for modelling an OGC Operation

Methods

new(xmlObj, serviceVersion) This method is used to instantiate an OWSOperation object
getName() Get name
getParameters() Get the list of parameters
getParameter(name) Get a given parameter

Note

Abstract class used internally by ows4R

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>
**_EWSRequest_**

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

---

**Description**

_EWSRequest_

**Usage**

_EWSRequest_

**Format**

_R6Class_ object.

**Value**

Object of _R6Class_ for modelling a generic OWS request

**Methods**

`new(op, type, url, request, user, pwd, namedParams, attrs, contentType, mimeType, logger)`

This method is used to instantiate a object for doing an OWS request

`getRequest()` Get the request payload

`getRequestHeaders()` Get the request headers

`getStatus()` Get the request status code

`getResponse()` Get the request response

`getException()` Get the exception (in case of request failure)

`getResult()` Get the result TRUE if the request is successful, FALSE otherwise

**Note**

Abstract class used internally by _ows4R_

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>
Description

OWSServiceIdentification

Usage

OWSServiceIdentification

Format

R6Class object.

Value

Object of R6Class for modelling an OGC Service Identification

Methods

new(xmlObj, serviceVersion) This method is used to instantiate a OWSServiceIdentification object
getName() Get service name
getTitle() Get service title
getAbstract() Get service abstract
getKeywords() Get service keywords
getOnlineResource() Get service online resource
getServiceType Get service type
getServiceTypeVersion Get service type version

Note

Abstract class used internally by ows4R

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>
**Description**

OWSServiceProvider

**Usage**

OWSServiceProvider

**Format**

*R6Class* object.

**Value**

Object of *R6Class* for modelling an OGC Service Provider

**Methods**

`new(xmlObj, version)` This method is used to instantiate a OWSServiceProvider object

`getProviderName()` Get the provider name

`getProviderSite()` Get the provision site

`getServiceContact()` Get the service contact, as object of class ISOResponsiblleParty from package *geometa*

**Note**

Abstract class used internally by *ows4R*

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>
Description

OWSUtils

Usage

OWSUtils

Format

R6Class object.

Value

set of OWS Utilities

Static methods

- getNamespaces() Get the namespaces associated to a given XML object
- findNamespace(namespaces, id, uri) Finds a namespace by id or by URI
- toBBOX(xmin, xmax, ymin, ymax) Creates a bbox matrix from min/max x/y coordinates
- findP4s(srsName, morphToESRI) Finds the PROJ4 string definition for a given srsName
- toCRS(srsName) Converts a srsName into a CRS object
- toEPSG(crs) Get the EPSG code from a CRS object
- getAspectRatio(bbox) Get the aspect ratio for a given bbox

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

Examples

```r
#toBBOX
bbox <- OWSUtils$toBBOX(-180,-90,180,90)

#toCRS
crs <- OWSUtils$toCRS("EPSG:4326")
```
### PropertyIsBetween

**Description**

PropertyIsBetween

**Usage**

PropertyIsBetween

**Format**

*R6Class* object.

**Value**

Object of *R6Class* for modelling an PropertyIsBetween

**Methods**

new(PropertyName, lower, upper) This method is used to instantiate an PropertyIsBetween

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**Examples**

```r
expr <- PropertyIsBetween$new(PropertyName = "property", lower = 1, upper = 10)
expr_xml <- expr$encode() #see how it looks like in XML
```

### PropertyIsEqualTo

**Description**

PropertyIsEqualTo

**Usage**

PropertyIsEqualTo

**Format**

*R6Class* object.
Value
Object of **R6Class** for modelling an **PropertyIsEqualTo**

Methods

`new(PropertyName, Literal, matchCase)` This method is used to instantiate an **PropertyIsEqualTo**

Author(s)
Emmanuel Blondel <emmanuel.blondel1@gmail.com>

Examples
```r
expr <- PropertyIsEqualTo$new(PropertyName = "property", Literal = "value")
expr_xml <- expr$encode() # see how it looks like in XML
```

---

**PropertyIsGreaterThan**  **PropertyIsGreaterThan**

Description
**PropertyIsGreaterThan**

Usage
**PropertyIsGreaterThan**

Format

**R6Class** object.

Value
Object of **R6Class** for modelling an **PropertyIsGreaterThan**

Methods

`new(PropertyName, Literal, matchCase)` This method is used to instantiate an **PropertyIsGreaterThan**

Author(s)
Emmanuel Blondel <emmanuel.blondel1@gmail.com>

Examples
```r
expr <- PropertyIsGreaterThan$new(PropertyName = "property", Literal = "value")
expr_xml <- expr$encode() # see how it looks like in XML
```
PropertyIsGreaterThanOrEqualTo

Description

PropertyIsGreaterThanOrEqualTo

Usage

PropertyIsGreaterThanOrEqualTo

Format

R6Class object.

Value

Object of R6Class for modelling an PropertyIsGreaterThanOrEqualTo

Methods

new(PropertyName, Literal, matchCase) This method is used to instantiate an PropertyIsGreaterThanOrEqualTo

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

Examples

expr <- PropertyIsGreaterThanOrEqualTo$new(PropertyName = "property", Literal = "value")
expr_xml <- expr$encode() #see how it looks like in XML

PropertyIsLessThan

Description

PropertyIsLessThan

Usage

PropertyIsLessThan
PropertyIsLessThanOrEqualTo

Format

R6Class object.

Value

Object of R6Class for modelling an PropertyIsLessThan

Methods

new(PropertyName, Literal, matchCase) This method is used to instantiate an PropertyIsLessThan

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

Examples

expr <- PropertyIsLessThan$new(PropertyName = "property", Literal = "value")
expr_xml <- expr$encode() #see how it looks like in XML

PropertyIsLessThanOrEqualTo

PropertyIsLessThanOrEqualTo

Description

PropertyIsLessThanOrEqualTo

Usage

PropertyIsLessThanOrEqualTo

Format

R6Class object.

Value

Object of R6Class for modelling an PropertyIsLessThanOrEqualTo

Methods

new(PropertyName, Literal, matchCase) This method is used to instantiate an PropertyIsLessThanOrEqualTo

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>
PropertyIsLike

Examples

expr <- PropertyIsLessThanOrEqualTo$new(PropertyName = "property", Literal = "value")
expr_xml <- expr$encode() # see how it looks like in XML

Description

PropertyIsLike

Usage

PropertyIsLike

Format

R6Class object.

Value

Object of R6Class for modelling an PropertyIsLike

Methods

ew(PropertyOperator, PropertyName, Literal, escapeChar, singleChar, wildCard, matchCase)
This method is used to instantiate an PropertyIsLike

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

Examples

expr <- PropertyIsLike$new(PropertyName = "property", Literal = "value")
expr_xml <- expr$encode() # see how it looks like in XML
PropertyIsNotEqualTo

Description

PropertyIsNotEqualTo

Usage

PropertyIsNotEqualTo

Format

R6Class object.

Value

Object of R6Class for modelling an PropertyIsNotEqualTo

Methods

new(PropertyName, Literal, matchCase) This method is used to instantiate an PropertyIsNotEqualTo

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

Examples

expr <- PropertyIsNotEqualTo$new(PropertyName = "property", Literal = "value")
expr_xml <- expr$encode() #see how it looks like in XML

PropertyIsNull

Description

PropertyIsNull

Usage

PropertyIsNull
**UnaryLogicOpType**

**Format**

*R6Class* object.

**Value**

Object of *R6Class* for modelling an PropertyIsNull

**Methods**

`new(PropertyName)` This method is used to instantiate an PropertyIsNull

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**Examples**

```r
expr <- PropertyIsNull$new(PropertyName = "property")
expr_xml <- expr$encode() #see how it looks like in XML
```

---

**Abstract super class of all the unary logical operation classes**
**WFSCapabilities**

**Description**

WFSCapabilities

**Usage**

WFSCapabilities

**Format**

*R6Class* object.

**Value**

Object of *R6Class* with methods for interfacing an OGC Web Feature Service Get Capabilities document.

**Methods**

- `new(url, version)` This method is used to instantiate a WFSGetCapabilities object
- `getFeatureTypes(pretty)` List the feature types available. If pretty is TRUE, the output will be an object of class `data.frame`
- `findFeatureTypeByName(name, exact)` Find feature type(s) by name.

**Note**

Class used to read a WFSCapabilities document. The use of WFSClient is recommended instead to benefit from the full set of capabilities associated to a WFS server.

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**Examples**

```r
#example based on CSW endpoint responding at http://localhost:8080/geoserver/wfs
caps <- WFSCapabilities$new("http://localhost:8080/geoserver/wfs", version = "1.1.1")
```
**WFSClent**

**Description**
WFSClent

**Usage**
WFSClent

**Format**

**R6Class** object.

**Value**
Object of **R6Class** with methods for interfacing an OGC Web Feature Service.

**Methods**

`new(url, serviceVersion, user, pwd, logger)` This method is used to instantiate a WFSClent with the url of the OGC service. Authentication (user/pwd) is not yet supported and will be added with the support of service transactional modes. By default, the logger argument will be set to **NULL** (no logger). This argument accepts two possible values: **INFO**: to print only **ows4R** logs, **DEBUG**: to print more verbose logs.

`getCapabilities()` Get service capabilities. Inherited from OWS Client

`reloadCapabilities()` Reload service capabilities

`describeFeatureType(typeName)` Get the description of a given featureType

`getFeatures(typeName, ...)` Retrieves the features for a given feature type.

`getFeatureTypes(pretty)` List the feature types available. If pretty is TRUE, the output will be an object of class **data.frame**

**Author(s)**
Emmanuel Blondel <emmanuel.blondel1@gmail.com>

**Examples**

# example based on a WFS endpoint responding at http://localhost:8080/geoserver/wfs
wfs <- WFSClent$new("http://localhost:8080/geoserver/wfs", serviceVersion = "1.1.1")

# get capabilities
caps <- wfs$getCapabilities()

# find feature type
ft <- caps$findFeatureTypeByName("mylayer")
if(length(ft)>0){
  data <- ft$getFeatures()
  data_with_filter <- ft$getFeatures(cql_filter = "somefilter")
}

#Advanced examples at https://github.com/eblondel/ows4R/wiki#wfs

---

**WFSDescribeFeatureType**

**WFSDescribeFeatureType**

---

**Description**

WFSDescribeFeatureType

**Usage**

WFSDescribeFeatureType

**Format**

*R6Class* object.

**Value**

Object of *R6Class* for modelling a WFS DescribeFeatureType request

**Methods**

new(op, url, version, typeName, logger, ...) This method is used to instantiate a WFSDescribeFeatureType object

**Note**

Abstract class used by *ows4R* to trigger a WFS DescribeFeatureType request

**Author(s)**

Emmanuel Blondel <emmanuel.blondel1@gmail.com>
WFSFeatureType

Description

WFSFeatureType

Usage

WFSFeatureType

Format

R6Class object.

Value

Object of R6Class modelling a WFS feature type

Methods

new(xmlObj, capabilities, version, logger) This method is used to instantiate a WFSFeatureType object

getName() Get feature type name

title() Get feature type title

gtAbstract() Get feature type abstract

gtKeywords() Get feature type keywords

gtDefaultCrs() Get feature type default CRS

gtBoundingBox() Get feature type bounding box

description(pretty) Get feature type description. If pretty is TRUE, the output will be an object of class data.frame

gtFeatures() Get features

Note

Abstract class used by ows4R

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>
WFSFeatureTypeElement

Description

WFSFeatureTypeElement

Usage

WFSFeatureTypeElement

Format

R6Class object.

Value

Object of R6Class modelling a WFS feature type element

Methods

new(xmlObj) This method is used to instantiate a WFSFeatureTypeElement object
getMinOccurs() Get min occurs
getMaxOccurs() Get max occurs
isNillable() Get TRUE if nillable, FALSE otherwise
ggetName() Get element name
ggetType() Get element type

Note

Abstract class used by ows4R

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>
Description

WFSGetFeature

Usage

WFSGetFeature

Format

R6Class object.

Value

Object of R6Class for modelling a WFS GetFeature request

Methods

new(op, url, version, typeName, logger, ...) This method is used to instantiate a WFSGetFeature object

Note

Abstract class used by ows4R to trigger a WFS DescribeFeatureType request

Author(s)

Emmanuel Blondel <emmanuel.blondel1@gmail.com>
Index

*Topic Abstract
OGCAbstractObject, 15

*Topic And
And, 3

*Topic BBOX
BBOX, 3

*Topic BinaryComparisonOpType
BinaryComparisonOpType, 4
PropertyIsEqualTo, 27
PropertyIsGreaterThanOrEqualTo, 29
PropertyIsLessThan, 29
PropertyIsLessThanOrEqualTo, 30

*Topic BinaryLogicOpType
And, 3
BinaryLogicOpType, 5
Or, 17

*Topic CSW
CSWCapabilities, 6
CSWClient, 7
CSWDescribeRecord, 9
CSWGetRecordById, 10
CSWGetRecords, 10
CSWHarvest, 11
CSWRecordProperty, 13
CSWTxn, 14

*Topic Capabilities
CSWCapabilities, 6

*Topic Common
OWSClient, 20

*Topic DescribeFeatureType
WFSDescribeFeatureType, 36

*Topic DescribeRecord
CSWDescribeRecord, 9

*Topic Expression
And, 3
BBOX, 3
BinaryComparisonOpType, 4

*Topic Not
Not, 15

*Topic OGC

40
And, 3
BBOX, 3
BinaryComparisonOpType, 4
BinaryLogicOpType, 5
CSWCapabilities, 6
CSWClient, 7
CSWConstraint, 8
CSWDescribeRecord, 9
CSWGetRecordById, 10
CSWGetRecords, 10
CSWHarvest, 11
CSWQuery, 12
CSWTransaction, 14
Not, 15
OGCAbstractObject, 15
OGCExpression, 16
OGCFilter, 17
Or, 17
OWSCapabilities, 19
OWSClient, 20
OWSGetCapabilities, 21
OWSOperation, 21
OWSOperationsMetadata, 22
OWSRequest, 23
OWSServiceIdentification, 24
OWSServiceProvider, 25
OWSUtils, 26
PropertyIsBetween, 27
PropertyIsEqualTo, 27
PropertyIsGreaterThanOrEqualTo, 29
PropertyIsGreaterThan, 28
PropertyIsLessThan, 29
PropertyIsLessThanOrEqualTo, 30
PropertyIsLike, 31
PropertyIsNotEqualTo, 32
PropertyIsNull, 32
UnaryLogicOpType, 33
WFSCapabilities, 34
WFSClient, 35
WFSDescribeFeatureType, 36
WFSFeatureType, 37
WFSFeatureTypeElement, 38
WFSGetFeature, 39
*Topic OWS
OWSCapabilities, 19
OWSClient, 20
OWSOperation, 21
OWSOperationsMetadata, 22
OWSServiceIdentification, 24
OWSServiceProvider, 25
OWSUtils, 26
*Topic Object
OGCAbstractObject, 15
*Topic Or
Or, 17
*Topic PropertyIsBetween
PropertyIsBetween, 27
*Topic PropertyIsEqualTo
PropertyIsEqualTo, 27
*Topic PropertyIsGreaterThanOrEqualTo
PropertyIsGreaterThanOrEqualTo, 29
*Topic PropertyIsGreaterThan
PropertyIsGreaterThan, 28
*Topic PropertyIsLessThan
PropertyIsLessThan, 29
*Topic PropertyIsLessThanOrEqualTo
PropertyIsLessThanOrEqualTo, 30
*Topic PropertyIsLike
PropertyIsLike, 31
*Topic PropertyIsNotEqualTo
PropertyIsNotEqualTo, 32
*Topic PropertyIsNull
PropertyIsNull, 32
*Topic Provider
OWSServiceProvider, 25
*Topic Query
CSWQuery, 12
*Topic RecordProperty
CSWRecordProperty, 13
*Topic Request
OWSRequest, 23
*Topic Service
OWSServiceIdentification, 24
OWSServiceProvider, 25
*Topic Transaction
CSWTransaction, 14
*Topic UnaryLogicOpType
Not, 15
UnaryLogicOpType, 33
*Topic Utils
OWSUtils, 26
*Topic WFS
WFSCapabilities, 34