Package ‘dataRetrieval’

July 30, 2021

Type Package

Title Retrieval Functions for USGS and EPA Hydrologic and Water Quality Data

Version 2.7.9


License CC0

Copyright This software is in the public domain because it contains materials that originally came from the United States Geological Survey, an agency of the United States Department of Interior.

Depends R (>= 3.5.0)

Imports httr (>= 1.0.0), curl, lubridate (>= 1.5.0), stats, utils, xml2, readr (>= 1.0.0), jsonlite

Suggests covr, dplyr, ggplot2, DT, gridExtra, knitr, rmarkdown, sf, testthat

Encoding UTF-8

BuildVignettes true

VignetteBuilder knitr

BugReports https://github.com/USGS-R/dataRetrieval/issues

URL https://pubs.usgs.gov/tm/04/a10/

RoxygenNote 7.1.1

NeedsCompilation no

Author Laura DeCicco [aut, cre] (<https://orcid.org/0000-0002-3915-9487>), Robert Hirsch [aut] (<https://orcid.org/0000-0002-4534-075X>),
R topics documented:

David Lorenz [aut],
Jordan Read [ctb],
Jordan Walker [ctb],
Lindsay Carr [ctb],
David Watkins [aut] (<https://orcid.org/0000-0002-7544-0700>),
David Blodgett [ctb],
Mike Johnson [aut] (<https://orcid.org/0000-0002-5288-8350>)

Maintainer Laura DeCicco <ldecicco@usgs.gov>

Repository CRAN

Date/Publication  2021-07-30 15:00:02 UTC

R topics documented:

addWaterYear ........................................ 3
calcWaterYear ....................................... 4
constructNWISURL .................................... 4
constructUseURL ...................................... 6
constructWQPURL ..................................... 7
countyCd ............................................ 8
countyCdLookup ..................................... 8
dataRetrieval ......................................... 9
findNLDI ............................................. 9
getQuerySummary ..................................... 11
getWebServiceData ................................... 12
get_nldi_sources .................................... 13
importNGWMN ......................................... 13
importRDB1 .......................................... 14
importWaterML1 ..................................... 16
importWaterML2 ..................................... 18
importWQP ............................................ 19
parameterCdFile ..................................... 20
readNGWMNdata ..................................... 20
readNGWMNlevels ................................... 22
readNGWMNsites .................................... 23
readNWISdata ....................................... 24
readNWISdv .......................................... 26
readNWISgw1 ....................................... 28
readNWISmeas ....................................... 30
readNWISpCode ..................................... 32
readNWISpeak ....................................... 33
readNWISqw ......................................... 34
readNWISrating ..................................... 37
readNWISsite ........................................ 38
readNWISstat ........................................ 40
readNWISuse ......................................... 41
readNWISuv ......................................... 43
readWQPdata ......................................... 45
addWaterYear

Description

Add a column to the dataRetrieval data frame with the water year. WQP queries will return a water year column for the start and end dates of the data.

Usage

addWaterYear(rawData)

Arguments

rawData

the daily- or unit-values dataset retrieved from NWISweb. Must have at least one of the following columns to add the new water year columns: 'dateTime', 'Date', 'ActivityStartDate', or 'ActivityEndDate'. The date column(s) can be character, POSIXct, Date. They cannot be numeric.

Value

data.frame with an additional integer column with "WY" appended to the date column name. For WQP, there will be 2 columns: 'ActivityStartDateWY' and 'ActivityEndDateWY'.

Examples

nwisData <- readNWISdv('04085427','00060','2012-01-01','2012-06-30')
nwisData <- addWaterYear(nwisData)

wqpData <- readWQPqw('USGS-01594440','01075','','')
wqpData <- addWaterYear(wqpData)
### calcWaterYear

**Extract WY from a date**

**Description**

Determine the correct water year based on a calendar date.

**Usage**

\[
\text{calcWaterYear}(\text{dateVec})
\]

**Arguments**

- **dateVec**
  - vector of dates as character ("YYYY-DD-MM"), Date, or POSIXct. Numeric does not work.

**Details**

This function calculates a water year based on the USGS definition that a water year starts on October 1 of the year before, and ends on September 30. For example, water year 2015 started on 2014-10-01 and ended on 2015-09-30. See the USGS definition at [https://water.usgs.gov/nwc/explain_data.html](https://water.usgs.gov/nwc/explain_data.html).

**Value**

numeric vector indicating the water year

**Examples**

\[
x \leftarrow \text{seq}(\text{as.Date}("2010-01-01"), \text{as.Date}("2010-12-31"), \text{by}="month")
\]

\[
\text{calcWaterYear}(x)
\]

\[
y \leftarrow \text{c}("2010-01-01", "1994-02", "1980", "2009-11-01", \text{NA})
\]

\[
\text{calcWaterYear}(y)
\]

---

### constructNWISURL

**Construct NWIS url for data retrieval**

**Description**

Imports data from NWIS web service. This function gets the data from here: [https://nwis.waterdata.usgs.gov/nwis/qwdata](https://nwis.waterdata.usgs.gov/nwis/qwdata) A list of parameter codes can be found here: [https://nwis.waterdata.usgs.gov/nwis/pmcodes/](https://nwis.waterdata.usgs.gov/nwis/pmcodes/) A list of statistic codes can be found here: [https://nwis.waterdata.usgs.gov/nwis/help/?read_file=stat&format=table](https://nwis.waterdata.usgs.gov/nwis/help/?read_file=stat&format=table)
Usage

```
constructNWISURL(
    siteNumbers,
    parameterCd = "00060",
    startDate = "",
    endDate = "",
    service,
    statCd = "00003",
    format = "xml",
    expanded = TRUE,
    ratingType = "base",
    statReportType = "daily",
    statType = "mean"
)
```

Arguments

- **siteNumbers**: string or vector of strings USGS site number. This is usually an 8 digit number.
- **parameterCd**: string or vector of USGS parameter code. This is usually an 5 digit number.
- **startDate**: character starting date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the earliest possible record.
- **endDate**: character ending date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the latest possible record.
- **service**: string USGS service to call. Possible values are "dv" (daily values), "uv" (unit/instantaneous values), "qw" (water quality data), "gwlevels" (groundwater), and "rating" (rating curve), "peak", "meas" (discrete streamflow measurements), "stat" (statistics web service BETA).
- **statCd**: string or vector USGS statistic code only used for daily value service. This is usually 5 digits. Daily mean (00003) is the default.
- **format**: string, can be "tsv" or "xml", and is only applicable for daily and unit value requests. "tsv" returns results faster, but there is a possibility that an incomplete file is returned without warning. XML is slower, but will offer a warning if the file was incomplete (for example, if there was a momentary problem with the internet connection). It is possible to safely use the "tsv" option, but the user must carefully check the results to see if the data returned matches what is expected. The default is therefore "xml".
- **expanded**: logical defaults to TRUE. If TRUE, retrieves additional information, only applicable for qw data.
- **ratingType**: can be "base", "corr", or "exsa". Only applies to rating curve data.
- **statReportType**: character Only used for statistics service requests. Time division for statistics: daily, monthly, or annual. Default is daily. Note that daily provides statistics for each calendar day over the specified range of water years, i.e. no more than 366 data points will be returned for each site/parameter. Use readNWISdata or readNWISdv for daily averages. Also note that 'annual' returns statistics for the calendar year. Use readNWISdata for water years. Monthly and yearly provide statistics for each month and year within the range individually.
constructUseURL

Construct URL for NWIS water use data service

**Description**
Reconstructs URLs to retrieve data from here: [https://waterdata.usgs.gov/nwis/wu](https://waterdata.usgs.gov/nwis/wu)

**Usage**
`constructUseURL(years, stateCd, countyCd, categories)`

**Arguments**
- `years` integer Years for data retrieval. Must be years ending in 0 or 5, or "ALL", which retrieves all available years.
- `stateCd` could be character (full name, abbreviation, id), or numeric (id)
- `countyCd` could be numeric (County IDs from countyCdLookup) or character ("ALL")
- `categories` character Two-letter category abbreviation(s)
constructWQPURL

Value
url string

Examples

url <- constructUseURL(years=c(1990,1995),stateCd="Ohio",countyCd = c(1,3), categories = "ALL")

constructWQPURL

Construct WQP url for data retrieval

Description

Construct WQP url for data retrieval. This function gets the data from here: https://www.waterqualitydata.us

Usage

constructWQPURL(siteNumbers, parameterCd, startDate, endDate, zip = TRUE)

Arguments

siteNumbers string or vector of strings USGS site number. This is usually an 8 digit number
parameterCd string or vector of USGS parameter code. This is usually an 5 digit number.
startDate character starting date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the earliest possible record.
endDate character ending date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the latest possible record.
zip logical to request data via downloading zip file. Default set to TRUE.

Value
url string

Examples

site_id <- '01594440'
startDate <- '1985-01-01'
endDate <- '',
pCode <- c("00060","00010")
url_wqp <- constructWQPURL(paste("USGS",site_id,sep="-"),
c('01075','00029','00453'),
startDate,endDate)
url_wqp
charNames <- c("Temperature",
"Temperature, sample",
"Temperature, water",
"Temperature, ground")
countyCdLookup

"Temperature, water, deg F"

obs_url_orig <- constructWQPURL(siteNumbers = c("IIDFG-41WSSPAHS", "USGS-02352560"),
                               parameterCd = charNames,
                               startDate,"

obs_url_orig

countyCd

US County Code Lookup Table

Description

Data pulled from https://www2.census.gov/geo/docs/reference/codes/files/national_county.txt on April 1, 2015.

Value

countyCd data frame.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STUSAB</td>
<td>character</td>
<td>State abbreviation</td>
</tr>
<tr>
<td>STATE</td>
<td>character</td>
<td>two-digit ANSI code</td>
</tr>
<tr>
<td>COUNTY</td>
<td>character</td>
<td>three-digit county code</td>
</tr>
<tr>
<td>COUNTY_NAME</td>
<td>character</td>
<td>County full name</td>
</tr>
<tr>
<td>COUNTY_ID</td>
<td>character</td>
<td>County id</td>
</tr>
</tbody>
</table>

Examples

head(countyCd)

countyCdLookup

County code look up

Description

Function to simplify finding county and county code definitions. Used in readNWISdata and readNWISuse.

Usage

countyCdLookup(state, county, outputType = "id")
Arguments

state could be character (full name, abbreviation, id), or numeric (id)
county could be character (name, with or without "County") or numeric (id)
outputType character can be "fullName", "tableIndex", "id", or "fullEntry".

Examples

id <- countyCdLookup(state = "WI", county = "Dane")
name <- countyCdLookup(state = "OH", county = 13, output = "fullName")
index <- countyCdLookup(state = "Pennsylvania", county = "ALLEGHENY COUNTY", output = "tableIndex")
fromIDs <- countyCdLookup(state = 13, county = 5, output = "fullName")
already_correct <- countyCdLookup(county = "51001")

dataRetrieval Retrieval functions for USGS and EPA data

Description

Package: dataRetrieval
Type: Package
License: Unlimited for this package, dependencies have more restrictive licensing.
Copyright: This software is in the public domain because it contains materials that originally came from the United States Government.
LazyLoad: yes

Details

Retrieval functions for USGS and EPA hydrologic and water quality data.
Please see https://pubs.er.usgs.gov/publication/tm4A10 for more information.

Author(s)

Laura De Cicco <ldecicco@usgs.gov>

findNLDI R Client for the Network Linked Data Index

Description

Provides a formal client to the USGS Network Linked Data Index.
Usage

```r
findNLDI(
  comid = NULL,
  nwis = NULL,
  wqp = NULL,
  huc12 = NULL,
  location = NULL,
  origin = NULL,
  nav = NULL,
  find = c("flowlines"),
  distance_km = 100,
  no_sf = FALSE
)
```

Arguments

- **comid**: numeric or character. An NHDPlusV2 COMID
- **nwis**: numeric or character. A USGS NWIS surface water siteID
- **wqp**: numeric or character. A water quality point ID
- **huc12**: numeric or character. A WBD HUC12 unit ID
- **location**: numeric vector. Coordinate pair in WGS84 SRS ordered lng/lat (X,Y)
- **origin**: named list. Specifying a feature type and ID (e.g. list("comid" = 101))
- **nav**: character vector. Where to navigate from the starting point. Options include along the upper mainstream (UM), upstream tributary (UT), downstream mainstream (DM) and downstream divergences (DD). You may select one or more of the abbreviations ("UM", "UT", DM", "DD").
- **find**: character vector. Define what resources to find along the navigation path(s) (see get_nldi_sources()$source). Can also include 'basin' or 'flowline', which will return the upstream basin of the starting feature or flowlines along the navigation respectively. The default is "flowlines". If you provide any other resource, AND want flowlines, then flowlines must be explicitly requested.
- **distance_km**: numeric. Define how far to look along the navigation path in kilometers (default = 100)
- **no_sf**: if available, should ‘sf’ be used for parsing, defaults to ‘TRUE’ if ‘sf’ is locally installed

Details

The function is useful for topology and location based feature discovery. A user must specify an origin feature, optional navigation direction(s) along the network, as well as features to identify along the navigated paths. Valid starting options can be given by one of the following arguments: comid, nwis, huc12, wqp, location, and start.

Value

A list of data.frames if sf is not installed, a list of sf objects if it is

---

**findNLDI**
### Examples

```r
# Find Features / Define origin features

## Find feature by COMID
findNLDI(comid = 101)

## Find feature by NWIS ID
findNLDI(nwis = '11120000')

## Find feature by WQP ID
findNLDI(wqp = 'USGS-04024315')

## Find feature by LOCATION
findNLDI(location = c(-115,40))

## GENERAL ORIGIN: COMID
findNLDI(origin = list("comid" = 101))

## GENERAL ORIGIN: WaDE
findNLDI(origin = list("wade" = 'CA_45206'))

# Navigation (flowlines will be returned if find is unspecified)
# UPPER MAINSTEM of USGS-11120000
findNLDI(nwis = '11120000', nav = "UM")

# MULTI-REQUEST
# UPPER MAINSTEM and TRIBUTARY of USGS-11120000
findNLDI(nwis = '11120000', nav = c("UT", "UM"))

# Discover Features(flowlines will not be returned unless included in find)

## Find feature(s) on the upper tributary of USGS-11120000
findNLDI(nwis = '11120000', nav = "UT", find = c("nwis", "wqp"))

## Find upstream basin boundary and of USGS-11120000
findNLDI(nwis = '11120000', find = "basin")

# Control Distance
## Limit search to 50 km
findNLDI(comid = 101, nav = "DM", find = c("nwis", "wqp", "flowlines"), distance_km = 50)
```

### Description

Getting header information from a WQP query.
Usage

getQuerySummary(url)

Arguments

url the query url

getWebServiceData  Function to return data from web services

Description

This function accepts a url parameter, and returns the raw data. The function enhances GET with more informative error messages.

Usage

getWebServiceData(obs_url, ...)

Arguments

obs_url character containing the url for the retrieval
... information to pass to header request

Value

raw data from web services

Examples

siteNumber <- "02177000"
startDate <- "2012-09-01"
endDate <- "2012-10-01"
offering <- '00003'
property <- '00060'
obs_url <- constructNWISURL(siteNumber,property,startDate,endDate,'dv')

rawData <- getWebServiceData(obs_url)
get_nldi_sources  

Get current NLDI offerings

Description

Used to query the current resources available through the NLDI

Usage

get_nldi_sources()

Value

data.frame

Examples

get_nldi_sources()

importNGWMN  

Function to return data from the National Ground Water Monitoring Network waterML2 format

Description

This function accepts a url parameter for a WaterML2 getObservation. This function is still under development, but the general functionality is correct.

Usage

importNGWMN(input, asDateTime = FALSE, tz = "UTC")

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>character or raw, containing the url for the retrieval or a path to the data file, or raw XML.</td>
</tr>
<tr>
<td>asDateTime</td>
<td>logical, if TRUE returns date and time as POSIXct, if FALSE, character</td>
</tr>
<tr>
<td>tz</td>
<td>character to set timezone attribute of dateTime. Default is &quot;UTC&quot;, and converts the date times to UTC, properly accounting for daylight savings times based on the data's provided time zone offset. Possible values to provide are &quot;America/New_York&quot;,&quot;America/Chicago&quot;,&quot;America/Denver&quot;,&quot;America/Los_Angeles&quot;, &quot;America/Anchorage&quot;, as well as the following which do not use daylight savings time: &quot;America/Honolulu&quot;,&quot;America/Jamaica&quot;, &quot;America/Managua&quot;,&quot;America/Phoenix&quot;, and &quot;America/Metlakatla&quot;. See also OlsonNames() for more information on time zones.</td>
</tr>
</tbody>
</table>
importRDB1

Value
mergedDF a data frame source, time, value, uom, uomTitle, comment, gmlID

Examples

"service=SOS","version=2.0.0", 
"observedProperty=urn:ogc:def:property:OGC:GroundWaterLevel", 
"responseFormat=text/xml", 
"featureOfInterest=VW_GWDP_GEOSERVER.USGS.403836085374401",sep="&")
data <- importNGWMN(obs_url)

importRDB1  Function to return data from the NWIS RDB 1.0 format

Description
This function accepts a url parameter that already contains the desired NWIS site, parameter code, statistic, startdate and enddate. It is not recommended to use the RDB format for importing multi-site data.

Usage

importRDB1(obs_url, asDateTime = TRUE, convertType = TRUE, tz = "UTC")

Arguments

obs_url  character containing the url for the retrieval or a file path to the data file.
asDateTime  logical, if TRUE returns date and time as POSIXct, if FALSE, Date
convertType  logical, defaults to TRUE. If TRUE, the function will convert the data to dates, datetimes, numerics based on a standard algorithm. If false, everything is returned as a character
tz  character to set timezone attribute of datatime. Default converts the datetimes to UTC (properly accounting for daylight savings times based on the data’s provided tz_cd column). Recommended US values include "UTC","America/New_York","America/Chicago","America/Denver","America/Los_Angeles", "America/Anchorage","America/Honolulu","America/Jamaica","America/Phoenix", and "America/Metlakatla". For a complete list, see https://en.wikipedia.org/wiki/List_of_tz_database_time_zones
importRDB1

Value

A data frame with the following columns:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>agency_cd</td>
<td>character</td>
<td>The NWIS code for the agency reporting the data</td>
</tr>
<tr>
<td>site_no</td>
<td>character</td>
<td>The USGS site number</td>
</tr>
<tr>
<td>datetime</td>
<td>POSIXct</td>
<td>The date and time of the value converted to UTC (if asDateTime = TRUE) or raw character string (if asDateTime = FALSE)</td>
</tr>
<tr>
<td>tz_cd</td>
<td>character</td>
<td>The time zone code for datetime</td>
</tr>
<tr>
<td>code</td>
<td>character</td>
<td>Any codes that qualify the corresponding value</td>
</tr>
<tr>
<td>value</td>
<td>numeric</td>
<td>The numeric value for the parameter</td>
</tr>
<tr>
<td>tz_cd_reported</td>
<td></td>
<td>The originally reported time zone</td>
</tr>
</tbody>
</table>

Note that code and value are repeated for the parameters requested. The names are of the form XD_P_S, where X is literal, D is an option description of the parameter, P is the parameter code, and S is the statistic code (if applicable). If a date/time (dt) column contained incomplete date and times, a new column of dates and time was inserted. This could happen when older data was reported as dates, and newer data was reported as a date/time.

There are also several useful attributes attached to the data frame:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>character</td>
<td>The url used to generate the data</td>
</tr>
<tr>
<td>queryTime</td>
<td>POSIXct</td>
<td>The time the data was returned</td>
</tr>
<tr>
<td>comment</td>
<td>character</td>
<td>Header comments from the RDB file</td>
</tr>
</tbody>
</table>

Examples

```r
site_id <- "02177000"
startDate <- "2012-09-01"
endDate <- "2012-10-01"
offering <- "00003"
property <- "00060"

obs_url <- constructNWISURL(site_id,property,
                             startDate,endDate,"dv",format="tsv")
data <- importRDB1(obs_url)

urlMultiPcodes <- constructNWISURL("04085427",c("00060","00010"),
                                   startDate,endDate,"dv",statCd=c("00003","00001"),"tsv")
multiData <- importRDB1(urlMultiPcodes)

unitDataURL <- constructNWISURL(site_id,property,
                                 "2020-10-30","2020-11-01","uv",format="tsv") #includes timezone switch
unitData <- importRDB1(unitDataURL, asDateTime=TRUE)
qwURL <- constructNWISURL(c("04024430","04024000"),
                          c("34247","30234","32104","34220"),
                          "2010-11-03","","qw",format="rdb")
```
importWaterML1 <- function(obs_url, asDateTime = FALSE, tz = "UTC")

Description
This function accepts a url parameter that already contains the desired NWIS site, parameter code, statistic, startdate and enddate.

Usage
importWaterML1(obs_url, asDateTime = FALSE, tz = "UTC")

Arguments
- obs_url: character or raw, containing the url for the retrieval or a file path to the data file, or raw XML.
- asDateTime: logical, if TRUE returns date and time as POSIXct, if FALSE, Date
- tz: character to set timezone attribute of datetime. Default converts the datetimes to UTC (properly accounting for daylight savings times based on the data’s provided tz_cd column). Recommended US values include "UTC","America/New_York","America/Chicago","America/Denver","America/Los_Angeles","America/Anchorage","America/Honolulu","America/Jamaica" and "America/Metlakatla". For a complete list, see https://en.wikipedia.org/wiki/List_of_tz_database_time_zones

Value
A data frame with the following columns:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>agency_cd</td>
<td>character</td>
<td>The NWIS code for the agency reporting the data</td>
</tr>
<tr>
<td>site_no</td>
<td>character</td>
<td>The USGS site number</td>
</tr>
<tr>
<td>POSIXct</td>
<td>character</td>
<td>The date and time of the value converted to UTC (if asDateTime = TRUE), or raw character string (if asDateTime = FALSE)</td>
</tr>
</tbody>
</table>
tz_cd character The time zone code for
code character Any codes that qualify the corresponding value
code numeric The numeric value for the parameter

Note that code and value are repeated for the parameters requested. The names are of the form X_D_P_S, where X is literal, D is an option description of the parameter, P is the parameter code, and S is the statistic code (if applicable).

There are also several useful attributes attached to the data frame:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>character</td>
<td>The url used to generate the data</td>
</tr>
<tr>
<td>siteInfo</td>
<td>data.frame</td>
<td>A data frame containing information on the requested sites</td>
</tr>
<tr>
<td>variableInfo</td>
<td>data.frame</td>
<td>A data frame containing information on the requested parameters</td>
</tr>
<tr>
<td>statisticInfo</td>
<td>data.frame</td>
<td>A data frame containing information on the requested statistics on the data</td>
</tr>
<tr>
<td>queryTime</td>
<td>POSIXct</td>
<td>The time the data was returned</td>
</tr>
</tbody>
</table>

See Also

renameNWISColumns

Examples

```r
site_id <- "02177000"
startDate <- "2012-09-01"
endDate <- "2012-10-01"
offering <- '00003'
property <- '00060'
obs_url <- constructNWISURL(site_id,property,startDate,endDate, service="dv")
data <- importWaterML1(obs_url, asDateTime=TRUE)

groundWaterSite <- "431049071324301"
startGW <- "2013-10-01"
endGW <- "2014-06-30"
groundwaterExampleURL <- constructNWISURL(groundWaterSite, NA, startGW,endGW, service="gwlevels")
groundWater <- importWaterML1(groundwaterExampleURL)
groundWater2 <- importWaterML1(groundwaterExampleURL, asDateTime=TRUE)

unitDataURL <- constructNWISURL(site_id,property,
                                 "2013-11-03","2013-11-03","uv")
unitData <- importWaterML1(unitDataURL,TRUE)

# Two sites, two pcodes, one site has two data descriptors:
siteNumber <- c('01480015','04085427')
obs_url <- constructNWISURL(siteNumber,c("00060","00010"),startDate,endDate,'dv')
data <- importWaterML1(obs_url)
data$dateTime <- as.Date(data$dateTime)
```
importWaterML2

Parse the WaterML2 timeseries portion of a waterML2 file

Description

Returns data frame columns of all information with each time series measurement; Anything de-

fined as a default, is returned as an attribute of that data frame.

Usage

importWaterML2(input, asDateTime = FALSE, tz = "UTC")

Arguments

input XML with only the wml2:MeasurementTimeseries node and children
asDateTime logical, if TRUE returns date and time as POSIXct, if FALSE, character
tz character to set timezone attribute of datetime. Default is an empty quote, which
converts the datetimes to UTC (properly accounting for daylight savings times
based on the data's provided time zone offset). Possible values are "America/New_York", "America/Chicago", "America/Denver", "America/Los_Angeles", "UTC".
importWQP

"America/Anchorage", "America/Honolulu", "America/Jamaica", "America/Managua", "America/Phoenix", and "America/Metlakatla"

Examples

```r
baseURL <- "https://waterservices.usgs.gov/nwis/dv/?format=waterml,2.0"
URL <- paste(baseURL, "sites=01646500",
        "startDT=2014-09-01",
        "endDT=2014-09-08",
        "statCd=00003",
        "parameterCd=00060", sep="&")
timesereies <- importWaterML2(URL, asDateTime=TRUE, tz="UTC")
```

importWQP

**Basic Water Quality Portal Data parser**

Description

Imports data from the Water Quality Portal based on a specified url.

Usage

```r
importWQP(obs_url, zip = TRUE, tz = "UTC", csv = FALSE)
```

Arguments

- **obs_url** character URL to Water Quality Portal
- **zip** logical to request data via downloading zip file. Default set to TRUE.
- **tz** character to set timezone attribute of datetime. Default is UTC (properly accounting for daylight savings times based on the data’s provided tz_cd column). Possible values include "America/New_York", "America/Chicago", "America/Denver", "America/Los_Angeles", "America/Anchorage", "America/Honolulu", "America/Jamaica", "America/Managua", "America/Phoenix", and "America/Metlakatla"
- **csv** logical. Is the data coming back with a csv or tsv format. Default is FALSE. Currently, the summary service does not support tsv, for other services tsv is the safer choice.

Value

.retval dataframe raw data returned from the Water Quality Portal. Additionally, a POSIXct dateTime column is supplied for start and end times, and converted to UTC. See [https://www.waterqualitydata.us/portal_userguide/](https://www.waterqualitydata.us/portal_userguide/) for more information.
See Also

readWQPdata, readWQPqw, whatWQPsites

Examples

# These examples require an internet connection to run

## Examples take longer than 5 seconds:

rawSampleURL <- constructWQPURL('USGS-01594440','01075','','')
rawSample <- importWQP(rawSampleURL)

rawSampleURL_NoZip <- constructWQPURL('USGS-01594440','01075','','', zip=FALSE)
rawSample2 <- importWQP(rawSampleURL_NoZip, zip=FALSE)

STORETex <- constructWQPURL('WIDNR_WQX-10032762','Specific conductance','','')
STORETdata <- importWQP(STORETex)

---

parameterCdFile List of USGS parameter codes

Description


Value

parameterData data frame with information about USGS parameters.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>parameter_cd</td>
<td>character</td>
<td>5-digit USGS parameter code</td>
</tr>
<tr>
<td>parameter_group_nm</td>
<td>character</td>
<td>USGS parameter group name</td>
</tr>
<tr>
<td>parameter_nm</td>
<td>character</td>
<td>USGS parameter name</td>
</tr>
<tr>
<td>casrn</td>
<td>character</td>
<td>Chemical Abstracts Service (CAS) Registry Number</td>
</tr>
<tr>
<td>srsname</td>
<td>character</td>
<td>Substance Registry Services Name</td>
</tr>
<tr>
<td>parameter_units</td>
<td>character</td>
<td>Parameter units</td>
</tr>
</tbody>
</table>

Examples

head(parameterCdFile[,1:2])

---

readNGWMNdata import data from the National Groundwater Monitoring Network

readNGWMNdata

Description

Only water level data and site locations and names are currently available through the web service.

Usage

readNGWMNdata(service, ..., asDateTime = TRUE, tz = "UTC")

Arguments

service
char Service for the request - "observation" and "featureOfInterest" are implemented.

... Other parameters to supply, namely siteNumbers or bbox

asDateTime logical if TRUE, will convert times to POSIXct format. Currently defaults to FALSE since time zone information is not included.

tz character to set timezone attribute of dateTime. Default is "UTC", and converts the date times to UTC, properly accounting for daylight savings times based on the data’s provided time zone offset. Possible values to provide are "America/New_York", "America/Chicago", "America/Denver", "America/Los_Angeles", "America/Anchorage", as well as the following which do not use daylight savings time: "America/Honolulu", "America/Jamaica", "America/Managua", "America/Phoenix", and "America/Metlakatla". See also OlsonNames() for more information on time zones.

Examples

# one site
site <- "USGS.430427089284901"
# oneSite <- readNGWMNdata(siteNumbers = site, service = "observation")

# multiple sites
sites <- c("USGS.272838082142201", "USGS.404159100494601", "USGS.401216080362703")
# Very slow:
# multiSiteData <- readNGWMNdata(siteNumbers = sites, service = "observation")
# attributes(multiSiteData)

# non-USGS site
# accepts colon or period between agency and ID
site <- "MBMG:702934"
# data <- readNGWMNdata(siteNumbers = site, service = "featureOfInterest")

# site with no data returns empty data frame
noDataSite <- "UTGS.401544112060301"
# noDataSite <- readNGWMNdata(siteNumbers = noDataSite, service = "observation")

# bounding box
#bboxSites <- readNGWMNdata(service = "featureOfInterest", bbox = c(30, -102, 31, 99))
# retrieve sites. Set asDateTime to false since one site has an invalid date
# Very slow:
#bboxData <- readNGWMNdata(service = "observation", siteNumbers = bboxSites$site[1:3],
readNGWMNlevels

Retrieve groundwater levels from the National Ground Water Monitoring Network [https://cida.usgs.gov/ngwmn/](https://cida.usgs.gov/ngwmn/).

**Description**

Retrieve groundwater levels from the National Ground Water Monitoring Network [https://cida.usgs.gov/ngwmn/](https://cida.usgs.gov/ngwmn/).

**Usage**

```r
readNGWMNlevels(siteNumbers, asDateTime = TRUE, tz = "UTC")
```

**Arguments**

- `siteNumbers` character Vector of feature IDs formatted with agency code and site number separated by a period or semicolon, e.g. USGS.404159100494601.
- `asDateTime` logical Should dates and times be converted to date/time objects, or returned as character? Defaults to TRUE. Must be set to FALSE if a site contains non-standard dates.
- `tz` character to set timezone attribute of dateTime. Default is "UTC", and converts the date times to UTC, properly accounting for daylight savings times based on the data's provided time zone offset. Possible values to provide are "America/New_York","America/Chicago", "America/Denver", "America/Los_Angeles", "America/Anchorage", as well as the following which do not use daylight savings time: "America/Honolulu", "America/Jamaica", "America/Managua", "America/Phoenix", and "America/Metlakatla". See also OlsonNames() for more information on time zones.

**Examples**

```r
#one site
site <- "USGS.430427089284901"
#oneSite <- readNGWMNlevels(siteNumbers = site)

#multiple sites
sites <- c("USGS:272838082142201", "USGS:404159100494601", "USGS:401216080362703")
#multiSiteData <- readNGWMNlevels(sites)

#non-USGS site
site <- "MBMG.103306"
#data <- readNGWMNlevels(siteNumbers = site, asDateTime = FALSE)
```
#site with no data returns empty data frame
noDataSite <- "UTGS.401544112060301"
#noDataSite <- readNGWMNlevels(siteNumbers = noDataSite)

---

**readNGWMNsites**


**Description**


**Usage**

readNGWMNsites(siteNumbers)

**Arguments**

- **siteNumbers** character Vector of feature IDs formatted with agency code and site number separated by a period or semicolon, e.g. USGS.404159100494601.

**Value**

A data frame the following columns: #'

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>site</td>
<td>char</td>
<td>Site FID</td>
</tr>
<tr>
<td>description</td>
<td>char</td>
<td>Site description</td>
</tr>
<tr>
<td>dec_lat_va, dec_lon_va</td>
<td>numeric</td>
<td>Site latitude and longitude</td>
</tr>
</tbody>
</table>

**Examples**

#one site
site <- "USGS.430427089284901"
oneSite <- readNGWMNsites(siteNumbers = site)

#non-USGS site
site <- "MBMG.103306"
siteInfo <- readNGWMNsites(siteNumbers = site)
readNWISdata

General Data Import from NWIS

Description

Returns data from the NWIS web service. Arguments to the function should be based on https://waterservices.usgs.gov service calls. See examples below for ideas of constructing queries.

Usage

readNWISdata(..., asDateTime = TRUE, convertType = TRUE, tz = "UTC")

Arguments

... see https://waterservices.usgs.gov/rest/Site-Service.html for a complete list of options. A list of arguments can also be supplied. One important argument to include is 'service'. Possible values are "iv" (for instantaneous), "iv_recent" (for instantaneous values within the last 120 days), "dv" (for daily values), "gwlevels" (for groundwater levels), "site" (for site service), "qw" (water-quality), "measurement", and "stat" (for statistics service). Note: "qw" and "measurement" calls go to: https://nwis.waterdata.usgs.gov/usa/nwis for data requests, and use different call requests schemes. The statistics service has a limited selection of arguments (see https://waterservices.usgs.gov/rest/Statistics-Service-Test-Tool.html).

asDateTime logical, if TRUE returns date and time as POSIXct, if FALSE, Date

convertType logical, defaults to TRUE. If TRUE, the function will convert the data to dates, datetimes, numerics based on a standard algorithm. If false, everything is returned as a character

tz character to set timezone attribute of dateTime. Default is "UTC", and converts the date times to UTC, properly accounting for daylight savings times based on the data's provided tz_cd column. Possible values to provide are "America/New_York","America/Chicago","America/Denver","America/Los_Angeles", "America/Anchorage", as well as the following which do not use daylight savings time: "America/Honolulu","America/Jamaica","America/Managua","America/Phoenix", and "America/Metlakatla". See also OlsonNames() for more information on time zones.

Value

A data frame with the following columns:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>agency</td>
<td>character</td>
<td>The NWIS code for the agency reporting the data</td>
</tr>
<tr>
<td>site</td>
<td>character</td>
<td>The USGS site number</td>
</tr>
<tr>
<td>dateTime</td>
<td>POSIXct</td>
<td>The date and time (if applicable) of the measurement, converted to UTC for unit value data. R only all...</td>
</tr>
<tr>
<td>tz_cd</td>
<td>character</td>
<td>The time zone code for dateTime column</td>
</tr>
</tbody>
</table>
The function `readNWISdata` is used to retrieve data from the NWIS database. The function accepts several arguments:

- `code` (character): Any codes that qualify the corresponding value.
- `value` (numeric): The numeric value for the parameter.

Note that code and value are repeated for the parameters requested. The names are of the form `X_D_P_S`, where `X` is literal, `D` is an option description of the parameter, `P` is the parameter code, and `S` is the statistic code (if applicable).

There are also several useful attributes attached to the data frame:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>character</td>
<td>The url used to generate the data</td>
</tr>
<tr>
<td>siteInfo</td>
<td>data.frame</td>
<td>A data frame containing information on the requested sites</td>
</tr>
<tr>
<td>variableInfo</td>
<td>data.frame</td>
<td>A data frame containing information on the requested parameters</td>
</tr>
<tr>
<td>statisticInfo</td>
<td>data.frame</td>
<td>A data frame containing information on the requested statistics on the data</td>
</tr>
<tr>
<td>queryTime</td>
<td>POSIXct</td>
<td>The time the data was returned</td>
</tr>
</tbody>
</table>

**See Also**

`renameNWISColumns`, `importWaterML1`, `importRDB1`

**Examples**

```r
# Examples not run for time considerations
dataTemp <- readNWISdata(stateCd="OH", parameterCd="00010", service="dv")
instFlow <- readNWISdata(sites="05114000", service="iv",
                          parameterCd="00060",
                          startDate="2014-05-01T00:00Z", endDate="2014-05-01T12:00Z")

instFlowCDT <- readNWISdata(sites="05114000", service="iv",
                            parameterCd="00060",
                            startDate="2014-05-01T00:00", endDate="2014-05-01T12:00",
                            tz="America/Chicago")

#Empty:
multiSite <- readNWISdata(sites=c("04025000","04072150"), service="iv",
                           parameterCd="00010")

#Not empty:
multiSite <- readNWISdata(sites=c("04025500","040263491"),
                           service="iv", parameterCd="00060")
bBoxEx <- readNWISdata(bBox=c(-83,36.5,-81,38.5), parameterCd="00010")

startDate <- as.Date("2013-10-01")
endDate <- as.Date("2014-09-30")
waterYear <- readNWISdata(bBox=c(-83,36.5,-81,38.5), parameterCd="00010",
                          service="dv", startDate=startDate, endDate=endDate)
siteInfo <- readNWISdata(stateCd="WI", parameterCd="00010",
                          hasDataTypeCd="iv", service="site")
temp <- readNWISdata(bBox=c(-83,36.5,-81,38.5), parameterCd="00010", service="site",
```
readNWISdv

Daily Value USGS NWIS Data Retrieval
Description

Imports data from NWIS web service. This function gets the data from here: https://waterservices.usgs.gov/

Usage

```r
readNWISdv(
  siteNumbers, 
  parameterCd, 
  startDate = "", 
  endDate = "", 
  statCd = "00003"
)
```

Arguments

- `siteNumbers` character USGS site number. This is usually an 8 digit number. Multiple sites can be requested with a character vector.
- `parameterCd` character of USGS parameter code(s). This is usually an 5 digit number.
- `startDate` character starting date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the earliest possible record. Date arguments are always specified in local time.
- `endDate` character ending date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the latest possible record. Date arguments are always specified in local time.
- `statCd` character USGS statistic code. This is usually 5 digits. Daily mean (00003) is the default.

Value

A data frame with the following columns:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>agency</td>
<td>character</td>
<td>The NWIS code for the agency reporting the data</td>
</tr>
<tr>
<td>site</td>
<td>character</td>
<td>The USGS site number</td>
</tr>
<tr>
<td>Date</td>
<td>Date</td>
<td>The date of the value</td>
</tr>
<tr>
<td>code</td>
<td>character</td>
<td>Any codes that qualify the corresponding value</td>
</tr>
<tr>
<td>value</td>
<td>numeric</td>
<td>The numeric value for the parameter</td>
</tr>
</tbody>
</table>

Note that code and value are repeated for the parameters requested. The names are of the form X_D_P_S, where X is literal, D is an option description of the parameter, P is the parameter code, and S is the statistic code (if applicable).

There are also several useful attributes attached to the data frame:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>character</td>
<td>The url used to generate the data</td>
</tr>
</tbody>
</table>
siteInfo  data.frame A data frame containing information on the requested sites
variableInfo data.frame A data frame containing information on the requested parameters
statisticInfo data.frame A data frame containing information on the requested statistics on the data
queryTime   POSIXct The time the data was returned

See Also
renameNWISColumns, importWaterML1

Examples

```r
site_id <- '04085427'
startDate <- '2012-01-01'
endDate <- '2012-06-30'
pCode <- '00060'
rawDailyQ <- readNWISdv(site_id, pCode, startDate, endDate)
rawDailyQAndTempMeanMax <- readNWISdv(site_id, c('00010', '00060'),
                      startDate, endDate, statCd=c('00001', '00003'))
rawDailyMultiSites <- readNWISdv(c('01491000', '01645000'), c('00010', '00060'),
                       startDate, endDate, statCd=c('00001', '00003'))
# Site with no data:
x <- readNWISdv("10258500","00060", "2014-09-08", "2014-09-14")
names(attributes(x))
attr(x, "siteInfo")
attr(x, "variableInfo")
site <- "05212700"
notActive <- readNWISdv(site, "00060", "2014-01-01", "2014-01-07")
```

---

## readNWISgw1

**Groundwater level measurements retrieval from USGS (NWIS)**

### Description

Reads groundwater level measurements from NWISweb. Mixed date/times come back from the service depending on the year that the data was collected. See [https://waterdata.usgs.gov/usa/nwis/gw](https://waterdata.usgs.gov/usa/nwis/gw) for details about groundwater. By default the returned dates are converted to date objects, unless convertType is specified as FALSE. Sites with non-standard date formats (i.e. lacking a day) can be affected (see examples). See [https://waterservices.usgs.gov/rest/GW-Levels-Service.html](https://waterservices.usgs.gov/rest/GW-Levels-Service.html) for more information.
**Usage**

```r
readNWISgwl(
  siteNumbers,
  startDate = "",
  endDate = "",
  parameterCd = NA,
  convertType = TRUE,
  tz = "UTC"
)
```

**Arguments**

- `siteNumbers` character USGS site number (or multiple sites). This is usually an 8 digit number
- `startDate` character starting date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the earliest possible record.
- `endDate` character ending date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the latest possible record.
- `parameterCd` character USGS parameter code. This is usually an 5 digit number. Default is "".
- `convertType` logical, defaults to TRUE. If TRUE, the function will convert the data to dates, datetimes, numerics based on a standard algorithm. If false, everything is returned as a character
- `tz` character to set timezone attribute of dateTime. Default is "UTC", and converts the date times to UTC, properly accounting for daylight savings times based on the data's provided tz_cd column. Possible values to provide are "America/New_York","America/Chicago","America/Denver","America/Los_Angeles","America/Anchorage", as well as the following which do not use daylight savings time: "America/Honolulu","America/Jamaica","America/Managua","America/Phoenix", and "America/Metlakatla". See also OlsonNames() for more information on time zones.

**Value**

A data frame with the following columns:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>agency_cd</td>
<td>character</td>
<td>The NWIS code for the agency reporting the data</td>
</tr>
<tr>
<td>site_no</td>
<td>character</td>
<td>The USGS site number</td>
</tr>
<tr>
<td>site_tp_cd</td>
<td>character</td>
<td>Site type code</td>
</tr>
<tr>
<td>lev_dt</td>
<td>Date</td>
<td>Date level measured</td>
</tr>
<tr>
<td>lev_tm</td>
<td>character</td>
<td>Time level measured</td>
</tr>
<tr>
<td>lev_tz_cd</td>
<td>character</td>
<td>Time datum</td>
</tr>
<tr>
<td>lev_va</td>
<td>numeric</td>
<td>Water level value in feet below land surface</td>
</tr>
<tr>
<td>sl_lev_va</td>
<td>numeric</td>
<td>Water level value in feet above specific vertical datum</td>
</tr>
<tr>
<td>lev_status_cd</td>
<td>character</td>
<td>The status of the site at the time the water level was measured</td>
</tr>
<tr>
<td>lev_agency_cd</td>
<td>character</td>
<td>The agency code of the person measuring the water level</td>
</tr>
</tbody>
</table>


readNWISmeas

Surface-water measurement data retrieval from USGS (NWIS)

Description


Usage

readNWISmeas(
    siteNumbers, 
    startDate = "", 
    endDate = "", 
    tz = "UTC", 
    expanded = FALSE, 
    convertType = TRUE
)
Arguments

- **siteNumbers**: character USGS site number (or multiple sites). This is usually an 8 digit number.
- **startDate**: character starting date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the earliest possible record.
- **endDate**: character ending date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the latest possible record.
- **tz**: character to set timezone attribute of dateTime. Default is "UTC", and converts the date times to UTC, properly accounting for daylight savings times based on the data's provided tz_cd column. Possible values to provide are "America/New_York", "America/Chicago", "America/Denver", "America/Los_Angeles", "America/Anchorage", as well as the following which do not use daylight savings time: "America/Honolulu", "America/Jamaica", "America/Managua", "America/Phoenix", and "America/Metlakatla". See also OlsonNames() for more information on time zones.
- **expanded**: logical. Whether or not (TRUE or FALSE) to call the expanded data.
- **convertType**: logical, defaults to TRUE. If TRUE, the function will convert the data to dates, datetimes, numerics based on a standard algorithm. If false, everything is returned as a character.

Value

A data frame with at least the following columns:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>agency_cd</td>
<td>character</td>
<td>The NWIS code for the agency reporting the data</td>
</tr>
<tr>
<td>site_no</td>
<td>character</td>
<td>The USGS site number</td>
</tr>
<tr>
<td>measurement_dt</td>
<td>POSIXct</td>
<td>The date and time (in POSIXct) of the measurement. Unless specified with the tz parameter, the this column is an incomplete, a measurement_dt_date and measurement_dt_time column are added to the returned data frame.</td>
</tr>
<tr>
<td>tz_cd</td>
<td>character</td>
<td>The time zone code for the measurement_dt column</td>
</tr>
</tbody>
</table>


There are also several useful attributes attached to the data frame:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>character</td>
<td>The url used to generate the data</td>
</tr>
<tr>
<td>queryTime</td>
<td>POSIXct</td>
<td>The time the data was returned</td>
</tr>
<tr>
<td>comment</td>
<td>character</td>
<td>Header comments from the RDB file</td>
</tr>
<tr>
<td>siteInfo</td>
<td>data.frame</td>
<td>A data frame containing information on the requested sites</td>
</tr>
<tr>
<td>tz_cd_reported</td>
<td></td>
<td>The originally reported time zone</td>
</tr>
</tbody>
</table>

See Also

- constructNWISURL
- importRDB1
readNWISpCode

USGS Parameter Data Retrieval

Description
Imports data from NWIS about measured parameter based on user-supplied parameter code or codes. This function gets the data from here: https://nwis.waterdata.usgs.gov/nwis/pmcodes

Usage
readNWISpCode(parameterCd)

Arguments
parameterCd character of USGS parameter codes (or multiple parameter codes). These are 5 digit number codes, more information can be found here: https://help.waterdata.usgs.gov/. To get a complete list of all current parameter codes in the USGS, use "all" as the input.

Value
parameterData data frame with the following information:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>parameter_cd</td>
<td>character</td>
<td>5-digit USGS parameter code</td>
</tr>
<tr>
<td>parameter_group_nm</td>
<td>character</td>
<td>USGS parameter group name</td>
</tr>
<tr>
<td>parameter_nm</td>
<td>character</td>
<td>USGS parameter name</td>
</tr>
<tr>
<td>casrn</td>
<td>character</td>
<td>Chemical Abstracts Service (CAS) Registry Number</td>
</tr>
<tr>
<td>srname</td>
<td>character</td>
<td>Substance Registry Services Name</td>
</tr>
<tr>
<td>parameter_units</td>
<td>character</td>
<td>Parameter units</td>
</tr>
</tbody>
</table>

See Also
importRDB1
Examples

```r
paramINFO <- readNWISpCode(c('01075', '00060', '00931'))
paramINFO <- readNWISpCode(c('01075', '00060', '00931', NA))
```

---

**readNWISpeak**

*Peak flow data from USGS (NWIS)*

**Description**

Reads peak flow from NWISweb. Data is retrieved from [https://waterdata.usgs.gov/nwis](https://waterdata.usgs.gov/nwis). In some cases, the specific date of the peak data is not known. This function will default to converting complete dates to a "Date" object, and converting incomplete dates to "NA". If those incomplete dates are needed, set the ‘asDateTime’ argument to FALSE. No dates will be converted to R Date objects.

**Usage**

```r
readNWISpeak(
  siteNumbers, 
  startDate = "", 
  endDate = "", 
  asDateTime = TRUE, 
  convertType = TRUE 
)
```

**Arguments**

- `siteNumbers` character USGS site number(or multiple sites). This is usually an 8 digit number.
- `startDate` character starting date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the earliest possible record.
- `endDate` character ending date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the latest possible record.
- `asDateTime` logical default to TRUE. When TRUE, the peak_dt column is converted to a Date object, and incomplete dates are removed. When FALSE, no columns are removed, but no dates are converted.
- `convertType` logical, defaults to TRUE. If TRUE, the function will convert the data to dates, datetimes, numerics based on a standard algorithm. If false, everything is returned as a character

**Value**

A data frame with the following columns:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>agency_cd</td>
<td>character</td>
<td>The NWIS code for the agency reporting the data</td>
</tr>
<tr>
<td>site_no</td>
<td>character</td>
<td>The USGS site number</td>
</tr>
</tbody>
</table>
Imports data from NWIS web service. This function gets the data from here: https://nwis.waterdata.usgs.gov/nwis/qwdata A list of parameter codes can be found here: https://nwis.waterdata.usgs.gov/nwis/pmcodes/ A list of statistic codes can be found here: https://nwis.waterdata.usgs.gov/nwis/help/?read_file=stat&format=table

### Examples

```r
site_ids <- c('01594440', '040851325')

data <- readNWISpeak(site_ids)
data2 <- readNWISpeak(site_ids, asDateTime=FALSE)
stations<-'06011000''
peakdata<-readNWISpeak(stations,convertType=FALSE)
```
Usage

```r
readNWISqw(
  siteNumbers,
  parameterCd,
  startDate = "",
  endDate = "",
  expanded = TRUE,
  reshape = FALSE,
  tz = "UTC"
)
```

Arguments

- **siteNumbers**: character of USGS site numbers. This is usually an 8 digit number
- **parameterCd**: character that contains the code for a parameter group, or a character vector of 5-digit parameter codes. See **Details**.
- **startDate**: character starting date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the earliest possible record. Date arguments are always specified in local time.
- **endDate**: character ending date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the latest possible record. Date arguments are always specified in local time.
- **expanded**: logical defaults to TRUE. If TRUE, retrieves additional information. Expanded data includes `remark_cd` (remark code), `result_va` (result value), `val_qual_tx` (result value qualifier code), `meth_cd` (method code), `dqi_cd` (data-quality indicator code), `rpt_lev_va` (reporting level), and `rpt_lev_cd` (reporting level type). If FALSE, only returns `remark_cd` (remark code) and `result_va` (result value). Expanded = FALSE will not give sufficient information for unbiased statistical analysis.
- **reshape**: logical, reshape the expanded data. If TRUE, then return a wide data frame with all water-quality in a single row for each sample. If FALSE (default), then return a long data frame with each water-quality result in a single row. This argument is only applicable to expanded data. Data requested using expanded=FALSE is always returned in the wide format.
- **tz**: character to set timezone attribute of dateTime. Default is "UTC", and converts the date times to UTC, properly accounting for daylight savings times based on the data’s provided tz_cd column. Possible values to provide are "America/New_York","America/Chicago","America/Denver","America/Los_Angeles", "America/Anchorage", as well as the following which do not use daylight savings time: "America/Honolulu","America/Jamaica","America/Managua","America/Phoenix", and "America/Metlakatla". See also OlsonNames() for more information on time zones.

Details

Valid parameter code groups are "All," or group codes:
Code  Description
INF  Information
PHY  Physical
INM  Inorganics, Major, Metals
INN  Inorganics, Major, Non-metals
NUT  Nutrient
MBI  Microbiological
BIO  Biological
IMN  Inorganics, Minor, Non-metals
IMM  Inorganics, Minor, Metals
TOX  Toxicity
OPE  Organics, pesticide
OPC  Organics, PCBs
OOT  Organics, other
RAD  Radiochemistry
SED  Sediment
POP  Population/community
OTH  Other
HAB  Habitat
ISO  Stable Isotopes

If more than one parameter group is requested, only sites that data for all requested groups are returned.

**Value**

A data frame with at least the following columns:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>agency_cd</td>
<td>character</td>
<td>The NWIS code for the agency reporting the data</td>
</tr>
<tr>
<td>site_no</td>
<td>character</td>
<td>The USGS site number</td>
</tr>
<tr>
<td>sample_dt</td>
<td>Date</td>
<td>The date the sample was collected</td>
</tr>
<tr>
<td>sample_tm</td>
<td>character</td>
<td>The reported sample collection time</td>
</tr>
<tr>
<td>startDateTime</td>
<td>POSIXct</td>
<td>Combining sample_dt and sample_tm, a date/time column is created, and converted into UTC (unless the tz argument specifies a different time zone)</td>
</tr>
<tr>
<td>endDateTime</td>
<td>POSIXct</td>
<td>If any sample_end_dt and sample_end_dt exist, this column is created similar to startDateTime</td>
</tr>
</tbody>
</table>

Further columns will be included depending on the requested output format (expanded = TRUE or FALSE). Columns that end in "_reported" are the originally reported timezones, but the "tz_cd" column defines the timezone of any POSIXct columns.

There are also several useful attributes attached to the data frame:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>character</td>
<td>The url used to generate the data</td>
</tr>
<tr>
<td>queryTime</td>
<td>POSIXct</td>
<td>The time the data was returned</td>
</tr>
<tr>
<td>comment</td>
<td>character</td>
<td>Header comments from the RDB file</td>
</tr>
<tr>
<td>siteInfo</td>
<td>data frame</td>
<td>A data frame containing information on the requested sites</td>
</tr>
</tbody>
</table>
readNWISrating

variableInfo  data frame  A data frame containing information on the requested parameters

See Also

readWQPdata, whatWQPsites, readWQPq, constructNWISURL

Description

Reads current rating table for an active USGS streamgage from NWISweb. Data is retrieved from https://waterdata.usgs.gov/nwis.

Usage

readNWISrating(siteNumber, type = "base", convertType = TRUE)

Arguments

siteNumber  character  USGS site number. This is usually an 8 digit number

type  character  can be "base", "corr", or "exsa"

convertType  logical  defaults to TRUE. If TRUE, the function will convert the data to dates, datetimes, numerics based on a standard algorithm. If false, everything is returned as a character

Value

A data frame. If type is "base," then the columns are INDEP, typically the gage height, in feet; DEP, typically the streamflow, in cubic feet per second; and STOR, where "*" indicates that the pair are a fixed point of the rating curve. If type is "exsa," then an additional column, SHIFT, is included that indicates the current shift in the rating for that value of INDEP. If type is "corr," then the columns are INDEP, typically the gage height, in feet; CORR, the correction for that value; and CORRINDEP, the corrected value for CORR.

If type is "base," then the data frame has an attribute called "RATING" that describes the rating curve is included.

There are also several useful attributes attached to the data frame:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>character</td>
<td>The url used to generate the data</td>
</tr>
<tr>
<td>queryTime</td>
<td>POSIXct</td>
<td>The time the data was returned</td>
</tr>
<tr>
<td>comment</td>
<td>character</td>
<td>Header comments from the RDB file</td>
</tr>
<tr>
<td>siteInfo</td>
<td>data.frame</td>
<td>A data frame containing information on the requested sites</td>
</tr>
<tr>
<td>RATING</td>
<td>character</td>
<td>Rating information</td>
</tr>
</tbody>
</table>
Note
Not all active USGS streamgages have traditional rating curves that relate flow to stage.

See Also
constructNWISURL, importRDB1

Examples

```r
site_id <- '01594440'

data <- readNWISrating(site_id, "base")
attr(data, "RATING")
```

---

**readNWISsite**  
*USGS Site File Data Retrieval*

**Description**
Imports data from USGS site file site. This function gets data from here: https://waterservices.usgs.gov/

**Usage**

```r
readNWISsite(siteNumbers)
```

**Arguments**

- `siteNumbers` character USGS site number (or multiple sites). This is usually an 8 digit number

**Value**
A data frame with at least the following columns:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>agency_cd</td>
<td>character</td>
<td>The NWIS code for the agency reporting the data</td>
</tr>
<tr>
<td>site_no</td>
<td>character</td>
<td>The USGS site number</td>
</tr>
<tr>
<td>station_nm</td>
<td>character</td>
<td>Site name</td>
</tr>
<tr>
<td>site_tp_cd</td>
<td>character</td>
<td>Site type</td>
</tr>
<tr>
<td>lat_va</td>
<td>numeric</td>
<td>DMS latitude</td>
</tr>
<tr>
<td>long_va</td>
<td>numeric</td>
<td>DMS longitude</td>
</tr>
<tr>
<td>dec_lat_va</td>
<td>numeric</td>
<td>Decimal latitude</td>
</tr>
<tr>
<td>dec_long_va</td>
<td>numeric</td>
<td>Decimal longitude</td>
</tr>
<tr>
<td>coord_meth_cd</td>
<td>character</td>
<td>Latitude-longitude method</td>
</tr>
<tr>
<td>coord_acy_cd</td>
<td>character</td>
<td>Latitude-longitude accuracy</td>
</tr>
<tr>
<td>coordDatum_cd</td>
<td>character</td>
<td>Latitude-longitude datum</td>
</tr>
<tr>
<td>decCoordDatum_cd</td>
<td>character</td>
<td>Decimal Latitude-longitude datum</td>
</tr>
</tbody>
</table>
district_cd character District code
state_cd character State code
county_cd character County code
country_cd character Country code
land_net_ds character Land net location description
map_nm character Name of location map
map_scale_fc character Scale of location map
alt_va numeric Altitude of Gage/land surface
alt_meth_cd character Method altitude determined
alt_acy_va numeric Altitude accuracy
alt_datum_cd character Altitude datum
huc_cd character Hydrologic unit code
basin_cd character Drainage basin code
topo_cd character Topographic setting code
instruments_cd character Flags for instruments at site
construction_dt character Date of first construction
inventory_dt character Date site established or inventoried
drain_area_va numeric Drainage area
contrib_drain_area_va numeric Contributing drainage area
tz_cd character Time Zone abbreviation
local_time_fg character Site honors Daylight Savings Time
reliability_cd character Data reliability code
gw_file_cd character Data-other GW files
nat_aqfr_cd character National aquifer code
aqfr_cd character Local aquifer code
aqfr_type_cd character Local aquifer type code
well_depth_va numeric Well depth
hole_depth_va numeric Hole depth
depth_src_cd character Source of depth data
project_no character Project number

There are also several useful attributes attached to the data frame:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>character</td>
<td>The url used to generate the data</td>
</tr>
<tr>
<td>queryTime</td>
<td>POSIXct</td>
<td>The time the data was returned</td>
</tr>
<tr>
<td>comment</td>
<td>character</td>
<td>Header comments from the RDB file</td>
</tr>
</tbody>
</table>

Examples

siteINFO <- readNWISsite('05114000')
siteINFORMulti <- readNWISsite(c('05114000','00423350'))
readNWISstat  

Site statistics retrieval from USGS (NWIS)

Description


Usage

readNWISstat(
  siteNumbers, 
  parameterCd, 
  startDate = "", 
  endDate = "", 
  convertType = TRUE, 
  statReportType = "daily", 
  statType = "mean"
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>siteNumbers</td>
<td>character USGS site number (or multiple sites). This is usually an 8 digit number.</td>
</tr>
<tr>
<td>parameterCd</td>
<td>character USGS parameter code. This is usually a 5 digit number.</td>
</tr>
<tr>
<td>startDate</td>
<td>character starting date for data retrieval in the form YYYY, YYYY-MM, or YYYY-MM-DD. Dates cannot be more specific than the statReportType, i.e. startDate for monthly statReportTypes cannot include days, and annual statReportTypes cannot include days or months. Months and days are optional for the daily statReportType. Default is &quot;&quot; which indicates retrieval for the earliest possible record. For daily data, this indicates the start of the period the statistics will be computed over.</td>
</tr>
<tr>
<td>endDate</td>
<td>character ending date for data retrieval in the form YYYY, YYYY-MM, or YYYY-MM-DD. Default is &quot;&quot; which indicates retrieval for the latest possible record. For daily data, this indicates the end of the period the statistics will be computed over. The same restrictions as startDate apply.</td>
</tr>
<tr>
<td>convertType</td>
<td>logical, defaults to TRUE. If TRUE, the function will convert the data to numerics based on a standard algorithm. Years, months, and days (if applicable) are also returned as numerics in separate columns. If convertType is false, everything is returned as a character.</td>
</tr>
<tr>
<td>statReportType</td>
<td>character time division for statistics: daily, monthly, or annual. Default is daily. Note that daily provides statistics for each calendar day over the specified range of water years, i.e. no more than 366 data points will be returned for each site/parameter. Use readNWISdata or readNWISdv for daily averages. Also note that 'annual' returns statistics for the calendar year. Use readNWISdata</td>
</tr>
</tbody>
</table>
for water years. Monthly and yearly provide statistics for each month and year within the range individually.

statType character type(s) of statistics to output for daily values. Default is mean, which is the only option for monthly and yearly report types. See the statistics service documentation at https://waterservices.usgs.gov/rest/Statistics-Service.html for a full list of codes.

Value

A data frame with the following columns:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>agency_cd</td>
<td>character</td>
<td>The NWIS code for the agency reporting the data</td>
</tr>
<tr>
<td>site_no</td>
<td>character</td>
<td>The USGS site number</td>
</tr>
<tr>
<td>parameter_cd</td>
<td>character</td>
<td>The USGS parameter code</td>
</tr>
</tbody>
</table>

Other columns will be present depending on statReportType and statType

See Also

constructNWISURL, importRDB1

Examples

```r
x1 <- readNWISstat(siteNumbers=c("02319394"),
                     parameterCd=c("00060"),
                     statReportType="annual")
#all the annual mean discharge data for two sites
x2 <- readNWISstat(siteNumbers=c("02319394","02171500"),
                     parameterCd=c("00010","00060"),
                     statReportType="annual")
#Request p25, p75, and mean values for temperature and discharge for the 2000s
#Note that p25 and p75 were not available for temperature, and return NAs
x <- readNWISstat(siteNumbers=c("02171500"),
                   parameterCd=c("00010","00060"),
                   statReportType="daily",
                   statType=c("mean","median"),
                   startDate="2000",endDate="2010")
```

readNWISUse  Water use data retrieval from USGS (NWIS)

Description

Retrieves water use data from USGS Water Use Data for the Nation. See https://waterdata.usgs.gov/nwis/wu for more information. All available use categories for the supplied arguments are retrieved.
Usage

```r
readNWISuse(
  stateCd,
  countyCd,
  years = "ALL",
  categories = "ALL",
  convertType = TRUE,
  transform = FALSE
)
```

Arguments

- **stateCd**: could be character (full name, abbreviation, id), or numeric (id). Only one is accepted per query.
- **countyCd**: could be character (name, with or without "County", or "ALL"), numeric (id), or code `NULL`, which will return state or national data depending on the `stateCd` argument. `ALL` may also be supplied, which will return data for every county in a state. Can be a vector of counties in the same state.
- **years**: integer Years for data retrieval. Must be years ending in 0 or 5. Default is all available years.
- **categories**: character categories of water use. Defaults to ALL. Specific categories must be supplied as two-letter abbreviations as seen in the URL when using the NWIS water use web interface. Note that there are different codes for national and state level data.
- **convertType**: logical defaults to TRUE. If TRUE, the function will convert the data to numerics based on a standard algorithm. Years, months, and days (if applicable) are also returned as numerics in separate columns. If `convertType` is false, everything is returned as a character.
- **transform**: logical only intended for use with national data. Defaults to FALSE, with data being returned as presented by the web service. If TRUE, data will be transformed and returned with column names, which will reformat national data to be similar to state data.

Value

A data frame with at least the year of record, and all available statistics for the given geographic parameters. County and state fields will be included as appropriate.

Examples

```r
#All data for a county
allegheny <- readNWISuse(stateCd = "Pennsylvania", countyCd = "Allegheny")

#Data for an entire state for certain years
ohio <- readNWISuse(years=c(2000,2005,2010), stateCd = "OH", countyCd = NULL)

#Data for an entire state, county by county
```
pr <- readNWISuse(years=c(2000,2005,2010),stateCd = "PR",countyCd="ALL")

#All national-scale data, transforming data frame to named columns from named rows
national <- readNWISuse(stateCd = NULL, countyCd = NULL, transform = TRUE)

#Washington, DC data
dc <- readNWISuse(stateCd = "DC",countyCd = NULL)

#data for multiple counties, with different input formatting
paData <- readNWISuse(stateCd = "42",countyCd = c("Allegheny County","BUTLER",1,"031"))

#retrieving two specific categories for an entire state
ks <- readNWISuse(stateCd = "KS", countyCd = NULL, categories = c("IT","LI"))

---

**readNWISuv**

**Instantaneous value data retrieval from USGS (NWIS)**

**Description**


**Usage**

```
readNWISuv(siteNumbers, parameterCd, startDate = "", endDate = "", tz = "UTC")
```

**Arguments**

- **siteNumbers**: character USGS site number (or multiple sites). This is usually an 8 digit number.
- **parameterCd**: character USGS parameter code. This is usually an 5 digit number.
- **startDate**: character starting date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the earliest possible record. Simple date arguments are specified in local time. See more information here: [https://waterservices.usgs.gov/rest/IV-Service.html](https://waterservices.usgs.gov/rest/IV-Service.html).
- **endDate**: character ending date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the latest possible record. Simple date arguments are specified in local time. See more information here: [https://waterservices.usgs.gov/rest/IV-Service.html](https://waterservices.usgs.gov/rest/IV-Service.html).
- **tz**: character to set timezone attribute of dateTime. Default is "UTC", and converts the date times to UTC, properly accounting for daylight savings times based on the data’s provided tz_cd column. Possible values to provide are "America/New_York","America/Chicago","America/Denver","America/Los_Angeles", "America/Phoenix", "America/Asheville", etc. Defaul
"America/Anchorage", as well as the following which do not use daylight savings time: "America/Honolulu", "America/Jamaica", "America/Managua", "America/Phoenix", and "America/Metlakatla". See also OlsonNames() for more information on time zones.

Value

A data frame with the following columns:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>agency_cd</td>
<td>character</td>
<td>The NWIS code for the agency reporting the data</td>
</tr>
<tr>
<td>site_no</td>
<td>character</td>
<td>The USGS site number</td>
</tr>
<tr>
<td>dateTime</td>
<td>POSIXct</td>
<td>The date and time of the value converted to UTC</td>
</tr>
<tr>
<td>tz_cd</td>
<td>character</td>
<td>The time zone code for dateTime</td>
</tr>
<tr>
<td>code</td>
<td>character</td>
<td>Any codes that qualify the corresponding value</td>
</tr>
<tr>
<td>value</td>
<td>numeric</td>
<td>The numeric value for the parameter</td>
</tr>
</tbody>
</table>

Note that code and value are repeated for the parameters requested. The names are of the form: X_D_P_S, where X is literal, D is an option description of the parameter, P is the parameter code, and S is the statistic code (if applicable).

There are also several useful attributes attached to the data frame:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>character</td>
<td>The url used to generate the data</td>
</tr>
<tr>
<td>siteInfo</td>
<td>data.frame</td>
<td>A data frame containing information on the requested sites</td>
</tr>
<tr>
<td>variableInfo</td>
<td>data.frame</td>
<td>A data frame containing information on the requested parameters</td>
</tr>
<tr>
<td>statisticInfo</td>
<td>data.frame</td>
<td>A data frame containing information on the requested statistics on the data</td>
</tr>
<tr>
<td>queryTime</td>
<td>POSIXct</td>
<td>The time the data was returned</td>
</tr>
</tbody>
</table>

See Also

renameNWISColumns, importWaterML1

Examples

```r
site_id <- '05114000'
parameterCd <- '00060'
startDate <- "2014-10-10"
endDate <- "2014-10-10"
rawData <- readNWISuv(site_id, parameterCd, startDate, endDate)

rawData_today <- readNWISuv(site_id, parameterCd, Sys.Date(), Sys.Date())

timeZoneChange <- readNWISuv(c('04024430', '04024000'), parameterCd, "2013-11-03", "2013-11-03")

centralTime <- readNWISuv(site_id, parameterCd,)
```
readWQPdata

"2014-10-10T12:00", "2014-10-10T23:59",
tz="America/Chicago")

# Adding 'Z' to the time indicates to the web service to call the data with UTC time:
GMTdata <- readNWISuv(site_id, parameterCd,
"2014-10-10T00:00Z", "2014-10-10T23:59Z")

---

readWQPdata  General Data Import from Water Quality Portal

**Description**

Imports data from Water Quality Portal web service. This function gets the data from here: https://www.waterqualitydata.us because it allows for other agencies rather than the USGS.

**Usage**

readWQPdata(..., querySummary = FALSE, tz = "UTC", ignore_attributes = FALSE)

**Arguments**

... see [https://www.waterqualitydata.us/webservices_documentation](https://www.waterqualitydata.us/webservices_documentation) for a complete list of options. A list of arguments can also be supplied.

querySummary logical to ONLY return the number of records and unique sites that will be returned from this query. This argument is not supported via the combined list from the ...argument

tz character to set timezone attribute of dateTime. Default is "UTC", and converts the date times to UTC, properly accounting for daylight savings times based on the data’s provided tz_cd column. Possible values to provide are "America/New_York", "America/Chicago", "America/Denver", "America/Los_Angeles", "America/Anchorage", as well as the following which do not use daylight savings time: "America/Honolulu", "America/Jamaica", "America/Managua", "America/Phoenix", and "America/Metlakatla". See also OlsonNames() for more information on time zones.

ignore_attributes logical to choose to ignore fetching site and parameter attributes. Default is FALSE.

**Value**

A data frame with at least the following columns:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrganizationIdentifer</td>
<td>character</td>
<td>A designator used to uniquely identify a unique business establishment.</td>
</tr>
<tr>
<td>OrganizationFormalName</td>
<td>character</td>
<td>The legal designator (i.e. formal name) of an organization.</td>
</tr>
<tr>
<td>Field</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ActivityIdentifier</td>
<td>character</td>
<td>Designator that uniquely identifies an activity within an organization.</td>
</tr>
<tr>
<td>ActivityTypeCode</td>
<td>character</td>
<td>The text describing the type of activity.</td>
</tr>
<tr>
<td>ActivityMediaName</td>
<td>character</td>
<td>Name or code indicating the environmental medium where the sample was taken.</td>
</tr>
<tr>
<td>ActivityMediaSubdivisionName</td>
<td>character</td>
<td>Name or code indicating the environmental matrix as a subdivision of the sample medium.</td>
</tr>
<tr>
<td>ActivityStartDate</td>
<td>character</td>
<td>The calendar date on which the field activity is started.</td>
</tr>
<tr>
<td>ActivityStartTime.Time</td>
<td>character</td>
<td>The time of day that is reported when the field activity began.</td>
</tr>
<tr>
<td>ActivityStartTime.TimeZoneCode</td>
<td>character</td>
<td>The time zone for which the time of day is reported.</td>
</tr>
<tr>
<td>ActivityEndDate</td>
<td>character</td>
<td>The calendar date when the field activity is completed.</td>
</tr>
<tr>
<td>ActivityEndTime.Time</td>
<td>character</td>
<td>The time of day that is reported when the field activity ended.</td>
</tr>
<tr>
<td>ActivityEndTime.TimeZoneCode</td>
<td>character</td>
<td>The time zone for which the time of day is reported.</td>
</tr>
<tr>
<td>ActivityDepthHeightMeasure.MeasureValue</td>
<td>character</td>
<td>A measurement of the vertical location (measured from a reference point) at which a result occurred.</td>
</tr>
<tr>
<td>ActivityDepthHeightMeasure.MeasureUnitCode</td>
<td>character</td>
<td>The code that represents the unit for measuring the item.</td>
</tr>
<tr>
<td>ActivityTopDepthHeightMeasure.MeasureValue</td>
<td>character</td>
<td>The code that represents the unit for measuring the item.</td>
</tr>
<tr>
<td>ActivityTopDepthHeightMeasure.MeasureUnitCode</td>
<td>character</td>
<td>The code that represents the unit for measuring the item.</td>
</tr>
<tr>
<td>ActivityBottomDepthHeightMeasure.MeasureValue</td>
<td>character</td>
<td>The code that represents the unit for measuring the item.</td>
</tr>
<tr>
<td>ActivityBottomDepthHeightMeasure.MeasureUnitCode</td>
<td>character</td>
<td>The code that represents the unit for measuring the item.</td>
</tr>
<tr>
<td>ProjectIdentifier</td>
<td>character</td>
<td>A name of the Organization conducting an activity.</td>
</tr>
<tr>
<td>ActivityConductingOrganizationText</td>
<td>character</td>
<td>A designator used to describe the unique name, number, or code for the conducting organization.</td>
</tr>
<tr>
<td>MonitoringLocationIdentifier</td>
<td>character</td>
<td>General comments concerning the activity.</td>
</tr>
<tr>
<td>ActivityCommentText</td>
<td>character</td>
<td>A code that designates the aquifer associated with groundwater samples.</td>
</tr>
<tr>
<td>HydrologicCondition</td>
<td>character</td>
<td>Hydrologic condition is the hydrologic condition that is represented by the sample collected.</td>
</tr>
<tr>
<td>HydrologicEvent</td>
<td>character</td>
<td>A hydrologic event that is represented by the sample collected.</td>
</tr>
<tr>
<td>SampleCollectionMethod.MethodIdentifier</td>
<td>character</td>
<td>The identification number or code assigned by the method publisher.</td>
</tr>
<tr>
<td>SampleCollectionMethod.MethodIdentifierContext</td>
<td>character</td>
<td>Identifies the source or data system that created or defined the identifier.</td>
</tr>
<tr>
<td>SampleCollectionMethod.MethodName</td>
<td>character</td>
<td>The title that appears on the method from the method publisher.</td>
</tr>
<tr>
<td>SampleCollectionEquipmentName</td>
<td>character</td>
<td>The name for the equipment used in collecting the sample.</td>
</tr>
<tr>
<td>ResultDetectionConditionText</td>
<td>character</td>
<td>The textual descriptor of a result.</td>
</tr>
<tr>
<td>CharacteristicName</td>
<td>character</td>
<td>The object, property, or substance which is evaluated or enumerated.</td>
</tr>
<tr>
<td>ResultSampleFractionText</td>
<td>character</td>
<td>The text name of the portion of the sample associated with results obtained from a physically-parted sample.</td>
</tr>
<tr>
<td>ResultMeasureValue</td>
<td>numeric</td>
<td>The reportable measure of the result for the chemical, microbiological, or other characteristic measured.</td>
</tr>
<tr>
<td>MeasureQualifierCode</td>
<td>character</td>
<td>A code used to identify any qualifying issues that affect the results.</td>
</tr>
<tr>
<td>ResultMeasure.MeasureUnitCode</td>
<td>character</td>
<td>The code that represents the unit for measuring the item.</td>
</tr>
<tr>
<td>ResultMeasure.Value</td>
<td>numeric</td>
<td>The reportable measure of the result for the chemical, microbiological, or other characteristic measured.</td>
</tr>
<tr>
<td>ResultStatusIdentifier</td>
<td>character</td>
<td>Indicates the acceptability of the result with respect to QA/QC criteria.</td>
</tr>
<tr>
<td>StatisticalBaseCode</td>
<td>character</td>
<td>The code for the method used to calculate derived results.</td>
</tr>
<tr>
<td>ResultValueTypeName</td>
<td>character</td>
<td>A name that qualifies the process which was used in the determination of the result value.</td>
</tr>
<tr>
<td>ResultWeightBasisText</td>
<td>character</td>
<td>The name that represents the form of the sample or portion of the sample which is associated with the result value.</td>
</tr>
<tr>
<td>ResultTimeBasisText</td>
<td>character</td>
<td>The period of time (in days) over which a measurement was made.</td>
</tr>
<tr>
<td>ResultTemperatureBasisText</td>
<td>character</td>
<td>The name that represents the controlled temperature at which a measurement was made.</td>
</tr>
<tr>
<td>ResultParticleSizeBasisText</td>
<td>character</td>
<td>User defined free text describing the particle size class.</td>
</tr>
<tr>
<td>PrecisionValue</td>
<td>character</td>
<td>A measure of mutual agreement among individual measurements.</td>
</tr>
<tr>
<td>ResultCommentText</td>
<td>character</td>
<td>Free text with general comments concerning the result.</td>
</tr>
<tr>
<td>USGSPCCode</td>
<td>character</td>
<td>5-digit number used in the US Geological Survey computerized data system to uniquely identify the result.</td>
</tr>
<tr>
<td>ResultDepthHeightMeasure.MeasureValue</td>
<td>character</td>
<td>A measurement of the vertical location (measured from a reference point) at which a result occurred.</td>
</tr>
<tr>
<td>ResultDepthHeightMeasure.MeasureUnitCode</td>
<td>character</td>
<td>The code that represents the unit for measuring the item.</td>
</tr>
<tr>
<td>ResultDepthAltitudeReferencePointText</td>
<td>character</td>
<td>The reference used to indicate the datum or reference used to establish the depth/altitude of a result.</td>
</tr>
<tr>
<td>SubjectTaxonomicName</td>
<td>character</td>
<td>The name of the organism from which a tissue sample was taken.</td>
</tr>
<tr>
<td>ResultParticleSizeBasisText</td>
<td>character</td>
<td>User defined free text describing the particle size class.</td>
</tr>
<tr>
<td>ResultTemperatureBasisText</td>
<td>character</td>
<td>The name that represents the controlled temperature at which a measurement was made.</td>
</tr>
<tr>
<td>ResultTimeBasisText</td>
<td>character</td>
<td>The period of time (in days) over which a measurement was made.</td>
</tr>
<tr>
<td>ResultWeightBasisText</td>
<td>character</td>
<td>The name that represents the form of the sample or portion of the sample which is associated with the result value.</td>
</tr>
<tr>
<td>ResultValueTypeName</td>
<td>character</td>
<td>A name that qualifies the process which was used in the determination of the result value.</td>
</tr>
<tr>
<td>ResultStatusIdentifier</td>
<td>character</td>
<td>Indicates the acceptability of the result with respect to QA/QC criteria.</td>
</tr>
<tr>
<td>StatisticalBaseCode</td>
<td>character</td>
<td>The code for the method used to calculate derived results.</td>
</tr>
<tr>
<td>ResultValueTypeName</td>
<td>character</td>
<td>A name that qualifies the process which was used in the determination of the result value.</td>
</tr>
<tr>
<td>ResultWeightBasisText</td>
<td>character</td>
<td>The name that represents the form of the sample or portion of the sample which is associated with the result value.</td>
</tr>
<tr>
<td>ResultTimeBasisText</td>
<td>character</td>
<td>The period of time (in days) over which a measurement was made.</td>
</tr>
<tr>
<td>ResultTemperatureBasisText</td>
<td>character</td>
<td>The name that represents the controlled temperature at which a measurement was made.</td>
</tr>
<tr>
<td>ResultParticleSizeBasisText</td>
<td>character</td>
<td>User defined free text describing the particle size class.</td>
</tr>
<tr>
<td>PrecisionValue</td>
<td>character</td>
<td>A measure of mutual agreement among individual measurements.</td>
</tr>
<tr>
<td>ResultCommentText</td>
<td>character</td>
<td>Free text with general comments concerning the result.</td>
</tr>
</tbody>
</table>
SampleTissueAnatomyName * character The name of the anatomy from which a tissue sample was taken.
ResultAnalyticalMethod.MethodIdentifier character The identification number or code assigned by the method publisher.
ResultAnalyticalMethod.MethodIdentifierContext character Identifies the source or data system that created or defined the identifier.
ResultAnalyticalMethod.MethodName character The title that appears on the method from the method publisher.
MethodDescriptionText * character A brief summary that provides general information about the method.
LaboratoryName character The name of Lab responsible for the result.
AnalysisStartDate character The calendar date on which the analysis began.
ResultLaboratoryCommentText character Remarks which further describe the laboratory procedures.
DetectionQuantitationLimitTypeName character Text describing the type of detection or quantitation level used in the analysis of a characteristic.
DetectionQuantitationLimitMeasure.MeasureValue numeric Constituent concentration that, when processed through the method, is considered to be different from a blank. Measure value is given in the units stored in DetectionQuantitationLimitMeasure.MeasureUnitCode.
DetectionQuantitationLimitMeasure.MeasureUnitCode character The code that represents the unit for measuring the item.
PreparationStartDate character The calendar date when the preparation/extraction of the sample for analysis began.
ActivityStartDateTime POSIXct Activity start date and time converted to POSIXct UTC.
ActivityEndDateTime POSIXct Activity end date and time converted to POSIXct UTC.

* = elements only in NWIS  + = elements only in STORET

There are also several useful attributes attached to the data frame:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>character</td>
<td>The url used to generate the data</td>
</tr>
<tr>
<td>siteInfo</td>
<td>data.frame</td>
<td>A data frame containing information on the requested sites</td>
</tr>
<tr>
<td>variableInfo</td>
<td>data.frame</td>
<td>A data frame containing information on the requested parameters</td>
</tr>
<tr>
<td>queryTime</td>
<td>POSIXct</td>
<td>The time the data was returned</td>
</tr>
</tbody>
</table>

Examples

```r
calendar <- "pH"
pHData <- readWQPdata(siteid="USGS-04024315", characteristicName=pH)
pHData_summary <- readWQPdata(bBox=c(-90.10,42.67,-88.64,43.35),
characteristicName=pH, querySummary=TRUE)
startDate <- as.Date("2013-01-01")
secchi.names = c("Depth, Secchi disk depth",
               "Depth, Secchi disk depth (choice list)",
               "Secchi Reading Condition (choice list)",
               "Water transparency, Secchi disc")
args <- list('startDateLo' = startDate,
      'startDateHi' = "2013-12-31",
      statecode="WI",
      characteristicName=secchi.names)
wqp.data <- readWQPdata(args)

args_2 <- list('startDateLo' = startDate,
      'startDateHi' = "2013-12-31",
      statecode="WI",
      characteristicName=secchi.names,
      querySummary=TRUE)
```
readWQPqw <- readWQPdata(args_2)

arg_3 <- list('startDateLo' = startDate, 'startDateHi' = "2013-12-31")
arg_4 <- list(statecode="WI", characteristicName=secchi.names)
wqp.summary <- readWQPdata(arg_3, arg_4, querySummary=TRUE)
wqp.summary_WI <- readWQPdata(arg_3, statecode="WI", characteristicName=secchi.names, querySummary=TRUE)

# querying by county
dailyLexingtonVA <- readWQPdata(statecode = "Virginia", countycode="Lexington", parameterCd = "00010")

---

**readWQPqw**

*Raw Data Import for Water Quality Portal*

**Description**

Imports data from the Water Quality Portal. This function gets the data from here: [https://www.waterqualitydata.us](https://www.waterqualitydata.us). There are four required input arguments: siteNumbers, parameterCd, startDate, and endDate. ParameterCd can either be a USGS 5-digit code, or a characteristic name. The sites can be either USGS, or other Water Quality Portal offered sites. It is required to use the 'full' site name, such as 'USGS-01234567'.

**Usage**

```r
readWQPqw(
  siteNumbers,
  parameterCd,
  startDate = "",
  endDate = "",
  tz = "UTC",
  querySummary = FALSE
)
```

**Arguments**

- **siteNumbers** character site number. This needs to include the full agency code prefix.
- **parameterCd** vector of USGS 5-digit parameter code or characteristicNames. Leaving this blank will return all of the measured values during the specified time period.
- **startDate** character starting date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the earliest possible record. Date arguments are always specified in local time.
endDate character ending date for data retrieval in the form YYYY-MM-DD. Default is "" which indicates retrieval for the latest possible record. Date arguments are always specified in local time.

tz character to set timezone attribute of dateTime. Default is "UTC", and converts the date times to UTC, properly accounting for daylight savings times based on the data’s provided tz_cd column. Possible values to provide are "America/New_York", "America/Chicago", "America/Denver", "America/Los_Angeles", "America/Anchorage", as well as the following which do not use daylight savings time: "America/Honolulu", "America/Jamaica", "America/Managua", "America/Phoenix", and "America/Metlakatla". See also OlsonNames() for more information on time zones.

querySummary logical to look at number of records and unique sites that will be returned from this query.

Value

A data frame with at least the following columns:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrganizationIdentifier</td>
<td>character</td>
<td>A designator used to uniquely identify a unique business establishment.</td>
</tr>
<tr>
<td>OrganizationFormalName</td>
<td>character</td>
<td>The legal designator (i.e. formal name) of an organization.</td>
</tr>
<tr>
<td>ActivityIdentifier</td>
<td>character</td>
<td>Designator that uniquely identifies an activity within an organization.</td>
</tr>
<tr>
<td>ActivityTypeCode</td>
<td>character</td>
<td>The text describing the type of activity.</td>
</tr>
<tr>
<td>ActivityMediaName</td>
<td>character</td>
<td>Name or code indicating the environmental medium where the sample was taken.</td>
</tr>
<tr>
<td>ActivityMediaSubdivisionName</td>
<td>character</td>
<td>Name or code indicating the environmental matrix as a subdivision of the sample media.</td>
</tr>
<tr>
<td>ActivityStartDate</td>
<td>character</td>
<td>The calendar date on which the field activity is started.</td>
</tr>
<tr>
<td>ActivityStartTime.Time</td>
<td>character</td>
<td>The time of day that is reported when the field activity began.</td>
</tr>
<tr>
<td>ActivityStartTime.TimeZoneCode</td>
<td>character</td>
<td>The time zone for which the time of day is reported.</td>
</tr>
<tr>
<td>ActivityEndDate</td>
<td>character</td>
<td>The calendar date when the field activity is completed.</td>
</tr>
<tr>
<td>ActivityEndTime.Time</td>
<td>character</td>
<td>The time of day that is reported when the field activity ended.</td>
</tr>
<tr>
<td>ActivityEndTime.TimeZoneCode</td>
<td>character</td>
<td>The time zone for which the time of day is reported.</td>
</tr>
<tr>
<td>ActivityDepthHeightMeasure.MeasureValue</td>
<td>character</td>
<td>A measurement of the vertical location (measured from reference point).</td>
</tr>
<tr>
<td>ActivityDepthHeightMeasure.MeasureUnitCode</td>
<td>character</td>
<td>The code that represents the unit for measuring the item.</td>
</tr>
<tr>
<td>ActivityDepthAltitudeReferencePointText</td>
<td>character</td>
<td>The reference used to indicate the datum or reference used.</td>
</tr>
<tr>
<td>ActivityTopDepthHeightMeasure.MeasureValue</td>
<td>character</td>
<td>A measurement of the upper vertical location of a vertical location.</td>
</tr>
<tr>
<td>ActivityTopDepthHeightMeasure.MeasureUnitCode</td>
<td>character</td>
<td>The code that represents the unit for measuring the item.</td>
</tr>
<tr>
<td>ActivityBottomDepthHeightMeasure.MeasureValue</td>
<td>character</td>
<td>A measurement of the lower vertical location of a vertical location.</td>
</tr>
<tr>
<td>ActivityBottomDepthHeightMeasure.MeasureUnitCode</td>
<td>character</td>
<td>The code that represents the unit for measuring the item.</td>
</tr>
<tr>
<td>ProjectIdentifier</td>
<td>character</td>
<td>A designator used to uniquely identify a data collection project.</td>
</tr>
<tr>
<td>ActivityConductingOrganizationText</td>
<td>character</td>
<td>A name of the Organization conducting an activity.</td>
</tr>
<tr>
<td>MonitoringLocationIdentifier</td>
<td>character</td>
<td>A designator used to describe the unique name, number, or code.</td>
</tr>
<tr>
<td>ActivityCommentText</td>
<td>character</td>
<td>General comments concerning the activity.</td>
</tr>
<tr>
<td>SampleAquifer *</td>
<td>character</td>
<td>A code that designates the aquifer associated with groundwater sampling.</td>
</tr>
<tr>
<td>HydrologicCondition *</td>
<td>character</td>
<td>Hydrologic condition is the hydrologic condition that is represented.</td>
</tr>
<tr>
<td>HydrologicEvent *</td>
<td>character</td>
<td>A hydrologic event that is represented by the sample collected.</td>
</tr>
<tr>
<td>SampleCollectionMethod.MethodIdentifier</td>
<td>character</td>
<td>The identification number or code assigned by the method publisher.</td>
</tr>
<tr>
<td>SampleCollectionMethod.MethodIdentifierContext</td>
<td>character</td>
<td>Identifies the source or data system that created or defined the method.</td>
</tr>
<tr>
<td>SampleCollectionMethod.MethodName</td>
<td>character</td>
<td>The title that appears on the method publisher.</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SampleCollectionEquipmentName</td>
<td>character</td>
<td>The name for the equipment used in collecting the sample.</td>
</tr>
<tr>
<td>ResultDetectionConditionText</td>
<td>character</td>
<td>The textual descriptor of a result.</td>
</tr>
<tr>
<td>CharacteristicName</td>
<td>character</td>
<td>The object, property, or substance which is evaluated or enumerated.</td>
</tr>
<tr>
<td>ResultSampleFractionText</td>
<td>character</td>
<td>The text name of the portion of the sample associated with results.</td>
</tr>
<tr>
<td>ResultMeasureValue</td>
<td>numeric</td>
<td>The reportable measure of the result for the chemical, microbiological, or</td>
</tr>
<tr>
<td>MeasureQualifierCode</td>
<td>character</td>
<td>A code used to identify any qualifying issues that affect the result.</td>
</tr>
<tr>
<td>ResultMeasure.MeasureUnitCode</td>
<td>character</td>
<td>The code that represents the unit for measuring the item.</td>
</tr>
<tr>
<td>ResultStatusIdentifier</td>
<td>character</td>
<td>Indicates the acceptability of the result with respect to QA/QC criteria.</td>
</tr>
<tr>
<td>StatisticalBaseCode</td>
<td>character</td>
<td>The code for the method used to calculate derived results.</td>
</tr>
<tr>
<td>ResultValueTypeName</td>
<td>character</td>
<td>A name that qualifies the process which was used in the determination.</td>
</tr>
<tr>
<td>ResultWeightBasisText</td>
<td>character</td>
<td>The name that represents the form of the sample or portion that was analyzed</td>
</tr>
<tr>
<td>ResultTimeBasisText</td>
<td>character</td>
<td>The period of time (in days) over which a measurement or observation was made</td>
</tr>
<tr>
<td>ResultTemperatureBasisText</td>
<td>character</td>
<td>The name that represents the controlled temperature at which the analysis was</td>
</tr>
<tr>
<td>ResultParticleSizeBasisText</td>
<td>character</td>
<td>User defined free text describing the particle size class of the sample.</td>
</tr>
<tr>
<td>PrecisionValue</td>
<td>character</td>
<td>A measure of mutual agreement among individual measurements.</td>
</tr>
<tr>
<td>ResultCommentText</td>
<td>character</td>
<td>Free text with general comments concerning the result.</td>
</tr>
<tr>
<td>USGSCode</td>
<td>character</td>
<td>A 5-digit number used in the US Geological Survey computerized data system.</td>
</tr>
<tr>
<td>ResultDepthHeightMeasure.MeasureValue</td>
<td>character</td>
<td>A measurement of the vertical location (measured from the reference point).</td>
</tr>
<tr>
<td>ResultDepthHeightMeasure.MeasureUnitCode</td>
<td>character</td>
<td>The code that represents the unit for measuring the vertical location.</td>
</tr>
<tr>
<td>ResultDepthAltitudeReferencePointText</td>
<td>character</td>
<td>The reference used to indicate the datum or reference used for vertical</td>
</tr>
<tr>
<td>SubjectTaxonomicName</td>
<td>character</td>
<td>The name of the organism from which a tissue sample was taken.</td>
</tr>
<tr>
<td>SampleTissueAnatomyName</td>
<td>character</td>
<td>The name of the anatomy from which a tissue sample was taken.</td>
</tr>
<tr>
<td>ResultAnalyticalMethod.MethodIdentifier</td>
<td>character</td>
<td>The identification number or code assigned by the method publisher.</td>
</tr>
<tr>
<td>ResultAnalyticalMethod.MethodIdentifierContext</td>
<td>character</td>
<td>Identifies the source or data system that created or defined the identifier.</td>
</tr>
<tr>
<td>ResultAnalyticalMethod/MethodName</td>
<td>character</td>
<td>The title that appears on the method from the method publisher.</td>
</tr>
<tr>
<td>MethodDescriptionText</td>
<td>character</td>
<td>A brief summary that provides general information about the analytical</td>
</tr>
<tr>
<td>LaboratoryName</td>
<td>character</td>
<td>The name of Lab responsible for the result.</td>
</tr>
<tr>
<td>AnalysisStartDate</td>
<td>character</td>
<td>The calendar date on which the analysis began.</td>
</tr>
<tr>
<td>ResultLaboratoryCommentText</td>
<td>character</td>
<td>Remarks which further describe the laboratory procedures used.</td>
</tr>
<tr>
<td>DetectionQuantitationLimitTypeName</td>
<td>character</td>
<td>Text describing the type of detection or quantitation level that was used.</td>
</tr>
<tr>
<td>DetectionQuantitationLimitMeasure.Value</td>
<td>numeric</td>
<td>Constituent concentration that, when processed through the method, was</td>
</tr>
<tr>
<td>DetectionQuantitationLimitMeasure.MeasureUnitCode</td>
<td>character</td>
<td>The code that represents the unit for measuring the constituent concentration</td>
</tr>
<tr>
<td>PreparationStartDate</td>
<td>character</td>
<td>The calendar date when the preparation/extraction of the sample was begun.</td>
</tr>
<tr>
<td>ActivityStartDateTime</td>
<td>POSIXct</td>
<td>Activity start date and time converted to POSIXct UTC.</td>
</tr>
<tr>
<td>ActivityEndDateTime</td>
<td>POSIXct</td>
<td>Activity end date and time converted to POSIXct UTC.</td>
</tr>
</tbody>
</table>

* = elements only in NWIS  + = elements only in STORET

There are also several useful attributes attached to the data frame:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>character</td>
<td>The url used to generate the data</td>
</tr>
<tr>
<td>siteInfo</td>
<td>data.frame</td>
<td>A data frame containing information on the requested sites</td>
</tr>
<tr>
<td>variableInfo</td>
<td>data.frame</td>
<td>A data frame containing information on the requested parameters</td>
</tr>
<tr>
<td>queryTime</td>
<td>POSIXct</td>
<td>The time the data was returned</td>
</tr>
</tbody>
</table>
renameNWISColumns

See Also

readWQPdata, whatWQPsites, readNWISqw, and importWQP

Examples

```r
rawPcode <- readWQPqw('USGS-01594440', '01075', ' ', ' ')
rawCharacteristicName <- readWQPqw('WIDNR_WQX-1032762', 'Specific conductance', ' ', ' ')
rawPHsites <- readWQPqw(c('USGS-05406450', 'USGS-05427949', 'WIDNR_WQX-133040'), 'pH', ' ', ' ')
nwisEx <- readWQPqw('USGS-04024000', c('34247', '30234', '32104', '34220'), ' ', ' ')

nwisEx.summary <- readWQPqw('USGS-04024000', c('34247', '30234', '32104', '34220'), ' ', '2012-12-20', querySummary=TRUE)
```

renameNWISColumns  renameColumns

## Description

Rename columns coming back from NWIS data retrievals. Daily and unit value columns have names derived from their data descriptor, parameter, and statistic codes. This function reads information from the header and the arguments in the call to to rename those columns.

## Usage

```r
renameNWISColumns(
  rawData,
  p00010 = "Wtemp",
  p00045 = "Precip",
  p00060 = "Flow",
  p00065 = "GH",
  p00095 = "SpecCond",
  p00300 = "DO",
  p00400 = "pH",
  p62611 = "GWL",
  p63680 = "Turb",
  p72019 = "WLBS",
  ...
)
```

## Arguments

- **rawData**  the daily- or unit-values datset retrieved from NWISweb.
- **p00010**  the base name for parameter code 00010.
- **p00045**  the base name for parameter code 00045.
- **p00060**  the base name for parameter code 00060.
renameNWISColumns

p00065  the base name for parameter code 00065.
p00095  the base name for parameter code 00095.
p00300  the base name for parameter code 00300.
p00400  the base name for parameter code 00400.
p62611  the base name for parameter code 62611.
p63680  the base name for parameter code 63680.
p72019  the base name for parameter code 72019.
...  named arguments for the base name for any other parameter code. The form of
the name must be like pXXXXX, where XXXX is the parameter code.

Value
A dataset like data with selected columns renamed.

Note
The following statistics codes are converted by renameNWISColumns.

00000  Instantaneous Value, suffix: Inst
00001  Maximum value, suffix: Max
00002  Minimum value, suffix: Min
00003  Mean value, no suffix
00006  Sum of values, suffix: Sum
00007  Modal value, suffix: Mode
00008  Median value, suffix: Median
00012  Equivalent mean value, suffix: EqMean
00021  Tidal high-high value, suffix: HiHiTide
00022  Tidal low-high value, suffix: LoHiTide
00023  Tidal high-low value, suffix: HiLoTide
00024  Tidal low-low value, suffix: LoLoTide

See Also
readNWISdv, readNWISuv

Examples

```r
siteWithTwo <- '01480015'
startDate <- "2012-09-01"
endDate <- "2012-10-01"
twoResults <- readNWISdv(siteWithTwo, "00060",startDate,endDate)
names(twoResults)
renamedCols <- renameNWISColumns(twoResults)
names(renamedCols)
```
#setAccess

```r
#Custom names:
newNames <- renameNWISColumns(twoResults, p00060="Discharge")
names(newNames)
```

---

## setAccess  
*Set data endpoint*

### Description
access Indicate which dataRetrieval access code you want to use options: c('public', 'internal')

### Usage
```r
setAccess(access = "public")
```

### Arguments
access code for data access. Options are: "public","internal","cooperator", or "USGS".
- "internal" represents Access=3 ...for a single water science center
- "USGS" represents Access=2 ...for all water science centers
- "cooperator" represents Access=1
- "public" represents Access=0, public access

### Author(s)
Luke Winslow, Jordan S Read

### Examples
```r
setAccess('internal')
setAccess('public')
```

---

## stateCd  
*US State Code Lookup Table*

### Description
Data pulled from https://www2.census.gov/geo/docs/reference/state.txt on April 1, 2015.

### Value
stateCd data frame.
### Name Type Description

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATE</td>
<td>character</td>
<td>FIPS State Code</td>
</tr>
<tr>
<td>STUSAB</td>
<td>character</td>
<td>Official United States Postal Service (USPS) Code</td>
</tr>
<tr>
<td>STATE_NAME</td>
<td>character</td>
<td>State Name</td>
</tr>
<tr>
<td>STATENS</td>
<td>character</td>
<td>Geographic Names Information System Identifier (GNISID)</td>
</tr>
</tbody>
</table>

### Examples

```r
head(stateCd)
```

```
stateCdLookup St ate code look up
```

### Description

Function to simplify finding state and state code definitions. Used in `readNWISdata` and `readWQPdata`.

### Usage

```r
stateCdLookup(input, outputType = "postal")
```

### Arguments

- **input**: could be character (full name, abbreviation, id), or numeric (id)
- **outputType**: character can be "postal","fullName","tableIndex", or "id".

### Examples

```r
fullName <- stateCdLookup("wi", "fullName")
abbrev <- stateCdLookup("Wisconsin", "postal")
id <- stateCdLookup("WI", "id")
name <- stateCdLookup(55, "fullName")
index <- stateCdLookup("WI", "tableIndex")
stateCd[index,]
stateCdLookup(c("West Virginia", "Wisconsin", 200, 55, "MN"))
```
whatNWISdata

Description
Imports a table of available parameters, period of record, and count. See https://waterservices.usgs.gov/rest/Site-Service.html for more information.

Usage
whatNWISdata(..., convertType = TRUE)

Arguments
... see https://waterservices.usgs.gov/rest/Site-Service.html for a complete list of options. A list of arguments can also be supplied.
convertType logical, defaults to TRUE. If TRUE, the function will convert the data to dates, datetimes, numerics based on a standard algorithm. If false, everything is returned as a character

Value
A data frame with the following columns:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>agency_cd</td>
<td>character</td>
<td>The NWIS code for the agency reporting the data</td>
</tr>
<tr>
<td>site_no</td>
<td>character</td>
<td>The USGS site number</td>
</tr>
<tr>
<td>station_nm</td>
<td>character</td>
<td>Site name</td>
</tr>
<tr>
<td>site_tp_cd</td>
<td>character</td>
<td>Site type</td>
</tr>
<tr>
<td>dec_lat_va</td>
<td>numeric</td>
<td>Decimal latitude</td>
</tr>
<tr>
<td>dec_long_va</td>
<td>numeric</td>
<td>Decimal longitude</td>
</tr>
<tr>
<td>coord_acy_cd</td>
<td>character</td>
<td>Latitude-longitude accuracy</td>
</tr>
<tr>
<td>dec_coord_datum_cd</td>
<td>character</td>
<td>Decimal Latitude-longitude datum</td>
</tr>
<tr>
<td>alt_va</td>
<td>character</td>
<td>Altitude of Gage or land surface</td>
</tr>
<tr>
<td>alt_acy_va</td>
<td>character</td>
<td>Altitude accuracy</td>
</tr>
<tr>
<td>alt_datum_cd</td>
<td>character</td>
<td>Altitude datum</td>
</tr>
<tr>
<td>huc_cd</td>
<td>character</td>
<td>Hydrologic unit code</td>
</tr>
<tr>
<td>data_type_cd</td>
<td>character</td>
<td>Data type</td>
</tr>
<tr>
<td>parm_cd</td>
<td>character</td>
<td>Parameter code</td>
</tr>
<tr>
<td>stat_cd</td>
<td>character</td>
<td>Statistical code</td>
</tr>
<tr>
<td>dd_nu</td>
<td>character</td>
<td>Internal database key</td>
</tr>
<tr>
<td>loc_web_ds</td>
<td>character</td>
<td>Additional measurement description</td>
</tr>
<tr>
<td>medium_grp_cd</td>
<td>character</td>
<td>Medium group code</td>
</tr>
<tr>
<td>parm_grp_cd</td>
<td>character</td>
<td>Parameter group code</td>
</tr>
<tr>
<td>srs_id</td>
<td>character</td>
<td>SRS ID</td>
</tr>
<tr>
<td>access_cd</td>
<td>character</td>
<td>Access code</td>
</tr>
<tr>
<td>begin_date</td>
<td>Date</td>
<td>Begin date</td>
</tr>
</tbody>
</table>
whatNWISsites

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>end_date</td>
<td>Date</td>
<td>End date</td>
</tr>
<tr>
<td>count_nu</td>
<td>integer</td>
<td>Record count</td>
</tr>
<tr>
<td>parameter_group_nm</td>
<td>character</td>
<td>Parameter group name</td>
</tr>
<tr>
<td>parameter_nm</td>
<td>character</td>
<td>Parameter name</td>
</tr>
<tr>
<td>casrn</td>
<td>character</td>
<td>Chemical Abstracts Service (CAS) Registry Number</td>
</tr>
<tr>
<td>srsname</td>
<td>character</td>
<td>Substance Registry Services</td>
</tr>
<tr>
<td>parameter_units</td>
<td>character</td>
<td>Parameter units</td>
</tr>
</tbody>
</table>

There are also several useful attributes attached to the data frame:

```
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>character</td>
<td>The url used to generate the data</td>
</tr>
<tr>
<td>comment</td>
<td>character</td>
<td>Header comments from the RDB file</td>
</tr>
<tr>
<td>queryTime</td>
<td>POSIXct</td>
<td>The time the data was returned</td>
</tr>
</tbody>
</table>
```

Examples

```r
availableData <- whatNWISdata(siteNumber = '05114000')
# To find just unit value ('instantaneous') data:
uvData <- whatNWISdata(siteNumber = '05114000', service="uv")
uvDataMulti <- whatNWISdata(siteNumber = c('05114000', '09423350'), service=c("uv","dv"))
flowAndTemp <- whatNWISdata(stateCd = "WI", service = "uv",
                           parameterCd = c("00060","00010"),
                           statCd = "00003")
```

Description

Returns a list of sites from the NWIS web service. This function gets the data from: [https://waterservices.usgs.gov/rest/Site-Test-Tool.html](https://waterservices.usgs.gov/rest/Site-Test-Tool.html). Mapper format is used.

Usage

```
whatNWISsites(...)  
```

Arguments

... see [https://waterservices.usgs.gov/rest/Site-Service.html](https://waterservices.usgs.gov/rest/Site-Service.html) for a complete list of options. A list (or lists) can also be supplied.

Value

A data frame with at least the following columns:
whatWQPdata

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>agency_cd</td>
<td>character</td>
<td>The NWIS code for the agency reporting the data</td>
</tr>
<tr>
<td>site_no</td>
<td>character</td>
<td>The USGS site number</td>
</tr>
<tr>
<td>station_nm</td>
<td>character</td>
<td>Station name</td>
</tr>
<tr>
<td>site_tp_cd</td>
<td>character</td>
<td>Site type code</td>
</tr>
<tr>
<td>dec_lat_va</td>
<td>numeric</td>
<td>Decimal latitude</td>
</tr>
<tr>
<td>dec_long_va</td>
<td>numeric</td>
<td>Decimal longitude</td>
</tr>
<tr>
<td>queryTime</td>
<td>POSIXct</td>
<td>Query time</td>
</tr>
</tbody>
</table>

There are also several useful attributes attached to the data frame:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>character</td>
<td>The url used to generate the data</td>
</tr>
<tr>
<td>queryTime</td>
<td>POSIXct</td>
<td>The time the data was returned</td>
</tr>
</tbody>
</table>

Examples

```r
siteListPhos <- whatNWISsites(stateCd="OH", parameterCd="00665")
oneSite <- whatNWISsites(sites="05114000")
```

WhatWQPdata

Data Available from Water Quality Portal

Description

Returns a list of sites from the Water Quality Portal web service. This function gets the data from: https://www.waterqualitydata.us. Arguments to the function should be based on https://www.waterqualitydata.us/webservices_documentation. The information returned from this function describes the available data at the WQP sites, and some metadata on the sites themselves.

Usage

```r
whatWQPdata(..., saveFile = tempfile())
```

Arguments

- `...` see https://www.waterqualitydata.us/webservices_documentation for a complete list of options. A list of arguments can also be supplied.
- `saveFile` path to save the incoming geojson output.

Value

A data frame with at least the following columns:
### Site Data Import from Water Quality Portal

**whatWQPsamples**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;type_a&quot;</td>
<td>character</td>
<td>Geojson type</td>
</tr>
<tr>
<td>&quot;features.type&quot;</td>
<td>character</td>
<td>Geojson feature type</td>
</tr>
<tr>
<td>&quot;type1&quot;</td>
<td>character</td>
<td>Geojson spatial type</td>
</tr>
<tr>
<td>&quot;coordinates&quot;</td>
<td>list</td>
<td>List of longitude/latitude</td>
</tr>
<tr>
<td>&quot;ProviderName&quot;</td>
<td>character</td>
<td>The name of the database that provided the data to the Water Quality Portal</td>
</tr>
<tr>
<td>&quot;OrganizationIdentifier&quot;</td>
<td>character</td>
<td>A designator used to uniquely identify a unique business establishment</td>
</tr>
<tr>
<td>&quot;OrganizationFormalName&quot;</td>
<td>character</td>
<td>The legal designator (i.e. formal name) of an organization.</td>
</tr>
<tr>
<td>&quot;MonitoringLocationIdentifier&quot;</td>
<td>character</td>
<td>A designator used to describe the unique name, number, or code assigned to the monitoring location.</td>
</tr>
<tr>
<td>&quot;MonitoringLocationName&quot;</td>
<td>character</td>
<td>The designator specified by the sampling organization for the site at which sampling or other activities are conducted.</td>
</tr>
<tr>
<td>&quot;MonitoringLocationTypeName&quot;</td>
<td>character</td>
<td>The descriptive name for a type of monitoring location.</td>
</tr>
<tr>
<td>&quot;HUCEightDigitCode&quot;</td>
<td>character</td>
<td>The 8 digit federal code used to identify the hydrologic unit of the monitoring location.</td>
</tr>
<tr>
<td>&quot;siteUrl&quot;</td>
<td>character</td>
<td>URL to site information</td>
</tr>
<tr>
<td>&quot;activityCount&quot;</td>
<td>numeric</td>
<td></td>
</tr>
<tr>
<td>&quot;resultCount&quot;</td>
<td>numeric</td>
<td></td>
</tr>
<tr>
<td>&quot;StateName&quot;</td>
<td>character</td>
<td>State name</td>
</tr>
<tr>
<td>&quot;CountyName&quot;</td>
<td>character</td>
<td>County name</td>
</tr>
</tbody>
</table>

### See Also

whatNWISsites

### Examples

```r
site1 <- whatWPdata(siteid="USGS-01594440")
type <- "Stream"
sites <- whatWPdata(countycode="US:55:025",siteType=type)
lakeSites <- whatWPdata(siteType = "Lake, Reservoir, Impoundment", statecode = "US:55")
```

### Description

Returns a list of sites from the Water Quality Portal web service. This function gets the data from: https://www.waterqualitydata.us. Arguments to the function should be based on https://www.waterqualitydata.us/webservices_documentation. The return from this function returns the basic metadata on WQP sites. It is generally faster than the whatWPdata function, but does not return information on what data was collected at the site.
whatWQPsamples

Usage

whatWQPsamples(...)
whatWQPmetrics(...)
whatWQPsites(...)
readWQPsummary(...)

Arguments

... see https://www.waterqualitydata.us/webservices_documentation for a complete list of options. A list of arguments can also be supplied.

Details

The readWQPsummary function has

Value

A data frame with at least the following columns:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrganizationIdentifier</td>
<td>character</td>
<td>A designator used to uniquely identify a unique business establishment.</td>
</tr>
<tr>
<td>OrganizationFormalName</td>
<td>character</td>
<td>The legal designator (i.e. formal name) of an organization.</td>
</tr>
<tr>
<td>MonitoringLocationIdentifier</td>
<td>character</td>
<td>A designator used to describe the unique name, number, or code.</td>
</tr>
<tr>
<td>MonitoringLocationName</td>
<td>character</td>
<td>The designator specified by the sampling organization for the site.</td>
</tr>
<tr>
<td>MonitoringLocationTypeName</td>
<td>character</td>
<td>The descriptive name for a type of monitoring location.</td>
</tr>
<tr>
<td>MonitoringLocationDescriptionText</td>
<td>character</td>
<td>Text description of the monitoring location.</td>
</tr>
<tr>
<td>HUCEightDigitCode</td>
<td>character</td>
<td>The 8 digit federal code used to identify the hydrologic unit.</td>
</tr>
<tr>
<td>DrainageAreaMeasure/MeasureValue</td>
<td>character</td>
<td>The drainage basin of a lake, stream, wetland, or estuary.</td>
</tr>
<tr>
<td>DrainageAreaMeasure/MeasureUnitCode</td>
<td>character</td>
<td>The code that represents the unit for measuring the item.</td>
</tr>
<tr>
<td>ContributingDrainageAreaMeasure/MeasureValue</td>
<td>character</td>
<td>The contributing drainage area of a lake, stream, wetland, or estuary.</td>
</tr>
<tr>
<td>ContributingDrainageAreaMeasure/MeasureUnitCode</td>
<td>character</td>
<td>The code that represents the unit for measuring the item.</td>
</tr>
<tr>
<td>LatitudeMeasure</td>
<td>numeric</td>
<td>The measure of the angular distance on a meridian north.</td>
</tr>
<tr>
<td>LongitudeMeasure</td>
<td>numeric</td>
<td>The measure of the angular distance on a meridian east.</td>
</tr>
<tr>
<td>SourceMapScaleNumeric</td>
<td>character</td>
<td>The number that represents the proportional distance on the map or photo.</td>
</tr>
<tr>
<td>HorizontalAccuracyMeasure/MeasureValue</td>
<td>character</td>
<td>The horizontal measure of the relative accuracy of the latitude or longitude.</td>
</tr>
<tr>
<td>HorizontalAccuracyMeasure/MeasureUnitCode</td>
<td>character</td>
<td>The code that represents the unit for measuring the item.</td>
</tr>
<tr>
<td>HorizontalCoordinateReferenceSystemDatumName</td>
<td>character</td>
<td>The name that describes the reference datum used in determining coordinates.</td>
</tr>
<tr>
<td>VerticalMeasure/MeasureValue</td>
<td>character</td>
<td>The vertical measure of elevation (i.e., the altitude), above or below a reference point.</td>
</tr>
<tr>
<td>VerticalMeasure/MeasureUnitCode</td>
<td>character</td>
<td>The code that represents the unit for measuring the item.</td>
</tr>
<tr>
<td>VerticalAccuracyMeasure/MeasureValue</td>
<td>character</td>
<td>The vertical measure of the relative accuracy of the latitude or longitude.</td>
</tr>
<tr>
<td>VerticalAccuracyMeasure/MeasureUnitCode</td>
<td>character</td>
<td>The code that represents the unit for measuring the item.</td>
</tr>
<tr>
<td>VerticalCollectionMethodName</td>
<td>character</td>
<td>The name that identifies the method used to determine the vertical measure.</td>
</tr>
<tr>
<td>VerticalCoordinateReferenceSystemDatumName</td>
<td>character</td>
<td>The name of the reference datum used to determine the vertical measure.</td>
</tr>
<tr>
<td>CountryCode</td>
<td>character</td>
<td>A code designator used to identify a primary geopolitical unit of the world.</td>
</tr>
<tr>
<td>StateCode</td>
<td>character</td>
<td>A code designator used to identify a principal administrative division.</td>
</tr>
</tbody>
</table>
CountyCode
AquiferName *
FormationTypeText *
AquiferTypeName *
ConstructionDateText *
WellDepthMeasure/MeasureValue *
WellDepthMeasure/MeasureUnitCode *
WellHoleDepthMeasure/MeasureValue *
WellHoleDepthMeasure/MeasureUnitCode *
queryTime

character A code designator used to identify a U.S. county or county equivalent.
character Name of the aquifer in which the well is completed.
character Name of the primary formation or soils unit, in which the well is completed.
character The type of aquifer, such as confined or unconfined.
character Date of construction when well was completed. May be year only.
character Depth below land surface datum (LSD) to the bottom of the hole on completion of drilling. Measure value is given in the units stored in WellDepthMeasure/MeasureUnitCode.
character The code that represents the unit for measuring the item.
character Depth below land surface datum (LSD) to the bottom of the hole on completion of drilling. Measure value is given in the units stored in WellHoleDepthMeasure/MeasureUnitCode.
character The code that represents the unit for measuring the item.
POSIXct Query time

* element is only in NWIS

See Also
whatNWISdata

Examples

```r
site1 <- whatWQPsamples(siteid="USGS-01594440")
type <- "Stream"
sites <- whatWQPsamples(countycode="US:55:025",siteType=type)

sites <- whatWQPsamples(countycode="US:55:025",siteType=type)
lakeSites <- whatWQPmetrics(siteType = "Lake, Reservoir, Impoundment", statecode = "US:55")

site1 <- whatWQPsites(siteid="USGS-01594440")
type <- "Stream"
sites <- whatWQPsites(countycode="US:55:025",
characteristicName = "Phosphorus",
siteType=type)

site1 <- readWQPsummary(siteid="USGS-07144100",
summaryYears=5,
dataProfile="periodOfRecord")
# Pretty slow:
state1 <- readWQPsummary(statecode="NJ",
dataProfile="periodOfRecord")
```


zeroPad

Pad string with leading zeros

Description

Function to pad a string with leading zeros. Useful for parameter codes and USGS site IDs.

Usage

zeroPad(x, padTo)

Arguments

x character

padTo number Final desired length of the character

Value

x character returned with leading zeros

Examples

pCode <- '10'
correctPCode <- zeroPad(pCode,5)
pCodes <- c('100','1000','0','12345','1565465465465465')
correctPCodes <- zeroPad(pCodes,5)
pCodeNA <- c(1,2,NA)
padPCodeNA <- zeroPad(pCodeNA,4)
Index

* IO
  renameNWISColumns, 51

* USGS
  constructNWISURL, 4
  countyCd, 8
  readNWISdv, 26
  readNWISpCode, 32
  readNWISqw, 34
  readNWISsite, 38
  readNWISuv, 43
  readWQPqw, 48
  stateCd, 53
  whatNWISdata, 55
  zeroPad, 61

* WQP
  constructWQPURL, 7
  readWQPdata, 45
  whatWQPdata, 57
  whatWQPsamples, 58

* countyCd
  countyCd, 8

* data
  constructNWISURL, 4
  constructWQPURL, 7
  readNWISdv, 26
  readNWISpCode, 32
  readNWISqw, 34
  readNWISsite, 38
  readNWISuv, 43
  readWQPdata, 45
  readWQPqw, 48
  whatNWISdata, 55
  whatWQPdata, 57
  whatWQPsamples, 58
  zeroPad, 61

* import
  constructNWISURL, 4
  constructWQPURL, 7
  readNWISdv, 26
  readNWISpCode, 32
  readNWISqw, 34
  readNWISsite, 38
  readNWISuv, 43
  whatNWISdata, 55
  whatWQPdata, 57
  whatWQPsamples, 58
  zeroPad, 61

* manip
  renameNWISColumns, 51

* nldi
  findNLDI, 9
  get_nldi_sources, 13

* service
  constructNWISURL, 4
  constructWQPURL, 7
  readNWISdv, 26
  readNWISpCode, 32
  readNWISqw, 34
  readNWISsite, 38
  readNWISuv, 43
  readWQPdata, 45
  readWQPqw, 48
  whatNWISdata, 55
  whatWQPdata, 57
  whatWQPsamples, 58
  zeroPad, 61

* stateCd
  stateCd, 53

* web
  constructNWISURL, 4
  constructWQPURL, 7
  readNWISdv, 26
  readNWISpCode, 32
  readNWISqw, 34
  readNWISsite, 38
  readNWISuv, 43