# Package ‘wqTools’

January 28, 2019

**Title** A Collection of R Tools for Utah Division of Water Quality

**Version** 0.0.0.9000

**Authors** Jake Vander Laan, Utah Division of Water Quality, jvander@utah.gov [aut, cre]

**Description** This package is intended to house R tools developed and for use by UDWQ staff as well as support the UDWQ irTools package.

**Depends** R (>= 3.4.4)

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.1.1

**Roxygen** list(markdown = TRUE)

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

**Repository** github.com/utah-dwq/wqTools

## R topics documented:

## `assignAUs` *Assign Utah assessment units to sites*

### Description

This function assigns assessment units to water quality portal type site objects (or data with site information attached). This can be done before or after assigning beneficial uses.

### Usage

```
assignAUs(x, lat = "LatitudeMeasure", long = "LongitudeMeasure")
```

### Arguments

| | |
|---|---|
| `x` | Input dataset. Must include latitude & longitude columns. |
| `lat` | Name of latitude column. Default matches WQP objects. |
| `long` | Name of longitude column. Default matches WQP objects. |

### Value

Returns the input data frame with assessment unit information appended.

### Examples

```
# Read a couple of sites from Mantua Reservoir
sites=readWQP(type="sites", siteid=c("UTAHDWQ_WQX-4900440","UTAHDWQ_WQX-4900470"))
sites_AUs=assignUses(sites)
```

## `assignUses` *Assign Utah beneficial use classes to sites*

### Description

This function assigns beneficial use classes to water quality portal type site objects (or data with site information attached).

### Usage

```
assignUses(x, lat = "LatitudeMeasure", long = "LongitudeMeasure",
  flatten = FALSE)
```

### Arguments

| | |
|---|---|
| `x` | Input dataset. Must include latitude & longitude columns. |
| `lat` | Name of latitude column. Default matches WQP objects. |
| `long` | Name of longitude column. Default matches WQP objects. |
| `flatten` | Logical. If FALSE (default), maintain use categorys as single comma separated column. If TRUE, use column and data are flattened by expanded use column. |

### Examples

```
# Read a couple of sites from Mantua Reservoir
sites=readWQP(type="sites", siteid=c("UTAHDWQ_WQX-4900440","UTAHDWQ_WQX-4900470"))
sites_uses=assignUses(sites)
sites_uses_flat=assignUses(sites, flatten=TRUE)
```

## au_poly *Utah's IR-specific assessment unit polygons*

### Description

Polygons containing assessment unit designations.

### Usage

```
au_poly
```

### Format

An sf type polygon shapefile

## buildMap *Build a site map of WQP sites or ECHO facilities*

### Description

Build a map of sample sites, facilities, or both. Map includes sites, beneficial use and assessment unit polygons, and satellite and topo baselayers. This is designed to work with column names as extracted from WQP or ECHO via udwqTools functions readWQP() and readECHO_fac(). Map will launch in default browser (or R-Studio's browser if using R-Studio). Site and assessment unit features are searchable by idetifier and name via the search button on the left side of the map. The most recently turned on layer is "on top" of the map. Only features on top will show their pop-up on click.

### Usage

```
buildMap(fac, sites, au_poly, bu_poly, ss_poly, search = c("sites",
  "aus"))
```

### Arguments

| | |
|---|---|
| `fac` | Facility locations queried via readECHO_fac. |
| `sites` | Site locations queried via readWQP(type="sites"). May also be a data file with WQP site information merged to it. |
| `au_poly` | Optional. Polygon file to be mapped as assessment units. Useful for mapping a subset of specific assessment units. If missing, the default state wide AU polygon is used. |
| `bu_poly` | Optional. Polygon file to be mapped as beneficial uses. Useful for mapping a subset of beneficial uses. If missing, the default state wide uses polygon is used. |

| | |
|---|---|
| ss_poly | Optional. Polygon file to be mapped as site specific standards. Useful for mapping a subset of ss polygons. If missing, the default state wide ss polygon is used. |
| search | Vector of objects to be made searchable. One or both of "sites" and "aus". Defaults to c("sites","aus"). Any other inputs are ignored. |

### Examples

```
# Read sites & facility locations
jr_sites=readWQP(type="sites",
siteid=c("UTAHDWQ_WQX-4994100","UTAHDWQ_WQX-4994120","UTAHDWQ_WQX-4991860",
 "UTAHDWQ_WQX-4994190","UTAHDWQ_WQX-4994172","UTAHDWQ_WQX-4994090",
 "UTAHDWQ_WQX-4992890","UTAHDWQ_WQX-4992880","UTAHDWQ_WQX-4992480",
 "UTAHDWQ_WQX-4992055","UTAHDWQ_WQX-4991940","UTAHDWQ_WQX-4991880"))
jr_fac=readECHO_fac(p_pid=c("UT0024392","UT0024384","UT0025852","UT0021725"))
#Build some maps
map1=buildMap(sites=jr_sites, fac=jr_fac) #define new object for use later
map1 #call generated map object to launch in browser
buildMap(sites=mantua_sites) #just sites, launch w/o generating map object in workspace
buildMap(fac=jr_fac) #just facilities
buildMap() #Build an empty map w/ just AU, BU, and SS std polys
#html maps can be saved via htmlwidgets package saveWidget(map1, file="your/path/map1.html")
```

## bu_poly *Utah's beneficial use polygon shapes*

### Description

Polygons containing beneficial use designations and water body type information. Used to assign uses or standards to site locations.

### Usage

```
bu_poly
```

### Format

An sf type polygon shapefile

## calcTSI *Calculate TSI values from input data*

### Description

This function calculates TSI values according to Utah's IR methods from input data containing values for of chlorophyll, total phosphorus, and secchi disk depth. Note that inputs for these parameters must be specified in units of ug/L, mg/L, and meters, respectively.

### Usage

```
calcTSI(x, in_format = "wide", chl_ugL = "chla", TP_mgL = "TP",
  SD_m = "SD")
```

### Arguments

`x` Input dataset

`in_format` One of "wide" or "flat" to specify data input format. Note that only wide format inputs are currently supported.

`chl_ugL` Name of chlorophyll-a variable in ug/L

`TP_mgL` Name of total phosphorus variable in mg/L

`SD_m` Name of secchi disk depth variable in m

### Examples

```
data(ul_trophic)
head(ul_trophic)
tsi=calcTSI(ul_trophic,chl_ugL="ChlA",TP_mgL="Phosphate.phosphorus.Total",SD_m="Depth.Secchi.disk.depth")
head(tsi)
plot(TSIchl~ChlA,tsi)
```

## `facToNum` *Convert factors to numeric equivalents*

### Description

Converts input object to number if class=="factor". If class !="factor", input object is returned un-altered.

### Usage

```
facToNum(x)
```

### Arguments

`x` Input vector object

## `readECHO_ec` *Read effluent chart data from EPA ECHO webservices*

### Description

This function extracts effluent chart data from EPA ECHO for multiple stations & combinations of parameters. All arguments are optional except p_id. At least one p_id must be specified.

### Usage

```
readECHO_ec(..., print = TRUE)
```

### Arguments

`...` additional arguments to be passed to ECHO query path. See https://echo.epa.gov/tools/web-services/effluent-charts#!/Effluent_Charts/get_eff_rest_services_download_effluent_chart optional arguments for effluent chart data reads. Note that arguments for output are ignored.

`print` Logical. If TRUE (default), print summary table of facilities & parameters returned.

`p_id` Permitted facility ID. Either a single text value (in quotes) or a vector of text values.

`parameter_code` Parameter code. Either a single text value (in quotes) or a vector of text values.

`start_date` Query start date in "mm/dd/yyyy" format.

`end_date` Query end date in "mm/dd/yyyy" format.

### Value

A flat data frame of EPA ECHO effluent chart data

### Examples

```
#Extract effluent chart data for facility UT0025241, all outfalls
UT0025241_ec=readECHO_ec(type="ec",p_id="UT0025241", start_date="01/01/2010", end_date="01/15/2019")
head(UT0025241_ec)

# Extract effluent total phosphorus data from outfall 001 for facility UT0025241
UT0025241_tp_001=readECHO_ec(p_id="UT0025241", parameter_code="00665", outfall="001")
UT0025241_tp_001_effluent=UT0025241_tp_001[UT0025241_tp_001$monitoring_location_desc=="Effluent Gross",]
head(UT0025241_tp_001_effluent)

# Extract flow through facility from UT0021717
UT0021717_flow=readECHO_ec(p_id="UT0021717", parameter_code="50050")

# Extract flow & TP from UT0025241 & UT0021717
tp_flow=readECHO_ec(p_id=c("UT0025241","UT0021717"), parameter_code=c("50050","00665"))
```

## `readECHO_fac` *Read facility information from EPA ECHO webservices*

### Description

This function extracts facility information from EPA ECHO based on argument inputs.

### Usage

```
readECHO_fac(type = "", ...)
```

### Arguments

`...` Additional arguments to be passed to ECHO query path. See https://echo.epa.gov/tools/web-services/facility-search-water#!/Facility_Information/get_cwa_rest_services_get_facility_info for optional arguments for facilities. Note that arguments for output are ignored.

### Value

A data frame of EPA ECHO facility information

### Examples

```
# Read facility locations in Utah
ut_fac=readECHO_fac(p_st="ut", p_act="y")
head(ut_fac)
# Read facility locations for two permit IDs
two_fac=readECHO_fac(p_pid=c("UT0021717","UT0025241"))
two_fac
```

## readWQP *Read EPA Water Quality Portal Data*

### Description

This function extracts water quality data from EPA's Water Quality Portal based on user arguemnt inputs. Note that connections to the WQP occassionally time out during download. This function tries to download requested files up to 10 times before exiting. All arguments except type are optional, but at least one should be provided to limit download size and prevent errors connecting to WQP. Note that some, but not all, special characters in characteristic names have been accounted if. If in doubt, use the WQP web interface to determine the appropriate sytax for odd characteristic names. This function coerces non-numeric values in ResultMeasureValue column (for result & narrowresult type queries). This may generate NA values with a warning for special characters.

### Usage

```
readWQP(type = "result", ..., print = TRUE)
```

### Arguments

| | |
|---|---|
| `type` | Data type to read. One of "result", "narrowresult", "sites", "activity", or "detquantlim". |
| `...` | additional arguments to be passed to WQP query path. See https://www.waterqualitydata.us/portal/ for optional arguments. |
| `print` | Logical. Print summary table of sites & characteristics (only for result or narrowresult types). |
| `start_date` | Query start date in "mm/dd/yyyy" format. |
| `end_date` | Query end date in "mm/dd/yyyy" format. |

### Value

A data frame of WQP data

### Examples

```
# Read some data from Mantua Reservoir (2016-2018)
nr=readWQP(type="narrowresult", siteid=c("UTAHDWQ_WQX-4900440","UTAHDWQ_WQX-4900470"),
  start_date="01/01/2016", end_date="12/31/2018")

# Read just Arsenic, Cadmium, and DO, all dates
nr=readWQP(type="narrowresult",
  siteid=c("UTAHDWQ_WQX-4900440","UTAHDWQ_WQX-4900470"),
  characteristicName=c("Arsenic","Cadmium","Dissolved oxygen (DO)"))

# Read all Total dissolved solids statewide (2016-2018) (& note statecode for Utah)
tds_sw=readWQP(type="result",
  statecode="US:49",
  characteristicName="Total dissolved solids",
  start_date="01/01/2016", end_date="12/31/2018",
  print=F)

# Read DWQ's sites
sites=readWQP(type="sites", statecode="US:49", organization="UTAHDWQ_WQX", siteType=c("Lake, Reservoir, Impo
plot(LatitudeMeasure~LongitudeMeasure, sites[sites$LatitudeMeasure>0 & sites$LongitudeMeasure<0,])
```

## readWQPbySite *Read WQP data by selecting sites in an interactive map*

### Description

This function allows the user to read WQP data by selecting desired sites in an interactive map and specifiying desired types of data and output.

### Usage

```
readWQPbySite(sites, map, mode = "click", types = c("narrowresult",
  "activity", "sites"), merge = T, ...)
```

### Arguments

| | |
|---|---|
| sites | Optional. A sites object containing lat/long columns named "LatitudeMeasure" & "LongitudeMeasure". If no sites object is provided, a Utah state-wide query of WQP will be generated for you. |
| map | Optional. A map object to use as a background for site selection. If no map object is provided, a basic map will be generated for you. |
| mode | Mode for map selection. One of "click" or "draw". Click allows site selection by clicking on individual sites. Draw allows site selection by polygons. |
| types | Vector of data types to read from WQP for selected sites. See ?wqTools::readWQP for options. |
| merge | Logical. If TRUE (default), merge all selected data types to single data frame. Merges are performed as left joins in order they are provided. If FALSE, return list of individual data objects. |
| ... | Other arguments to be passed to readWQP (e.g. start_date, end_date, characteristicName, etc.). See ?wqTools::readWQP for more info. |

### Examples

```
wqp_data=readWQPbySite()
```

## `ss_poly` *Utah's site-specific standard polygon shapes*

### Description

Polygons containing site-specific standard designations and information.

### Usage

```
ss_poly
```

### Format

An sf type polygon shapefile

## `ul_trophic` *Utah Lake trophic data*

### Description

Utah Lake trophic data

### Usage

```
data(ul_trophic)
```

### Format

A data.frame with 729 rows and 15 columns

# Index