Package 'irTools'

April 19, 2019

Title R-tools for generating Utah's Integrated Report on water quality

Version 0.0.0.9000

Authors

Jake Vander Laan, Utah Division of Water Quality, jvander@utah.gov [aut, cre], Elise Hinman, Utah Division of Water Quality, ehinman@utah.gov [aut]

Description

This package is a collection of R-tools developed by Utah Division of Water Quality to enhance and automate Utah's Integrated Report on water quality. The package includes tools for downloading data from EPA's Water Quality Portal and Attains, prepping raw data for WQ assessments, and WQ assessment and reporting tools.

Depends R (¿= 3.4.4)

YEAR 2018

COPYRIGHT HOLDER Utah Division of Water Quality

License MIT + file LICENSE

RoxygenNote 6.1.1

Encoding UTF-8

Imports sf,
shiny,
shinyjs,
jsonlite,
openxlsx,
leaflet,
wqTools

Remotes utah-dwq/wqTools

R topics documented:

`applyScreenTable` *Apply screens to WQP data by selected table*

Description

Joins activity review inputs (detConditionTable, labNameActivityTable, activityMediaNameTable, masterSiteTable, paramTransTable, & activityCommentTable) to WQP data to apply decisions from input files to data.

Usage

```
applyScreenTable(data, translation_wb, sheetname, flag_col_name,
  com_col_name, startRow = 1, na_dup_err = TRUE)
```

Arguments

`data`	A merged WQP result object. Must include both narrowresult & activity files. May also be a post-fillMaskedValues() results object. Note: re-application of edited domain tables to an already screened dataset is not advised, and changes to the domain table likely will not be reflected in a re-screened dataset due to merging conventions.
`translation_wb`	Full path and filename for IR translation workbook
`sheetname`	Name of sheet in workbook holding desired screening decisions
`flag_col_name`	Name to rename IR_FLAG column to.
`com_col_name`	Name to rename IR_COMMENT column to.
`startRow`	Row to start reading excel sheet from (in case additional headers have been added). Defaults to 1.
`na_dup_err`	Logical. If TRUE (default), exit function with error if IR_FLAG values are NA or if duplicates detected in combinations in the domain table for which InData=="Y". Set to FALSE to apply a screen table without checking for NA values in IR_FLAG.

Value

A data.frame object of WQP data with merged columns from input screening tables.

`assessEColi` *Assess E.coli data at the year/site level.*

Description

Compares E.coli data to 30-day and max criterion standards using Scenarios A, B, and C, and assigns e. coli assessment categories to each site.

Usage

```
assessEColi(data, rec_season = TRUE, SeasonStartDate = "05-01",
  SeasonEndDate = "10-31")
```

Arguments

`data`	A prepped dataframe object (likely the ecoli object within the prepped_data list–prepped_data$ecoli) containing e.coli data at the site/use/parameter level with standards assigned.
`rec_season`	Logical. If TRUE, restricts assessments to recreation season data.
`SeasonStartDate`	A string in the form "mm-dd" to define beginning of rec season over which to perform assessments.
`SeasonEndDate`	A string in the form "mm-dd" to define end of rec season over which to perform assessments.

Value

Returns list with three objects: assessments from all Scenarios on all data, and ecoli assessments aggregated over scenario and year and rolled up to site level.

`assessExcCounts` *Perform assessments based on sample & exceedance counts (conventional & toxic assessments)*

Description

Performs site level (site-parameter-use-criterion specific) water quality assessments based on sample & exceedance counts by applying IR method flowcharts with user specified inputs.

Usage

```
assessExcCounts(data, min_n, max_exc_count = NA, max_exc_pct = NA,
  max_exc_count_id = NA, max_exc_pct_id = NA, id_cols = c())
```

Arguments

data	Site-use-parameter-criterion specific sample and exceedance counts(i.e. output from countExceedances())
min_n	Minimum sample size for sufficient data assessment (sample counts ¿= min_n are considered sufficient)
max_exc_count	Maximum allowable exceedance count for full support (exceedance counts ¿ max_exc_count are considered not supporting) - one of max_exc_count or max_exc_pct must be specified
max_exc_pct	Maximum allowable exceedance percentage for full support (exceedance pcts ¿ max_exc_pct are considered not supporting - one of max_exc_count or max_exc_pct must be specified
max_exc_count_id	Maximum allowable exceedance count for insufficient data with exceedances (for sites/parameters with insufficient data, exceedance counts ¿ max_exc_count_id are considered insufficient data with exceedances) - one of max_exc_count_id or max_exc_pct_id must be specified
max_exc_pct_id	Maximum allowable exceedance percentage for insufficient data with exceedances (for sites/parameters with insufficient data, exceedance pcts ¿ max_exc_count_id are considered insufficient data with exceedances) - one of max_exc_count_id or max_exc_pct_id must be specified

Details

When percentage arguments are supplied, specified percentages are used to calculate the maximum allowable number of whole samples exceeding a standard for that site/parameter to be considered supporting. Note that the min_n argument is ¿= and all other arguments are ¿.

Value

Returns input data frame with site-use-parameter-criterion specific assessment categories appended.

---

assessHFDO *Run HFDO assessments*

Description

Performs high frequency dissolved oxygen assessments per IR assessment methods. This includes checking for data sufficiency, calculating daily minima and averages, and 7-day/30-day moving averages.

Usage

```
assessHFDO(data, min_n = 10)
```

Arguments

data	HFDO data–for the time being is a test dataset built by EH using "hfdo_prep.R".
min_n	Minimum sample count for completing an assessment. This is used to generate the necessary number of consecutive days to calculate a moving average (averaging period length + min_n - 1).

Value

List object containing data used for minimum, 7-day, and 30-day assessments, full list of assessments by type, and a rollup to site-use assessments.

`assessLakeProfiles` *Run lake profile assessments*

Description

Performs lake profile assessments per IR assessment methods. This includes identifying stratified profiles & applying appropriate assessment methods for stratified & non stratified profiles.

Usage

```
assessLakeProfiles(data, do_crit = list(`3A` = 4, `3B` = 3),
  temp_crit = list(`3A` = 20, `3B` = 27), uses_assessed = c("3A",
  "3B"))
```

Arguments

`data`	Lake profiles object returned by dataPrep step.
`do_crit`	List of beneficial use classes and associated dissolved oxygen criteria to use for assessment. Defaults to list("3A"=5, "3B"=3). This excludes chronic & ELS present criteria per assessment methods. Sites with site specific criteria will be assessed regardless of criteria specified in this argument. Objects in this list should match the uses_assessed argument.
`temp_crit`	List of beneficial use classes and associated water temperature criteria to use for assessment. Defaults to list("3A"=20, "3B"=27). This excludes chronic & ELS present criteria per assessment methods. Sites with site specific criteria will be assessed regardless of criteria specified in this argument. Objects in this list should match the uses_assessed argument.
`uses_assessed`	Vector of beneficial uses to be assessed for lake profiles. Defaults to 3A & 3B uses.

Value

Returns a list of lake profile assessment dataframes. profile_asmnts_mlid_param contains site/parameter level profile assessments, profile_asmnts_individual contains assessments for each individual profile, profile_criteria contains the criteria used for the profile assessment (excluding any site-specific criteria that may have occurred in the input dataset), profiles_long contains profile data in long format including the numeric criterion associated with each parameter, profiles_wide contains profile data cast to wide format.

assignCriteria *Assign numeric criteria to WQP data*

Description

Assigns general & site-specific numeric WQ criteria to WQP data. Does not calculate criteria w/ correction factors. See calcCriteria (function in development).

Usage

```
assignCriteria(data, crit_wb, crit_sheetname, ss_sheetname,
  crit_startRow = 1, ss_startRow = 1, rm_nocrit = TRUE)
```

Arguments

data	A merged, parameter translated, spatially referenced WQP result object. Must be post parameter translation step.
crit_wb	Full path and filename for workbook containing criteria.
crit_sheetname	Name of sheet in workbook holding criteria to be assigned.
ss_sheetname	Name of sheet in workbook holding site-specific criteria to be assigned. Should typically contain parameters corresponding to those in criteria table.
crit_startRow	Row to start reading criteria table excel sheet from (in case headers have been added). Defaults to 1.
ss_startRow	Row to start reading site specific criteria table excel sheet from (in case headers have been added). Defaults to 1.
rm_nocrit	Logical. If TRUE (default), remove records w/o numeric criteria before returning. If FALSE, function will pass through all records with flattened uses & any matching criteria - useful for passing through non-assessed parameters for other analyses.

Value

A flattened & expanded sample x use x parameter x criterion data.frame.

autoValidateWQPsites *Auto-validate USEPA WQP stations*

Description

Performs auto-validation on previously queried WQP stations combined with the existing master site list. Auto-validates any new sites (i.e. those not currently in the master site list) and appends them into existing master site list flagged for acceptance, rejection, or review. Checks for any new site types in new data. A warning message and a list of new site types is printed if new site types are encountered. Also re-auto-validates any sites in the master site list that have previously only undergone auto-validation (to account for any changes in auto validation process) and checks existing master site list for changes in AUs, selected site types, or property boundaries.

Usage

```
autoValidateWQPsites(sites_object, master_site_file, waterbody_type_file,
  polygon_path, outfile_path, correct_longitude = FALSE)
```

Arguments

sites_object	Sites object queried from WQP to be reviewed.
master_site_file	Full path and filename of master site list as generated by autoValidateWQPsites function and manual site review application (.csv).
waterbody_type_file	Full path and filename of list of waterbody types and their associated IR_FLAGs (.csv)
polygon_path	Full path to folder containing, AU, land ownership, and Utah state boundary 1000 m buffer polygon shapefiles.
outfile_path	Path for file outputs.
ownership_assess	Vector of property ownership labels to be considered in assessment process. Defaults to c("Federal","Private","State") and exclude "Tribal".

Value

Exports a new, undated master site list to the outfile_path. If one already exists in outfile_path, it is moved to the 'edit_logs' folder and renamed with the system date.

---

countExceedances *Calculate sample and exceedance counts (conventional & toxic assessments)*

Description

Compares water quality result values to standards to calculates sample and exceedance counts. This is geared towards conventional and toxic assessments.

Usage

```
countExceedances(data, group_vars = c("IR_MLID", "R317Descrp", "IR_Lat",
  "IR_Long", "ASSESS_ID", "AU_NAME", "BeneficialUse", "BEN_CLASS",
  "R3172ParameterName", "CriterionLabel", "SSC_MLID", "SSC_StartMon",
  "SSC_EndMon", "AsmntAggFun"), agg_exc = FALSE, agg_exc_as_n = TRUE)
```

Arguments

data	A prepped water quality portal data object (i.e. output from dataPrep()). Must include IR_Value and ActivityStartDate.
group_vars	Vector of column names on which to group data when calculating sample counts and exceedances. This should not include any factors that prevent aggregation to site-scale assessments (e.g. date, time, etc.), but should include any columns that indicate a unique standard (e.g. season, acute v. chronic, etc.). See default for recommended.

agg_exc — Logical. If FALSE (default), individual samples are compared to the criterion as with non-aggregated criteria to count exceedances. If TRUE (default) aggregate samples by AsmntAggFun prior to counting exceedances.

agg_exc_as_n — Logical. If FALSE exceedance/support of aggregate water quality criteria (e.g. seasonal means) is indicated as 1 or 0 in ExcCount column. Only used if agg_exc==TRUE. If TRUE (default), set the ExcCount value of aggregate water quality criteria to equal the associated sample count. In this case, ExcCount==SampleCount indicates an exceedance and ExcCount==0 indicates no exceedance. This allows sample and exceedance counts for these types of criteria to pass through assessExc using the same input arguments.

Value

Returns sample and exceedance counts aggregated by grouping variables.

---

dataPrep *Final data prep step before assessments*

Description

Performs unit conversions & daily data aggregations. Also checks native vs. target activities and fractions, dissolved vs. total value checks, river/stream depth & flow checks, and generates value based data flags.

Usage

```
dataPrep(data, translation_wb, unit_sheetname = "unitConvTable", crit_wb,
  cf_formulas_sheetname, startRow_unit = 1, startRow_formulas = 1,
  split_agg_tds = TRUE)
```

Arguments

data — A merged, translated, and numeric criteria assigned WQP results R-object. Target units for conversions are defined by units associated with assigned numeric critera.

translation_wb — Full path and filename for IR translation workbook (.xlsx).

unit_sheetname — Name of sheet in workbook holding IR unit conversion table. Defaults to "unitConvTable".

crit_wb — Full path and filename for workbook containing criteria.

cf_formulas_sheetname — Name of sheet in criterion workbook holding conversion factors and criterion formulas for criteria dependent on CFs.

startRow_unit — Row to start reading the unit conversion table excel sheet from (in case headers have been added). Defaults to 1.

startRow_formulas — Row to start reading the formulas table from (in case headers have been added). Defaults to 1.

split_agg_tds — Logical. If TRUE (default) split off TDS records w/ function assigned in AsmntAggFun into separate output. If FALSE, these records are passed through to conventionals output.

Value

A list of objects ready for assessments.

`fillMaskedValues` *Select quantitation limits, determine detection condition, and fill masked values as appropriate.*

Description

This function selects a single upper and lower quantitation limit for each record in result (if one exists in detquantlim), determines detection condition for each record based on the presence/absence of a result value, upper or lower quant limits, and whether the result value is above or below quant limits or ¡=0.

Usage

```
fillMaskedValues(results, detquantlim, translation_wb,
  detsheetname = "detLimitTypeTable", unitsheetname = "unitConvTable",
  detstartRow = 3, unitstartRow = 1, unitstartCol = 1,
  lql_fac = 0.5, uql_fac = 1)
```

Arguments

`results`	A WQP results (must include narrow result, merged, wide objects OK) R-object name.
`detquantlim`	A WQP detection/quantitation limit file R-object. Should be matching complement to WQP results input.
`translation_wb`	Full path and filename for IR translation workbook (.xlsx).
`detsheetname`	Name of sheet in workbook holding detection limit type names and ranked prioritizations table. Defaults to "detLimitTypeTable".
`unitsheetname`	Name of sheet in workbook holding unit conversion table. Defaults to "unitConvTable".
`detstartRow`	Row to start reading the detLimitTypeTable excel sheet from (in case headers have been added). Defaults to 3.
`unitstartRow`	Row to start reading the unitConvTable excel sheet from (in case headers have been added). Defaults to 1.
`unitstartCol`	Column to start reading the unitConvTable excel sheet from (in case headers have been added). Defaults to 1.
`lql_fac`	Numeric - factor by which to multiply lower quantitation limit type values when filling masked data or other non-detects (e.g. below lql values). Default = 0.5.
`uql_fac`	Numeric - factor by which to multiply upper quantitation limit type values when filling masked data or other over limit values. Default = 1.

Value

Returns a data frame with new columns of selected limits, filled values, units, and detection condition appended to results input. In IR_DetCond column, "ND"=non-detect, "OD"=over detection, "NRV"=no result value & no ranked detection limit, "DET"=detection.

`readAttains` *Read data from EPA's ATTAINS database*

Description

This function pulls information from EPA ATTAINS database based on submitted arguments via ATTAINS web service. Any ATTAINS web service compatible argument can be submitted to this funciton. Depending on type, at least one argument may be required. See https://link-to-be-obtained.gov for a list of required and compatible arguments. The function is essentially an ATTAINS specific wrapper for jsonlite::fromJSON. It generates the appropriate web path to connect to ATTAINS web service and converts JSON to R object.

Usage

```
readAttains(type = "assessments", ...)
```

Arguments

`type`	ATTAINS data type to read. One of: "assessmentUnits", "assessments", "actions", or "domains".
`...`	Additional arguments to be passed to ATTAINS web service path.
`stateCode`	Two letter state code (for all pulls where type != "assessments")
`state`	Two letter state code (for read type == "assessments")
`reportingCycle`	Four digit year of interest. Must be specified for type == "assessments" (NOTE: EPA working on update that will return the most recent cycle).

Value

Returns a data.frame or list of queried ATTAINS data. Note that some results may contain additional lists of results that can be further flattened.

Examples

```
#Read Utah assessment unit information
UT_AUs=readAttains(type="assessmentUnits", stateCode="UT")

#Read Utah actions
UT_actions=readAttains(type="actions", stateCode="UT")

#Read Utah assessments
UT_assessments=readAttains(type="assessments", state="UT", reportingCycle=2016)
```

`readWQPFiles` *Read WQP Files*

Description

Creates list object containing data frames needed to run irTools. Removes duplicates and creates objects containing orphan records. Identifies non-numeric values in numeric columns and allows user to edit values and save to new object.

Usage

```
readWQPFiles(file_select, narrowresult_file, sites_file, activity_file,
  detquantlim_file, orph_check = TRUE)
```

Arguments

`file_select`	Logical. If TRUE, navigation window will pop up for manual selection of files. If FALSE, function requires full path names to files.
`narrowresult_file`	Full path and filename of narrowresult file queried from WQP to be reviewed (.csv).
`sites_file`	Full path and filename of sites file queried from WQP to be reviewed (.csv).
`activity_file`	Full path and filename of activity file queried from WQP to be reviewed (.csv).
`detquantlim_file`	Full path and filename of detquantlim file queried from WQP to be reviewed (.csv).
`orph_check`	Logical. Specifies whether function should perform orphan check between narrowresult and sites, activity, and detquantlim. If TRUE, produces list object containing orphan rows.

Value

List containing dataframes needed to run irTools package, as well as orphan records and edited non-numeric data tables.

`rollUp` *Summarize assessments to different spatial resolutions*

Description

Summarizes and aggregates assessments at one spatial resolution to another spatial resolution. For example, a collection of assessments conducted at the site/use/parameter level can be rolled up to the AU/use/parameter level or to the AU/use/level.

Usage

```
rollUp(data, group_vars = c("ASSESS_ID", "BeneficialUse",
  "R3172ParameterName"), expand_uses = TRUE, print = TRUE)
```

Arguments

data	A list of assessment output objects to be rolled up to a different spatial resolution.
group_vars	Vector of column names on which to group data for assessment rollups. Defaults to aggregate by ASSESS_ID, BeneficialUse, and R3172ParameterName.
expand_uses	Logical. If TRUE (default), uses are expanded in the output to include all uses associated with group_vars, including unassessed groups which are marked as 'NA' in output dataframe column AssessCat. If FALSE, only assessed groups are included in the output.
print	Logical. If TRUE (default) print summary table of applicable assessment categories & parameters.

Value

Returns dataframe with assessment categories for each AU/BenUse/R3172ParameterName.

---

runSiteValApp *Run site validation shiny app Runs site validation shiny app embedded in UT IR tools package.*

Description

Run site validation shiny app Runs site validation shiny app embedded in UT IR tools package.

Usage

```
runSiteValApp(master_site_file, polygon_path, edit_log_path,
  reasons_flat_file)
```

---

updateCommentTable *Identify New WQP comment combinations and append them into IR Parameter Translation Workbook for review (sheet name activityCommentTable)*

Description

This function identifies new ResultLaboratoryCommentCode/ResultLaboratoryCommentText/ActivityComme
combinations in WQP data for which an IR_FLAG has not yet been defined and appends them into the translation workbook for review.

Usage

```
updateCommentTable(data, translation_wb,
  sheetname = "activityCommentTable", startRow = 1, startCol = 1)
```

Arguments

`data`	A WQP results (must include activity, merged objects OK) R-object or the full path and filename to a WQP result file (.csv). If input is a path/file name, datatype must be specified as "filepath".
`translation_wb`	Full path and filename for IR translation workbook (.xlsx).
`sheetname`	Name of sheet in workbook holding comment table. Defaults to "activityCommentTable".
`startRow`	Row to start reading the activityCommentTable excel sheet from (in case headers have been added). Defaults to 1.
`startCol`	Col to start writing the activityCommentTable excel sheet to (to the right of all formula based columns). Defaults to 1.

Value

Appends any new combinations in WQP data to translation_wb for review. This updates the input translation_wb with those new rows with system date in the "DateAdded" column.

`updateDetCondLimTables` *Update detection condition and limit type tables*

Description

This function identifies new ResultDetectionConditionText & DetectionQuantitationLimitTypeName values in WQP data for which are not in IR translation tables and appends them into the translation workbook for review. New ResultDetectionConditionText & DetectionQuantitationLimitTypeName values generated by this function are derived from those in detquantlim that match (via merge) records in results input (this allows subsetting of these lists by other previously generated data screens if desired).

Usage

```
updateDetCondLimTables(results, detquantlim, translation_wb,
  detConditionTable_sheetname = "detConditionTable",
  detConditionTable_startRow = 3, detConditionTable_startCol = 1,
  detLimitTypeTable_sheetname = "detLimitTypeTable",
  detLimitTypeTable_startRow = 3, detLimitTypeTable_startCol = 1)
```

Arguments

`results`	A WQP results (must include narrow result, merged, wide objects OK) R-object name.
`detquantlim`	A WQP detection/quantitation limit file R-object. Should be matching complement to WQP results input. New values derived from those that successfully merge to results input.
`translation_wb`	Full path and filename for IR translation workbook (.xlsx).
`detConditionTable_sheetname`	Name of sheet in workbook holding result detection condition names table. Defaults to "detConditionTable".

detConditionTable_startRow
: Row to start reading the detConditionTable excel sheet from (in case headers have been added). Defaults to 3.

detConditionTable_startCol
: Col to start writing the detConditionTable excel sheet to (to the right of all formula based columns). Defaults to 1.

detLimitTypeTable_sheetname
: Name of sheet in workbook holding detection limit type names and ranked prioritizations table. Defaults to "detLimitTypeTable".

detLimitTypeTable_startRow
: Row to start reading the detLimitTypeTable excel sheet from (in case headers have been added). Defaults to 3.

detLimitTypeTable_startCol
: Col to start writing the detLimitTypeTable excel sheet to (to the right of all formula based columns). Defaults to 1.

Value

Appends any new values in WQP data to translation_wb for review. This updates the input translation_wb with those new rows.

---

updateLabActMediaTables

Update labNameActivityTable & activityMediaNameTable

Description

This function identifies new LaboratoryName/ActivityTypeCode combinations & ActivityMediaName/ActivityMediaSubdivisionName combinations in WQP data for which an IR_FLAG has not yet been defined and appends them into the translation workbook for review.

Usage

```
updateLabActMediaTables(data, translation_wb,
  labNameActivityTable_sheetname = "labNameActivityTable",
  labNameActivityTable_startCol = 1, labNameActivityTable_startRow = 1,
  activityMediaNameTable_sheetname = "activityMediaNameTable",
  activityMediaNameTable_startCol = 1,
  activityMediaNameTable_startRow = 1)
```

Arguments

data
: A WQP results (must include activity, merged objects OK) R-object

translation_wb
: Full path and filename for IR translation workbook (.xlsx).

labNameActivityTable_sheetname
: Name of sheet in workbook holding lab name / activity table. Defaults to "labNameActivityTable".

labNameActivityTable_startCol
: Col to start writing the labNameActivityTable excel sheet to (to the right of all formula based columns). Defaults to 3.

`labNameActivityTable_startRow`
: Row to start reading the labNameActivityTable excel sheet from (in case headers have been added). Defaults to 1.

`activityMediaNameTable_sheetname`
: Name of sheet in workbook holding activity / media/submedia name table. Defaults to "activityMediaNameTable".

`activityMediaNameTable_startCol`
: Col to start writing the activityMediaNameTable excel sheet to (to the right of all formula based columns). Defaults to 3.

`activityMediaNameTable_startRow`
: Row to start reading the activityMediaNameTable excel sheet from (in case headers have been added). Defaults to 1.

Value

Appends any new combinations in WQP data to translation_wb for review. This updates the input translation_wb with those new rows with system date in the "DateAdded" column.

`updateParamTrans` *Update paramTransTable, paramFractionGroup, & WQPParamCASID tables in translation workbook*

Description

This function identifies parameter/fraction/unit/method combinations in WQP data for which an IR translation has not yet been defined and appends them to the translation table for review.

Usage

```
updateParamTrans(data, detquantlim = detquantlim, translation_wb,
  paramTransTable_sheetname = "paramTransTable",
  paramTransTable_startRow = 4, paramTransTable_startCol = 16,
  WQPParamCASID_sheetname = "WQPParamCASID",
  WQPParamCASID_startRow = 4, WQPParamCASID_startCol = 1,
  paramFractionGroup_sheetname = "paramFractionGroup",
  paramFractionGroup_startRow = 3, paramFractionGroup_startCol = 3)
```

Arguments

`data` A WQP results (merged or narrow result) R-object

`detquantlim` WQP detection/quantitation limit object

`translation_wb` Full path and filename for IR translation workbook

`paramTransTable_sheetname`
: Name of sheet in workbook holding parameter translation table. Defaults to "paramTransTable".

`paramTransTable_startRow`
: Row to start reading the paramTransTable excel sheet from (in case additional headers have been added). Defaults to 4.

paramTransTable_startCol
: Col to start writing the paramTransTable excel sheet to (to the right of all formula based columns). Defaults to 16.

WQPParamCASID_sheetname
: Name of sheet in workbook holding parameter/CAS ID table. Defaults to "WQPParamCASID".

WQPParamCASID_startRow
: Row to start reading the WQPParamCASID excel sheet from (in case additional headers have been added). Defaults to 3.

WQPParamCASID_startCol
: Col to start writing the WQPParamCASID excel sheet to (to the right of all formula based columns). Defaults to 1.

paramFractionGroup_sheetname
: Name of sheet in workbook holding the WQP fraction names. Defaults to "paramFractionGroup".

paramFractionGroup_startRow
: Row to start reading the paramFractionGroup excel sheet from (in case additional headers have been added). Defaults to 3.

paramFractionGroup_startCol
: Col to start writing the paramFractionGroup excel sheet to (to the right of all formula based columns, note col w/ header 'spacer' in table). Defaults to 3.

Value

Appends any new parameter/fraction/unit combinations in WQP data to translation_wb. This updates the input translation_wb with those new rows with system date in the "DateAdded" column.

`updateUnitConvTable` *Update IR unit conversion table*

Description

Updates IR unit conversion table based on combinations of native and target units as defined by numeric criteria tables.

Usage

```
updateUnitConvTable(data, translation_wb, sheetname = "unitConvTable",
  startRow = 1, startCol = 1)
```

Arguments

data
: A merged WQP results-activity R-object that has had detection limit values filled (passed through fillMaskedValues function) and numeric criteria assigned (passed through assignCriteria function).

translation_wb
: Full path and filename for IR translation workbook (.xlsx).

sheetname
: Name of sheet in workbook holding IR unit conversion table. Defaults to "unitConvTable".

`startRow` Row to start reading the unit conversion excel sheet from (in case headers have been added). Defaults to 1.

`startCol` Column to start reading the unit conversion excel sheet from (in case columns have been added). Defaults to 1.

Value

Updates unit conversion table in translation_wb with any new combinations of native and target units.

Index