Package: TAMMsupport (via r-universe)

November 14, 2024

Title Streamline working with Terminal Area Management Modules

Version 0.1.0

Description A convenient tool for interfacing with Terminal Area Manamagement Modules (TAMMs) in R environments.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Imports cli, dplyr, janitor, kableExtra, openxlsx, purrr, quarto, readxl, rlang, stats, stringr, tibble, tidyr, tidyxl, utils, xldiff

Remotes cbedwards-dfw/xldiff

Depends R (>= 2.10)

LazyData true

Config/pak/sysreqs libfontconfig1-dev libfreetype6-dev make libicu-dev libpng-dev libxml2-dev

Repository https://framverse.r-universe.dev

RemoteUrl https://github.com/cbedwards-dfw/TAMMsupport

RemoteRef HEAD

RemoteSha 9e7c64b7781768ceef5cf265ddf63d2d826c67cd

Contents

`chunk_formater_percenter` *Format chunks of dataframe to present as %s, round digits*

Description

Format chunks of dataframe to present as %s, round digits

Usage

```
chunk_formater_percenter(df, block.ranges, percent.digits = 1)
```

Arguments

`df`	dataframe of sheet to apply formatting to.
`block.ranges`	vector of characters specifying blocks of cells (in excel nomenclature) to format as %s
`percent.digits`	Decimal place to round to in percent

Value

Formatted version of df

`chunk_formater_rounder` *Format chunks of dataframe to round digits*

Description

Format chunks of dataframe to round digits

Usage

```
chunk_formater_rounder(df, block.ranges, digits = 1)
```

Arguments

df	dataframe of sheet to apply formatting to.
block.ranges	vector of characters specifying blocks of cells (in excel nomenclature) to format as %s
digits	Decimal place to round to.

Value

Formatted version of df

`clean_limiting_stock` *Generates clean read of TAMM limiting stock sheet*

Description

Effectively a wrapper function for read_limiting_stock with some formatting added in. Filters to unmarked naturals, just present ER values.

Usage

```
clean_limiting_stock(filename)
```

Arguments

filename	Name (and path) for TAMM files

Value

dataframe

filter_tamm_wa *Filters a dataframe to Washington State fisheries.*

Description

Adapted from framrsquared::filter_wa(), but (a) only for Chinook (at present), and (b) uses TAMM fishery ids, so includes ids 72 and 73. Includes code for COHO based on FRAM ids. `filter_tamm_wa()` uses attributes to specify chinook or coho. To directly filter for chinook or coho, use `filter_tamm_wa_chin()` or `filter_tamm_wa_coho()`.

Usage

```
filter_tamm_wa(.data)

filter_tamm_wa_chin(.data)

filter_tamm_wa_coho(.data)
```

Arguments

`.data`	Dataframe generated within this package

Details

a `fishery_id` column name.

Examples

```
## Not run:
fram_dataframe |> filter_wa()

## End(Not run)
```

find_color_matches *Identify all cells in Excel sheet or sheets with matching cell colors*

Description

Designed for tracking the status of TAMM inputs, which are typically color coded. Searches through specified sheet or sheets, looking for cells whose colors match the `target_cell` argument. Returns a condensed list of those entries.

Usage

```
find_color_matches(file, target_cell, sheets = "Input Page")
```

Arguments

file	Character string of excel file name, including path if relevant
target_cell	Character string of excel address of cell with the cell shading of interest (e.g., "B4")
sheets	character string or character vector of sheets to search through

Value

tibble summarizing entries with the color of interest, including sheet, row and column ids, and row name (entry of first column in sheet)

Examples

```
## Not run:
file = "chin tamms/Chin2023_Final.xlsx"
find_color_matches(file, "B20")

## End(Not run)
```

---

fishery_chinook_fram *Chinook fishery information*

Description

Mapping of fishery_id to fishery name for Chinook salmon, taken from the `Fishery` table of the Chinook FRAM database.

Usage

```
fishery_chinook_fram
```

Format

A data frame with 74 rows and 5 columns:

species Species name

version_number

fishery_id Chinook fishery id number in FRAM

fishery_name Chinook fishery name in FRAM

fishery_title consistent and more human readable version of `fishery_name`

Source

2024 Pre-Season Chinook DB.mdb

fishery_coho_fram *Coho fishery information*

Description

Mapping of fishery_id to fishery name for Coho salmon, taken from the `Fishery` table of the Coho FRAM database.

Usage

```
fishery_coho_fram
```

Format

A data frame with 198 rows and 5 columns:

species Species name

version_number

fishery_id Coho fishery id number in FRAM

fishery_name Coho fishery name in FRAM

fishery_title consistent and more human readable version of `fishery_name`

Source

2024NOF_CohoFRAMdatabase_distribution.mdb

fishery_renamer *relabel inconsistent fishery names in FRAM/TAMM*

Description

With argument `sep = TRUE`, will return data frame with separate columns for the area and the the "class" (Net, Troll, Sport)

Usage

```
fishery_renamer(x, sep = FALSE)
```

Arguments

`x`	character vector of fishery names from TAMM or FRAM
`sep`	Should we return a dataframe with separate column for class of fishing? Defaults to FALSE

Value

#vector with cleaned fishery names OR dataframe with cleaned name (`$full`), area without fishery type (`$area`) and type of fishery (`$class`).

`format_key_tamm_sheets_chin` *Modify list of Chinook TAMM spreadsheet dataframes to facilitate comparison.*

Description

Modify list of Chinook TAMM spreadsheet dataframes to facilitate comparison.

Usage

```
format_key_tamm_sheets_chin(
  dat,
  percent.digits = 1,
  numeric.digits = 1,
  numeric.digits.small = 4
)
```

Arguments

`dat`	list of dataframes corresponding to the overview, limiting stock, and inputs tabs. Must be named `$overview`, `$limiting`, and `$input`
`percent.digits`	Optional, number of decimal places to round percentages to before comparison. Defaults to 1.
`numeric.digits`	Optional, number of decimal places to round non-percentage numerics to before comparison. Applied to numbers that are expected to have natural units of whole numbers (e.g. numbers of fish). Defaults to 1.
`numeric.digits.small`	Optional, number of decimal places to round non-percentage numerics to before comparison. Applied to numbers that are expected to be small (e.g. rates, proportions) Defaults to 4.

Value

list of dataframes with same structure as `dat`, contents modified.

`format_key_tamm_sheets_coho` *Modify list of Coho TAMM spreadsheet dataframes to facilitate comparison.*

Description

Modify list of Coho TAMM spreadsheet dataframes to facilitate comparison.

Usage

```
format_key_tamm_sheets_coho(
  dat,
  percent.digits = 2,
  numeric.digits = 1,
  numeric.digits.small = 4
)
```

Arguments

`dat`	list of dataframes corresponding to the overview, limiting stock, and inputs tabs. Must be named `$overview`, `$limiting`, and `$input`
`percent.digits`	Optional, number of decimal places to round percentages to before comparison. Defaults to 1.
`numeric.digits`	Optional, number of decimal places to round non-percentage numerics to before comparison. Applied to numbers that are expected to have natural units of whole numbers (e.g. numbers of fish). Defaults to 1.
`numeric.digits.small`	Optional, number of decimal places to round non-percentage numerics to before comparison. Applied to numbers that are expected to be small (e.g. rates, proportions) Defaults to 4.

Value

list of dataframes with same structure as `dat`, contents modified.

`format_overview` *Format TAMM overview for easy printing*

Description

Format TAMM overview for easy printing

Usage

```
format_overview(dat.overview)
```

Arguments

dat.overview	dataframe of TAMM overview page, as produced by read_overview()

Value

list with the formatted data, a vector of indent information, and a vector of bolding information.

---

fun_percenter *Format vector of mixed characters to present numbers as percents*

Description

Intended for internal use within formatting functions. Note that percents in excel are read in as proportions in R – this makes them percents to make saved files more readable.

Usage

```
fun_percenter(x, percent.digits)
```

Arguments

x	Character vector, presumably containing some entries that are numbers in character form
percent.digits	Number of digits to round to after converting to percents

Value

character vector with individual entries converted to percentages and rounded as appropriate.

---

fun_rounder *Format vector of mixed characters to round numbers to specified digits*

Description

Intended for internal use within formatting functions.

Usage

```
fun_rounder(x, digits)
```

Arguments

x	Character vector, presumably containing some entries that are numbers in character form
digits	Number of digits to round numeric items to

---

kable_overview *Print table of overview using kable and kableExtra*

Description

Uses output of format_overview()

Usage

```
kable_overview(dat.overview, ind.indent, col.bold)
```

Arguments

dat.overview	formatted data, first item of format_overview() output
ind.indent	Vector of indices that need indenting to match TAMM overview formatting. Second item of format_overview() output
col.bold	vector of names for bolding to match TAMM overview formatting. Third item of format_overview() output.

Value

html table

---

quick_er_format *Quick helper function to format vectors that contain text and ERs*

Description

Quick helper function to format vectors that contain text and ERs

Usage

```
quick_er_format(x)
```

Arguments

x	character vector

`read_key_tamm_sheets_chin` *Read Chinook TAMM files to extract the key sheets*

Description

Read Chinook TAMM files to extract the key sheets

Usage

```
read_key_tamm_sheets_chin(xlsxFile)
```

Arguments

`xlsxFile`	#Tamm file name

Value

List of dataframes: `$overview`, `$limiting`, and `$input`

`read_key_tamm_sheets_coho` *Read Chinook TAMM files to extract the key sheets*

Description

Read Chinook TAMM files to extract the key sheets

Usage

```
read_key_tamm_sheets_coho(xlsxFile)
```

Arguments

`xlsxFile`	#Tamm file name

Value

List of dataframes: `$two`, `$tami`, and `$wacoast`

read_limiting_stock *Read TAMM limiting stock complete tab*

Description

Reads in the table, adds fishery type information.

Usage

```
read_limiting_stock(filename, longform = FALSE)
```

Arguments

filename	Tamm name (and path)
longform	Should results be in long form (good for R stuff) (TRUE) or replicate the structure of the TAMM sheet (FALSE). Logical, defaults to FALSE.

Value

data frame summarizing the TAMM limiting stock tab.

read_management_chin *Extract and format management criterion from Chinook TAMM*

Description

Extract and format management criterion from Chinook TAMM

Usage

```
read_management_chin(file)
```

Arguments

file	filename, including filepath if appropriate

Value

Tibble, with $management_name giving the name of the management criteria as specified in the TAMM, $management_criteria giving a list of dataframes with the associated management criteria, formatted appropriately, and $notes giving a list of character vectors with notes (if any) to help interpret the criteria.

Examples

```
## Not run:
cur.management = read_management_chin("Code Inputs/Pre-Season/TAMMs/2013_Final W_BP7.1.xlsx")

## End(Not run)
```

`read_overview` *Read overview sheet from TAMM and return key stock*

Description

Read overview sheet from TAMM and return key stock

Usage

```
read_overview(path)
```

Arguments

`path`	filename (including path) to Chinook TAMM

Value

dataframe

`read_overview_complete` *Read overview sheet from TAMM and return all stock*

Description

The stock column of the overview tab uses R-unfriendly approaches to store season, overall stock name, and individual stocks in the `stock` column of the TAMM. `read_overview_complete` splits these into separate columns: `season`, `primary_stock`, and `stock`. For three stock (Green, Puyallup, Skokomish), the tamm presents model predictions across two rows with a single stock name due to merged cells. The `stock` column in the output preserves this information; the entry corresponding to the second row has the suffix "_row2".

Usage

```
read_overview_complete(path)
```

Arguments

`path`	filename (including path) to Chinook TAMM

Value

Dataframe containing all infomration from cells A2:H34 of the "ER_ESC_Overview_New" sheet.

`read_tnt_allocation_chin` **[Experimental]**

Description

Extract Treaty and Nontreaty numbers from "2A_CU&M_H+N" TAMM sheet. For Elwha and Dungeoness, separate allocations are calculated using column O of the JDF tab

Usage

```
read_tnt_allocation_chin(xlsxFile)
```

Arguments

`xlsxFile`	Character vector. Filename (including path) for chinook TAMM

Details

Currently provides dataframe with `stock.original` column identifying the stock based on their names in the TAMM, and `stock.clean` with more human-readable names (which are generally consistent with other stock names, like those used to label the management criterion).

Value

dataframe with the treaty and nontreaty mortalities at each stock.

`reformat_management` *Convenience function for formatting TAMM management criterion chunks*

Description

Internal tool to simplify converting TAMM sheet to stock management criterion lists. See `read_management_chin()` for example use.

Usage

```
reformat_management(
  df,
  cols.percent,
  rows.percent = NULL,
  rows.header = 2,
  notes = NULL
)
```

Arguments

df	dataframe trimmed from TAMM management page to just the relevant stock's criterion.
cols.percent	Column numbers for columns that should be converted to percents.
rows.percent	Character vector of row sub-stock names identifying rows (if any) that should be translated to percents. Defaults to NULL. Developed to accomodate "CERC (SUS)" in Snohomish.
rows.header	How many rows are used to make up the header? Defaults to 2, but sometimes header block in TAMM is 1 row.
notes	Optional. Character vector, each item of which is a separate note associated with this stock. Generally based on comments in excel doc, can also feed in "extra" rows from TAMM that contain caveats.

Value

Stock management criterion list, containing dataframe extracted from TAMM and notes hard-coded in based on TAMM cell comments.

---

stock_chinook_fram *Chinook stock information*

Description

Mapping of stock_id to stock name for Chinook salmon, taken from the `Stock` table of the Chinook FRAM database.

Usage

```
stock_chinook_fram
```

Format

A data frame with 78 rows and 7 columns:

species Species name

stock_version

stock_id Chinook stock id number in FRAM

production_region_number

management_unit_number

stock_name Chinook stock name in FRAM

stock_long_name `stock_name` but more human readable

Source

2024 Pre-Season Chinook DB.mdb

stock_coho_fram *Coho stock information*

Description

Mapping of stock_id to stock name for Coho salmon, taken from the `Stock` table of the Coho FRAM database.

Usage

```
stock_coho_fram
```

Format

A data frame with 78 rows and 7 columns:

species Species name

stock_version

stock_id Coho stock id number in FRAM

production_region_number

management_unit_number

stock_name Coho stock name in FRAM

stock_long_name `stock_name` but more human readable

Source

2024NOF_CohoFRAMdatabase_distribution.mdb

tamm_compare *Generate summary comparison of any number of TAMM files*

Description

Intended use case is to compare low, mid, and high ocean option.

Usage

```
tamm_compare(
  tamm.names,
  tamm.path = getwd(),
  fisheries = NULL,
  clean = TRUE,
  overwrite = TRUE
)
```

Arguments

tamm.names	character vector of the three tamm files to compare (including .xlsx suffix).
tamm.path	Absolute path to directory containing the tamm file. here::here() can be useful in identifying appropriate path. Character atomic; defaults to current working directory.
fisheries	Optional, numeric vector of fishery IDs to filter to before plotting fishery-specific ER for each stock. Defaults to NULL (no filtering).
clean	Should the intermediate .qmd files used to make the report be deleted afterwards? Logical, defaults to TRUE. Set to FALSE to explore the .qmd files underlying the report.
overwrite	Should the intermediate .qmd files or the final report be overwritten if those files already exist?; Logical, defaults to TRUE.

Value

Nothing

Examples

```
## Not run:
tamm.path <- "C:/Users/edwc1477/Documents/WDFW FRAM team work/NOF material/NOF 2024/FRAM"
tamm.names <- c("Chin1024.xlsx", "Chin1124.xlsx", "Chin1224.xlsx")
tamm_compare3(tamm.names = tamm.names, tamm.path = tamm.path)

## End(Not run)
```

---

tamm_compare3 *Generate summary comparison of 3 TAMM files*

Description

Intended use case is to compare low, mid, and high ocean option.

Usage

```
tamm_compare3(tamm.names, tamm.path = getwd(), clean = TRUE, overwrite = TRUE)
```

Arguments

tamm.names	character vector of the three tamm files to compare (including .xlsx suffix).
tamm.path	Absolute path to directory containing the tamm file. here::here() can be useful in identifying appropriate path. Character atomic; defaults to current working directory.
clean	Should the intermediate .qmd files used to make the report be deleted afterwards? Logical, defaults to TRUE. Set to FALSE to explore the .qmd files underlying the report.
overwrite	Should the intermediate .qmd files or the final report be overwritten if those files already exist?; Logical, defaults to TRUE.

Value

Nothing

Examples

```
## Not run:
tamm.path <- "C:/Users/edwc1477/Documents/WDFW FRAM team work/NOF material/NOF 2024/FRAM"
tamm.names <- c("Chin1024.xlsx", "Chin1124.xlsx", "Chin1224.xlsx")
tamm_compare3(tamm.names = tamm.names, tamm.path = tamm.path)

## End(Not run)
```

`tamm_diff` *Compare key sheets of TAMM files*

Description

Compare key sheets of TAMM files

Usage

```
tamm_diff(
  filename.1,
  filename.2,
  results.name,
  percent.digits = 1,
  numeric.digits = 1,
  numeric.digits.small = 4,
  dim.override = FALSE,
  wrap.text = FALSE
)
```

Arguments

`filename.1`	name of first TAMM file to compare. Include file path if file is not in working directory.
`filename.2`	name of second TAMM file to compare. Include file path if file is not in working directory.
`results.name`	name of output sheets. Include file path if save location is not in working directory.
`percent.digits`	Number of decimals to round percentages to before comparing. Defaults to 1.
`numeric.digits`	Number of decimals to round numbers to before comparing. Applied to cells which expect to be whole numbers (e.g. #s of fish). Defaults to 1.
`numeric.digits.small`	Number of decimals to round numbers to before comparing. Applied to cells which expect to be small decimals. Defaults to 4.

dim.override	Should we force comparisons even if one or more of the sheets don't have matching dimensions between the two files? Defaults to FALSE.
wrap.text	Should specific cells with long contents (e.g., input "Fishery Description" cells) use text wrapping? Defaults to FALSE

Details

If TAMM formatting is changes (e.g. the addition of rows, etc), make changes in the following areas:

- `read_key_tamm_sheets_SPECIES()`: The `range` argument in each `read_excel` call should change to match the new dimensions of each sheet.
- `format_key_tamm_sheets_SPECIES()`: These functions designate groups of cells to be Depending on the sheets that change, any amount of the content here may need to change. In the case of the inputs tab section, the cell ranges can be reported directly. Remember that we are separately designating cells which should be rounded to the nearest `numeric.digits` (generally measures of fish) and those that should be rounded to the nearest `numeric.digits.small` (generall proportions and rates; typically values that are less than 1.). Note that the earlier code was written before the more flexible `chunk_formater_percenter` and `chunk_formater_rounder` had been developed. If rewriting, lean into those tools, as they will streamline designating regions of cells for the various rounding criterion.
- `tamm_format_SHEETNAME()` (for the sheet formatting functions relevant to that species): each individual sheet has custom formatting to generally match the corresponding TAMM sheets. If the locations of cells move, the changes to font size, addition of borders, etc, will also need to move. Note that `tamm_format_limiting` and `tamm_format_overview` were written before the development of the more flexible `add_cell_borders`, or the combined use of `purrr::map` and `cell_range_translate`. Look to `tamm_format_input` for relatively inputting of formatting. Consider developing other helper functions as needed (esp for merging).

Examples

```
## Not run:
tamm_diff(
  filename.1 = here("FRAM/Chin1124.xlsx"),
  filename.2 = here("NOF 2/Chin2524.xlsx"),
  results.name = here("Chin 1124 vs Chin 2524.xlsx")
)

## End(Not run)
```

---

tamm_diff_chin *Compare key sheets of Chinook TAMM files*

Description

Compare key sheets of Chinook TAMM files

Usage

```
tamm_diff_chin(
  filename.1,
  filename.2,
  results.name,
  percent.digits = 1,
  numeric.digits = 1,
  numeric.digits.small = 4,
  dim.override = FALSE,
  wrap.text = FALSE
)
```

Arguments

filename.1	name of first TAMM file to compare. Include file path if file is not in working directory.
filename.2	name of second TAMM file to compare. Include file path if file is not in working directory.
results.name	name of output sheets. Include file path if save location is not in working directory.
percent.digits	Number of decimals to round percentages to before comparing. Defaults to 1.
numeric.digits	Number of decimals to round numbers to before comparing. Applied to cells which expect to be whole numbers (e.g. #s of fish). Defaults to 1.
numeric.digits.small	Number of decimals to round numbers to before comparing. Applied to cells which expect to be small decimals. Defaults to 4.
dim.override	Should we force comparisons even if one or more of the sheets don't have matching dimensions between the two files? Defaults to FALSE.
wrap.text	Should specific cells with long contents (e.g., input "Fishery Description" cells) use text wrapping? Defaults to FALSE

---

tamm_diff_coho *Compare key sheets of Coho TAMM files*

Description

Compare key sheets of Coho TAMM files

Usage

```
tamm_diff_coho(
  filename.1,
  filename.2,
  results.name,
```

```
  percent.digits = 2,
  numeric.digits = 1,
  numeric.digits.small = 4,
  dim.override = FALSE,
  wrap.text = FALSE
)
```

Arguments

filename.1	name of first TAMM file to compare. Include file path if file is not in working directory.
filename.2	name of second TAMM file to compare. Include file path if file is not in working directory.
results.name	name of output sheets. Include file path if save location is not in working directory.
percent.digits	Number of decimals to round percentages to before comparing. Defaults to 1.
numeric.digits	Number of decimals to round numbers to before comparing. Applied to cells which expect to be whole numbers (e.g. #s of fish). Defaults to 1.
numeric.digits.small	Number of decimals to round numbers to before comparing. Applied to cells which expect to be small decimals. Defaults to 4.
dim.override	Should we force comparisons even if one or more of the sheets don't have matching dimensions between the two files? Defaults to FALSE.
wrap.text	Should specific cells with long contents (e.g., input "Fishery Description" cells) use text wrapping? Defaults to FALSE

---

tamm_format_sheets *Apply final formatting to diff'd TAMM sheets*

Description

Functions that add formatting to broadly replicate the formatting of the TAMM sheets. Colored foregrounds have been toned down to help change highlighting pop, and some of the complicated or superfluous formatting has been skipped.

Usage

```
tamm_format_overview(wb, diff.sheet, tabname = "overview")

tamm_format_limiting(wb, tabname = "limiting")

tamm_format_input(wb, tabname = "input", wrap.text = FALSE)

tamm_format_wacoast(wb, tabname = "wacoast")
```

Arguments

wb	openxlsx workbook object containing $overview, limiting and $input sheets with the contents of diffing two TAMM files.
diff.sheet	For tamm_format_overview, the output of xldiff::sheet_comp of the overview tab. Used to programmatically bold the appropriate ERs based on the management objective.
tabname	Name of sheet in the wb to be modified. Defaults to correct value.
wrap.text	Should specific cells with long content use text wrapping? Defaults to FALSE.

---

tamm_report *Generate report of figures from TAMM file*

Description

Generate report of figures from TAMM file

Usage

```
tamm_report(
  tamm.name,
  tamm.path = getwd(),
  clean = TRUE,
  overwrite = TRUE,
  additional.children = NULL,
  additional.support.files = NULL
)
```

Arguments

tamm.name	Name of tamm file (including .xlsx suffix). Character atomic
tamm.path	Absolute path to directory containing the tamm file. here::here() can be useful in identifying appropriate path. Character atomic; defaults to current working directory.
clean	Should the intermediate .qmd files used to make the report be deleted afterwards? Logical, defaults to TRUE. Set to FALSE to explore the .qmd files underlying the report.
overwrite	Should the intermediate .qmd files or the final report be overwritten if those files already exist?; Logical, defaults to TRUE.
additional.children	Optional argument with filepath(s) to additional quarto child documents (see Details). Defaults to NULL.
additional.support.files	Optional argument with filepath(s) to additional files needed by additional quarto documents. Files will be copied into the same directory as the TAMM for easy reading/use by additional.children quarto files, and then deleted after the report is generated

Details

This function generates a summary report of a single chinook TAMM (Terminal Area Management Module) file. The common use case is to call this function for a tamm, and then view the resulting html report (which will be generated in the same folder as the TAMM). However, this function also provides substantial flexibility in the form of optional user-created child quarto documents, which requires some additional context to develop and use.

`tamm_report` is implemented using a parameterized quarto report and a child quarto document, which are included in the package. The resulting report includes some broadly useful content – currently, a replication of the overview TAMM sheet and bar charts showing the breakdown of exploitation rates by fishery for each stock listed in the `limiting stock complete` TAMM sheet. However, individuals or organizations may consistently want additional, specific visualizations based on their own needs. For example, individual tribes may want to visualize impacts on a single stock, but use AEQ instead of ER rates. The `tamm_report` function is designed to integrate these extra components with the `additional.children` argument.

The basic principle is that the individual or group can write a reusable "child" .qmd file, see this explanation using Rmarkdown), which will effectively be inserted into the main report. As an example, we might create a simple additional child .qmd file called `extra-tamm-child.qmd` to include a small table summarizing a few key aspects of the Nooksack Earlies stock. The contents of that file are provided in the example below.

When we call `tamm_report()`, we can include as an argument `additional.children =` with the file name (including file path) for our new file. In this way we can easily include this extra content any time we create a tamm report. If desired, multiple child files can be created, and `additional.children` can take a character vector with each child file name.

When writing child documents, formatting and chunks work akin to copy-pasting a section of a quarto file out of the main file. (To easily access the main quarto file, run tamm_report() with `clean = FALSE`, which will leave `tamm-visualizer.qmd` and `tamm-visualizer-child.qmd` files in the folder with the TAMM file). The child document will have access to key objects

- `dat.all`: the output of `read_limiting_stock` with `longform = TRUE` called on the TAMM;
- `dat.overview`: the output of of `read_overview` called on the TAMM;
- and `dat`: the specific filtered and formatted version of the limiting stock data generated in the `limiting stock tab` chunk of `tamm-visualizer.qmd` and used in generating the unmarked natural stock exploitation rate figures. When developing a child document, it may be useful to create in the working environment versions of `dat.all` and `dat.overview` from an example TAMM to facilitate writing appropriate filters, plot-making code, etc. The primary quarto report loads in the tidyverse, framrsquared, kableExtra, and TAMMsupport packages, but you can include additional packages for the child documents as needed by adding `library(packagename` calls just like in any other context.

If the `additional.children` files need to read contents of additional files (or call children .qmd files of their own to streamline looping over stocks or fisheries), the file paths to these supplemental files can be provided in the `additional.support files` argument. These files will be copied into the same folder as the .qmd files, so the children quarto files can be written to access the supplemental files without dealing with file paths.

For questions/support in developing child documents to extend the report, contact Collin Edwards at WDFW.

Value

Nothing

Examples

```
## Not run:
##### CONTENTS FOR extra-tamm-child.qmd #####----------------------------
## Test section

Hypothetical world in which we want a table summarizing the impacts of three
fisheries on the "Nooksack Earlies" stock.

```{r}
kable(dat |> filter(stock == "Nooksack Earlies") |> filter(fishery_id %in% 16:18))
```
END CONTENTS FOR extra-tamm-child.qmd #####--------------------------

tamm.path = "C:/Users/edwc1477/Documents/WDFW FRAM team work/NOF material/NOF 2024/NOF 2"
tamm.name = "Chin2124.xslx"
tamm_report(tamm.name = "Chin2124.xlsx", tamm.path = tamm.path)
Now with our extra content, presuming `extra-tamm-child.qmd` is saved in our documents folder
tamm_report(tamm.name = "Chin2124.xlsx", tamm.path = tamm.path,
  additional.children = "C:/Users/JohnDoe/Documents/extra-tamm-child.qmd")

End(Not run)
```

---

timestep_chinook_fram *Chinook timestep information stock information*

Description

Mapping of timestep numbers to dates in the year, taken from the `TimeStep` table of the Chinook FRAM database

Usage

```
timestep_chinook_fram
```

Format

A data frame with 4 rows and 5 columns:

species Species name

version_number

time_step_id id number for the time step

time_step_name Span of each timestep. Timesteps start on the first of the month, and end on the last of the month. Note that `Oct-Apr-2` is highlighting that timestep 4 runs from October of the current year to April of the NEXT year (equivalent to timestep 1 of the following year).

time_step_title `time_step_name`, but months are written out

Source

2024 Pre-Season Chinook DB.mdb

`timestep_coho_fram` *Coho timestep information stock information*

Description

Mapping of timestep numbers to dates in the year, taken from the `TimeStep` table of the Chinook FRAM database

Usage

```
timestep_coho_fram
```

Format

A data frame with 5 rows and 5 columns:

species Species name

version_number

time_step_id id number for the time step

time_step_name Span of each timestep. Timesteps start on the first of the month, and end on the last of the month.

time_step_title `time_step_name`, but months are written out consistently

Source

2024NOF_CohoFRAMdatabase_distribution.mdb

`validate_data_frame` *Confirm object is dataframe*

Description

Confirm object is dataframe

Usage

```
validate_data_frame(
  x,
  ...,
  arg = rlang::caller_arg(x),
  call = rlang::caller_env()
)
```

Arguments

x	Object to check
...	additional arguments
arg	name of argument in calling function, identified programmatically using rlang
call	name of calling function, identified programmatically using rlang

Index