

Package: presentresults (via r-universe)

September 17, 2026

Title Helpers to present results into table or graphs

Version 0.0.0.9002

Description The package provides wrappers to turn results from a long format into wide or graphics. For results in a long format, it will work with an analysis key which have the following format
analysis_type @/@ dependent_variable %/%
dependent_variable_value @/@ independent_variable %/%
independent_variable_value.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Suggests knitr, rmarkdown, testthat (>= 3.0.0), vdiff, withr

Config/testthat/edition 3

Imports analysistools, dplyr, ggplot2, glue, magrittr, openxlsx, purrr, rlang, scales, stringr, tidyr, tidyselect

Remotes impact-initiatives/analysistools

Depends R (>= 4.3.0)

LazyData true

URL <https://impact-initiatives.github.io/presentresults/>

VignetteBuilder knitr

Config/pak/sysreqs cmake make libicu-dev libuv1-dev libssl-dev libnode-dev

Repository <https://impact-initiatives.r-universe.dev>

Date/Publication 2024-10-28 13:24:07 UTC

RemoteUrl <https://github.com/impact-initiatives/presentresults>

RemoteRef HEAD

RemoteSha 6219392a2983db6c90b5e7cb557db5e052b76fed

Contents

add_label_analysis_key 2

add_label_columns 3

add_label_columns_to_results_table 4

create_group_clusters 4

create_ipc_table 5

create_label_dictionary 9

create_table_for_map 11

create_table_group_x_variable 12

create_table_variable_x_group 13

create_xlsx_group_x_variable 14

create_xlsx_variable_x_group 15

impact_colors 16

impact_palettes 16

presentresults_MSNA_template_data 17

presentresults_MSNA2024_dictionary 17

presentresults_MSNA2024_kobotool_fixed 18

presentresults_MSNA2024_kobotool_template 18

presentresults_MSNA2024_labelled_results_table 18

presentresults_MSNA2024_results_table 19

presentresults_resultstable 19

review_kobo_labels 19

theme_barplot 20

theme_impact 21

unite_labels 22

verify_grep_AinB 23

verify_numbers_values 23

Index 25

add_label_analysis_key	
	<i>Helper to add the label analysis key</i>

Description

Helper to add the label analysis key

Usage

add_label_analysis_key(results)

Arguments

results	results table with the label_* columns
---------	--

Value

results with label analysis key

add_label_columns	<i>Helper to add label columns</i>
-------------------	------------------------------------

Description

Helper to add label columns

Usage

```
add_label_columns(
  results_table,
  columns_var_to_convert,
  dictionary_survey,
  dictionary_survey_name_column = "name",
  dictionary_choices,
  dictionary_choices_survey_column = "name_survey",
  dictionary_choices_choices_column = "name_choices"
)
```

Arguments

`results_table` Result object with an analysis key.

`columns_var_to_convert` set of columns to add labels.

`dictionary_survey` Dictionary with name and label to use. Should be created with `create_label_dictionary`.

`dictionary_survey_name_column` name column in the dictionary.

`dictionary_choices` Dictionary with list_name, name and label to use. Should be created with `create_label_dictionary`.

`dictionary_choices_survey_column` name column for the survey variable column in the dictionary.

`dictionary_choices_choices_column` name column for the choices value column in the dictionary.

Value

Results table with label_* columns

```
add_label_columns_to_results_table
```

Add labels to the result table

Description

Add labels to the result table

Usage

```
add_label_columns_to_results_table(results_table, dictionary)
```

Arguments

`results_table` Result object with an analysis key.
`dictionary` Dictionary created with `create_label_dictionary`

Value

results table with label columns

Examples

```
add_label_columns_to_results_table(  
  results_table = presentresults_MSNA2024_results_table,  
  dictionary = presentresults_MSNA2024_dictionary  
)
```

```
create_group_clusters
```

Create number of cluster and number of hh surveyed per group/strata

Description

Create number of cluster and number of hh surveyed per group/strata

Usage

```
create_group_clusters(  
  results_table,  
  analysis_key = "analysis_key",  
  dataset,  
  cluster_name = NULL  
)
```

Arguments

results_table results table with analysis key
 analysis_key String with the name of the analysis key. Default is "analysis_key"
 dataset dataset
 cluster_name String with the name of the cluster id in the dataset.

Value

A dataframe with number of HH and cluster in each group

Examples

```
create_group_clusters(
  results_table = presentresults_resultstable,
  dataset = presentresults_MSNA_template_data,
  cluster_name = "cluster_id"
)
```

create_ipc_table	<i>Creates a table for the IPC</i>
------------------	------------------------------------

Description

Create a table from a results table with a key analysis in a format to be shared with the IPC TWG.

Usage

```
create_ipc_table(
  results_table,
  analysis_key = "analysis_key",
  dataset,
  cluster_name = NULL,
  fcs_cat_var = "fsl_fcs_cat",
  fcs_cat_values = c("Poor", "Borderline", "Acceptable"),
  fcs_set = c("fsl_fcs_cereal", "fsl_fcs_legumes", "fsl_fcs_veg", "fsl_fcs_fruit",
    "fsl_fcs_meat", "fsl_fcs_dairy", "fsl_fcs_sugar", "fsl_fcs_oil"),
  hhs_cat_var = "fsl_hhs_cat_ipc",
  hhs_cat_values = c("None", "No or Little", "Moderate", "Severe", "Very Severe"),
  hhs_cat_yesno_set = c("fsl_hhs_nofoodhh", "fsl_hhs_sleephungry", "fsl_hhs_alldaynight"),
  hhs_value_yesno_set = c("yes", "no"),
  hhs_cat_freq_set = c("fsl_hhs_nofoodhh_freq", "fsl_hhs_sleephungry_freq",
    "fsl_hhs_alldaynight_freq"),
  hhs_value_freq_set = c("rarely", "sometimes", "often"),
  rcsi_cat_var = "fsl_rcsi_cat",
  rcsi_cat_values = c("No to Low", "Medium", "High"),
```

```

rcsi_set = c("fsl_rcsi_lessquality", "fsl_rcsi_borrow", "fsl_rcsi_mealsize",
  "fsl_rcsi_mealadult", "fsl_rcsi_mealnb"),
lcsi_cat_var = "fsl_lcsi_cat",
lcsi_cat_values = c("None", "Stress", "Crisis", "Emergency"),
lcsi_set = c("fsl_lcsi_stress1", "fsl_lcsi_stress2", "fsl_lcsi_stress3",
  "fsl_lcsi_stress4", "fsl_lcsi_crisis1", "fsl_lcsi_crisis2", "fsl_lcsi_crisis3",
  "fsl_lcsi_emergency1", "fsl_lcsi_emergency2", "fsl_lcsi_emergency3"),
lcsi_value_set = c("yes", "no_had_no_need", "no_exhausted", "not_applicable"),
with_hdds = TRUE,
hdds_cat = "fsl_hdds_cat",
hdds_cat_values = c("Low", "Medium", "High"),
hdds_set = c("fsl_hdds_cereals", "fsl_hdds_tubers", "fsl_hdds_veg", "fsl_hdds_fruit",
  "fsl_hdds_meat", "fsl_hdds_eggs", "fsl_hdds_fish", "fsl_hdds_legumes",
  "fsl_hdds_dairy", "fsl_hdds_oil", "fsl_hdds_sugar", "fsl_hdds_condiments"),
hdds_value_set = c("yes", "no"),
with_fclc = FALSE,
fclc_matrix_var = "fslcm_phase",
fclc_matrix_values = c("Phase 1 FCLC", "Phase 2 FCLC", "Phase 3 FCLC", "Phase 4 FCLC",
  "Phase 5 FCLC"),
fc_matrix_var = "fsl_fc_phase",
fc_matrix_values = c("Phase 1 FC", "Phase 2 FC", "Phase 3 FC", "Phase 4 FC",
  "Phase 5 FC"),
other_variables = NULL,
stat_col = "stat",
proportion_name = "prop_select_one",
mean_name = "mean"
)

```

Arguments

results_table	results table with analysis key
analysis_key	String with the name of the analysis key. Default is "analysis_key"
dataset	dataset used to create the analysis results to calculate the number of clusters or number of survey.
cluster_name	string with the name of column of the cluster in the dataset. Default is NULL, it will calculate the number of interviews only.
fcs_cat_var	The string with the name of the Food Consumption Score. Default is "fsl_fcs_cat"
fcs_cat_values	String with the options of the Food Consumption Score. Default is c("Poor", "Borderline", "Acceptable")
fcs_set	String for the Food Consumption Score questions set. Default is c("fcs_cereal", "fcs_pulses", "fcs_milk", "fcs_meat", "fcs_veg", "fcs_fruit", "fcs_oil", "fcs_sugar", "fcs_spices")
hhs_cat_var	The string with the name of the Household Hunger Scale. Default is "fsl_hhs_cat_ipc"
hhs_cat_values	String with the options of the Household Hunger Scale. Default is c("None", "No or Little", "Moderate", "Severe", "Very Severe")

hhs_cat_yesno_set	String for the Household Hunger Scale yes-no questions set. Default is c("fsl_hhs_nofoodhh", "fsl_hhs_sleephungry", "fsl_hhs_alldaynight")
hhs_value_yesno_set	String for the values of the Household Hunger Scale yes-no questions set. Default is c("yes", "no")
hhs_cat_freq_set	String for the Household Hunger Scale frequency questions set. Default is c("fsl_hhs_nofoodhh_freq", "fsl_hhs_sleephungry_freq", "fsl_hhs_alldaynight_freq")
hhs_value_freq_set	String for the values of the Household Hunger Scale frequency questions set. Default is c("rarely", "sometimes", "often")
rcsi_cat_var	The string with the name of the reduced Coping Strategy Index. Default is "fsl_rcsi_cat"
rcsi_cat_values	String with the options of the reduced Coping Strategy Index. Default is c("No to Low", "Medium", "High")
rcsi_set	String for the reduced Coping Strategy Index questions set. Default is c("fsl_rcsi_lesquality", "fsl_rcsi_borrow", "fsl_rcsi_mealsize", "fsl_rcsi_mealadult", "fsl_rcsi_mealnb")
lcsi_cat_var	The string with the name of the Livelihood Coping Strategy Index. Default is "fsl_lcsi_cat"
lcsi_cat_values	String with the options of the Livelihood Coping Strategy Index. Default is c("None", "Stress", "Crisis", "Emergency")
lcsi_set	String for the Livelihood Coping Strategy Index questions set. Default is c("fsl_lcsi_stress1", "fsl_lcsi_stress2", "fsl_lcsi_stress3", "fsl_lcsi_stress4", "fsl_lcsi_crisis1", "fsl_lcsi_crisis2", "fsl_lcsi_crisis3", "fsl_lcsi_emergency1", "fsl_lcsi_emergency2", "fsl_lcsi_emergency3")
lcsi_value_set	String for the values of the Livelihood Coping Strategy Index questions set. Default is c("yes", "no_had_no_need", "no_exhausted", "not_applicable")
with_hdds	TRUE or FALSE, whether to include the FCLC and FC values.
hdds_cat	String with the name of the Household Dietary Diversity Score. Default is "fsl_hdds_cat"
hdds_cat_values	String with the options of the Household Dietary Diversity Score. Default is c("Low", "Medium", "High")
hdds_set	String for the Household Dietary Diversity Score. Default is c("fsl_hdds_cereals", "fsl_hdds_tubers", "fsl_hdds_veg", "fsl_hdds_fruit", "fsl_hdds_meat", "fsl_hdds_eggs", "fsl_hdds_fish", "fsl_hdds_legumes", "fsl_hdds_dairy", "fsl_hdds_oil", "fsl_hdds_sugar", "fsl_hdds_condiments")
hdds_value_set	String for the values of the Household Dietary Diversity Score questions set. Default is c("yes", "no")
with_fclc	TRUE or FALSE, whether to include the FCLC and FC values. Default is set to FALSE.
fclc_matrix_var	String with the name of the food consumption livelihood matrix from FEWS-NET. Default is "fclcm_phase"

fcllc_matrix_values	String with the options of the food consumption livelihood matrix Default is c("Phase 1 FCLC", "Phase 2 FCLC", "Phase 3 FCLC", "Phase 4 FCLC", "Phase 5 FCLC")
fc_matrix_var	String with the name of the food consumption matrix from FEWSNET. Default is "fsl_fc_phase"
fc_matrix_values	String with the options of the food consumption matrix. Default is c("Phase 1 FC", "Phase 2 FC", "Phase 3 FC", "Phase 4 FC", "Phase 5 FC")
other_variables	String for the names of other variables to include. Default is NULL
stat_col	String for the name of the column with the mean or the proportion. Default is "stat"
proportion_name	String how a proportion is called in the analysis key. Default is "prop_select_one"
mean_name	String how a mean is called in the analysis key. Default is "mean"

Details

For arguments that are *_values or *_set, and other_variables, the order of appearance will be the order in the table.

Value

a list with:

- a wide table with groups of interest in the rows, and the variables in the columns in a format that can be shared to the IPC TWG. This table should be pass into create_xlsx_group_x_variable
- the dataset that was provided.

Examples

```
no_nas_presentresults_resultstable <- presentresults_resultstable %>%
  dplyr::filter(!(analysis_type == "prop_select_one" & is.na(analysis_var_value)))

create_ipc_table(
  results_table = no_nas_presentresults_resultstable,
  dataset = presentresults_MSNA_template_data,
  cluster_name = "cluster_id",
  fcs_cat_var = "fcs_cat",
  fcs_cat_values = c("low", "medium", "high"),
  fcs_set = c(
    "fs_fcs_cereals_grains_roots_tubers",
    "fs_fcs_beans_nuts",
    "fs_fcs_dairy",
    "fs_fcs_meat_fish_eggs",
    "fs_fcs_vegetables_leaves",
    "fs_fcs_fruit",
    "fs_fcs_oil_fat_butter",
    "fs_fcs_sugar",
```

```

        "fs_fcs_condiment"
    ),
    hhs_cat_var = "hhs_cat",
    hhs_cat_values = c("none", "slight", "moderate", "severe", "very_severe"),
    hhs_cat_yesno_set = c("fs_hhs_nofood_yn", "fs_hhs_sleephungry_yn", "fs_hhs_daynoteating_yn"),
    hhs_cat_freq_set = c("fs_hhs_nofood_freq", "fs_hhs_sleephungry_freq", "fs_hhs_daynoteating_freq"),
    hhs_value_freq_set = c("rarely_1_2", "sometimes_3_10", "often_10_times"),
    rcsi_cat_var = "rcsi_cat",
    rcsi_cat_values = c("low", "medium", "high"),
    rcsi_set = c("rCSIlessQlty", "rCSIBorrow", "rCSIMealSize", "rCSIMealAdult", "rCSIMealNb"),
    lcsi_cat_var = "lcs_cat",
    lcsi_cat_values = c("none", "stress", "emergency", "crisis"),
    lcsi_set = c(
        "liv_stress_lcsi_1",
        "liv_stress_lcsi_2",
        "liv_stress_lcsi_3",
        "liv_stress_lcsi_4",
        "liv_crisis_lcsi_1",
        "liv_crisis_lcsi_2",
        "liv_crisis_lcsi_3",
        "liv_emerg_lcsi_1",
        "liv_emerg_lcsi_2",
        "liv_emerg_lcsi_3"
    ),
    with_hdds = FALSE
)

```

```
create_label_dictionary
```

Create a dictionary for the labeling results

Description

Create a dictionary for the labeling results

Usage

```

create_label_dictionary(
  kobo_survey_sheet,
  kobo_choices_sheet,
  label_column = "label::english",
  analysis_type_dictionary = NULL,
  results_table = NULL
)

```

Arguments

kobo_survey_sheet
 KOBO survey sheet to be used.

`kobo_choices_sheet` KOBO choices sheet to be used.

`label_column` label column from the KOBO tools to be used.

`analysis_type_dictionary` Analysis type dictionary, a data frame with analysis type and label_analysis_type to be used. By default parameters is set to NULL. It will use a default dataframe. See section `analysis_type_dictionary` for more details.

`results_table` result object with an analysis key. Default is NULL, it will be used with `review_kobo_labels`.

Value

A list with 3 dataframes: `dictionary_survey`, `dictionary_choices`, `analysis_type_dictionary`

analysis_type_dictionary

The default analysis dictionary is created like that.

```
data.frame(analysis_type = c("prop_select_one",
                             "prop_select_multiple",
                             "mean",
                             "ratio",
                             "median"),
           label_analysis_type = c("Proportion (single choice)",
                                   "Proportion (multiple choice)",
                                   "Mean",
                                   "Ratio",
                                   "Median"))
```

Examples

```
create_label_dictionary(
  kobo_survey_sheet = presentresults_MSNA2024_kobotool_fixed$kobo_survey,
  kobo_choices_sheet = presentresults_MSNA2024_kobotool_fixed$kobo_choices
)
```

```
french_dictionary <- data.frame(
  analysis_type = c(
    "prop_select_one",
    "prop_select_multiple",
    "mean",
    "ratio",
    "median"
  ),
  label_analysis_type = c(
    "Proportion (Choix unique)",
    "Proportion (Choix multiple)",
    "Moyenne",
```

```

        "Ratio",
        "Médiane"
    )
)

create_label_dictionary(
  kobo_survey_sheet = presentresults_MSNA2024_kobotool_fixed$kobo_survey,
  kobo_choices_sheet = presentresults_MSNA2024_kobotool_fixed$kobo_choices,
  label_column = "label::french",
  analysis_type_dictionary = french_dictionary
)

```

create_table_for_map *Create a table for MSNA Indicator Maps 1.2 tool*

Description

Create a table for MSNA Indicator Maps 1.2 tool

Usage

```

create_table_for_map(
  results_table,
  group_var_value_column = "group_var_value",
  analysis_var_column = "analysis_var",
  stat_column = "stat",
  number_classes = 5
)

```

Arguments

results_table Results table from analysistools filtered for only value per admin.

group_var_value_column Name of the column with the group/dependent variable values. Default is "group_var_value".

analysis_var_column Name of the column with the analysis/independent variable names. Default is "analysis_var".

stat_column Name of the column with the stat. Default is "stat".

number_classes Number of classes for the map. It will convert percentages to classes. Default value is 5.

Details

There can be 5 or 6 classes as follow:

5 classes:

- 1 : 0
- 2 : <= 25 %
- 3 : <= 50 %
- 4 : <= 75 %
- 5 : <= 100 %
- NA: Anything else

6 classes:

- 1 : 0
- 2 : <= 20 %
- 3 : <= 40 %
- 4 : <= 60 %
- 5 : <= 80 %
- 6 : <= 100 %
- NA: Anything else

Value

A wide table with group variables in rows and the indicators coded in their classes in the columns.

Examples

```
presentresults::presentresults_MSNA2024_results_table |>
  dplyr::filter(
    analysis_var == "wash_drinking_water_source_cat",
    analysis_var_value == "surface_water",
    group_var == "admin1"
  ) |>
  create_table_for_map()
```

create_table_group_x_variable

Create a wide table with indicators in the columns

Description

Create a wide table with indicators in the columns

Usage

```
create_table_group_x_variable(
  results_table,
  analysis_key = "analysis_key",
  value_columns = c("stat", "stat_low", "stat_upp")
)
```

Arguments

results_table	result object with an analysis key
analysis_key	name of the columns of the analysis key, as character vector. Default is "analysis_key"
value_columns	names of the columns with the stats of interest, as character vector. Default is c("stat", "stat_low", "stat_upp")

Value

a wide table with grouping variables as rows and analysed variables as columns

Examples

```
create_table_group_x_variable(presentresults_resultstable, value_columns = "stat")
```

```
create_table_variable_x_group
```

Turns a long format table into a wide format

Description

Turns a long format table into a wide format

Usage

```
create_table_variable_x_group(
  results_table,
  analysis_key = "analysis_key",
  value_columns = c("stat", "stat_low", "stat_upp"),
  list_for_excel = FALSE
)
```

Arguments

results_table	results table with analysis key
analysis_key	analysis key following this description "analysis_type @/@ dependent_variable %/% dependent_variable_value @/@ independent_variable %/% independent_variable_value"
value_columns	string containing the names of the columns with the stats to export
list_for_excel	Default is FALSE, the function will return a dataframe. If set to TRUE, it will return a list of dataframe with the grouping variable as name. This format makes it easier to write excel files with different tab.

Value

a data frame in a wide format with the analysis type, analysis variable, analysis variable value and the group variable value as columns. If list_for_excel is set to TRUE, it will return a list per grouping variable.

Examples

```
presentresults_resultstable %>% create_table_variable_x_group("analysis_key", "stat")
```

```
create_xlsx_group_x_variable
```

Write a table group by variable into Excel

Description

Write a table group by variable into Excel

Usage

```
create_xlsx_group_x_variable(
  table_group_x_variable,
  table_name = "group_x_table",
  dataset_name = "dataset",
  file_path,
  table_sheet = "table_group_x_variable",
  dataset_sheet = "dataset",
  write_file = TRUE,
  overwrite = FALSE
)
```

Arguments

table_group_x_variable	a table create by create_table_group_x_variable
table_name	string with the name of table to write. It will only be used if it is part of a list. Default is "group_x_table".
dataset_name	string with the name of dataset to write. It will only be used if it is part of a list. Default is "dataset".
file_path	File path, it should contains the file name
table_sheet	string with the name of the sheet to write the table, default is "table_group_x_variable"
dataset_sheet	string with the name of the sheet to write the dataset, default is "dataset"
write_file	Default is TRUE, it will write the file. If set to FALSE, it will return a workbook object from openxlsx
overwrite	Default is FALSE, it will overwrite the file if set to TRUE.

Value

An excel file formatted.

Examples

```
## Not run:
presentresults_resultstable %>%
  create_table_group_x_variable() %>%
  create_xlsx_group_x_variable(file_path = "mytable.xlsx")

## End(Not run)
```

```
create_xlsx_variable_x_group
```

Write a table variable by group into Excel

Description

Write a table variable by group into Excel

Usage

```
create_xlsx_variable_x_group(
  table_group_x_variable,
  file_path = NULL,
  table_name = "variable_x_group_table",
  value_columns = c("stat", "stat_low", "stat_upp"),
  total_columns = NULL,
  readme_sheet_name = "readme",
  table_sheet_name = "variable_x_group_table",
  overwrite = FALSE
)
```

Arguments

table_group_x_variable	a table create by create_table_variable_x_group
file_path	File names and path. Default is NULL which will return a workbook instead of an excel file.
table_name	string with the name of table to write. It will only be used if it is part of a list. Default is "variable_x_group_table".
value_columns	string containing the names of the columns with the stats to export
total_columns	string containing the names of the columns with the totals (n, n_total, n_weighted, n_total_weighted, etc.) to export
readme_sheet_name	string with the name of the sheet to write the read me page, default is "readme"

table_sheet_name string with the name of the sheet to write the table, default is "variable_x_group_table"

overwrite Default is FALSE, it will overwrite the file if set to TRUE.

Value

An excel file formatted.

Examples

```
## Not run:
presentresults_resultstable %>%
  create_table_variable_x_group() %>%
  create_xlsx_variable_x_group(file_path = "mytable.xlsx")

## End(Not run)
```

impact_colors	<i>IMPACT colors hex code</i>
---------------	-------------------------------

Description

IMPACT colors hex code

Usage

impact_colors

Details

IMPACT colors hex code

impact_palettes	<i>IMPACT palettes</i>
-----------------	------------------------

Description

IMPACT palettes

Usage

impact_palettes

Details

IMPACT palettes hex code

presentresults_MSNA_template_data
<i>MSNA dataset generated by xslxform</i>

Description

Datasets for example

Usage

presentresults_MSNA_template_data

Examples

presentresults_MSNA_template_data
presentresults_resultstable

presentresults_MSNA2024_dictionary
<i>MSNA2024 dictionary</i>

Description

Dictionary to be used to label results

Usage

presentresults_MSNA2024_dictionary

Details

An example of dictionary to be used in add_label_columns_to_results_table() create_label_dictionary(kobo_survey_sheet = presentresults_MSNA2024_kobotool_fixed\$kobo_survey, kobo_choices_sheet = presentresults_MSNA2024_kobotool_fix results_table = presentresults_MSNA2024_results_table)

presentresults_MSNA2024_kobotool_fixed	<i>results table example</i>
Description	
KOBO Template 2024 V10 without duplicated labels and names.	
Usage	
presentresults_MSNA2024_kobotool_fixed	
Details	
KOBO template 2024 with composite indicators information without duplicated labels and names.	
presentresults_MSNA2024_kobotool_template	<i>KOBO template for MSNA 2024 with composite indicators information</i>
Description	
KOBO Template 2024 V10	
Usage	
presentresults_MSNA2024_kobotool_template	
Details	
KOBO template 2024 with composite indicators information	
presentresults_MSNA2024_labelled_results_table	<i>MSNA 2024 results table with labels.</i>
Description	
MSNA 2024 labelled results table	
Usage	
presentresults_MSNA2024_labelled_results_table	

presentresults_MSNA2024_results_table	<i>results table example for MSNA 2024 tools</i>
---------------------------------------	--

Description

Results table from MSNA2024 template

Usage

presentresults_MSNA2024_results_table

presentresults_resultstable	<i>results table example</i>
-----------------------------	------------------------------

Description

results table example

Usage

presentresults_resultstable

review_kobo_labels	<i>Review if there are duplication of kobo name and labels in the kobo tools</i>
--------------------	--

Description

Review if there are duplication of kobo name and labels in the kobo tools

Usage

```
review_kobo_labels(  
  kobo_survey_sheet,  
  kobo_choices_sheet,  
  label_column = "label::english",  
  exclude_type = c("begin_group", "end_group", "begin_repeat", "end_repeat", "note"),  
  results_table = NULL  
)
```

Arguments

- kobo_survey_sheet kobo survey sheet. It must contain type and name.
- kobo_choices_sheet kobo choices sheet. It must contain list_name and name.
- label_column Column name with the label to be used, default is "label::english"
- exclude_type Types to exclude in the review, default is c("begin_group", "end_group", "begin_repeat", "end_repeat", "note")
- results_table Results table with group_var and analysis_var columns (names of the different variables). The table will be used to review only the names and label that will appear.

Value

A data frame with the duplicated cases and the reasons. Empty if there is no duplication.

Examples

```
review_kobo_labels(  
  kobo_survey_sheet = presentresults_MSNA2024_kobotool_template$kobo_survey,  
  kobo_choices_sheet = presentresults_MSNA2024_kobotool_template$kobo_choices,  
)  
review_kobo_labels(  
  kobo_survey_sheet = presentresults_MSNA2024_kobotool_template$kobo_survey,  
  kobo_choices_sheet = presentresults_MSNA2024_kobotool_template$kobo_choices,  
  results_table = presentresults_MSNA2024_results_table  
)
```

theme_barplot	<i>Barplot theme</i>
---------------	----------------------

Description

theme_barplot will fill the colors with the palette, put the y-axis to 0 to 100 %

Usage

```
theme_barplot(palette = impact_palettes$reach_palette)
```

Arguments

- palette color palette to be used in scale_fill_manual

Value

ggplot2 plot with filled colors with the palette, put the y-axis to 0 to 100 %

Examples

```
data_to_plot <- presentresults::presentresults_MSNA2024_labelled_results_table |>
  dplyr::filter(
    analysis_var == "wash_drinking_water_source_cat",
    group_var == "hoh_gender"
  )

data_to_plot %>%
  ggplot2::ggplot() +
    ggplot2::geom_col(ggplot2::aes(x = label_analysis_var_value,
                                   y = stat,
                                   fill = label_group_var_value),
                     position = "dodge") +
    ggplot2::labs(title = stringr::str_wrap(unique(data_to_plot$indicator),50),
                  x = stringr::str_wrap(unique(data_to_plot$label_analysis_var),50),
                  fill = stringr::str_wrap(unique(data_to_plot$label_group_var),20)) +
    theme_barplot()
```

 theme_impact

Theme for IMPACT Initiative

Description

It will set:

- theme_minimal,
- color of the text in REACH gray (see impact_colors\$reach_gray),
- Title in bold and with the color of the initiative (reach: impact_colors\$red, impact: impact_colors\$blue, and agora: impact_colors\$bordeaux)

Usage

```
theme_impact(initiative = "reach")
```

Arguments

initiative Name of the initiative, should be "reach", "impact" or "agora".

Value

ggplot2 plot with theme_minimal, bold title and color of the initiative.

Examples

```
data_to_plot <- presentresults::presentresults_MSNA2024_labelled_results_table |>
  dplyr::filter(
    analysis_var == "wash_drinking_water_source_cat",
    group_var == "hoh_gender"
  )

data_to_plot %>%
  ggplot2::ggplot() +
    ggplot2::geom_col(ggplot2::aes(x =label_analysis_var_value,
                                   y = stat,
                                   fill = label_group_var_value),
                      position = "dodge") +
    ggplot2::labs(title = stringr::str_wrap(unique(data_to_plot$indicator),50),
                  x = stringr::str_wrap(unique(data_to_plot$label_analysis_var),50),
                  fill = stringr::str_wrap(unique(data_to_plot$label_group_var),20)) +
    theme_impact("reach")
```

unite_labels	<i>Unite labels columns</i>
--------------	-----------------------------

Description

Unite labels columns

Usage

```
unite_labels(key_table)
```

Arguments

key_table a key table built with create_analysis_key_table

Value

a table with label_analysis_var, label_analysis_var_value, label_group_var, and label_group_var_value united and with a %/%% as separator

verify_grep_AinB	<i>Verify that which value of a vector is present in another vector</i>
------------------	---

Description

Verify that which value of a vector is present in another vector

Usage

```
verify_grep_AinB(.A, .B)
```

Arguments

.A	String of values to check
.B	Vector of string to check

Value

a vector of the length of values_to_check with TRUE or FALSE if the value appears at least once in .B

Examples

```
verify_grep_AinB(c("hhs_cat", "fsc_cat"), presentresults_resultstable$analysis_key)
```

verify_numbers_values	<i>Verify that a given variable set as the expected number of values.</i>
-----------------------	---

Description

Verify that a given variable set as the expected number of values.

Usage

```
verify_numbers_values(var_name, values_set, expected_number)
```

Arguments

var_name	The name of the variable as string.
values_set	Vector with a the set of values.
expected_number	Expected numbers of unique value (excluding NA)

Value

If the number of unique value is different than the expected, it will show a warning.

Examples

```
verify_numbers_values("my_var", c("low", "borderline", "acceptable"), 3)
verify_numbers_values("my_var", c("low", "borderline", "acceptable", NA), 3)
verify_numbers_values("my_var", c("low", "acceptable", NA), 3)
verify_numbers_values("my_var", c("none", "low", "borderline", "acceptable"), 3)
```

Index

```
* datasets
    impact_colors, 16
    impact_palettes, 16
    presentresults_MSNA2024_dictionary, 17
    presentresults_MSNA2024_kobotool_fixed, 18
    presentresults_MSNA2024_kobotool_template, 18
    presentresults_MSNA2024_labelled_results_table, 18
    presentresults_MSNA2024_results_table, 19
    presentresults_MSNA_template_data, 17
    presentresults_resultstable, 19

    presentresults_MSNA2024_results_table, 19
    presentresults_MSNA_template_data, 17
    presentresults_resultstable, 19

    review_kobo_labels, 19
    theme_barplot, 20
    theme_impact, 21
    unite_labels, 22
    verify_grep_AinB, 23
    verify_numbers_values, 23

add_label_analysis_key, 2
add_label_columns, 3
add_label_columns_to_results_table, 4

create_group_clusters, 4
create_ipc_table, 5
create_label_dictionary, 9
create_table_for_map, 11
create_table_group_x_variable, 12
create_table_variable_x_group, 13
create_xlsx_group_x_variable, 14
create_xlsx_variable_x_group, 15

impact_colors, 16
impact_palettes, 16

presentresults_MSNA2024_dictionary, 17
presentresults_MSNA2024_kobotool_fixed, 18
presentresults_MSNA2024_kobotool_template, 18
presentresults_MSNA2024_labelled_results_table, 18
```