

Package: tlShiny (via r-universe)

November 1, 2024

Type Package

Title Supplies essential functions to Teaching Lab dashboards

Version 0.1.0

Author Duncan Gates

Maintainer Duncan Gates <duncan.gates123@gmail.com>

Description A bunch of random functions I use in developing dashboards
Needs to vastly reduce the number of dependencies at the moment.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Imports dplyr, forcats, ggplot2, ggtext, glue, gt, magrittr, purrr,
rlang, stringr, tidyr, treemapify, waffle

Depends R (>= 2.10)

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Repository <https://teachinglab.r-universe.dev>

RemoteUrl <https://github.com/teachinglab/tlShiny>

RemoteRef HEAD

RemoteSha c2e6a7f0240f2c9014cf6635e8431737c19573fd

Contents

agree_strongly_agree	3
calc_nps	3
check_email_approved	4
check_email_domain	4
contact_lead_graph	5
course_feedback_graph	5

drop1	6
end_coaching_feedback_graph	6
file.path2	7
get_percent_positive_student_survey	7
grade_ipg	7
gtable_extract_grob	8
gtable_remove_grob	8
gt_percent_n	9
gt_theme_tl	10
highlight_fun	10
html_wrap	11
know_assess_summary	12
know_assess_summary_detailed	12
make_ipg_ela_summary_chart	13
make_ipg_fsot_summary_chart	13
make_ipg_math_summary_chart	14
make_student_work_chart_circle	14
make_student_work_chart_people	15
make_teacher_curriculum_perceptions	15
make_teacher_curriculum_usage	16
make_teacher_lesson_usage	16
make_teacher_perceptions_peer_relationships	17
make_teacher_perceptions_school_leaders	17
na_df	18
neg_cond_filter	18
notin	19
no_data_plot_currently	19
no_data_plot_custom	19
no_data_plot_filters	20
ongoing_coaching_feedback_graph	20
quote_viz	21
remove_empty_tl	22
scale_colour_tl	22
session_feedback_graph	24
square	24
stop_words	25
student_bar_chart	26
subsites	26
survey_monkey	27
theme_tl	27
tl_palette	29
tl_palette2	29
tl_pal_blue	30
tl_select_percent	30

agree_strongly_agree *Agree/Strongly agree*

Description

Gets the percent that agree and strongly agree

Usage

```
agree_strongly_agree(data, question)
```

Arguments

data	the data
question	a string - the question to get the percentage for

Value

a string

Examples

```
## Not run:  
plot_agree |>  
  agree_strongly_agree(question = "x")  
  
## End(Not run)
```

calc_nps *Calculate nps score*

Description

Calculate nps score

Usage

```
calc_nps(x)
```

Arguments

x	A vector of nps scores
---	------------------------

Value

Returns the nps score With formula ‘

check_email_approved *Verify email from vector*

Description

Verifies that an email is in a list

Usage

```
check_email_approved(email, approved_emails_list)
```

Arguments

email	the email provided
approved_emails_list	the email to check the list for

Value

TRUE or FALSE

check_email_domain *Verify email domain*

Description

Verifies that an email is of a specified domain

Usage

```
check_email_domain(email, domain)
```

Arguments

email	the email provided
domain	the domain to check that it is from

Value

TRUE or FALSE

contact_lead_graph *Contact Lead Graph Summary*

Description

Returns a barchart for selections in the relevant questions of the contact lead survey

Usage

```
contact_lead_graph(data)
```

Arguments

data the data to use

Value

prints a ggplot object

course_feedback_graph *Course survey feedback graph dependent on race and content area*

Description

Returns a barchart for selections in the relevant questions of the end of coaching survey

Usage

```
course_feedback_graph(data)
```

Arguments

data the data to use

Value

prints a ggplot object

drop1

HTML/CSS Button Content Expander

Description

Creates a button that will expand or hide content

Usage

```
drop1(before = T, options, envir, name)
```

Arguments

before	button default, collapsed/not collapsed
options	unclear
envir	also unclear
name	chunk name

Value

html wrapper

end_coaching_feedback_graph

End of Coaching feedback graph dependent on race and content area

Description

Returns a barchart for selections in the relevant questions of the end of coaching survey

Usage

```
end_coaching_feedback_graph(data)
```

Arguments

data	the data to use
------	-----------------

Value

prints a ggplot object

file.path2	<i>File path</i>
------------	------------------

Description

Gives the file path without double slash bug

Usage

```
file.path2(..., fsep = .Platform$file.sep)
```

Arguments

...	The file path
fsep	the file separation

Value

fp a file path

get_percent_positive_student_survey	<i>Get get_percent_positive_student_survey(data) \itemdatathe data to use a tibble Uses column names and finds the</i>
-------------------------------------	--

grade_ipg	<i>Grade IPG Data</i>
-----------	-----------------------

Description

function for grading different parts of the ipg forms

Usage

```
grade_ipg(x, type = "character")
```

Arguments

x	the data
type	character or numeric

Value

a percentage of correct either by checking 3 or 4 or yes

`gtable_extract_grob` *gtable_extract_grob*

Description

Helper function to extract a grob from gtables by name.

Usage

```
gtable_extract_grob(g, pattern = "guide-box")
```

Arguments

<code>g</code>	the table to extract the grob
<code>pattern</code>	grob name or pattern to match

Value

`g`, a grob matching the specified pattern

`gtable_remove_grob` *gtable_remove_grob*

Description

Helper function to remove grobs by name, from gtables

Usage

```
gtable_remove_grob(g, pattern = "guide-box")
```

Arguments

<code>g</code>	gtable with the grob removed
<code>pattern</code>	grob name or pattern to match

Value

`g`, with pattern removed.

gt_percent_n	<i>GT or ggplot maker</i>
--------------	---------------------------

Description

makes a gt table with percent and n colored

Usage

```
gt_percent_n(  
  df,  
  column,  
  custom_title,  
  no_title = T,  
  base_font = 10,  
  heading_font = 14,  
  custom_column_name = "",  
  viz_type = "gt"  
)
```

Arguments

df	the data frame
column	the column to get count and percent from
custom_title	the title for the table
no_title	make the table have no title
base_font	overall table font size
heading_font	title font size
custom_column_name	a custom name for the column
viz_type	gt by default, also has ggplot options like pie chart, waffle, or treemap

Value

a gt table

gt_theme_tl	<i>Create Teaching Lab theme to a gt table</i>
-------------	--

Description

Create Teaching Lab theme to a gt table

Usage

```
gt_theme_tl(
  data,
  all_caps = F,
  align = "center",
  base_font = 16,
  heading_font = 20,
  ...
)
```

Arguments

data	An existing gt table object
all_caps	Whether or not to capitalize titles
align	Align options are "left", "center", "right"
base_font	the font size
heading_font	the title font size
...	Optional additional arguments to gt::table_options()

Value

Creates a gt theme as a pipeable function

Examples

```
mtcars |> utils::head() |> gt::gt() |> tlShiny::gt_theme_tl()
```

highlight_fun	<i>Word highlighting</i>
---------------	--------------------------

Description

Provides html formatted highlighting

Usage

```
highlight_fun(data, highlight = tlShiny::find_highlight(data))
```

Arguments

data the data to highlight
highlight the words to highlight

Details

```
#' @title Word highlighting #' @description Finds most common words in string #' @param string
the string to evaluate #' @param n the number of words to find #' @param print whether or not
to print the highlighted words #' @return a vector of strings #' #' @export find_highlight <- function(string, n = 3, print = F) stop_words <- tShiny::stop_words |> dplyr::bind_rows(data.frame(word = tShiny::na_df, lexicon = "TL_NA"))
txt_df <- data.frame(column_name = "txt") |>
highlight <- string |> na.omit() |> as.data.frame() |> dplyr::rename() |> tidytext::unnest_tokens(word, txt) |> # This makes sure to get rid of numbers in consideration for highlighting # By making sure
as.numeric returns NA on words dplyr::filter(is.na(as.numeric(word))) |> # Get a count of words and
sort dplyr::count(word, sort = T) |> # Get rid of generic (stop) words dplyr::anti_join(stop_words)
|> # Get a user-specified number of words, or the default 3 utils::head(n) |> # Make this a vector
dplyr::pull(word) |> # Suppress warnings from the as.numeric call suppressWarnings() |> suppressMessages()
if (print == T) print(highlight)
return(highlight)
```

Value

a vector of strings

html_wrap

HTML Text Wrapping

Description

Takes a string and inserts
 at the requested intervals

Usage

```
html_wrap(string, n = 40)
```

Arguments

string the string
n the width of the string before a
 tag

Value

the same string with
 inserted at the requested interval

Examples

```
html_wrap("a random string that has about 40 characters in it")
```

```
know_assess_summary
```

Knowledge Assessment Graph Summary

Description

Creates a graph specifically for Knowledge Assessments in mid year reports

Usage

```
know_assess_summary(data, know_assess_filter)
```

Arguments

`data` the data
`know_assess_filter` the knowledge assessment to make plot for

Value

a ggplot

```
know_assess_summary_detailed
```

Knowledge Assessment Graph Question-level Summary

Description

Creates a graph specifically for Knowledge Assessments Scored by Question from Qualtrics

Usage

```
know_assess_summary_detailed(data, know_assess)
```

Arguments

`data` the data
`know_assess` the knowledge assessment to make plot for

Value

a ggplot

`make_ipg_ela_summary_chart`*ELA IPG Chart Summary Maker*

Description

Creates a chart to summarise the ELA IPG based on the round selected

Usage

```
make_ipg_ela_summary_chart(  
  data,  
  round = "Baseline (first observation of the year)"  
)
```

Arguments

<code>data</code>	the data
<code>round</code>	the round of the IPG for which data should be pulled (can be one of "Baseline (first observation of the year)", "Mid-year (middle of service, if applicable)", "End of year (last observation of the year)", "Other", or "Ongoing")

Value

a ggplot

`make_ipg_fsot_summary_chart`*FSOT IPG Chart Summary Maker*

Description

Creates a chart to summarise the FSOT IPG based on the round selected

Usage

```
make_ipg_fsot_summary_chart(  
  data,  
  round = "Baseline (first observation of the year)"  
)
```

Arguments

<code>data</code>	the data
<code>round</code>	the round of the IPG for which data should be pulled (can be one of "Baseline (first observation of the year)", "Mid-year (middle of service, if applicable)", "End of year (last observation of the year)", "Other", or "Ongoing")

Value

a ggplot

make_ipg_math_summary_chart

Math IPG Chart Summary Maker

Description

Creates a chart to summarise the Math IPG based on the round selected

Usage

```
make_ipg_math_summary_chart(  
  data,  
  round = "Baseline (first observation of the year)"  
)
```

Arguments

data	the data
round	the round of the IPG for which data should be pulled (can be one of "Baseline (first observation of the year)", "Mid-year (middle of service, if applicable)", "End of year (last observation of the year)", "Other", or "Ongoing")

Value

a ggplot

make_student_work_chart_circle

Student Work Circle Charts

Description

Chart to show the

Usage

```
make_student_work_chart_circle(data)
```

Arguments

data	the data for the plotter to use, should include all columns of interest
------	---

Value

a ggplot

`make_student_work_chart_people`
Student Work People Chart

Description

Chart to show the

Usage

`make_student_work_chart_people(data)`

Arguments

`data` the data for the plotter to use, should include all columns of interest

Value

a ggplot

`make_teacher_curriculum_perceptions`
Teacher Curriculum Perceptions Graph

Description

Creates a chart to summarise the teacher perceptions of curriculum

Usage

`make_teacher_curriculum_perceptions(data)`

Arguments

`data` the data

Value

a ggplot

make_teacher_curriculum_usage

Teacher Curriculum Usage Graph

Description

Creates a chart to summarise the Teacher Curriculum Usage from the Educator Survey

Usage

```
make_teacher_curriculum_usage(data)
```

Arguments

data the data

Value

a ggplot

make_teacher_lesson_usage

Teacher Use of Lessons Graph

Description

Creates a chart to summarise the teacher use of lessons from district or school-adopted materials

Usage

```
make_teacher_lesson_usage(data)
```

Arguments

data the data

Value

a ggplot

`make_teacher_perceptions_peer_relationships`
Teacher Perceptions Peer Relationships Graph

Description

Creates a chart to summarise the teacher perceptions of of peer relationships

Usage

`make_teacher_perceptions_peer_relationships(data)`

Arguments

`data` the data

Value

a ggplot

`make_teacher_perceptions_school_leaders`
Teacher School Leader Perceptions Graph

Description

Creates a chart to summarise the teacher perceptions of school leaders

Usage

`make_teacher_perceptions_school_leaders(data)`

Arguments

`data` the data

Value

a ggplot

na_df	<i>A list of NA type words to ignore in qualitative responses</i>
-------	---

Description

A list of NA type words to ignore in qualitative responses

Usage

```
na_df
```

Format

A character vector

neg_cond_filter	<i>Negative Conditional Filter</i>
-----------------	------------------------------------

Description

Conditionally filters value given that it is not the first parameter, for use in shiny apps

Usage

```
neg_cond_filter(data, if_not_this, filter_this, dat_filter)
```

Arguments

data	the dataframe to apply filter
if_not_this	If value is not this
filter_this	Filter for this
dat_filter	Data column object to filter

Value

filtered dataframe

notin	<i>Find elements x not in a vector y</i>
-------	--

Description

Find elements x not in a vector y

Usage

```
x %!in% y
```

Arguments

x	A vector of what shouldn't exist
y	A vector to check against

Value

Returns elements not in vector

no_data_plot_currently	<i>No Data Plot</i>
------------------------	---------------------

Description

A plot that says no data available yet this year

Usage

```
no_data_plot_currently
```

Format

An object of class gg (inherits from ggplot) of length 11.

no_data_plot_custom	<i>No Data Plot Adjustable Title</i>
---------------------	--------------------------------------

Description

A plot that says no data available by default, but can be adjusted to say whatever you want

Usage

```
no_data_plot_custom(title = "No data available!")
```

no_data_plot_filters *No Data Plot*

Description

A plot that says no data available, and to check your filters

Usage

```
no_data_plot_filters
```

Format

An object of class gg (inherits from ggplot) of length 11.

ongoing_coaching_feedback_graph

Ongoing Coaching feedback graph dependent on race and content area

Description

Returns a barchart for selections in the relevant questions of the end of coaching survey

Usage

```
ongoing_coaching_feedback_graph(data)
```

Arguments

data the data to use

Value

prints a ggplot object

`quote_viz`*Quote Visualization*

Description

takes a dataframe and makes a gt table or ggplot that shows a quote

Usage

```
quote_viz(  
  data,  
  text_col = colnames(data)[1],  
  viz_type = "gt",  
  custom_highlight = NULL,  
  n = 3,  
  print = T,  
  width = 60,  
  title = NULL,  
  suppress_warnings = T,  
  align = "center",  
  save = T,  
  ...  
)
```

Arguments

<code>data</code>	the dataframe
<code>text_col</code>	columns to create table for
<code>viz_type</code>	ggplot or gt visualization
<code>custom_highlight</code>	a vector, optional custom highlighting
<code>n</code>	integer, number of words to auto-highlight
<code>print</code>	T, whether or not to print the highlighted words to console
<code>width</code>	The width of the table generated
<code>title</code>	the title of the ggplot or gt
<code>suppress_warnings</code>	T/F suppression of warnings
<code>align</code>	the table alignment: "left", "center", "right"
<code>...</code>	Arguments passed onto the gt table

Value

a ggplot/gt that visualizes text

Examples

```
## Not run:
df <- tlShiny::survey_monkey
colnames(df)[1] <- "What learning are you excited to try?"
quote_viz(
  data = df,
  text_col = "What learning are you excited to try?",
  viz_type = "gt",
  title = "Responses from Survey Monkey"
)

## End(Not run)
```

remove_empty_tl *Remove Empty Rows*

Description

Filters out all rows that contain entirely NA

Usage

```
remove_empty_tl(data)
```

Arguments

data the dataframe to apply filter

Value

a dataframe with no empty rows

scale_colour_tl *Discrete color & fill scales based on the Teaching Lab palette*

Description

See [tl_palette\(\)](#).

Usage

```
scale_colour_tl(n, color = "blue", ...)

scale_color_tl(n, color = "blue", ...)

scale_fill_tl(
  n,
  color = c("blue", "orange", "purple", "green", "teal", "tl_colors"),
  ...
)
```

Arguments

n	the number of colors
color	the color from <code>tl_palette</code>
...	Arguments passed on to <code>ggplot2::discrete_scale</code>
aesthetics	The names of the aesthetics that this scale works with.
scale_name	[Deprecated] The name of the scale that should be used for error messages associated with this scale.
palette	A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take (e.g., <code>scales::pal_hue()</code>).
name	The name of the scale. Used as the axis or legend title. If <code>waiver()</code> , the default, the name of the scale is taken from the first mapping used for that aesthetic. If <code>NULL</code> , the legend title will be omitted.
breaks	One of: <ul style="list-style-type: none"> • <code>NULL</code> for no breaks • <code>waiver()</code> for the default breaks (the scale limits) • A character vector of breaks • A function that takes the limits as input and returns breaks as output. Also accepts rlang <code>lambda</code> function notation.
labels	One of: <ul style="list-style-type: none"> • <code>NULL</code> for no labels • <code>waiver()</code> for the default labels computed by the transformation object • A character vector giving labels (must be same length as breaks) • An expression vector (must be the same length as breaks). See <code>?plot-math</code> for details. • A function that takes the breaks as input and returns labels as output. Also accepts rlang <code>lambda</code> function notation.
limits	One of: <ul style="list-style-type: none"> • <code>NULL</code> to use the default scale values • A character vector that defines possible values of the scale and their order • A function that accepts the existing (automatic) values and returns new ones. Also accepts rlang <code>lambda</code> function notation.

`na.translate` Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify `na.translate = FALSE`.

`na.value` If `na.translate = TRUE`, what aesthetic value should the missing values be displayed as? Does not apply to position scales where NA is always placed at the far right.

`drop` Should unused factor levels be omitted from the scale? The default, TRUE, uses the levels that appear in the data; FALSE includes the levels in the factor. Please note that to display every level in a legend, the layer should use `show.legend = TRUE`.

`guide` A function used to create a guide or its name. See [guides\(\)](#) for more information.

`call` The call used to construct the scale for reporting messages.

`super` The super class to use for the constructed scale

`session_feedback_graph`

End of session feedback graph dependent on race and content area

Description

Returns a barchart for selections in the relevant questions of the end of coaching survey

Usage

```
session_feedback_graph(data)
```

Arguments

`data` the data to use

Value

prints a ggplot object

`square`

Calculate the square of a number

Description

This function computes the square of a numeric input.

Usage

```
square(x)
```


Arguments

x A numeric input.

Value

The square of the input.

Examples

```
tlShiny::square(2)
tlShiny::square(3.5)
```

stop_words *Various lexicons for English stop words*

Description

English stop words from three lexicons, as a data frame. The snowball and SMART sets are pulled from the tm package. Note that words with non-ASCII characters have been removed.

Usage

```
stop_words
```

Format

A data frame with 1149 rows and 2 variables:

word An English word

lexicon The source of the stop word. Either "onix", "SMART", or "snowball"

Source

- <http://www.lextek.com/manuals/onix/stopwords1.html>
- <https://www.jmlr.org/papers/volume5/lewis04a/lewis04a.pdf>
- <http://snowball.tartarus.org/algorithms/english/stop.txt>

student_bar_chart	<i>Student percent agree/strongly agree</i>
-------------------	---

Description

Automatically dodged bar chart for student data

Usage

```
student_bar_chart(
  data,
  col_select,
  agree_select,
  string_remove,
  title,
  legend_position = c(0.8, 0.25)
)
```

Arguments

data	the data for the plotter to use, should include all columns of interest
col_select	the columns to select with ‘tidyselect::contains’
agree_select	the type of agree/strongly agree to select, for example also often/always
string_remove	NULL by default, provides an optional string removal
title	the title for the plot
legend_position	c(0.8, 0.25) by default, adjustable

Value

a ggplot object

subsites	<i>The current subsites selection</i>
----------	---------------------------------------

Description

All Subsites as currently named in Qualtrics Reference Survey

Format

A vector of subsite names

Source

<https://docs.google.com/spreadsheets/d/11j1o9UeWxZGwunhDb24hZBwAKc5b8ZKM9AYNWZaUyZY/edit#gid=243942212>

survey_monkey	<i>A survey monkey text selection</i>
---------------	---------------------------------------

Description

Survey monkey reviews from the course survey in response to the question What is the learning from this course that you are most excited about trying out?, as a data frame.

Usage

```
survey_monkey
```

Format

A data frame with 87 rows and 1 variable:

‘What is the learning from this course that you are most excited about trying out?’ A deidentified review in each row

@source <https://www.surveymonkey.com/r/TLendofcourse>

theme_tl	<i>Teaching Lab Custom Ggplot2 Theme</i>
----------	--

Description

Teaching Lab Custom Ggplot2 Theme

Usage

```
theme_tl(  
  base_family = "Calibri",  
  base_size = 14,  
  strip_text_family = base_family,  
  strip_text_size = 15,  
  plot_title_family = "Calibri",  
  plot_title_size = 20,  
  plot_title_margin = 10,  
  subtitle_family = "Roboto",  
  subtitle_size = 15,  
  subtitle_margin = 15,  
  caption_family = "Roboto",  
  caption_size = 11,  
  caption_margin = 10,  
  axis_title_family = "Calibri",  
  axis_title_size = 12,
```

```

axis_title_just = "mm",
axis_text_size = 10.5,
dark = FALSE,
grid = TRUE,
axis = FALSE,
ticks = FALSE,
markdown = FALSE,
legend = F
)

```

Arguments

base_family	base font family
base_size	base font size
strip_text_family	facet label font family
strip_text_size	facet label text size
plot_title_family	plot title family
plot_title_size	plot title font size
plot_title_margin	plot title margin
subtitle_family	plot subtitle family
subtitle_size	plot subtitle size
subtitle_margin	plot subtitle margin
caption_family	plot caption family
caption_size	plot caption size
caption_margin	plot caption margin
axis_title_family	axis title font family
axis_title_size	axis title font size
axis_title_just	axis title font justification blmcrt
axis_text_size	axis text size
dark	dark mode TRUE, FALSE
grid	panel grid (TRUE, FALSE, or a combination of X, x, Y, y)
axis	axis TRUE, FALSE, [xy]
ticks	ticks TRUE, FALSE
markdown	enabled ggtext markdown styling TRUE, FALSE
legend	default no legend with F

Note

It requires installing Roboto, Calibri fonts unless you change the font parameters

<https://www.google.com/fonts>

tl_palette

Teaching Lab Color Palette Maker

Description

Teaching Lab Color Palette Maker

Usage

```
tl_palette(  
  color = c("blue", "orange", "purple", "green", "teal", "tl_colors"),  
  theme = c("dark"),  
  n = 6,  
  base_color_start = NULL  
)
```

Arguments

color	the color palette to generate
theme	if theme is light or dark
n	number of colors to generate
base_color_start	the base color of the palette to start with, ramping towards the color option

Value

color ramp palette function

tl_palette2

Teaching Lab Color Palette Maker

Description

Teaching Lab Color Palette Maker

Usage

```
tl_palette2(n = 6, base_color_start = NULL, end_color_start = NULL)
```

Arguments

n number of colors to generate
 base_color_start the base color of the palette to start with, ramping from the color option
 base_color_end the base color of the palette to end with, ramping towards the color option

Value

color ramp palette function

tl_pal_blue	<i>TL Default Blue Palette</i>
-------------	--------------------------------

Description

TL Default Blue Palette

Usage

tl_pal_blue

Format

An object of class character of length 8.

tl_select_percent	<i>Get the percent of a column that equals specific values</i>
-------------------	--

Description

Automatically scaled stacked bar chart with TL theming

Usage

tl_select_percent(data, percent_equal)

Arguments

data the data for the plotter to use, should include all columns of interest
 percent_equal string inputs to find the percent of the column that equals those values

Value

a percentage as a string

Index

- * **datasets**
 - na_df, 18
 - no_data_plot_currently, 19
 - no_data_plot_filters, 20
 - stop_words, 25
 - survey_monkey, 27
 - tl_pal_blue, 30
- %!in%(notin), 19
- agree_strongly_agree, 3
- calc_nps, 3
- check_email_approved, 4
- check_email_domain, 4
- contact_lead_graph, 5
- course_feedback_graph, 5
- drop1, 6
- end_coaching_feedback_graph, 6
- file.path2, 7
- get_percent_positive_student_survey, 7
- ggplot2::discrete_scale, 23
- grade_ipg, 7
- gt_percent_n, 9
- gt_theme_tl, 10
- gtable_extract_grob, 8
- gtable_remove_grob, 8
- guides(), 24
- highlight_fun, 10
- html_wrap, 11
- know_assess_summary, 12
- know_assess_summary_detailed, 12
- lambda, 23
- make_ipg_ela_summary_chart, 13
- make_ipg_fsot_summary_chart, 13
- make_ipg_math_summary_chart, 14
- make_student_work_chart_circle, 14
- make_student_work_chart_people, 15
- make_teacher_curriculum_perceptions, 15
- make_teacher_curriculum_usage, 16
- make_teacher_lesson_usage, 16
- make_teacher_perceptions_peer_relationships, 17
- make_teacher_perceptions_school_leaders, 17
- na_df, 18
- neg_cond_filter, 18
- no_data_plot_currently, 19
- no_data_plot_custom, 19
- no_data_plot_filters, 20
- notin, 19
- ongoing_coaching_feedback_graph, 20
- quote_viz, 21
- remove_empty_tl, 22
- scale_color_tl (scale_colour_tl), 22
- scale_colour_tl, 22
- scale_fill_tl (scale_colour_tl), 22
- scales::pal_hue(), 23
- session_feedback_graph, 24
- square, 24
- stop_words, 25
- student_bar_chart, 26
- subsites, 26
- survey_monkey, 27
- theme_tl, 27
- tl_pal_blue, 30
- tl_palette, 29
- tl_palette(), 22

`tl_palette2`, [29](#)

`tl_select_percent`, [30](#)