Package 'tinysnapshot'

Type Package

Title Snapshots for Unit Tests using the 'tinytest' Framework

Version 0.1.0

Maintainer Vincent Arel-Bundock <vincent.arel-bundock@umontreal.ca>

Description Snapshots for unit tests using the 'tinytest' framework for R. Includes expectations to test base R and 'ggplot2' plots as well as console output from print().

License GPL (>= 3)

URL https://github.com/vincentarelbundock/tinysnapshot

BugReports https://github.com/vincentarelbundock/tinysnapshot/issues

Imports diffobj, magick (>= 2.7.4), tinytest (>= 1.4.1)

Suggests fontquiver, ggplot2, ragg, rsvg, svglite

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Vincent Arel-Bundock [aut, cre]
    (<https://orcid.org/0000-0003-2042-7063>)

Repository CRAN

Date/Publication 2025-03-16 19:50:02 UTC

Contents

---

tinysnapshot-package *Snapshots for Unit Tests using the 'tinytest' Framework*

Description

Snapshots for unit tests using the 'tinytest' framework for R. Includes expectations to test base R and 'ggplot2' plots as well as console output from print().

Package Content

Index of help topics:

```
expect_equivalent_images
                    Test if two image files are equivalent
expect_snapshot_plot  Test if the new plot matches a target
                    (snapshot) plot
expect_snapshot_print  Test if printed output matches a target
                    printout
tinysnapshot-package  Snapshots for Unit Tests using the 'tinytest'
                    Framework
```

Maintainer

Vincent Arel-Bundock <vincent.arel-bundock@umontreal.ca>

Author(s)

Vincent Arel-Bundock [aut, cre] (<https://orcid.org/0000-0003-2042-7063>)

---

expect_equivalent_images

Test if two image files are equivalent

Description

Test if two image files are equivalent

Usage

```
expect_equivalent_images(
  current,
  target,
  tol = getOption("tinysnapshot_tol", default = 0),
  metric = getOption("tinysnapshot_metric", default = "AE"),
  fuzz = getOption("tinysnapshot_fuzz", default = 0),
  style = getOption("tinysnapshot_plot_diff_style", default = c("old", "new", "diff")),
```

```
    review = getOption("tinysnapshot_plot_review", default = TRUE),
    diffpath = NULL
)
```

Arguments

`current`	path to an image file
`target`	path to an image file
`tol`	distance estimates larger than this threshold will trigger a test failure. Scale depends on the `metric` argument. With the default `metric="AE"` (absolute error), the `tolerance` corresponds roughly to the number of pixels of difference between the plot and the reference image.
`metric`	string with a metric from `magick::metric_types()` such as `"AE"` or `"phash"`.
`fuzz`	relative color distance between 0 and 100 to be considered similar.
`style`	A character vector to control the panels of the diff image saved to file. The order and number of entries controls the side-by-side panels. Allowable values are: "old", "new", "diff".
`review`	logical. TRUE if a diff plot should be saved to file for review when the expectation fails.
`diffpath`	path where to save an image which shows the differences between `current` and `target`. NULL means that the diff image is not saved.

Value

A `tinytest` object. A `tinytest` object is a `logical` with attributes holding information about the test that was run

---

expect_snapshot_plot   *Test if the new plot matches a target (snapshot) plot*

Description

This expectation can be used with `tinytest` to check if the new plot matches a target plot.

When the expectation is checked for the first time, the expectation fails and a reference plot is saved to the `inst/tinytest/_tinysnapshot` folder.

When the expectation fails, the reference plot, the new plot, and a diff are saved to the `inst/tinytest/label` folder. Call the `review()` function to compare.

To update a snapshot, delete the reference file from the `_tinysnapshot` folder and run the test suite again.

See the package README file or website for detailed examples.

Usage

```
expect_snapshot_plot(
  current,
  label,
  width = getOption("tinysnapshot_width", default = NULL),
  height = getOption("tinysnapshot_height", default = NULL),
  tol = getOption("tinysnapshot_tol", default = 0),
  metric = getOption("tinysnapshot_metric", default = "AE"),
  fuzz = getOption("tinysnapshot_fuzz", default = 0),
  device = getOption("tinysnapshot_device", default = "svg"),
  device_args = getOption("tinysnapshot_device_args", default = list()),
  style = getOption("tinysnapshot_plot_diff_style", default = c("old", "new", "diff")),
  review = getOption("tinysnapshot_plot_review", default = TRUE),
  os = getOption("tinysnapshot_os", default = Sys.info()["sysname"])
)
```

Arguments

`current`	an object of class `ggplot` or a function which returns a base R plot.
`label`	a string to identify the snapshot (alpha-numeric, hyphens, or underscores). Each plot in the test suite must have a unique label.
`width`	of the snapshot. PNG default: 480 pixels. SVG default: 7 inches.
`height`	of the snapshot. PNG default: 480 pixels. SVG default: 7 inches.
`tol`	distance estimates larger than this threshold will trigger a test failure. Scale depends on the `metric` argument. With the default `metric="AE"` (absolute error), the `tolerance` corresponds roughly to the number of pixels of difference between the plot and the reference image.
`metric`	string with a metric from `magick::metric_types()` such as `"AE"` or `"phash"`.
`fuzz`	relative color distance between 0 and 100 to be considered similar.
`device`	"svg", "png", "ragg" or "svglite"
`device_args`	list of arguments to pass to the device call (e.g., `user_fonts` for `svglite` device).
`style`	A character vector to control the panels of the diff image saved to file. The order and number of entries controls the side-by-side panels. Allowable values are: "old", "new", "diff".
`review`	logical. TRUE if a a diff plot should be saved to file for review when the expectation fails.
`os`	character vector of operating systems on which the test should be run (e.g., "Windows", "Linux", "Darwin"). Tests are skipped when no element of the vector matches the output of: `Sys.info()["sysname"]`

Value

A `tinytest` object. A `tinytest` object is a `logical` with attributes holding information about the test that was run

---

expect_snapshot_print   *Test if printed output matches a target printout*

Description

This expectation can be used with `tinytest` to check if the new plot matches a target plot.

When the expectation is checked for the first time, the expectation fails and a reference text file is saved to the `inst/tinytest/_tinysnapshot` folder.

To update a snapshot, delete the reference file from the `_tinysnapshot` folder and run the test suite again.

See the package README file or website for detailed examples.

Usage

```
expect_snapshot_print(
  current,
  label,
  mode = getOption("tinysnapshot_mode", default = "unified"),
  format = getOption("tinysnapshot_format", default = "ansi256"),
 ignore_white_space = getOption("tinysnapshot_ignore_white_space", default = FALSE),
  fn_current = getOption("tinysnapshot_fn_current", default = identity),
  fn_target = getOption("tinysnapshot_fn_target", default = identity),
  ...
)
```

Arguments

`current`	an object which returns text to the console when calling `print(x)`‘
`label`	a string to identify the snapshot (alpha-numeric, hyphens, or underscores). Each plot in the test suite must have a unique label.
`mode`	"unified", "sidebyside", "context", or "auto". See `?diffobj::diffPrint`
`format`	"raw", "ansi8", "ansi256", "html", or "auto". See `?diffobj::diffPrint`
`ignore_white_space`	
	TRUE to ignore horizontal white space and empty lines.
`fn_current`	A function to apply to the current output before comparison.
`fn_target`	A function to apply to the target output before comparison.
`...`	Additional arguments are passed to `diffobj::diffPrint()`

Value

A `tinytest` object. A `tinytest` object is a `logical` with attributes holding information about the test that was run

Index