Package ‘hySpc.testthat’

June 24, 2020

Type Package

Title 'testthat' Unit Test Enhancements

Version 0.2.1

Maintainer Claudia Beleites <Claudia.Beleites@chemometrix.gmbh>

Description Enhance package 'testthat' by allowing tests to be attached to the function/object they test. This allows to keep functional and unit test code together.

License MIT + file LICENSE

Encoding UTF-8

LazyLoad yes

LazyData yes

Depends testthat

RoxygenNote 7.1.0

Collate 'test.R' 'gettest.R' 'test-fun.R' 'unittest.R'

NeedsCompilation no

Author Claudia Beleites [aut, cre],
Erick Oduniyi [aut]

Repository CRAN

Date/Publication 2020-06-24 13:00:02 UTC

R topics documented:

get_test .................................................. 2

test<- ................................................... 3

test_fun ................................................ 3

unittest ............................................... 4

Index 5
get_test

Get test that is attached to object as "test" attribute

Description

The returned function can be executed. If that is done within a testthat::Reporter, the results will be reported accordingly. Without reporter, the test runs silently if successful and stops if it fails.

Usage

get_test(object)

Arguments

object to which the test is attached

Value

the test (function)

Examples

```r
f <- function(x) x^2
test(f) <- function() {
  context("f")
  test_that("correct result for complex number", {
    expect_equal(f(1i), -1 + 0i)
  })
}
get_test(f)
```

```r
library(testthat)
## execute the test
get_test(f)()
with_reporter("summary", get_test(f)())
```
**test<-**  
*Attach unit tests to objects*

**Description**
This function attaches unit tests in value to an object (typically a function) as an attribute "test".

**Usage**
```r
test(f) <- value
```

**Arguments**
- `f`: the function (object) to which the tests are to be attached
- `value`: the test code, a function with no parameters

**Value**
`f` with the test attached as attribute "test"

**Examples**
```r
f <- function(x) x^2

test(f) <- function() {
  context("f")
  test_that("correct result for complex number", {
    expect_equal(f(1i), -1 + 0i)
  })
}
```

---

**test_fun**  
*Run test attached to function*

**Description**
Execute test attached to a function with `testthat::Reporter`.

**Usage**
```r
test_fun(object, reporter = "minimal")
```

**Arguments**
- `object`: to which the test is attached
- `reporter`: `testthat::Reporter` to use
### Description

If `testthat::testthat-package` is available, run the unit tests and display the results.

### Usage

```r
unittest(ns, standalone = TRUE, reporter = "progress")
```

### Arguments

- **ns**
  - namespace (package) to test

- **standalone**
  - if TRUE, appropriate `testthat::Reporters` are used, if FALSE the execution assumes reporters are started already.

- **reporter**
  - `testthat::Reporter` to be used.

### Value

Invisibly returns a data.frame with the test results

### Author(s)

Claudia Beleites

### Examples

```r
unittest("hySpc.testthat")
```
Index

*Topic **programming**
  unittest, 4

*Topic **utilities**
  unittest, 4

get_test, 2

test<-, 3
test_fun, 3
testthat::Reporter, 2–4
testthat::testthat-package, 4

unittest, 4