Package 'switchcase'

July 23, 2025

Type Package

Title A Simple and Flexible Switch-Case Construct for the 'R' Language

Version 0.1.1

Maintainer Joachim Zuckarelli <joachim@zuckarelli.de>

Description

Provides a switch-case construct for 'R', as it is known from other programming languages. It allows to test multiple, similar conditions in an efficient, easy-to-read manner, so nested if-else constructs can be avoided. The switch-case construct is designed as an 'R' function that allows to return values depending on which condition is met and lets the programmer flexibly decide whether or not to leave the switch-case construct after a case block has been executed.

License GPL-3

BugReports https://github.com/jsugarelli/switchcase/issues

URL https://github.com/jsugarelli/switchcase/

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

NeedsCompilation no

Author Joachim Zuckarelli [aut, cre] (ORCID:
<https://orcid.org/0000-0002-9280-3016>)

Repository CRAN

Date/Publication 2020-05-17 01:30:03 UTC

Contents

---

alt *Building alterantive 'case' branches for the switch-case construct*

Description

The alt() function is used to build alternative 'case' branches for the switch-case construct. Each alternative branch consist of four elements: The test condition, the code that is executed if the condition evaluates to TRUE, an optional return value which switchCase() will return if the condition is met and after the branch code has ben executed, and an optional logical value indicating if the switch-case construct will be left after this branch has been executed.

Usage

```
alt(condition, code.if, return.val = NULL, break.case = TRUE)
```

Arguments

condition	The condition that the test expression (expr argument in the call of the switchCase() function) needs to fulfill.
code.if	The code that is executed if the condition is met. If the code stretches over multiple R statements then use curly brackets to enclose the whole code segment.
return.val	A value that is returned if the condition is met and after the code in code.if has been executed. See switchCase() for more details on return values.
break.case	A logical value indicating if the switch-case construct will be left after the condition of this alternative 'case' branch of the switch-case construct has been met and the code in code.if has been executed. Default is TRUE, i.e. exiting the switch-case construct after this branch. See switchCase() for more details on breaking out of a switch-case construct.

Value

An alternative branch that can be used in a switch-case construct (technically, a list object).

See Also

Other switchcase: switchCase()

Examples

```
alt(.(..expr > 10), .(cat("The result is larger than 10.")), "10plus")
```

---

switchCase   *The switch-case construct*

Description

switchCase() provides the functionality of a typical switch-case statement as it is available in other programming languages.

Usage

```
switchCase(expr, ..., break.case = TRUE)
```

Arguments

expr   The expression which is evaluated and the value of which is tested in the alternative 'case' code branches.

...   The alternative code branches ('case' blocks). Each alternative is produced with the [alt]() function and is basically a list with up to four elements: (1) The expression against which the expr argument is tested (if the resulting condition is TRUE then the code of this alternative is executed) - if this first element of the alternative is NULL then this alternative is the default of the switch-case construct and will be executed if no condition of any other alternative evaluate to TRUE; (2) the code of the alternative that is executed if the condition of this alternative is fulfilled; (3) an optional return value that will be returned by switchCase() if the condition of this alternative evaluates to TRUE and after the alternative's code has been executed (default is NULL which means nothing is returned); (4) an optional logical value (with default TRUE) that indicates if the switch-case construct will be left after the code of this alternative has been executed, or if the subsequent alternatives are tested as well. This value overrides any option provided via the break.case argument.

break.case An optional logical value with default TRUE indicating if the switch-case construct will be left after the condition of one of the alternative is fulfilled and this alternative's code has been executed. This can be overwritten with the fourth element of an alternative 'case' branch produced with the [alt]() function and provided via the ... argument.

Details

The expr argument is the expression that is evaluated and tested. Using the ..expr placeholder, this expression can be used in the first element (the test condition) of an alternative branch provided via the ... argument. If, for example, the first element of an alternative branch is ..expr > 10 then the condition of this alternative 'case' will be met if the value of swichCase()'s expr argument is larger than 10. Each alternative 'case' branch can return a value if its condition is fulfilled. This value needs to be provided as the third element of the alternative 'case' branch (the third argument of the [alt]() function that is used to create alternative branches) and is NULL by default resulting in no return whatsoever. If multiple branches are executed (e.g. because break.case = FALSE) and want to return a value then only the 'case' branch that is execxuted *last* will return its value. A default

'case' which is executed if no other alternative branch of the switch-case statement is applicable can be modeled by setting the first element of an alternative (which would normally contain the test condition) to NULL. If multiple defaults are found, only the first one is executed. Technical note: All code is executed in the enviroment from which switchCase() is called so it is easy to access variables and functions from there.

Value

The return value of the 'case' alternative that is executed last. If there is no applicable alternative or no applicable alternative returns a value then nothing is retuned.

See Also

Other switchcase: [alt](../)()

Examples

```
# Calculation and interpretation of a person's body-mass index (mass in kilograms)
bmi <- function(mass, size) {
ind <- mass / size^2
switchCase(
  ind,
  alt(
    ..expr <= 15,
    { cat("Your body mass index is ", ind, " which is very severely underweight.\n") },
    "very severely underweight"
  ),
  alt(
    ..expr > 15 & ..expr <= 16,
    { cat("Your body mass index is ", ind, " which is severely underweight.\n") },
  "severely underweight"
  ),
  alt(
    ..expr > 16 & ..expr <= 18.5,
    { cat("Your body mass index is ", ind, " which is underweight.\n") },
    "underweight"
  ),
  alt(
    ..expr > 18.5 & ..expr <= 25,
    { cat("Your body mass index is ", ind, " which is normal.\n") },
    "normal"
  ),
  alt(
    ..expr > 25 & ..expr <= 30,
    { cat("Your body mass index is ", ind, " which is overweight.\n") },
    "overweight"
  ),
  alt(
    ..expr > 30 & ..expr <= 35,
    { cat("Your body mass index is ", ind, " which is moderately obese.\n") },
    "moderately obese"
  ),
```

```
    alt(
      ..expr > 35 & ..expr <= 40,
      { cat("Your body mass index is ", ind, " which is severely obese.\n") },
      "severly obese"
    ),
    alt(
      ..expr > 40,
      { cat("Your body mass index is ", ind, " which is Very severely obese.\n") },
      "very severly obese"
    )
  )
}
bmi.person1 <- bmi(82.5, 1.79)
cat("Person1 turned out to be ", bmi.person1)
```

Index