Labelling data is typically a task for end-users and is applied in own scripts or functions rather than in packages. However, sometimes it can be useful for both end-users and package developers to have a flexible way to add variable and value labels to their data. In such cases, [quasiquotation](https://adv-r.hadley.nz/quasiquotation.html" \t "_blank) is helpful.

This vignette demonstrate how to use quasiquotation in *[sjlabelled](https://strengejacke.github.io/sjlabelled/" \t "_blank)* to label your data.

**Adding value labels to variables using quasiquotation**

Usually, [set\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_labels.html" \t "_blank) can be used to add value labels to variables. The syntax of this function is easy to use, and [set\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_labels.html" \t "_blank)allows to add value labels to multiple variables at once, if these variables share the same value labels.

In the following examples, we will use the [frq()](https://strengejacke.github.io/sjmisc/reference/frq.html" \t "_blank) function, that shows an extra **label**-column containing *value labels*, if the data is labelled. If the data has *no* value labels, this column is not shown in the output.

[library](https://www.rdocumentation.org/packages/base/topics/library?tap_a=5644-dce66f&tap_s=10907-287229)(sjlabelled)

[library](https://www.rdocumentation.org/packages/base/topics/library?tap_a=5644-dce66f&tap_s=10907-287229)(sjmisc) # for frq()-function

[library](https://www.rdocumentation.org/packages/base/topics/library?tap_a=5644-dce66f&tap_s=10907-287229)(rlang)

# unlabelled data

dummies <- [data.frame](https://www.rdocumentation.org/packages/base/topics/data.frame?tap_a=5644-dce66f&tap_s=10907-287229)(

dummy1 = [sample](https://www.rdocumentation.org/packages/base/topics/sample?tap_a=5644-dce66f&tap_s=10907-287229)(1:3, 40, replace = TRUE),

dummy2 = [sample](https://www.rdocumentation.org/packages/base/topics/sample?tap_a=5644-dce66f&tap_s=10907-287229)(1:3, 40, replace = TRUE),

dummy3 = [sample](https://www.rdocumentation.org/packages/base/topics/sample?tap_a=5644-dce66f&tap_s=10907-287229)(1:3, 40, replace = TRUE)

)

# set labels for all variables in the data frame

test <- [set\_labels](https://strengejacke.github.io/sjlabelled/reference/set_labels.html)(dummies, labels = [c](https://www.rdocumentation.org/packages/base/topics/c?tap_a=5644-dce66f&tap_s=10907-287229)("low", "mid", "hi"))

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$dummy1, "labels")

#> low mid hi

#> 1 2 3

frq(test, dummy1)

#>

#> # dummy1

#> # total N=40 valid N=40 mean=2.23 sd=0.86

#>

#> val label frq raw.prc valid.prc cum.prc

#> 1 low 11 27.5 27.5 27.5

#> 2 mid 9 22.5 22.5 50.0

#> 3 hi 20 50.0 50.0 100.0

#> NA NA 0 0.0 NA NA

# and set same value labels for two of three variables

test <- [set\_labels](https://strengejacke.github.io/sjlabelled/reference/set_labels.html)(

dummies, dummy1, dummy2,

labels = [c](https://www.rdocumentation.org/packages/base/topics/c?tap_a=5644-dce66f&tap_s=10907-287229)("low", "mid", "hi")

)

frq(test)

#>

#> # dummy1

#> # total N=40 valid N=40 mean=2.23 sd=0.86

#>

#> val label frq raw.prc valid.prc cum.prc

#> 1 low 11 27.5 27.5 27.5

#> 2 mid 9 22.5 22.5 50.0

#> 3 hi 20 50.0 50.0 100.0

#> NA NA 0 0.0 NA NA

#>

#> # dummy2

#> # total N=40 valid N=40 mean=2.10 sd=0.74

#>

#> val label frq raw.prc valid.prc cum.prc

#> 1 low 9 22.5 22.5 22.5

#> 2 mid 18 45.0 45.0 67.5

#> 3 hi 13 32.5 32.5 100.0

#> NA NA 0 0.0 NA NA

#>

#> # dummy3

#> # total N=40 valid N=40 mean=1.98 sd=0.83

#>

#> val frq raw.prc valid.prc cum.prc

#> 1 14 35.0 35.0 35.0

#> 2 13 32.5 32.5 67.5

#> 3 13 32.5 32.5 100.0

#> 0 0.0 NA NA

[val\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_labels.html) does the same job as [set\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_labels.html" \t "_blank), but in a different way. While [set\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_labels.html" \t "_blank) requires variables to be specified in the ...-argument, and labels in the labels-argument, [val\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_labels.html" \t "_blank) requires both to be specified in the ....

[val\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_labels.html) requires *named* vectors as argument, with the *left-hand side* being the name of the variable that should be labelled, and the *right-hand side* containing the labels for the values.

test <- [val\_labels](https://strengejacke.github.io/sjlabelled/reference/set_labels.html)(dummies, dummy1 = [c](https://www.rdocumentation.org/packages/base/topics/c?tap_a=5644-dce66f&tap_s=10907-287229)("low", "mid", "hi"))

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$dummy1, "labels")

#> low mid hi

#> 1 2 3

# remaining variables are not labelled

frq(test)

#>

#> # dummy1

#> # total N=40 valid N=40 mean=2.23 sd=0.86

#>

#> val label frq raw.prc valid.prc cum.prc

#> 1 low 11 27.5 27.5 27.5

#> 2 mid 9 22.5 22.5 50.0

#> 3 hi 20 50.0 50.0 100.0

#> NA NA 0 0.0 NA NA

#>

#> # dummy2

#> # total N=40 valid N=40 mean=2.10 sd=0.74

#>

#> val frq raw.prc valid.prc cum.prc

#> 1 9 22.5 22.5 22.5

#> 2 18 45.0 45.0 67.5

#> 3 13 32.5 32.5 100.0

#> 0 0.0 NA NA

#>

#> # dummy3

#> # total N=40 valid N=40 mean=1.98 sd=0.83

#>

#> val frq raw.prc valid.prc cum.prc

#> 1 14 35.0 35.0 35.0

#> 2 13 32.5 32.5 67.5

#> 3 13 32.5 32.5 100.0

#> 0 0.0 NA NA

Unlike [set\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_labels.html" \t "_blank), [val\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_labels.html" \t "_blank) allows the user to add *different* value labels to different variables in one function call. Another advantage, or difference, of [val\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_labels.html" \t "_blank) is it’s flexibility in defining variable names and value labels by using quasiquotation.

**Add labels that are stored in a vector**

To use quasiquotation, we need the **rlang** package to be installed and loaded. Now we can have labels in a character vector, and use !! to unquote this vector.

labels <- [c](https://www.rdocumentation.org/packages/base/topics/c?tap_a=5644-dce66f&tap_s=10907-287229)("low\_quote", "mid\_quote", "hi\_quote")

test <- [val\_labels](https://strengejacke.github.io/sjlabelled/reference/set_labels.html)(dummies, dummy1 = !! labels)

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$dummy1, "labels")

#> low\_quote mid\_quote hi\_quote

#> 1 2 3

**Define variable names that are stored in a vector**

The same can be done with the names of *variables* that should get new value labels. We then need !! to unquote the variable name and := as assignment.

variable <- "dummy2"

test <- [val\_labels](https://strengejacke.github.io/sjlabelled/reference/set_labels.html)(dummies, !! variable := [c](https://www.rdocumentation.org/packages/base/topics/c?tap_a=5644-dce66f&tap_s=10907-287229)("lo\_var", "mid\_var", "high\_var"))

# no value labels

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$dummy1, "labels")

#> NULL

# value labels

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$dummy2, "labels")

#> lo\_var mid\_var high\_var

#> 1 2 3

**Both variable names and value labels are stored in a vector**

Finally, we can combine the above approaches to be flexible regarding both variable names and value labels.

variable <- "dummy3"

labels <- [c](https://www.rdocumentation.org/packages/base/topics/c?tap_a=5644-dce66f&tap_s=10907-287229)("low", "mid", "hi")

test <- [val\_labels](https://strengejacke.github.io/sjlabelled/reference/set_labels.html)(dummies, !! variable := !! labels)

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$dummy3, "labels")

#> low mid hi

#> 1 2 3

**Adding variable labels using quasiquotation**

[set\_label()](https://strengejacke.github.io/sjlabelled/reference/set_label.html) is the equivalent to [set\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_labels.html" \t "_blank) to add variable labels to a variable. The equivalent to [val\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_labels.html" \t "_blank) is [var\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_label.html" \t "_blank), which works in the same way as [val\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_labels.html" \t "_blank). In case of *variable* labels, a label-attribute is added to a vector or factor (instead of a labels-attribute, which is used for *value* labels).

The following examples show how to use [var\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_label.html" \t "_blank) to add variable labels to the data. We demonstrate this function without further explanation, because it is actually very similar to [val\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_labels.html" \t "_blank).

dummy <- [data.frame](https://www.rdocumentation.org/packages/base/topics/data.frame?tap_a=5644-dce66f&tap_s=10907-287229)(

a = [sample](https://www.rdocumentation.org/packages/base/topics/sample?tap_a=5644-dce66f&tap_s=10907-287229)(1:4, 10, replace = TRUE),

b = [sample](https://www.rdocumentation.org/packages/base/topics/sample?tap_a=5644-dce66f&tap_s=10907-287229)(1:4, 10, replace = TRUE),

c = [sample](https://www.rdocumentation.org/packages/base/topics/sample?tap_a=5644-dce66f&tap_s=10907-287229)(1:4, 10, replace = TRUE)

)

# simple usage

test <- [var\_labels](https://strengejacke.github.io/sjlabelled/reference/set_label.html)(dummy, a = "first variable", c = "third variable")

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$a, "label")

#> [1] "first variable"

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$b, "label")

#> NULL

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$c, "label")

#> [1] "third variable"

# quasiquotation for labels

v1 <- "First variable"

v2 <- "Second variable"

test <- [var\_labels](https://strengejacke.github.io/sjlabelled/reference/set_label.html)(dummy, a = !! v1, b = !! v2)

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$a, "label")

#> [1] "First variable"

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$b, "label")

#> [1] "Second variable"

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$c, "label")

#> NULL

# quasiquotation for variable names

x1 <- "a"

x2 <- "c"

test <- [var\_labels](https://strengejacke.github.io/sjlabelled/reference/set_label.html)(dummy, !! x1 := "First", !! x2 := "Second")

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$a, "label")

#> [1] "First"

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$b, "label")

#> NULL

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$c, "label")

#> [1] "Second"

# quasiquotation for both variable names and labels

test <- [var\_labels](https://strengejacke.github.io/sjlabelled/reference/set_label.html)(dummy, !! x1 := !! v1, !! x2 := !! v2)

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$a, "label")

#> [1] "First variable"

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$b, "label")

#> NULL

[attr](https://www.rdocumentation.org/packages/base/topics/attr?tap_a=5644-dce66f&tap_s=10907-287229)(test$c, "label")

#> [1] "Second variable"

**Conclusion**

As we have demonstrated, [var\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_label.html" \t "_blank) and [val\_labels()](https://strengejacke.github.io/sjlabelled/reference/set_labels.html" \t "_blank) are one of the most flexible and easy-to-use ways to add value and variable labels to our data. Another advantage is the consistent design of all functions in **sjlabelled**, which allows seamless integration into pipe-workflows.