|  |  |
| --- | --- |
| merge {base} | R Documentation |

**Merge Two Data Frames**

**Description**

Merge two data frames by common columns or row names, or do other versions of database *join* operations.

**Usage**

merge(x, y, ...)

## Default S3 method:

merge(x, y, ...)

## S3 method for class 'data.frame'

merge(x, y, by = intersect(names(x), names(y)),

by.x = by, by.y = by, all = FALSE, all.x = all, all.y = all,

sort = TRUE, suffixes = c(".x",".y"),

incomparables = NULL, ...)

**Arguments**

|  |  |
| --- | --- |
| x, y | data frames, or objects to be coerced to one. |
| by, by.x, by.y | specifications of the columns used for merging. See ‘Details’. |
| all | logical; all = L is shorthand for all.x = L and all.y = L, where L is either [TRUE](http://127.0.0.1:14695/library/base/help/TRUE) or FALSE. |
| all.x | logical; if TRUE, then extra rows will be added to the output, one for each row in x that has no matching row in y. These rows will have NAs in those columns that are usually filled with values from y. The default is FALSE, so that only rows with data from both x and y are included in the output. |
| all.y | logical; analogous to all.x. |
| sort | logical. Should the result be sorted on the by columns? |
| suffixes | a character vector of length 2 specifying the suffixes to be used for making unique the names of columns in the result which are not used for merging (appearing in by etc). |
| incomparables | values which cannot be matched. See [match](http://127.0.0.1:14695/library/base/help/match). This is intended to be used for merging on one column, so these are incomparable values of that column. |
| ... | arguments to be passed to or from methods. |

**Details**

merge is a generic function whose principal method is for data frames: the default method coerces its arguments to data frames and calls the "data.frame" method.

By default the data frames are merged on the columns with names they both have, but separate specifications of the columns can be given by by.x and by.y. The rows in the two data frames that match on the specified columns are extracted, and joined together. If there is more than one match, all possible matches contribute one row each. For the precise meaning of ‘match’, see [match](http://127.0.0.1:14695/library/base/help/match).

Columns to merge on can be specified by name, number or by a logical vector: the name "row.names" or the number 0 specifies the row names. If specified by name it must correspond uniquely to a named column in the input.

If by or both by.x and by.y are of length 0 (a length zero vector or NULL), the result, r, is the *Cartesian product* of x and y, i.e., dim(r) = c(nrow(x)\*nrow(y), ncol(x) + ncol(y)).

If all.x is true, all the non matching cases of x are appended to the result as well, with NA filled in the corresponding columns of y; analogously for all.y.

If the columns in the data frames not used in merging have any common names, these have suffixes (".x" and ".y" by default) appended to try to make the names of the result unique. If this is not possible, an error is thrown.

The complexity of the algorithm used is proportional to the length of the answer.

In SQL database terminology, the default value of all = FALSE gives a *natural join*, a special case of an *inner join*. Specifying all.x = TRUE gives a *left (outer) join*, all.y = TRUE a *right (outer) join*, and both (all = TRUE a *(full) outer join*. DBMSes do not match NULL records, equivalent to incomparables = NA in **R**.

**Value**

A data frame. The rows are by default lexicographically sorted on the common columns, but for sort = FALSE are in an unspecified order. The columns are the common columns followed by the remaining columns in x and then those in y. If the matching involved row names, an extra character column called Row.names is added at the left, and in all cases the result has ‘automatic’ row names.

**Note**

This is intended to work with data frames with vector-like columns: some aspects work with data frames containing matrices, but not all.

Currently long vectors are not accepted for inputs, which are thus restricted to less than 2^31 rows. Prior to **R** 3.2.0 that restriction also applied to the result (and still does for 32-bit platforms).

**See Also**

[data.frame](http://127.0.0.1:14695/library/base/help/data.frame), [by](http://127.0.0.1:14695/library/base/help/by), [cbind](http://127.0.0.1:14695/library/base/help/cbind).

[dendrogram](http://127.0.0.1:14695/library/base/help/dendrogram) for a class which has a merge method.

**Examples**

## use character columns of names to get sensible sort order

authors <- data.frame(

surname = I(c("Tukey", "Venables", "Tierney", "Ripley", "McNeil")),

nationality = c("US", "Australia", "US", "UK", "Australia"),

deceased = c("yes", rep("no", 4)))

books <- data.frame(

name = I(c("Tukey", "Venables", "Tierney",

"Ripley", "Ripley", "McNeil", "R Core")),

title = c("Exploratory Data Analysis",

"Modern Applied Statistics ...",

"LISP-STAT",

"Spatial Statistics", "Stochastic Simulation",

"Interactive Data Analysis",

"An Introduction to R"),

other.author = c(NA, "Ripley", NA, NA, NA, NA,

"Venables & Smith"))

(m1 <- merge(authors, books, by.x = "surname", by.y = "name"))

(m2 <- merge(books, authors, by.x = "name", by.y = "surname"))

stopifnot(as.character(m1[, 1]) == as.character(m2[, 1]),

all.equal(m1[, -1], m2[, -1][ names(m1)[-1] ]),

dim(merge(m1, m2, by = integer(0))) == c(36, 10))

## "R core" is missing from authors and appears only here :

merge(authors, books, by.x = "surname", by.y = "name", all = TRUE)

## example of using 'incomparables'

x <- data.frame(k1 = c(NA,NA,3,4,5), k2 = c(1,NA,NA,4,5), data = 1:5)

y <- data.frame(k1 = c(NA,2,NA,4,5), k2 = c(NA,NA,3,4,5), data = 1:5)

merge(x, y, by = c("k1","k2")) # NA's match

merge(x, y, by = "k1") # NA's match, so 6 rows

merge(x, y, by = "k2", incomparables = NA) # 2 rows