Introduction to R and Tidyverse Cheatsheet

The tables below consist of valuable functions and commands that will help you through this module.

Each table represents a different library/tool and the corresponding commands.

> Please note that these tables are not intended to tell you all the information you need to know about each command.
>
> The hyperlinks found in each piece of code will take you to the documentation for further information on the usage of each command.
> Please be aware that the documentation will generally provide information about the given function's most current version (or a recent version, depending on how often the documentation site is updated).
> This will usually (but not always!) match what you have installed on your machine.
> If you have a different version of R or other R packages, the documentation may differ from what you have installed.

Table of Contents

Base `R`

Read the Base `R` documentation.

Library/Package	Piece of code	What it's called	What it does
Base `R`	`library()`	Library	Loads and attaches additional packages to the R environment.
Base `R`	`<-`	Assignment operator	Assigns a name to something in the R environment.
Base `R`	`|>`	Pipe operator	Funnels an object from the output of one function to input of the next function
Base `R`	`c()`	Combine	Combines values into a vector or list.
Base `R`	`%in%`	"in" logical operator	Checks if the given value(s) on the left side of the operator are in the vector or other R object defined on the right side of the operator. It returns a logical `TRUE` or `FALSE` statement. This resource also provides a helpful explanation about its usage.
Base `R`	`rm(x)`	Remove	Removes object(s) `x` from your environment.
Base `R`	`==, <=, >=, !=`	Relational Operators	These are binary operators which allow for the comparison of values in an object.
Base `R`	`str(x)`	Object Structure	Gets a summary of the object `x` structure.
Base `R`	`class(x)`	Object Class	Returns the type of the values in object `x`.
Base `R`	`nrow(x)`; `ncol(x)`	Number of Rows; Number of Columns	Get the number of rows and the number of columns in an object `x`, respectively.
Base `R`	`length(x)`	Length	Returns how long the object `x` is.
Base `R`	`min(x)`	Minimum	Returns the minimum value of all values in an object `x`.
Base `R`	`sum(x)`	Sum	Returns the sum of all values (values must be integer, numeric, or logical) in object `x`.
Base `R`	`mean(x)`	Mean	Returns the arithmetic mean of all values (values must be integer or numeric) in object `x` or logical vector `x`.
Base `R`	`log(x)`	Logarithm	Gives the natural logarithm of object `x`. `log2(x)` can be used to give the logarithm of the object in base 2. Or the base can be specified as an argument.
Base `R`	`head()`; `tail()`	Head; Tail	Returns the top 6 (`head()`) or bottom 6 (`tail()`) an of an object in the environment by default. You can specify how many rows you want by including the `n = argument.

Library/Package	Piece of code	What it's called	What it does
Base `R`	`factor(x)` or `as.factor(x)`	Factor	Coerces object x into a factor (which is used to represent categorical data). This function can be used to coerce object `x` into other data types, i.e., `as.character`, `as.numeric`, `as.data.frame`, `as.matrix`, etc.
Base `R`	`levels(x)`	Levels attributes	Returns or sets the value of the levels in an object `x`.
Base `R`	`summary(x)`	Object summary	Returns a summary of the values in object `x`.
Base `R`	`data.frame()`	Data Frame	Creates a data frame where the named arguments will be the same length.
Base `R`	`sessionInfo()`	Session Information	Returns the R version information, the OS, and the attached packages in the current R session.
Base `R`	`file.path()`	File path	Constructs the path to a desired file.
Base `R`	`dir()`	Directory	Lists the names of the files and/or directories in the named directory.
Base `R`	`getwd()`	Get working directory	Finds the current working directory.
Base `R`	`setwd()`	Set working directory	Changes the current working directory.
Base `R`	`dir.exists()`	Directory exists	Checks the file path to see if the directory exists there.
Base `R`	`dir.create()`	Create directory	Creates a directory at the specified path.
Base `R`	`apply()`	Apply	Returns a vector or list of values after applying a specified function to values in each row/column of an object.
Base `R`	`round()`	Round	Rounds the values of an object to the specified number of decimal places (default is 0).
Base `R`	`names()`	Names	Gets or sets the names of an object.
Base `R`	`colnames()`	Column names	Gets or sets the column names of a matrix or data frame.
Base `R`	`all.equal()`	All equal	Checks if two R objects are nearly equal.
Base `R`	`all()`	All	Checks if all of the values are `TRUE` in a logical vector.
Base `R`	`t()`	Transpose	Returns the transpose of a matrix or data frame. If given a data frame, returns a matrix.

`tidyverse`

Read the `tidyverse` package documentation, as well as the philosophy behind the `tidyverse`.

`dplyr`

Read the `dplyr` package documentation, and a vignette on its usage.

Library/Package	Piece of code	What it's called	What it does
`dplyr` / `magrittr`	`%>%`	Pipe operator	Funnels an object from the output of one function to input of the next function (used like the base pipe `|>` found in `R` 4.1 and later, but can be used in earlier versions of `R`)
`dplyr`	`filter()`	Filter	Returns a subset of rows matching the conditions of the specified logical argument
`dplyr`	`arrange()`	Arrange	Reorders rows in ascending order. `arrange(desc())` would reorder rows in descending order.
`dplyr`	`select()`	Select	Selects columns that match the specified argument
`dplyr`	`mutate()`	Mutate	Adds a new column that is a function of existing columns
`dplyr`	`summarize()`	Summarize	Summarizes multiple values in an object into a single value. This function can be used with other functions to retrieve a single output value for the grouped values. `summarize` and `summarise` are synonyms in this package. However, note that this function does not work in the same manner as the base R `summary` function.
`dplyr`	`rename()`	Rename	Renames designated columns while keeping all variables of the data.frame
`dplyr`	`group_by()`	Group By	Groups data into rows that contain the same specified value(s)
`dplyr`	`inner_join()`	Inner Join	Joins data from two data frames, retaining only the rows that are in both datasets.

`ggplot2`

Read the `ggplot2` package documentation, an overall reference for `ggplot2` functions, and a vignette on the usage of the `ggplot2` aesthetics.
Additional vignettes are available from the "Articles" dropdown menu on this webpage.

Library/Package	Piece of code	What it's called	What it does
`ggplot2`	`ggplot()`	GG Plot	Begins a plot that is finished by adding layers.
`ggplot2`	`aes()`	Aesthetic Mappings	Designates how variables in the data object are mapped to the visual properties of the ggplot.
`ggplot2`	`geom_boxplot()`	Boxplot	Creates a boxplot when added as a layer to a `ggplot()` object.
`ggplot2`	`geom_density()`	Density Plot	Creates a smoothed plot when added as a layer to a `ggplot()` object based on the computed density estimate.
`ggplot2`	`geom_point()`	Scatterplot	Creates a scatterplot when added as a layer to a `ggplot()` object.
`ggplot2`	`geom_line()`	Line plot	Creates a line plot when added as a layer to a `ggplot()` object by connecting the points in order of the x axis variable.
`ggplot2`	`geom_hline()`	Horizontal line	Annotates a plot with a horizontal line when added as a layer to a `ggplot()` object
`ggplot2`	`geom_vline()`	Vertical line	Annotates a plot with a vertical line when added as a layer to a `ggplot()` object
`ggplot2`	`theme_classic()`	Classic Theme	Displays `ggplot` without gridlines. The `ggtheme` documentation has descriptions on additional themes that can be used.
`ggplot2`	`labs()`	Labels	Modify labels (axis, title, legends) on a `ggplot()` object.
`ggplot2`	`xlab()` ; `ylab()` ; `ggtitle()`	X Axis Labels; Y Axis Labels; GG Title	Alternative individual functions to add individual plot labels: x-axis, y-axis, and title, respectively.
`ggplot2`	`facet_wrap()`	Facet Wrap	Plots individual graphs using specified variables to subset the data.
`ggplot2`	`ggsave()`	GG Save	Saves the last plot in working directory.
`ggplot2`	`last_plot()`	Last plot	Returns the last plot produced.

`readr`, `fs`, `tibble` `tidyr`

Read the `readr` package documentation and a vignette on its usage.
Read the `fs` package documentation.
Read the `tibble` package documentation and a vignette on its usage.
Read the `tidyr` package documentation and a vignette on its usage.

Library/Package	Piece of code	What it's called	What it does
`readr`	`read_tsv()`	Read TSV	Reads in a TSV file from a specified file path. This function can be tailored to read in other common types of files, e.g. `read_csv()`, `read_rds()`, etc.
`fs`	`dir_create()`	Create directory	Create a directory, unless the directory already exists.
`tibble`	`column_to_rownames()`	Column to Rownames	Transforms an existing column called by a string into the rownames.
`tibble`	`rownames_to_column()`	Rownames to Column	Transforms the rownames of a data frame into a column (which is added to the start of the data frame). The string supplied as an argument will be the name of the new column.
`tidyr`	`pivot_longer()`	Pivot Longer	Lengthens a data frame by increasing the number of rows and decreasing the number of columns.