

# Day 4 Cheatsheet

## Data Summarization

### Functions

Library/Package	Piece of code	Example of usage	What it does
Base R	<code>min(x)</code>	<code>min(x)</code>	Returns the minimum value of all values in an object <code>x</code> .
Base R	<code>sum(x)</code>	<code>sum(x)</code>	Returns the sum of all values (values must be integer, numeric, or logical) in object <code>x</code> .
Base R	<code>mean(x)</code>	<code>mean(x)</code>	Returns the arithmetic mean of all values (values must be integer or numeric) in object <code>x</code> or logical vector <code>x</code> .
Base R	<code>log(x)</code>	<code>log(x)</code>	Gives the natural logarithm of object <code>x</code> . <code>log2(x)</code> can be used to give the logarithm of the object in base 2. Or the base can be specified as an argument.
Base R	<code>range(x)</code>	<code>range(x)</code>	Gives the min and max for object <code>x</code> .
Base R	<code>sd(x)</code>	<code>sd(x)</code>	Gives the standard deviation for object <code>x</code> .
Base R	<code>sqrt(x)</code>	<code>sqrt(x)</code>	Gives the square root for object <code>x</code> .
Base R	<code>quantile(x)</code>	<code>quantile(x, probs = .5)</code>	Produces sample quantiles corresponding to the given probabilities <code>x</code> .
Base R	<code>summary(x)</code>	<code>summary(x)</code>	Returns a summary of the values in object <code>x</code> .
dplyr	<code>pull()</code>	<code>x_vect &lt;- df %&gt;% pull(x)</code>	Extract a single column into vector form. <code>pull()</code> is very handy before summary functions like <code>mean()</code> , <code>sum()</code> , etc.

Library/Package	Piece of code	Example of usage	What it does
dplyr	summarize()	df <- df %>% summarize(mean_x = mean(x))	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. <b>summarize</b> and <b>summarise</b> are synonyms in this package. However, note that this function does not work in the same manner as the base R <b>summary</b> function.
dplyr	distinct()	df %>% distinct(factor_name)	Display unique/distinct rows from a data frame or tibble
dplyr	n_distinct()	x_vect %>% n_distinct()	Counts the number of unique/distinct combinations in a set of one or more vectors.
dplyr	count()	df %>% count(factor_name)	Count the number of groups in a factor variable of a data frame or tibble
dplyr	group_by()	df %>% group_by(factor_name)	Groups data into rows that contain the same specified value(s)
dplyr	ungroup()	df %>% ungroup()	Undo a grouping that was done by <b>group_by()</b>
Base R	unique()	unique(df)	Returns a vector, data frame or array like x but with duplicate elements/rows removed.
Base R	rowSums()	rowSums(df)	Calculates sums for each row
Base R	colSums()	colSums(df)	Calculates sums for each column
Base R	rowMeans()	rowMeans(df)	Calculates means for each row
Base R	colMeans()	colMeans(df)	Calculates means for each column

- Many summarizing functions (e.g., **mean()**, **sum()**) have the argument **na.rm = TRUE**. This can be used to ignore missing data.

## Data Classes

### Major concepts

- **Character** - strings or individual characters, quoted
- **Numeric** - any real number(s)
- **Double** - a special subset of numeric that contains fractional values.
- **Integer** - any integer(s)/whole numbers
- **Factor** - categorical/qualitative variables
- **Logical** - variables composed of TRUE or FALSE
- **Date/POSIXct** - represents calendar dates and times
- **matrix** - Two-dimensional class of data where all rows and columns consist of the same data type.
- **data frame** - Two-dimensional class of data where all columns can be of different data types.
- **list** - Can be of varying dimensions and can hold any kind of data type. Can hold vectors, strings, matrices, models, list of other lists.

### Functions

Library/Package	Piece of code	Example of usage	What it does
Base R	<code>as.numeric(x)</code>	<code>as.numeric(x)</code>	Coerces object x into numeric class. This type of function can be used to coerce object x into other data types, i.e., <code>as.character</code> , <code>as.numeric</code> , <code>as.data.frame</code> , <code>as.matrix</code> , <code>as.Date</code> etc.
lubridate	<code>ymd(x)</code>	<code>ymd("2024-01-31")</code>	Coerces character object x into date class. The format of the character object determines the function to use. Other examples include <code>mdy()</code> , <code>dmy()</code> , etc.

- `lubridate` is a powerful, widely used R package from “tidyverse” family to work with Date / POSIXct class objects

\* This format was adapted from the cheatsheet format from AlexsLemonade.