

# 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> .
Base R	<code>rowSums()</code>	<code>rowSums(df)</code>	Calculates sums for each row
Base R	<code>colSums()</code>	<code>colSums(df)</code>	Calculates sums for each column
Base R	<code>rowMeans()</code>	<code>rowMeans(df)</code>	Calculates means for each row
Base R	<code>colMeans()</code>	<code>colMeans(df)</code>	Calculates means for each column

Library/Package	Piece of code	Example of usage	What it does
dplyr	<code>summarize()</code>	<code>df &lt;- df %&gt;% summarize(mean_x = mean(x))</code>	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. <code>summarize</code> and <code>summarise</code> are synonyms in this package. However, note that this function does not work in the same manner as the base R <code>summary</code> function.
dplyr	<code>across()</code>	<code>df %&gt;% summarize(across( c('col_a', 'col_b'), ~ sum(.x)))</code>	Use the across function with summarize to summarize across multiple columns of your data.
Base R	<code>unique()</code>	<code>unique(df)</code>	Returns a vector, data frame or array like x but with duplicate elements/rows removed.
Base R	<code>table()</code>	<code>table(x)</code>	Builds a contingency table of the counts at each combination of factor levels.
dplyr	<code>count()</code>	<code>df %&gt;% count(factor_name)</code>	Count the number of groups in a factor variable of a data frame or tibble
dplyr	<code>group_by()</code>	<code>df %&gt;% count(factor_name)</code>	Groups data into rows that contain the same specified value(s)
dplyr	<code>ungroup()</code>	<code>df %&gt;% count(factor_name)</code>	Undo a grouping that was done by <code>group_by()</code>
Base R	<code>plot()</code>	<code>plot(x, y)</code>	Creates a scatterplot of x and y vector data
Base R	<code>boxplot()</code>	<code>boxplot(x, y)</code>	Creates a boxplot of y against levels of x
Base R	<code>hist()</code>	<code>hist(x)</code>	Creates a histogram of x
Base R	<code>density()</code>	<code>plot(density(x))</code>	Creates a kernel density plot of x when used with <code>plot()</code>

## 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>factor(x)</code> or <code>as.factor(x)</code>	Factor	Coerces object <code>x</code> into a factor (which is used to represent categorical data). This function can be used to coerce object <code>x</code> 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.
Base R	<code>levels(x)</code>	<code>levels(factor_obj)</code>	Returns or sets the value of the levels in an object <code>x</code> .
Base R	<code>rep()</code>	<code>rep(1:3)</code>	Replicates the values in <code>x</code> to make a vector.
Base R	<code>seq()</code>	<code>seq(from = 0, to = 1, by = 0.2)</code>	Creates a vector of a sequence of numbers based on the specified arguments.

- `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.