INT_PROG (R) :: CHEATSHEET

```
> print(data)
  col1 col2     col3  col4
1    1    A Manzana  TRUE
2    2    B  Banana FALSE
3    3    C  Cereza  TRUE
4    4    A Damasco  TRUE
5    5    B     Uva FALSE
```

```
> str(data)
'data.frame':   5 obs. of  4 variables:
 $ col1: num  1 2 3 4 5
 $ col2: Factor w/ 3 levels "A","B","C": 1 2 3 1 2
 $ col3: chr  "Manzana" "Banana" "Cereza" "Damasco"
 $ col4: logi  TRUE FALSE TRUE TRUE FALSE
```

> summarise(data, nombre = max(col1))
Summarise data into single row of values

> count(data, col3)
Count number of rows with each unique value of variable (with or without weights)

> group_by(data, col2)
Group data into rows with the same value of Species

> ungroup(data)
Remove grouping information from data frame

> mutate(data, nombre = col1 * 2)
Compute and append/replace one or more columns

> filter(data, col4 == FALSE)
Extract rows that meet logical criteria

> distinct(data)
Remove duplicate rows

> sample_n(data, 10, replace = TRUE)
Randomly select n rows

> select(data, col1, col3)
Select columns by name or helper function

> separate(data, col3, c("coln1","coln2"), sep=...)
Separate one column into several

> unite(data, nombre, c(col1,col2, sep=...)
Unite several columns into one

> arrange(data, col1)
Order rows by values of a column

> rename(data, wcol = col1)
Rename the columns of a data frame

> rename(data, wcol = col1)
Rename the columns of a data frame

> pivot_longer(data, c(col1,col4))
"Lengthen" data by collapsing several columns into two. Column names move to a new "name" column and values to a new "value" column

> pivot_wider(data, "name", "value")
"Widen" data by expanding two columns into several. One column provides the new column names, the other the values

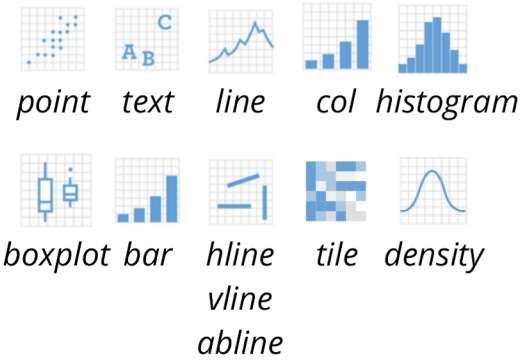
> replace_na(data, 0)
Specify a value to replace NA in selected columns

> top_n(data, 2, col1)
> slice_max(data, 2, n=col1)
Select and order top n entries

> ggplot(data, aes(x = col1, y = col2), ...)
Begins a plot that you finish by adding layers to. Add one geom function per layer

Aesthetics: x, y, color, fill, shape, group, linetype, size, label, ...

> ... + geom_X()



> ...+ facet_wrap(~ col2)
> ...+ facet_grid(~ col2)
Wrap facets into a rectangular layout

> ... + labs(x = "X", y = "Y", title ="Title", subtitle = "Subtitle", caption = "Caption", alt = "Alt", ...)
Label the elements of plot

> ... + scale_X_Y()
X = x, y, color, fill, linetype, shape, ...
Y = discrete, continuous, manual, gradient, ...
Set scales of plot

> ... + coord_flip()
Flip cartesian coordinates by switching x and y aesthetic mappings

> str_detect(string, pattern)
Detect the presence of a pattern match in a string

> str_which(string, pattern)
Find the indexes of strings that contain a pattern match

> str_count(string, pattern)
Count the number of matches in a string

> str_sub(string, start = 1, end = 1)
Extract substrings from a character vector

> str_extract(string, pattern)
Return the first NA pattern match found in each string, as a vector

> str_match(string, pattern)
Return the first pattern match found in each string, as a matrix with a column for each group in pattern

> str_length(string)
The width of strings

> str_to_lower(string)
> str_to_upper(string)
Convert strings to lower/upper case

> str_c(..., sep = "", collapse = "")
Join multiple strings into a single string

> str_split(string, pattern)
Split a vector of strings into a list of substrings