String Manipulation: Takeaways ⤴

Syntax

FINDING NUMBERS IN A STRING

- Extract numbers in a string:

```
library(readr)
`Total Grads - % of cohort` = parse_number(`Total Grads - % of cohort`)
```

- Extract numbers from multiple variables at once:

```
library(readr)
data_frame %>%
mutate(across(`first column in range`:`last column in range`, parse_number))
```

SPLITING STRING INTO SEVERAL COLUMNS

```
library(tidyr)
dataframe %>%
separate(col = the_name_of_the_column_to_separate,
         into = c("col name 1", "col name 2", "col name 3"), # the names of new columns
         sep = "\n") # the separator of strings
```

SUBSETTING STRINGS

- Subset a character string from left to right:

```
library(stringr)
Vector_2  <- Vector_1 %>%
str_sub(5, 7)
```
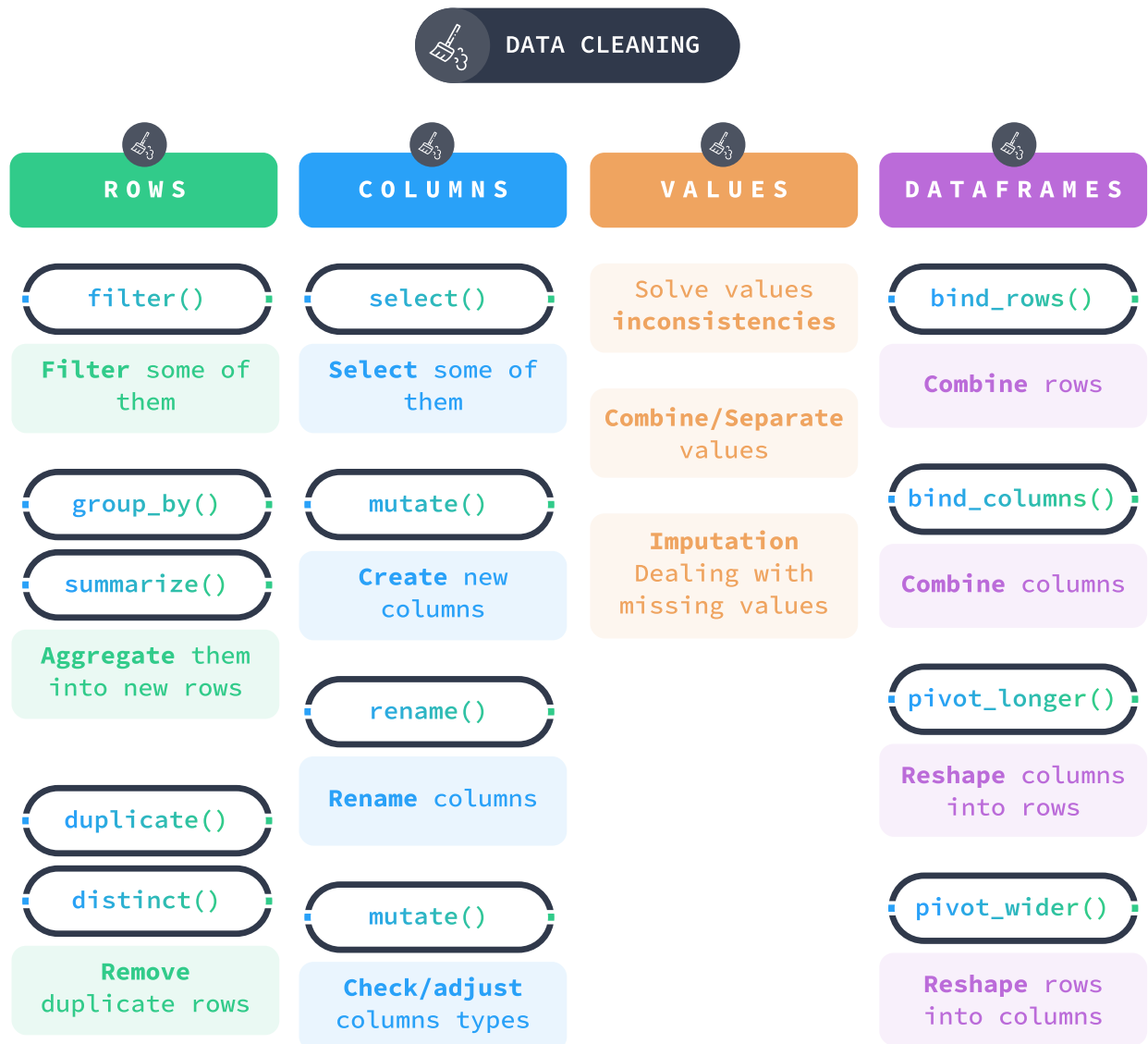
- Subset a character string from right to left:

```
library(stringr)
Vector_2  <- Vector_1 %>%
str_sub(-4, -6)
```

Concepts

- Data is optimally organized for use with tidyverse tools when it is "tidy":
  - Variables in columns
  - Observations in rows

• Values in cells

DATA CLEANING

ROWS	COLUMNS	VALUES	DATAFRAMES

ROWS

filter()

Filter some of them

group_by()

summarize()

Aggregate them into new rows

duplicate()

distinct()

Remove duplicate rows

COLUMNS

select()

Select some of them

mutate()

Create new columns

rename()

Rename columns

mutate()

Check/adjust columns types

VALUES

Solve values **inconsistencies**

Combine/Separate values

Imputation Dealing with missing values

DATAFRAMES

bind_rows()

Combine rows

bind_columns()

Combine columns

pivot_longer()

Reshape columns into rows

pivot_wider()

Reshape rows into columns

Resources

• [Documentation for parse_number](#)