

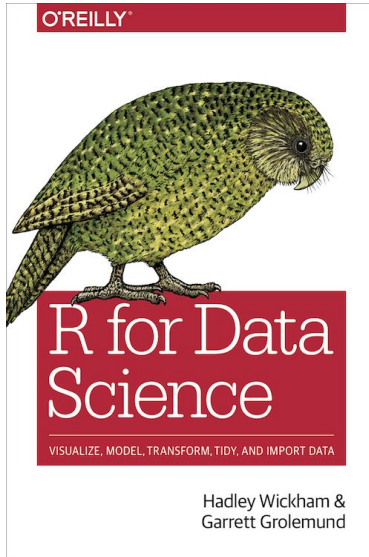


DATA TRANSFORMATION

R & dplyr

- Steps in Exploratory Data Analysis
- First Steps with R and RStudio
- Our Tool Set
- Data loading with `{readr}` / data management with `{tibble}`
- Data transformation with `{dplyr}`
 - Select columns
 - Filter rows
 - Sort rows
 - Add or change columns
 - Aggregate rows
- Exercise

RECOMMENDED LITERATURE



Wickham, Hadley, and Garrett G. Grolemund. R for Data Science: Import, Tidy, Transform, Visualize, and Model Data. 2nd edition, O'Reilly, 2023. Online verfügbar: <https://r4ds.hadley.nz/>

→ Chapter 4 in the online version

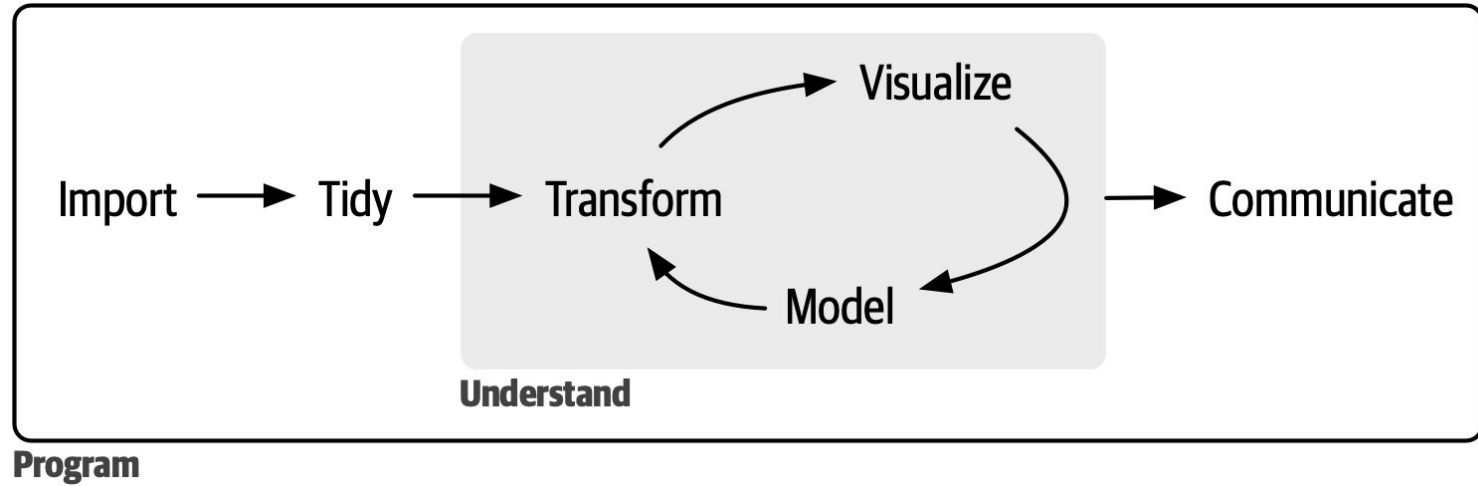


Sauer, Sebastian. Moderne Datenanalyse mit R. Springer Gabler, 2019.

→ Chapter 7

STEPS IN EXPLORATORY DATA ANALYSIS

STEPS IN EXPLORATORY DATA ANALYSIS



Source: Wickham, Hadley, and Garrett Grolemund. R for Data Science: Import, Tidy, Transform, Visualize, and Model Data. First edition, O'Reilly, 2016. URL: <https://r4ds.hadley.nz/diagrams/data-science/base.png>

FIRST STEPS WITH R & RStudio

FIRST STEPS WITH R AND RSTUDIO

DESKTOP OR CLOUD

Download, Installation R and RStudio

alternatively

Registration and Login RStudio Cloud



Walkthrough RStudio

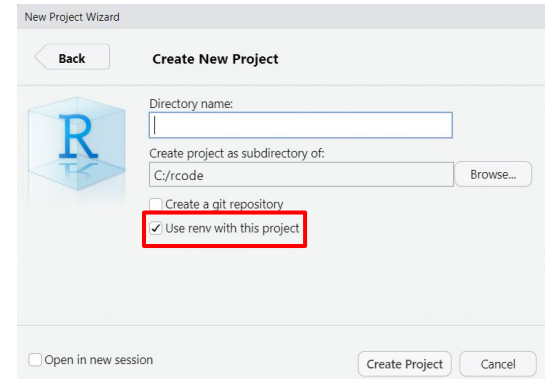
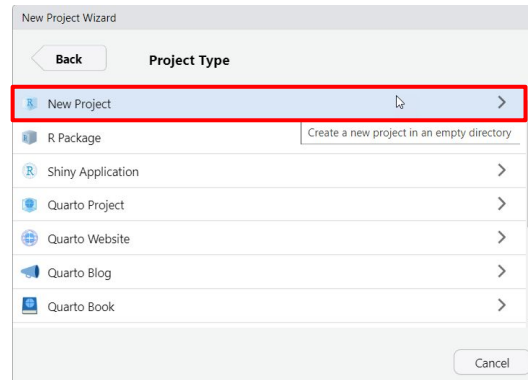
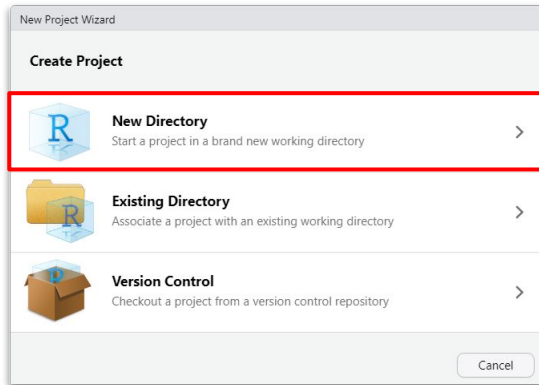
- Console and script editor
- Installing packages
- Projects
- Environment
- Previews
- Getting Help

FIRST STEPS WITH R AND RSTUDIO

CREATE A NEW PROJECT

All code examples for this course are hosted publicly on GitHub

- File → New Project → New Directory
- Choose a location on your computer and enter the name for the new directory
- Check “Use renv with this project”



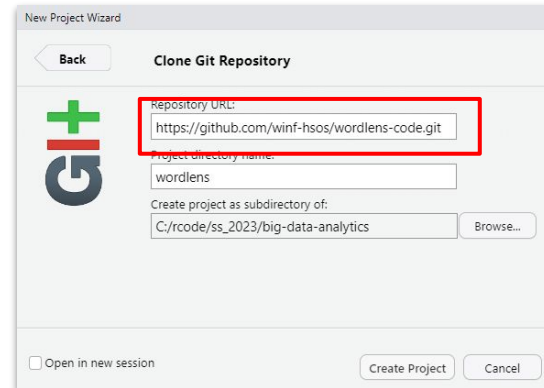
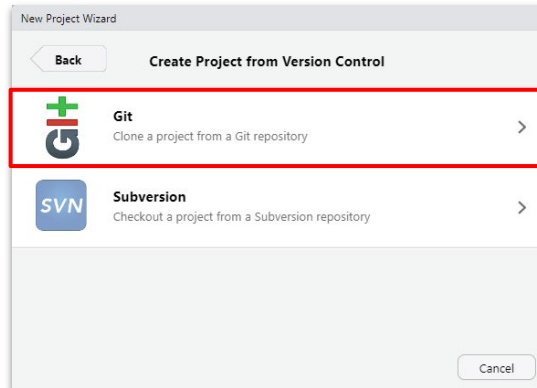
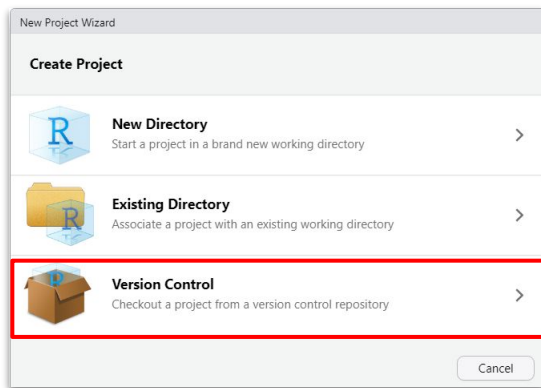
FIRST STEPS WITH R AND RSTUDIO

CHECKOUT GITHUB REPO

All code examples for this course are hosted publicly on GitHub

- File → New Project → Version Control → Git
- Paste the repository's URL and choose a location on your computer:

```
https://github.com/winf-hsos/<name_of_repo>.git
```



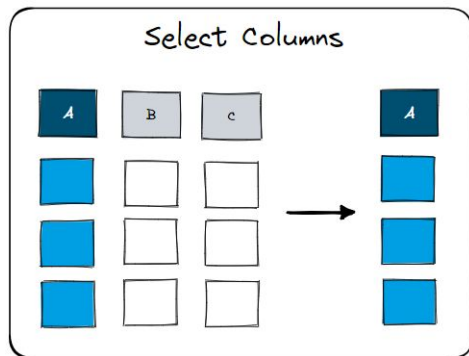
OUR TOOLSET

OUR TOOLSET

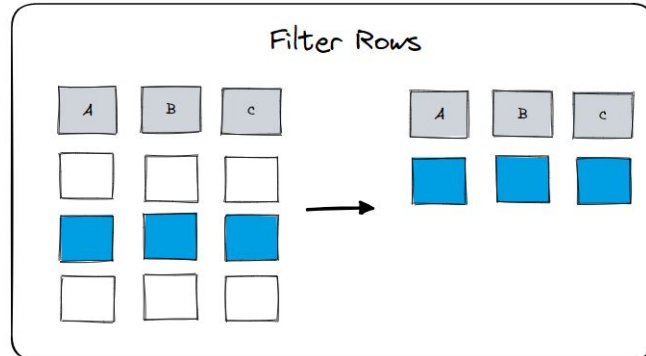
- Data loading, e.g., with `{readr}` or `{readxl}`
- Data management with `{tibble}`
- Data transformation with `{dplyr}`
 - `select()`
 - `filter()`
 - `arrange()`
 - `mutate()` / `transmute()`
 - `summarise()` / `group_by()`
- Data visualization with `{ggplot2}`
- Working Environment(s)
 - R & Python
 - RStudio
 - Databricks (*for Big Data*)



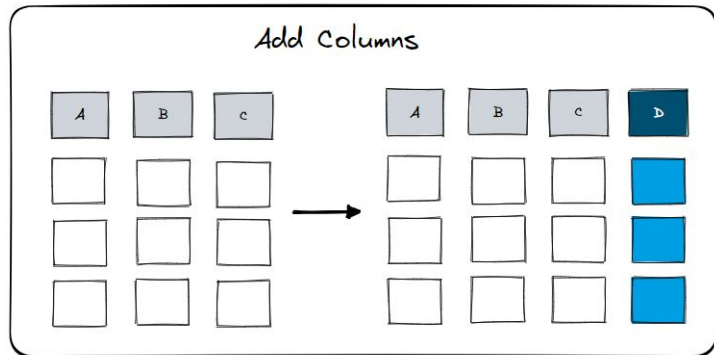
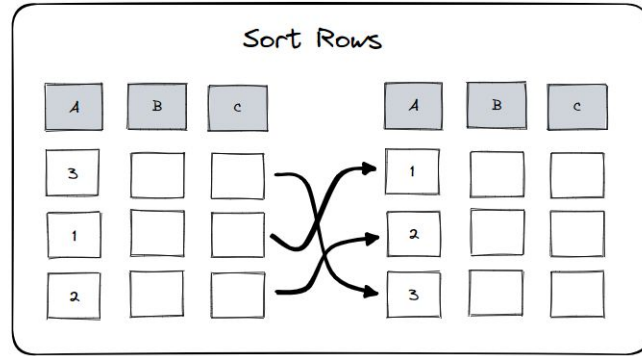
select



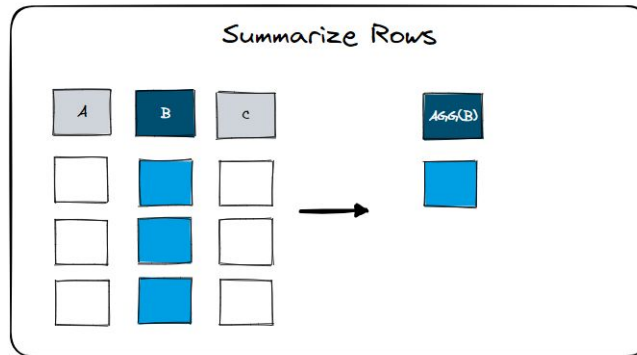
filter



arrange



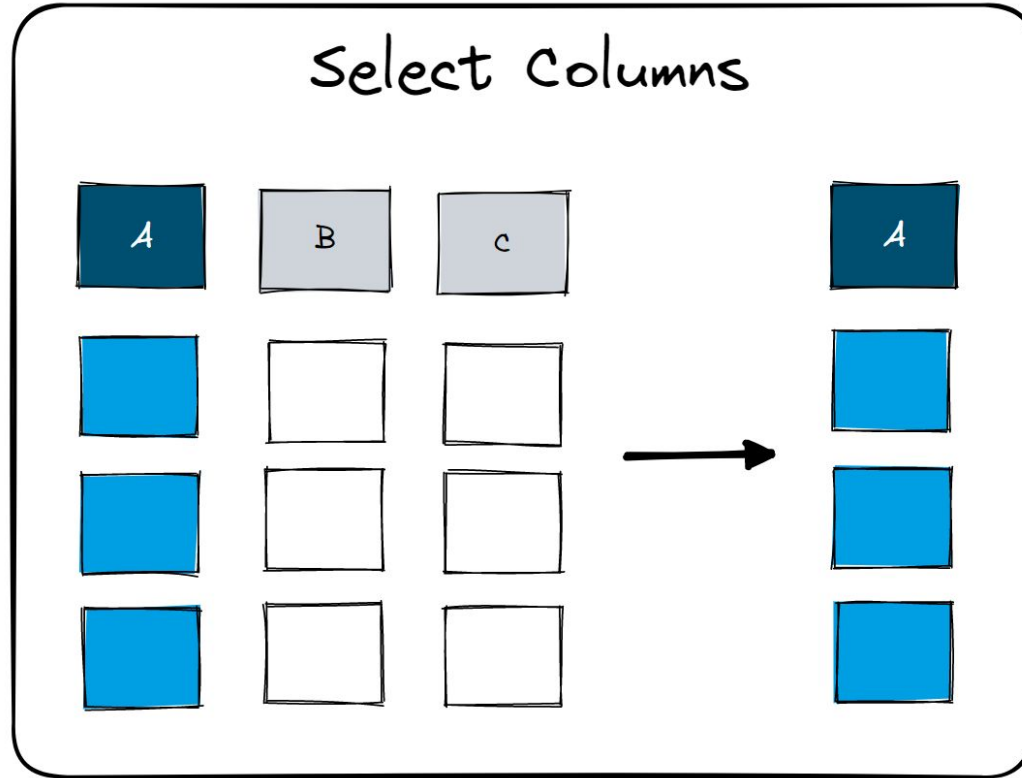
mutate
transmute



group_by
summarize

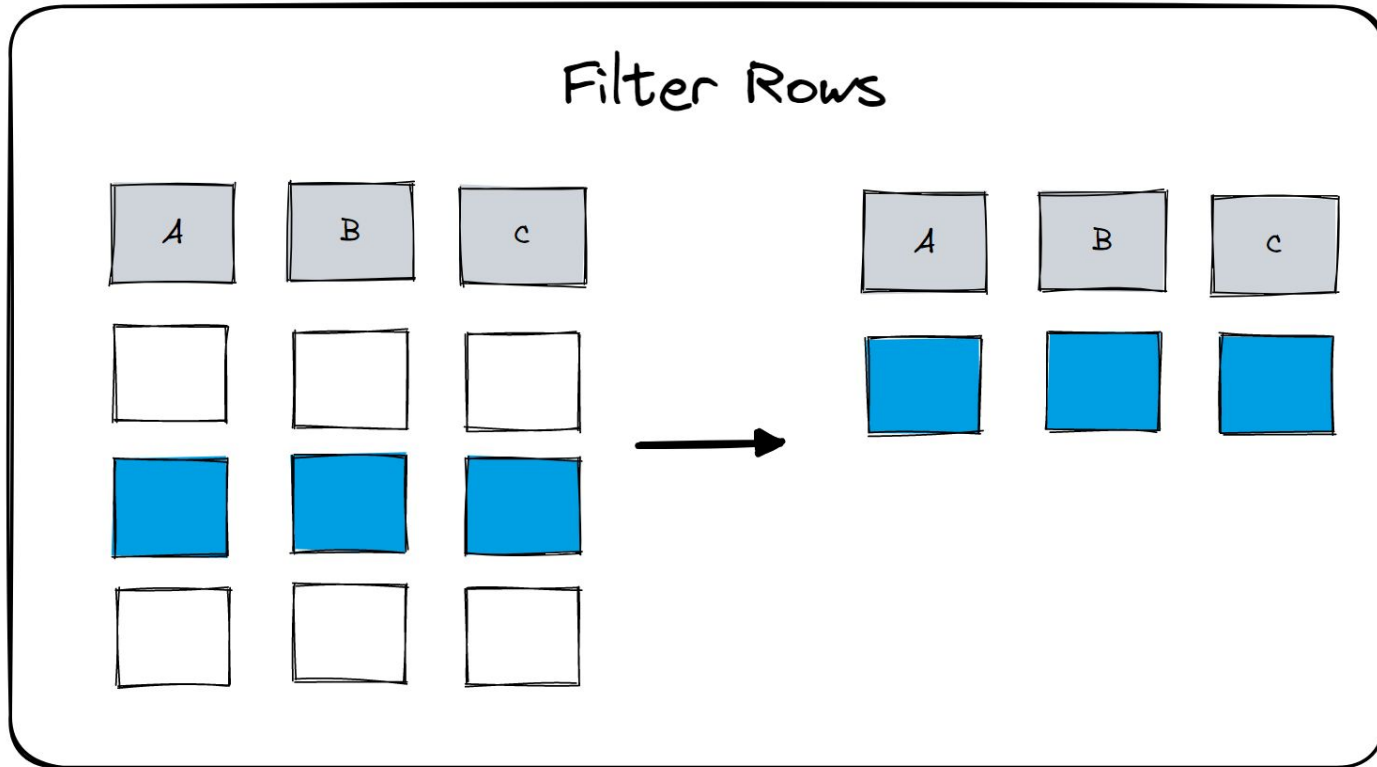
SELECT

REDUCING COLUMNS



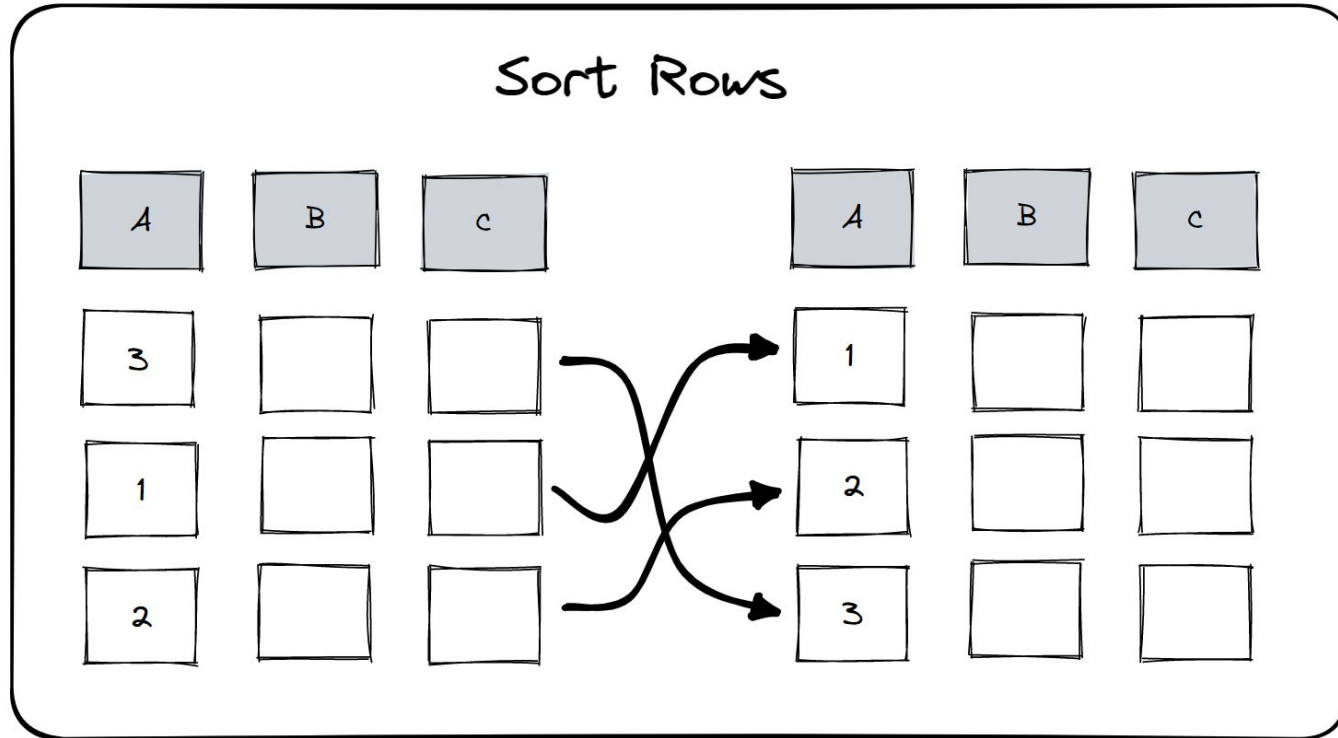
FILTER

REDUCING ROWS



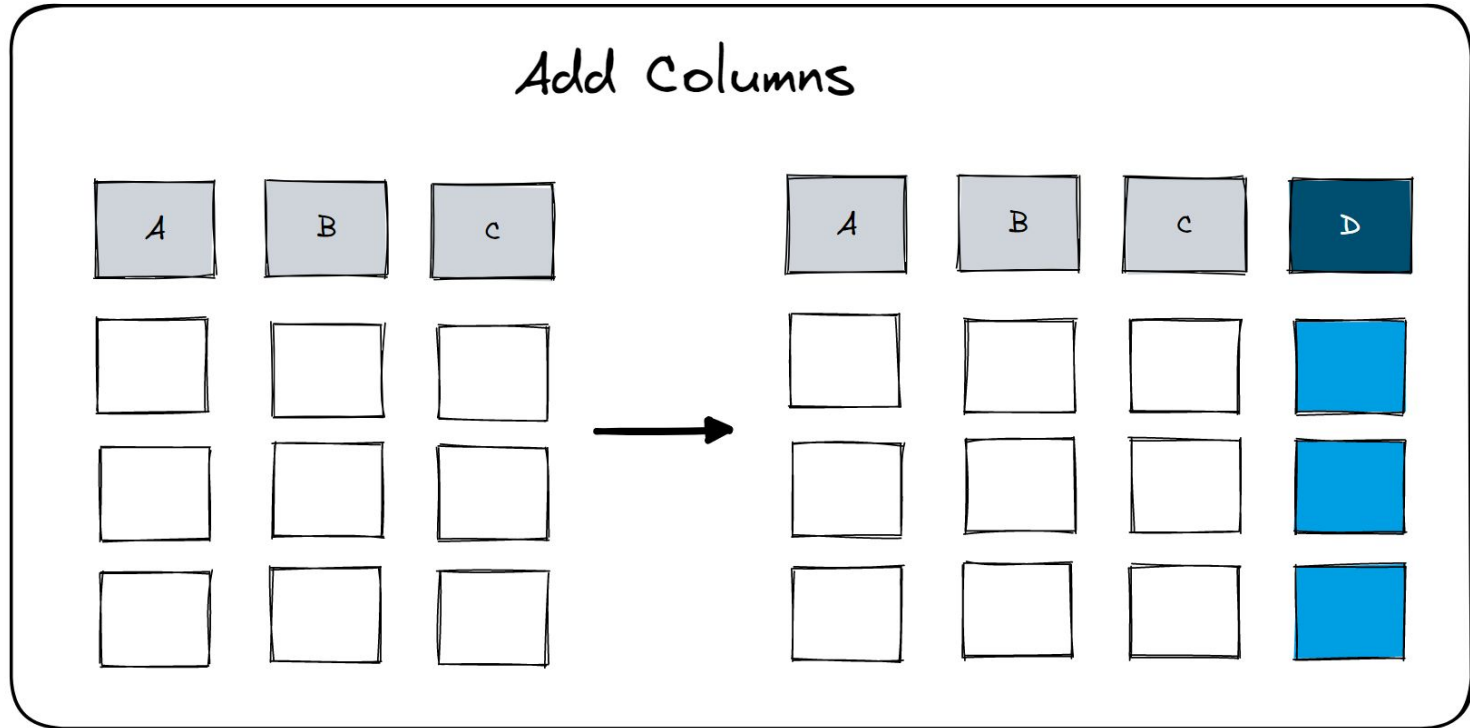
ARRANGE

SORTING ROWS



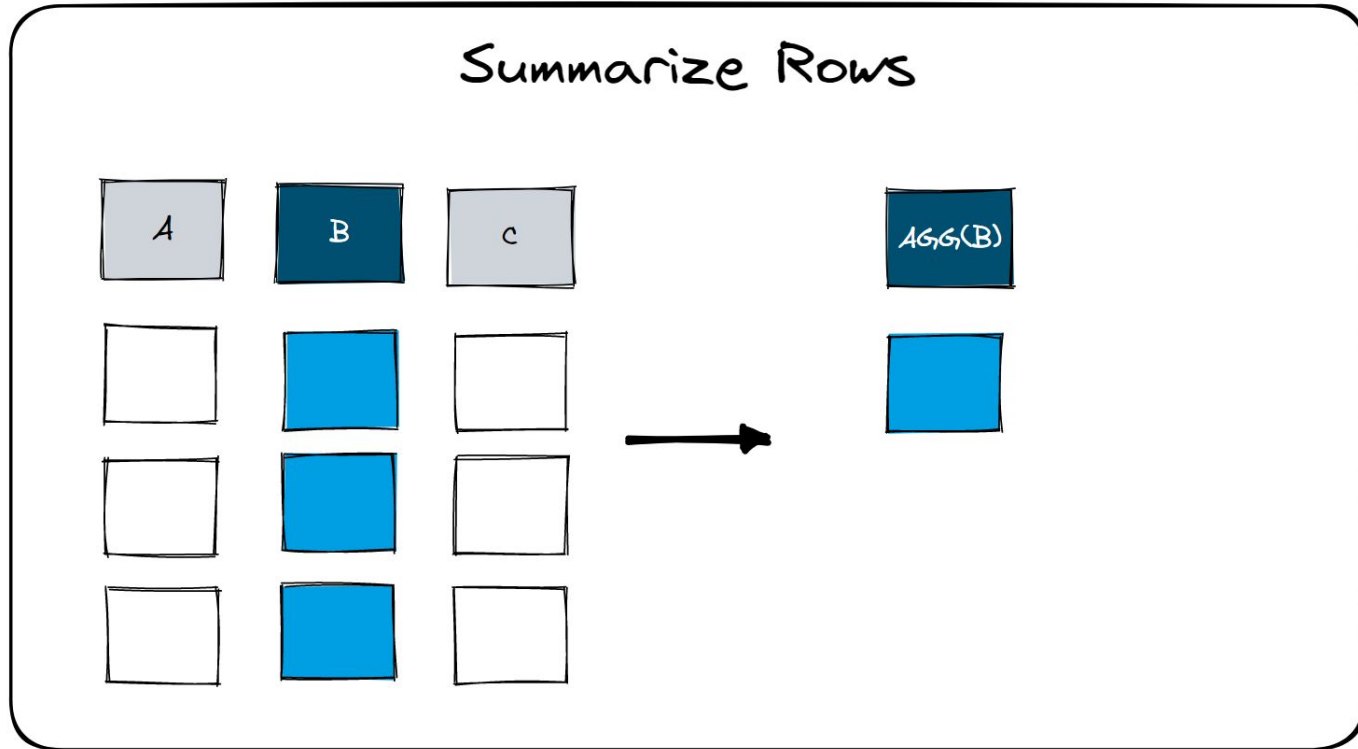
MUTATE

ADD NEW COLUMNS



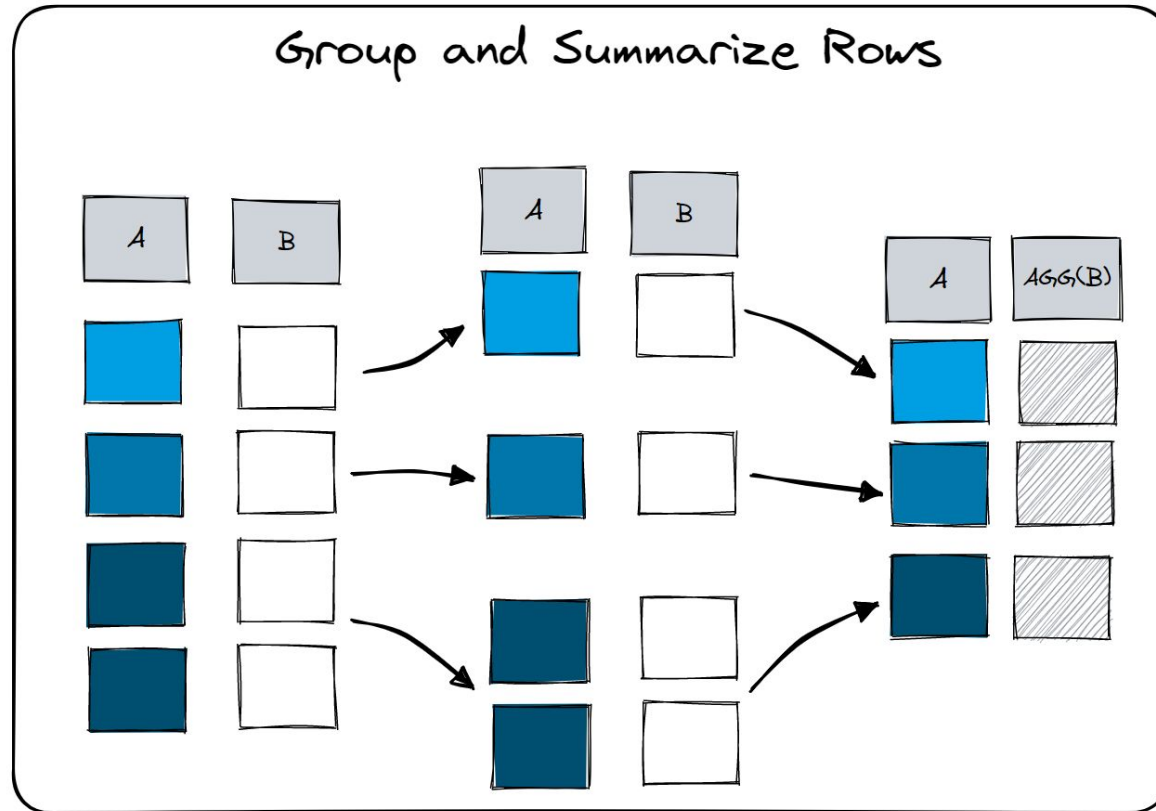
SUMMARIZE

SUMMARIZE ROWS



GROUP & SUMMARIZE

GROUP BY VARIABLE AND SUMMARIZE



DATA LOADING

{readr}

DATA LOADING

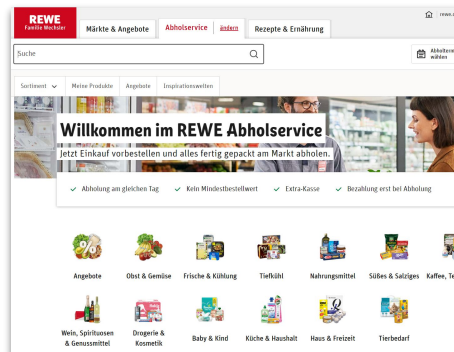
EXERCISE DATA

- Data loading with `{readr}` and `{readxl}` (Excel, CSV), `{jsonlite}` (JSON), or `readRDS` (R-format)
- `{janitor}` and `clean_names` for better column names
- Introductory data sets:

Campusbier Sales Orders (CSV)



REWE Online Products (CSV)



Politician's Tweets (JSON / RDS)



DATA MANAGEMENT

{tibble}

- Manage data with data frames and `{tibble}`
- Tibbles as **modern data frames**
 - Better printing
 - No string conversion into factors
 - No **rownames**
 - Original column names are kept when loading a tibble
 - Lazy processing

*Tibbles or data frames? Both are like
tables in a spreadsheet... just in R*

DATA TRANSFORMATION

`{dplyr}`

- Select specific columns with `{dplyr}`
 - `select`
 - By name
 - By name pattern (`starts_with`, `ends_with`, `contains`)
 - By position or index (`last_col`)
 - By set (`all_of`, `any_of`)
 - By data type (`where(is.numeric)`)
 - White vs. blacklist (!)

- Reduce rows with `{dplyr}`
 - `filter`
 - Simple filter conditions (`==`, `!=`, `<`, `>`)
 - Multiple conditions (`&`, `|`, `!`, `xor`)
 - Set operators (`%in%`)
 - Missing values (`NA`, `is.na`)
 - Simple text searches (`str_detect`)

- Sort results with `{dplyr}`
 - `arrange`
 - Ascending order by one or more columns
 - Descending order (`desc` or `-`)

DATA TRANSFORMATION

ADD OR CHANGE COLUMNS

- Add new calculated columns with `{dplyr}`
 - `mutate`
 - Add new calculated columns (+, -, /, *, %, ^, `paste0`)
 - Vectorized calculations (`mean`, `sum`, `max`, `min`, `lag`, `lead`)
 - Keep only used columns (`.keep = "used"`)
 - Determine position of new columns with `.before` and `.after`
 - `transmute`
 - Add new columns and remove all others (sometimes what we want)

- Summarize data with `{dplyr}`
 - `count`, `tally`, `distinct` for quick aggregations
 - `summarize`
 - Aggregate data using functions (`mean`, `median`, `quantile`, `sd`, `IQR`, `mad`, `sum`, `max`, `min`, `n`, `n_distinct`, `first`, `nth`, `last`)
 - `group_by`
 - Create groups by which to aggregate
 - The `janitor` package with `tabyl` for quick percentages and cross-tables

EXERCISE

You are new as a managing director in the Campusbier project and are supposed to get a first impression of the business. All you have are two datasets: `orders.csv` and `line_items.csv`.

- How do you approach this unknown dataset?
- With a partner, come up with at least 3 questions you want to ask the data! Look at the available columns for this!
- Create R commands to answer the questions (without visualization yet)!