

teaching_material_book

Stefan Thoma

```
r format(Sys.time(), '%d %B, %Y')
```

Table of contents

Preface	3
1 Introduction	4
1.1 Structure of teal apps	4
2 Data	6
2.1 Aim of this chapter	6
2.2 Setup	6
2.3 What data do we use	7
2.4 Getting started	9
2.4.1 Data	9
2.4.2 Pre-processing	11
2.5 Define the app	12
3 Summary	15
References	16

Preface

This is a collection of materials used for the introductory content of teal found on the [pharmaverse youtube channel](#).

1 + 1

[1] 2

1 Introduction

1.1 Structure of teal apps

teal is a shiny-based modular framework. It provides an interactive user interface allowing users to customize tables and visualizations facilitating the exploration and analysis of clinical trial data. Dynamic analysis and display control, data filtering and code reproducibility are included in each module.

teal modules are laid out in three panels.

- The **Left Panel**, referred to as the encoding panel, includes elements used to adjust the output interactively.
- The **Center Panel**, referred to as the output panel, where the tables and visualizations are displayed.
- The **Right Panel**, referred to as the data filter panel, includes elements used to subset the data interactively.

The layout is pre-set within each module. However, user can decide overall tab order and nesting. See **?@fig-layout** for the typical teal app structure.

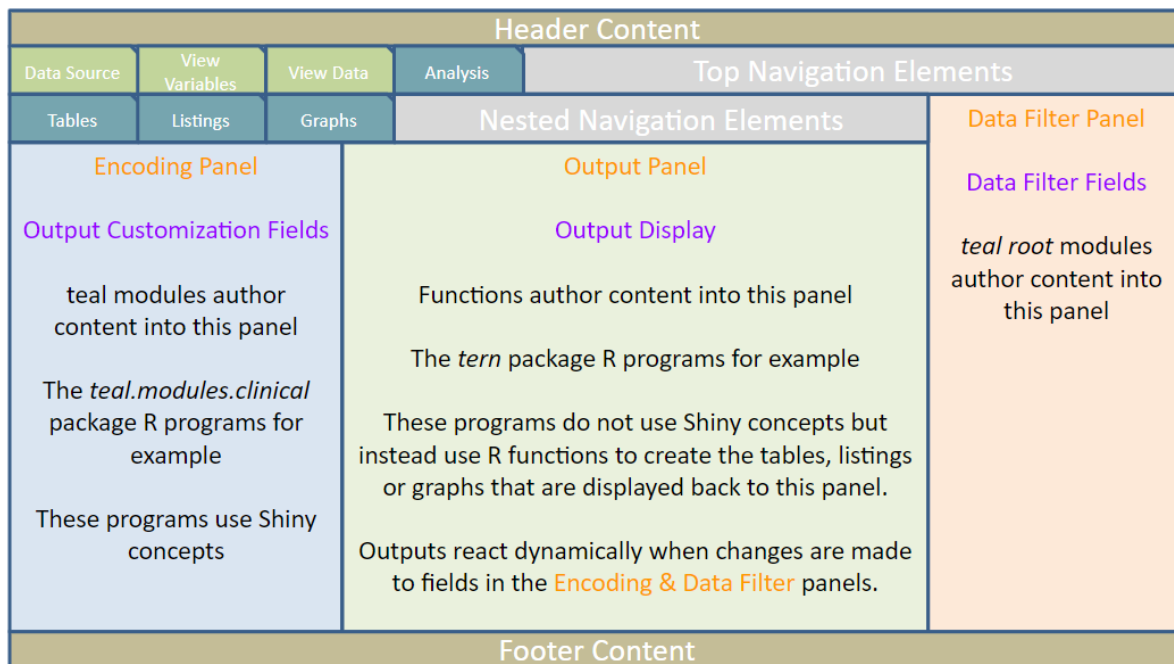


Figure 1.1: Typical teal app layout

2 Data

2.1 Aim of this chapter

2.2 Setup

In this tutorial we will be working with realistic looking but simulated data.

```
if (!require("remotes")) install.packages("remotes")

# install teal
remotes::install_github("insightsengineering/teal@*release")
# install teal.modules.general
remotes::install_github("insightsengineering/teal.modules.general@*release")

library(random.cdisc.data)
library(teal.data)
```

Loading required package: shiny

```
library(scda)
```

There are no scda.XXXX libraries installed, like scda.2021.
Please install an scda database to take full advantage of the scda package.

```
library(teal.modules.general)
```

Loading required package: ggmosaic

Loading required package: ggplot2

Loading required package: shinyTree

Loading required package: teal

Loading required package: teal.transform

Loading required package: magrittr

You are using teal version 0.12.0

```
library(sparkline)
```

```
library(teal)  
library(dplyr)
```

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

filter, lag

The following objects are masked from 'package:base':

intersect, setdiff, setequal, union

2.3 What data do we use

We first load two synthetic data sets. 'random.cdisc.data::cadsl' is a Subject Level Analysis Dataset (ADSL) with one record (row) per subject. The unique identifier per subject is stored in the variable USUBJID. According to [CDISC](#) the main purpose of ADSL is to provide a “(...) source for denominators for populations of interest, stratification variables, and other important subject subgroups”.

```
ADSL <- random.cdisc.data::cadsl
```

Data Structure for Adverse Event Analysis

```
ADAE <- random.cdisc.data::cadae
ADAE
```

```
# A tibble: 1,934 x 92
```

	STUDYID	USUBJID	SUBJID	SITEID	AGE	AGEU	SEX	RACE	ETHNIC	COUNTRY	DTHFL
	<chr>	<chr>	<chr>	<chr>	<int>	<fct>	<fct>	<fct>	<fct>	<fct>	<fct>
1	AB12345	AB12345-B~	id-134	BRA-1	47	YEARS	M	WHITE	NOT H~	BRA	Y
2	AB12345	AB12345-B~	id-134	BRA-1	47	YEARS	M	WHITE	NOT H~	BRA	Y
3	AB12345	AB12345-B~	id-134	BRA-1	47	YEARS	M	WHITE	NOT H~	BRA	Y
4	AB12345	AB12345-B~	id-134	BRA-1	47	YEARS	M	WHITE	NOT H~	BRA	Y
5	AB12345	AB12345-B~	id-141	BRA-1	35	YEARS	F	WHITE	NOT H~	BRA	N
6	AB12345	AB12345-B~	id-141	BRA-1	35	YEARS	F	WHITE	NOT H~	BRA	N
7	AB12345	AB12345-B~	id-141	BRA-1	35	YEARS	F	WHITE	NOT H~	BRA	N
8	AB12345	AB12345-B~	id-141	BRA-1	35	YEARS	F	WHITE	NOT H~	BRA	N
9	AB12345	AB12345-B~	id-141	BRA-1	35	YEARS	F	WHITE	NOT H~	BRA	N
10	AB12345	AB12345-B~	id-141	BRA-1	35	YEARS	F	WHITE	NOT H~	BRA	N

```
# ... with 1,924 more rows, and 81 more variables: INVID <chr>, INVNAM <chr>,
#   ARM <fct>, ARMCD <fct>, ACTARM <fct>, ACTARMCD <fct>, TRT01P <fct>,
#   TRT01A <fct>, TRT02P <fct>, TRT02A <fct>, REGION1 <fct>, STRATA1 <fct>,
#   STRATA2 <fct>, BMRKR1 <dbl>, BMRKR2 <fct>, ITTFL <fct>, SAFFL <fct>,
#   BMEASIFL <fct>, BEP01FL <fct>, AEWITHFL <fct>, RANDDT <date>,
#   TRTSDTM <dtm>, TRTEDTM <dtm>, TRT01SDTM <dtm>, TRT01EDTM <dtm>,
#   TRT02SDTM <dtm>, TRT02EDTM <dtm>, AP01SDTM <dtm>, AP01EDTM <dtm>, ...
```

Analysis data-set for time to event.

```
ADTTE <- random.cdisc.data::cadtte
ADTTE
```

```
# A tibble: 2,000 x 68
```

	STUDYID	USUBJID	SUBJID	SITEID	AGE	AGEU	SEX	RACE	ETHNIC	COUNTRY	DTHFL
	<chr>	<chr>	<chr>	<chr>	<int>	<fct>	<fct>	<fct>	<fct>	<fct>	<fct>
1	AB12345	AB12345-B~	id-105	BRA-1	38	YEARS	M	BLAC~	HISPA~	BRA	N
2	AB12345	AB12345-B~	id-105	BRA-1	38	YEARS	M	BLAC~	HISPA~	BRA	N
3	AB12345	AB12345-B~	id-105	BRA-1	38	YEARS	M	BLAC~	HISPA~	BRA	N
4	AB12345	AB12345-B~	id-105	BRA-1	38	YEARS	M	BLAC~	HISPA~	BRA	N
5	AB12345	AB12345-B~	id-105	BRA-1	38	YEARS	M	BLAC~	HISPA~	BRA	N
6	AB12345	AB12345-B~	id-134	BRA-1	47	YEARS	M	WHITE	NOT H~	BRA	Y
7	AB12345	AB12345-B~	id-134	BRA-1	47	YEARS	M	WHITE	NOT H~	BRA	Y
8	AB12345	AB12345-B~	id-134	BRA-1	47	YEARS	M	WHITE	NOT H~	BRA	Y
9	AB12345	AB12345-B~	id-134	BRA-1	47	YEARS	M	WHITE	NOT H~	BRA	Y


```

10 AB12345 AB12345-B~ id-134 BRA-1      47 YEARS M      WHITE NOT H~ BRA      Y
# ... with 1,990 more rows, and 57 more variables: INVID <chr>, INVNAM <chr>,
#   ARM <fct>, ARMCD <fct>, ACTARM <fct>, ACTARMCD <fct>, TRT01P <fct>,
#   TRT01A <fct>, TRT02P <fct>, TRT02A <fct>, REGION1 <fct>, STRATA1 <fct>,
#   STRATA2 <fct>, BMRKR1 <dbl>, BMRKR2 <fct>, ITTFL <fct>, SAFFL <fct>,
#   BMEASIFL <fct>, BEP01FL <fct>, AEWITHFL <fct>, RANDDT <date>,
#   TRTSDTM <dtm>, TRTEDTM <dtm>, TRT01SDTM <dtm>, TRT01EDTM <dtm>,
#   TRT02SDTM <dtm>, TRT02EDTM <dtm>, AP01SDTM <dtm>, AP01EDTM <dtm>, ...

```

2.4 Getting started

The most crucial function of the `teal` package(s) is the `teal::init()` function, which is structured as follows:

```

init(
  data,
  modules,
  title = NULL,
  filter = list(),
  header = tags$p("Add Title Here"),
  footer = tags$p("Add Footer Here"),
  id = character(0)
)

```

The `init` function sets up a shiny app that consists of teal modules. Let's go through the arguments:

2.4.1 Data

In the `data` argument we can define one or more dataframes for the application. If more than one dataframes are specified, they should be combined as a list, e.g. `data = list(ADSL, ADTR)`.

If you work with ADaM datasets, the package `teal.data` provides helper functions with which the `data` argument of `init()` can be specified, — the `cdisc_data()` and the `cdisc_dataset()` functions. They allow the teal app to know the merge key variables of the datasets. The `cdisc_data` function returns an S6 object.



2.4.2 Pre-processing

Generally, it is recommended to collect all data pre-processing steps in a separate R-script, which can then directly be applied to the `'cdisc_data()'` output.

Let's see an example of this:

||| we write a proper rscript file that does some data processing, cool would be something useful.

```
file_example <- tempfile(fileext = ".R")
writeLines(
  text = c(
    "ADSL <- ADSL %>%
      dplyr::mutate(newdata = 1)"
  ),
  con = file_example
)

cdisc_ADSSL <-
  cdisc_dataset(dataname = "ADSSL", x = ADSL)

our_cdisc_data_mutated <- cdisc_ADSSL %>%
  mutate_dataset(script = file_example)
get_raw_data(our_cdisc_data_mutated)$newdata
```

[illegible]

```

ADSL_dataset <- dataset(
  dataname = "ADSL",
  x = ADSL,
  label = "AdAM subject-level dataset"
)
ADSL_mutated <- ADSL_dataset %>%
  mutate_dataset(code = "ADSL$new_variable <- 1")

ADSL_mutated$get_raw_data()$new_variable[1]

```

[1] 1

```

file_example <- tempfile(fileext = ".R")
writeLines(
  text = c(
    "ADSL <- ADSL %>%
      dplyr::mutate(new_variable = new_variable * 2)"
  ),
  con = file_example
)

ADSL_mutated <- ADSL_mutated %>%
  mutate_dataset(script = file_example)

ADSL_mutated$get_raw_data()$new_variable[1]

```

[1] 2

2.5 Define the app

```

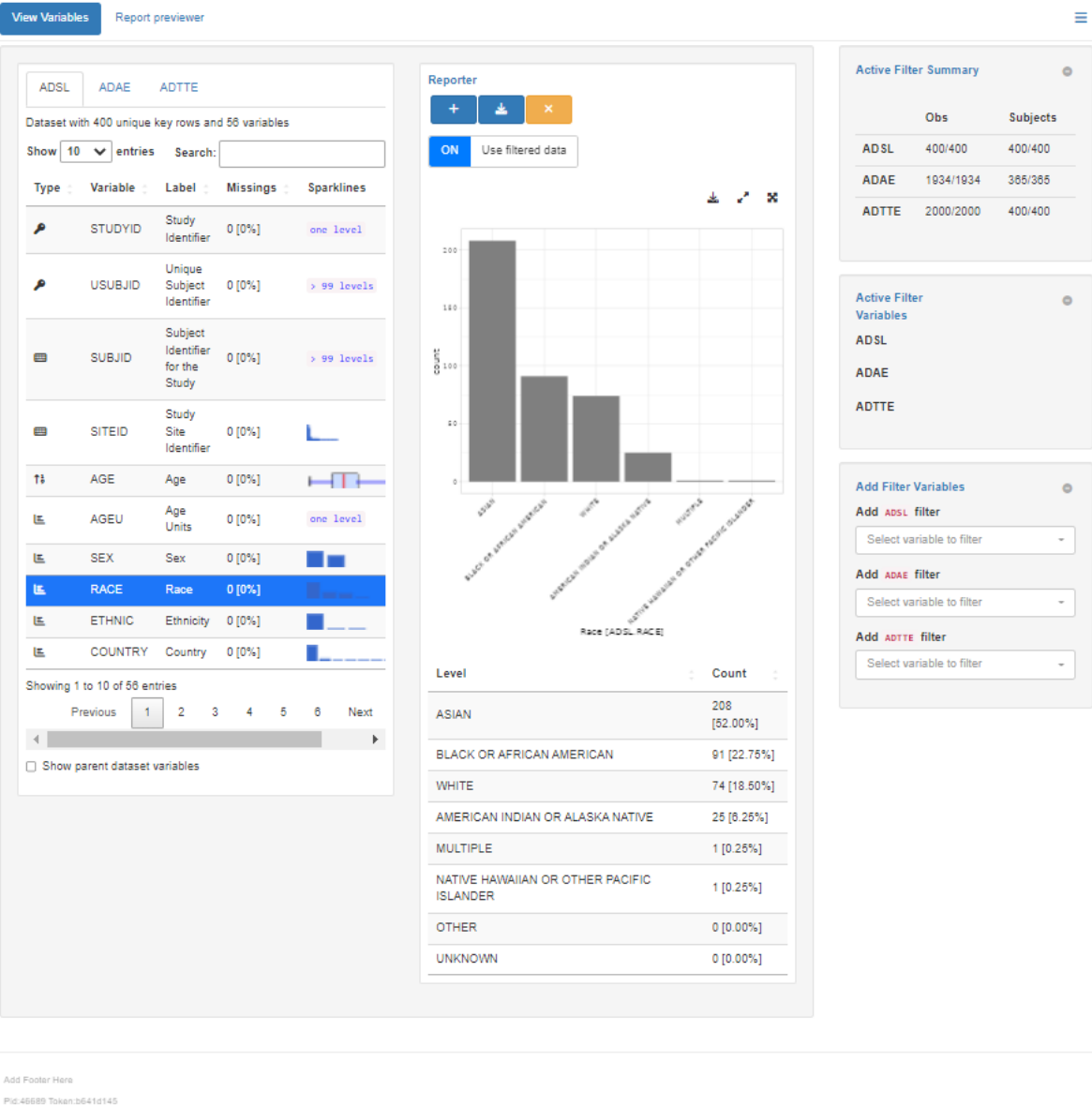
app <- teal::init(data = cdisc_data(
  cdisc_dataset("ADSL", ADSL),
  cdisc_dataset("ADAE", ADAE),
  cdisc_dataset("ADTTE", ADTTE)
  # <<additional dataset code>>
),
modules = modules(tm_variable_browser(label = "View Variables")),
header = "My first application")

```

```
shinyApp(app$ui, app$server)
```

The output should look something like this:

My first application



Add Footer Here

Plot:46689 Token:3641d145

Figure 2.1: Data App preview

3 Summary

In summary, this book has no content whatsoever.

$1 + 1$

[1] 2

References