R Code – Best practices

Ken Mwai

1 – Naming conventions

- R has no standardised naming conventions
- Always choose a naming convention to work with; for example
  - all lowercase: e.g. adjustcolor
  - underscore separated: e.g. numeric_version
  - lowerCamelCase: e.g. addTaskCallback
  - UpperCamelCase: e.g. SignatureMethod
- **Avoid SPACES** while naming files

“There are only two hard things in Computer Science: cache invalidation and naming things.” — Phil Karlton

- Strive for names that are concise and meaningful
- R file names should be meaningful and end in .R.

Object names

- Variable and function names should be lowercase.

```
# Good
day_one
day_1

# Bad
first_day_of_the_month
DayOne
dayone
djm1
```

- avoid using names of existing functions and variables.

2 – Files organisation

- File organisation makes code and data analysis project readable
- Data should be seperated from codes
- Documents should be seperated from codes
- Use project facility of RStudio each time you start working on a new project

3 - organise the code within each file

- Start each file with a comment saying who wrote it and when, what it contains, and how it fits into the larger program
- Load all required packages
- Source required data files if any

```
#-------------------------------------------
## I-Star Introduction
## Ken Mwai - May 2021
#-------------------------------------------
#-------------------------------------------
# 0 - Load librairies
#-------------------------------------------
library(dplyr)
library(ggplot2)
#-------------------------------------------
# 1 - Source Data
#-------------------------------------------
df1 <- read_csv("data/my_data.csv")
#-------------------------------------------
# 2 - Start my code
#-------------------------------------------
mean(mtcars$mpg)
```

3 – Syntax

- Place spaces around all infix operators `(=, +, -, <-, etc.)`.
- Use `<-`, not `=`, for object assignment in R.
- Use comments to mark off sections of code.
- Comment your code with care. Comments should explain the why, not the what
- Each line of a comment should begin with the comment symbol and a single space
- Keep your lines less than 80 characters.

```
# This is a comment
# Good
# Object assignment in R
x <- 10

#Bad
x=10
```

```r
# Good
average <- mean(feet / 12 + inches, na.rm = TRUE)

# Bad
average<-mean(feet/12+inches,na.rm=TRUE)
```

Use <-, not =, for assignment.

```
# Good
x <- 5
# Bad
x = 5
```

Task

Take a first look at the data. Useful functions are `dim()`, `head()`, `str()` and `summary()`.

Solution

```
dim(mtcars)
```

```
## [1] 32 11
```

```
head(mtcars)
```

```
##                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

```
str(mtcars)
```