R’s quirks

> sample(4:4, 5, replace = TRUE)
[1] 4 4 4 4 4
[1] 4 2 4 1 2
Raw Data Collected

Data is Processed

Clean Dataset

Exploratory Data Analysis

Models & Algorithms

Make Decisions

Communicate Visualize Report

Data Product
Who should read this book?

• Intermediate R programmers who want to dive deeper into R and learn new strategies for solving diverse problems.

• Programmers from other languages who are learning R and want to understand why R works the way it does.
Challenges

• strong motivation to understand the content
• a good memory to bear all the details in mind
• discipline to stay focused on the content
• patience to finish

Exercises

1. Fix each of the following common data frame subsetting errors:

   ```
   mtcars[mtcars$cyl == 4, ]
   mtcars[-1:4, ]
   mtcars[mtcars$cyl <= 5]
   mtcars[mtcars$cyl == 4 | 6, ]
   ```

2. Why does `x <- 1:5; x[NA]` yield five missing values? (Hint: why is it different from `x[NA_real_]`?)

3. What does `upper.tri()` return? How does subsetting a matrix with it work? Do we need any additional subsetting rules to describe its behaviour?

   ```
   x <- outer(1:5, 1:5, FUN = "*")
   x[upper.tri(x)]
   ```
• need to **write** code anyway
• solve **every** exercise just once
• possibility to **share** solutions
• **backup** for later
Bookdown to the rescue ;-)  

- provides a **good structure**  
- almost **no overhead**  
- **easy to publish** on [bookdown.org](http://bookdown.org) (private or public)
S3 objects
S3 objects are made up of atomic vectors, arrays, and lists, so you can always pull apart an S3 object using the techniques described above and the knowledge you gain from `str()`. 

S4 objects
There are also two additional subsetting operators that are needed for S4 objects: `@` (equivalent to `$`), and `slot()` (equivalent to `[[`). `@` is more restrictive than `$` in that it will return an error if the slot does not exist. These are described in more detail in the OO field guide.

Exercises
1. Fix each of the following common data frame subsetting errors:
   ```r
   mtcars[mtcars$cyl = 4, ]
   mtcars[1:4, ]
   mtcars[mtcars$cyl <= 5]
   mtcars[mtcars$cyl = 4 | 6, ]
   ```

2. Why does `x <- 1:5; x[NA]` yield five missing values? (Hint: why is it different from `x[NA_real,]?`)

3. What does `upper.tri()` return? How does subsetting a matrix with it work? Do we need any additional subsetting rules to describe its behaviour?
   ```r
   x <- outer(1:5, 1:5, FUN = "*")
   x[upper.tri(x)]
   ```


5. Implement your own function that extracts the diagonal entries from a matrix.

2.1 Data types
1. Q: Fix each of the following common data frame subsetting errors:
   ```r
   mtcars[mtcars$cyl = 4, ]
   mtcars[-1:4, ]
   mtcars[mtcars$cyl <= 5]
   mtcars[mtcars$cyl = 4 | 6, ]
   ```

2. Q: Why does `x <- 1:5; x[NA]` yield five missing values? (Hint: why is it different from `x[NA_real,]?`)
   A: NA is of class logical, so `x[NA]` becomes recycled to `x[NA, NA, NA, NA, NA]`. Since subsetting an atomic with `NA` leads to an `NA`, you will get 5 of them returned. (Note that the recycling won’t happen, if you subset with `NA_real`, `NA_integer`, `NA_character` or `NA_complex`. In fact the latter gives an error).

3. Q: What does `upper.tri()` return? How does subsetting a matrix with it work? Do we need any additional subsetting rules to describe its behaviour?
Progress (09/16 – 04/17)

Solved exercises (70%)

1. Data structures
2. Subsetting
3. Functions
4. OO field guide
5. Environments
6. Exceptions and debugging
7. Functional programming
8. Functionals
9. Function operators
10. Non standard evaluation
11. Expressions
12. Domain specific languages
13. Performance
14. Profiling
15. Memory
16. Rcpp
Advanced R Solutions - Bookdown
https://bookdown.org/Tazinho/Advanced-R-Solutions/
01.05.2017 - This book aims to contribute solutions to Hadley Wickham's book Advanced R. It is planned to finish until July 2017. The code can be found on ...

GitHub - peterhurford/adv-r-book-solutions: Solutions for the Advanced ...
Community

solutions advanced r

Advanced R Solutions - Bookdown
https://bookdown.org/Tazinho/Advanced-R-Solutions/ - Diese Seite übersetzen
01.05.2017 - This book aims to contribute solutions to Hadley Wickham’s book Advanced R. It is planned to finish until July 2017. The code can be found on ...

GitHub - peterhurford/adv-r-book-solutions: Solutions for the Advanced ... 
4.2 S4

1. Q: Which S4 generic has the most methods defined for it? Which S4 class has the most methods associated with it?
   A:

2. Q: What happens if you define a new S4 class that doesn’t “contain” an existing class? (Hint: read about virtual classes in ?Classes.)
   A:

3. Q: What happens if you pass an S4 object to an S3 generic? What happens if you pass an S3 object to an S4 generic? (Hint: read ?setOldClass for the second case.)
   A:

4.3 RC
Next steps

Finish the book:
• merge Peter's and Robert's solutions
• solve more exercises
• review & spellcheck

Update with new “Advanced R” version:
http://adv-r.hadley.nz/

Smaller enhancements:
• toggle option to show/hide answers
• PDF version
• tests
Thank you!

https://bookdown.org/Tazinho/Advanced-R-Solutions/

✉️ malte.grosser@gmail.com