



## Outtro

LearningSpoonsR

2018-07-19

## Wrap-up

# Main

## 1 Week 1 (Introduction to R)

- M11-intro
- M12-base
- M13-quiz

## 2 Week 2 (Carseats)

- M21-dplyr\_ggplot\_rmd
- M22-Carseats (dep: M21)

## 3 Week 3 (WordCloud & FD & Shiny)

- M31-ggplot\_wordcloud
- M32-flexdashboard (dep: M31)
- M33-shiny (dep: M31-M32)

## 4 Week 4 (API & map & CaseStudy)

- M41-kma&automation
- M42-dart (dep: M41)
- M43-apis (dep: M41-M42)
- M44-subway
- M45-retail
- M46-retail-shiny (dep: M45)

## 5 Week 5 (Other Topics)

- M51-tidyr
- M52-timeSeries
- M53-kospi200 (dep: M52)
- M54-longevity
- M55-python
- M56-outtro

# References

## 1 Ref-rmdTemplates

- M81-pdf
- M82-beamer

## 2 Ref-internal

- M91-ggplot2-50
- M92-Happy Birthday  
(google translate API)
- M93-R and Python merging

## 3 Ref-external (Hadley)

- A Layered Grammar of Graphics
- ggmap Spatial Visualization with ggplot2
- The Split-Apply-Combine Strategy for Data

## 4 Ref-external (R Books)

- Advanced R
- Grammar of Graphics
- ISLR Seventh Printing
- R 공식 메뉴얼 번역  
(introduction to R)

## Future You

# Future You

- ❶ 프로그램의 목적을 정하고 구현
- ❷ 질문하고, 대답하고, 공유하고, 가르치고...
- ❸ 방어적인 프로그래밍 (Defensive Programming)
- ❹ 벡터 단위의 프로그래밍 (Vectorized Programming)
- ❺ 기타

## 1. 프로그램의 목적을 정하고 구현

Find what should it do first, then ask how!

- 돌아가고, 느리고, 무식하더라도 **한 번** 가보는 것이 중요.
- 프로그램 경험으로 기하적으로 빠른 프로그래머가 되어감.

Try 1-2-3 and generalize it.

- “1”에 많은 시간을 쓰는 것을 두려워 말아야
- 레고를 만들듯이, 큐브를 맞추듯이...
- 공통점과 차이점을 발견하여 함수, 클래스, 패키지를 작성
- 새로운 도구를 익혀나감 (Python, Graphics, 통계...)

## 2. 질문하고, 대답하고, 공유하고, 가르치고...

### 질문하고 대답할 곳

- Rubber duck (Today-you)
- 강사 이메일
- Classmates (facebook group)
- 친구, 동료 들...

### Minimal example of your problem

- 내 질문을 더 효과적으로 할 수 있는 방법은 무엇인가?
- 검색? 질문하는 글의 작성? Youtube?

### 일 잘하는 사람

- 새로운 문제를 찾아낼 수 있는 사람
- 필요한 지식과 자원을 찾아낼 수 있는 사람
- 문제와 결론을 공유할 수 있는 사람



### 3. 방어적인 프로그래밍 (Defensive Programming)

#### Styling

- 꼼꼼하고 프로페셔널하게...
- Font, Spacing, 일관성, 문서화

#### Naming

- Function = 동사 = Do something
- Variables = 명사 = Exists

#### Commenting

- 왜 code가 존재하는가
- 이 code는 무엇을 하는가
- Future-you에게 message를 보냄

## 4. 벡터 단위의 프로그래밍 (Vectorized Programming)

### 사고와 코드의 단위를 크게

- Element가 아닌 Variable로
- Variable이 아닌 data.frame으로
- data.frame이 아닌 database로

## 5. 기 타

- fb그룹 - 댓글/Like/답답/질문/의견 독려
- 일하시다가 디버깅이나 아이디어도 질문 독려
- (재)수강 문의 등 전병관 매니저에게 문의

# Blank

“A person who never made a mistake never tried anything new” - A. Einstein

“It is often necessary to do it in the wrong way first to learn the right way.” - Hadley Wickham (from **R Packages**)

```
"hello"
```

```
## [1] "hello"
```