R4DS(2e) 3. Data transformation - v0.3

한상곤(sangkon@pusan.ac.kr / sigmadream@gmail.com)

2023.09.13, updated: 2023.12.05

`flights`는 `tibble`로 구성되어 있고, `tidyverse`가 몇 가지 일반적인 문제를 회피하기 위해 사용하는 특수한 유형의 데이터 프레임입니다. tibble과 `data.frame`의 가장 중요한 차이점은 인쇄 방식입니다. tibble 큰 데이터 집합을 위해 설계되었기 때문에 처음 몇 개의 행과 한 화면에 맞는 열만 표시합니다. 모든 것을 볼 수 있는 몇 가지 옵션이 있습니다.

```
flights
```

```
## # A tibble: 336,776 x 19
##     year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
##    <int> <int> <int>    <int>          <int>     <dbl>    <int>          <int>
##  1  2013     1     1      517            515         2      830            819
...
```

RStudio를 사용하는 경우 가장 편리한 옵션은 스크롤 및 필터링이 가능한 대화형 보기를 열어주는 `View(flights)`일 것입니다. 그렇지 않으면 `print(flights, width = Inf)`를 사용하여 모든 열을 표시하거나 `glimpse()`를 사용할 수 있습니다. 두 보기 모두에서 변수 이름 뒤에는 각 변수의 유형을 알려주는 약어가 있습니다: 는 정수의 약자, 은 실수의 약자, 은 문자(문자열)의 약자, 은 날짜-시간의 약자입니다. 이러한 변수가 중요한 이유는 열에서 수행할 수 있는 작업이 열의 '유형'에 따라 크게 달라지기 때문입니다.

```
glimpse(flights)
```

```
## Rows: 336,776
## Columns: 19
## $ year           <int> 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2~
## $ month          <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
...
```

이제 데이터 조작을 위해 dplyr 주요 함수를 배워보겠습니다. 각 함수의 기본적인 작동 방식에 대해서 간략하게 소개하겠습니다.

1. 첫 번째 인수는 항상 데이터 프레임입니다.
2. 후속 인수는 일반적으로 변수 이름(따옴표 없이)을 사용하여 작업할 열을 설명합니다.
3. 출력은 항상 새 데이터 프레임입니다.

```
flights |>
  filter(dest == "IAH") |>
  group_by(year, month, day) |>
  summarize(
    arr_delay = mean(arr_delay, na.rm = TRUE)
  )
```

```
## # A tibble: 365 x 4
## # Groups:   year, month [12]
##     year month   day arr_delay
##    <int> <int> <int>     <dbl>
...
```

dplyr의 함수는 대상에 따라 행, 열, 그룹 또는 테이블의 네 가지 그룹으로 구성됩니다. 차례대로 살펴보겠습니다.