Use of Pipe oprator

Sadaf Zuhra

2024-09-17

Combining Multiple Operations with the Pipe

```
library(tidyverse)
library(nycflights13)
by_dest <- group_by(flights, dest)
delay <- summarize(by_dest,
 count = n(),
  dist = mean(distance, na.rm = TRUE),
 delay = mean(arr_delay, na.rm = TRUE))

delay <- filter(delay, count > 20, dest != "HNL")
```

There's another way to tackle the same problem with the pipe, %>%:

```
library(tidyverse)
delays <- flights %>%
 group_by(dest) %>%
 summarize(
 count = n(),
 dist = mean(distance, na.rm = TRUE),
 delay = mean(arr_delay, na.rm = TRUE)
 ) %>%
 filter(count >  20, dest != "HNL")
```

This focuses on the transformations, not what's being transformed, which makes the code easier to read. You can read it as a series of imperative statements: group, then summarize, then filter. As sug- gested by this reading, a good way to pronounce %>% when reading code is "then".

Missing Values

```
library(tidyverse)
flights %>%
 group_by(year, month, day) %>%
 summarize(mean = mean(dep_delay))
```

```
## # A tibble: 365 x 4
## # Groups:   year, month [12]
##     year month   day  mean
##    <int> <int> <int> <dbl>
##  1  2013     1     1    NA
##  2  2013     1     2    NA
##  3  2013     1     3    NA
##  4  2013     1     4    NA
##  5  2013     1     5    NA
##  6  2013     1     6    NA
##  7  2013     1     7    NA
##  8  2013     1     8    NA
##  9  2013     1     9    NA
## 10  2013     1    10    NA
## # i 355 more rows
```

```
flights %>%
 group_by(year, month, day) %>%
 summarize(mean = mean(dep_delay, na.rm = TRUE))
```

```
## # A tibble: 365 x 4
## # Groups:   year, month [12]
##     year month   day  mean
##    <int> <int> <int> <dbl>
##  1  2013     1     1 11.5
##  2  2013     1     2 13.9
##  3  2013     1     3 11.0
##  4  2013     1     4  8.95
##  5  2013     1     5  5.73
##  6  2013     1     6  7.15
##  7  2013     1     7  5.42
##  8  2013     1     8  2.55
##  9  2013     1     9  2.28
## 10  2013     1    10  2.84
## # i 355 more rows
```

```
library(tidyverse)
not_cancelled <- flights %>%
 filter(!is.na(dep_delay), !is.na(arr_delay))
```

```
library(tidyverse)
not_cancelled %>%
 group_by(year, month, day) %>%
 summarize(mean = mean(dep_delay))
```

```
## # A tibble: 365 x 4
## # Groups:   year, month [12]
##     year month   day  mean
##    <int> <int> <int> <dbl>
##  1  2013     1     1 11.4
##  2  2013     1     2 13.7
##  3  2013     1     3 10.9
```

```
##  4  2013     1     4  8.97
##  5  2013     1     5  5.73
##  6  2013     1     6  7.15
##  7  2013     1     7  5.42
##  8  2013     1     8  2.56
##  9  2013     1     9  2.30
## 10  2013     1    10  2.84
## # i 355 more rows
```