dplyr

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(hflights)
```

tbls

as_tibble()

Convert data.frame to tibble.

```
hflights <- as_tibble(hflights)

hflights
```

```
## # A tibble: 227,496 x 21
##     Year Month DayofMonth DayOfWeek DepTime ArrTime UniqueCarrier FlightNum
##  * <int> <int>      <int>     <int>   <int>   <int> <chr>             <int>
##  1  2011     1          1         6    1400    1500 AA                  428
##  2  2011     1          2         7    1401    1501 AA                  428
##  3  2011     1          3         1    1352    1502 AA                  428
##  4  2011     1          4         2    1403    1513 AA                  428
##  5  2011     1          5         3    1405    1507 AA                  428
##  6  2011     1          6         4    1359    1503 AA                  428
##  7  2011     1          7         5    1359    1509 AA                  428
##  8  2011     1          8         6    1355    1454 AA                  428
##  9  2011     1          9         7    1443    1554 AA                  428
## 10  2011     1         10         1    1443    1553 AA                  428
## # ... with 227,486 more rows, and 13 more variables: TailNum <chr>,
## #   ActualElapsedTime <int>, AirTime <int>, ArrDelay <int>,
## #   DepDelay <int>, Origin <chr>, Dest <chr>, Distance <int>,
## #   TaxiIn <int>, TaxiOut <int>, Cancelled <int>, CancellationCode <chr>,
## #   Diverted <int>
```

```
class(hflights)
```

```
## [1] "tbl_df"     "tbl"        "data.frame"
```

chaning labels

```
unique(hflights$UniqueCarrier)
```

```
##  [1] "AA" "AS" "B6" "CO" "DL" "OO" "UA" "US" "WN" "EV" "F9" "FL" "MQ" "XE"
## [15] "YV"
```

```
lut <- c("AA" = "American", "AS" = "Alaska", "B6" = "JetBlue", "CO" = "Continental",
         "DL" = "Delta", "OO" = "SkyWest", "UA" = "United", "US" = "US_Airways",
         "WN" = "Southwest", "EV" = "Atlantic_Southeast", "F9" = "Frontier",
         "FL" = "AirTran", "MQ" = "American_Eagle", "XE" = "ExpressJet", "YV" = "Mesa")

# Add the Carrier column to hflights
hflights$Carrier <- lut[hflights$UniqueCarrier]

unique(hflights$Carrier)
```

```
##  [1] "American"           "Alaska"             "JetBlue"
##  [4] "Continental"        "Delta"              "SkyWest"
##  [7] "United"             "US_Airways"         "Southwest"
## [10] "Atlantic_Southeast" "Frontier"           "AirTran"
## [13] "American_Eagle"     "ExpressJet"         "Mesa"
```

```
glimpse(hflights$UniqueCarrier)
```

```
##  chr [1:227496] "AA" "AA" "AA" "AA" "AA" "AA" "AA" "AA" "AA" "AA" "AA" ...
```

```
glimpse(hflights$Carrier)
```

```
##  chr [1:227496] "American" "American" "American" "American" "American" ...
```

Change the labels in the CancellationCode column. This column lists reasons why a flight was cancelled using a non-informative alphabetical code.

```
# The lookup table
lut <- c("A" = "carrier", "B" = "weather", "C" = "FFA", "D" = "security", "E" = "not cancelled")

# Add the Code column
hflights$Code <- lut[hflights$CancellationCode]

#glimpse it
unique(hflights[, c("CancellationCode", "Code")])
```

```
## # A tibble: 5 x 2
##   CancellationCode Code
##   <chr>            <chr>
## 1 ""               <NA>
## 2 A                carrier
## 3 B                weather
## 4 C                FFA
## 5 D                security
```

5 verbs

select()

returns a subset of the columns,

```
# Print out a tbl with the four columns of hflights related to delay
select(hflights, 'ActualElapsedTime', 'AirTime', 'ArrDelay', 'DepDelay')
```

```
## # A tibble: 227,496 x 4
##    ActualElapsedTime AirTime ArrDelay DepDelay
##  *             <int>   <int>    <int>    <int>
##  1                60      40      -10        0
##  2                60      45       -9        1
##  3                70      48       -8       -8
##  4                70      39        3        3
##  5                62      44       -3        5
##  6                64      45       -7       -1
##  7                70      43       -1       -1
##  8                59      40      -16       -5
##  9                71      41       44       43
## 10                70      45       43       43
## # ... with 227,486 more rows
```

```
# Print out the columns Origin up to Cancelled of hflights
select(hflights, 'Origin':'Cancelled')
```

```
## # A tibble: 227,496 x 6
##    Origin Dest  Distance TaxiIn TaxiOut Cancelled
##  * <chr>  <chr>    <int>  <int>   <int>     <int>
##  1 IAH    DFW        224      7      13         0
##  2 IAH    DFW        224      6       9         0
##  3 IAH    DFW        224      5      17         0
##  4 IAH    DFW        224      9      22         0
##  5 IAH    DFW        224      9       9         0
##  6 IAH    DFW        224      6      13         0
##  7 IAH    DFW        224     12      15         0
##  8 IAH    DFW        224      7      12         0
##  9 IAH    DFW        224      8      22         0
## 10 IAH    DFW        224      6      19         0
## # ... with 227,486 more rows
```

Helper functions

dplyr comes with a set of helper functions that can help you select groups of variables inside a select() call:

- **starts_with("X")**: every name that starts with "X",
- **ends_with("X")**: every name that ends with "X",
- **contains("X")**: every name that contains "X",
- **matches("X")**: every name that matches "X", where "X" can be a regular expression,
- **num_range("x", 1:5)**: the variables named x01, x02, x03, x04 and x05,
- **one_of(x)**: every name that appears in x, which should be a character vector.

Pay attention here: When you refer to columns directly inside select(), you don't use quotes. If you use the helper functions, you do use quotes.

filter()

return a subset of the rows,

```
# All flights that traveled 3000 miles or more
x1 <- filter(hflights, Distance >= 3000)

# All flights flown by JetBlue, Southwest, or Delta
```

```r
x2 <- filter(hflights, UniqueCarrier %in% c("JetBlue", "Southwest", "Delta"))

# All flights where taxiing took longer than flying
x3 <- filter(hflights, TaxiIn + TaxiOut > AirTime)
```

arrange()

that reorders the rows according to single or multiple variables,

```r
# Definition of dtc
dtc <- filter(hflights, Cancelled == 1, !is.na(DepDelay))

# Arrange dtc by departure delays
arrange(dtc, DepDelay)
```

```
## # A tibble: 68 x 23
##     Year Month DayofMonth DayOfWeek DepTime ArrTime UniqueCarrier FlightNum
##    <int> <int>      <int>     <int>   <int>   <int> <chr>             <int>
##  1  2011     7         23         6     605      NA F9                  225
##  2  2011     1         17         1     916      NA XE                 3068
##  3  2011    12          1         4     541      NA US                  282
##  4  2011    10         12         3    2022      NA MQ                 3724
##  5  2011     7         29         5    1424      NA CO                 1079
##  6  2011     9         29         4    1639      NA OO                 2062
##  7  2011     2          9         3     555      NA MQ                 3265
##  8  2011     5          9         1     715      NA OO                 1177
##  9  2011     1         20         4    1413      NA UA                  552
## 10  2011     1         17         1     831      NA WN                    1
## # ... with 58 more rows, and 15 more variables: TailNum <chr>,
## #   ActualElapsedTime <int>, AirTime <int>, ArrDelay <int>,
## #   DepDelay <int>, Origin <chr>, Dest <chr>, Distance <int>,
## #   TaxiIn <int>, TaxiOut <int>, Cancelled <int>, CancellationCode <chr>,
## #   Diverted <int>, Carrier <chr>, Code <chr>
```

```r
# Arrange dtc so that cancellation reasons are grouped
arrange(dtc, CancellationCode)
```

```
## # A tibble: 68 x 23
##     Year Month DayofMonth DayOfWeek DepTime ArrTime UniqueCarrier FlightNum
##    <int> <int>      <int>     <int>   <int>   <int> <chr>             <int>
##  1  2011     1         20         4    1413      NA UA                  552
##  2  2011     1          7         5    2028      NA XE                 3050
##  3  2011     2          4         5    1638      NA AA                 1121
##  4  2011     2          8         2    1057      NA CO                  408
##  5  2011     2          1         2    1508      NA OO                 5812
##  6  2011     2         21         1    2257      NA OO                 1111
##  7  2011     2          9         3     555      NA MQ                 3265
##  8  2011     3         18         5     727      NA UA                  109
##  9  2011     4          4         1    1632      NA DL                    8
## 10  2011     4          8         5    1608      NA WN                    4
## # ... with 58 more rows, and 15 more variables: TailNum <chr>,
## #   ActualElapsedTime <int>, AirTime <int>, ArrDelay <int>,
## #   DepDelay <int>, Origin <chr>, Dest <chr>, Distance <int>,
## #   TaxiIn <int>, TaxiOut <int>, Cancelled <int>, CancellationCode <chr>,
```

```
## #   Diverted <int>, Carrier <chr>, Code <chr>
```

```
# Arrange dtc according to carrier and departure delays
arrange(dtc, UniqueCarrier, DepDelay)
```

```
## # A tibble: 68 x 23
##     Year Month DayofMonth DayOfWeek DepTime ArrTime UniqueCarrier FlightNum
##    <int> <int>      <int>     <int>   <int>   <int> <chr>             <int>
##  1  2011     8         18         4    1808      NA AA                 1294
##  2  2011     2          4         5    1638      NA AA                 1121
##  3  2011     7         29         5    1424      NA CO                 1079
##  4  2011     1         26         3    1703      NA CO                  410
##  5  2011     8         11         4    1320      NA CO                 1669
##  6  2011     7         25         1    1654      NA CO                 1422
##  7  2011     1         26         3    1926      NA CO                  310
##  8  2011     3         31         4    1016      NA CO                  586
##  9  2011     2          8         2    1057      NA CO                  408
## 10  2011     4          4         1    1632      NA DL                    8
## # ... with 58 more rows, and 15 more variables: TailNum <chr>,
## #   ActualElapsedTime <int>, AirTime <int>, ArrDelay <int>,
## #   DepDelay <int>, Origin <chr>, Dest <chr>, Distance <int>,
## #   TaxiIn <int>, TaxiOut <int>, Cancelled <int>, CancellationCode <chr>,
## #   Diverted <int>, Carrier <chr>, Code <chr>
```

Reverse order of arrange

```
# Arrange according to carrier and decreasing departure delays
arrange(hflights, UniqueCarrier, desc(DepDelay))
```

```
## # A tibble: 227,496 x 23
##     Year Month DayofMonth DayOfWeek DepTime ArrTime UniqueCarrier FlightNum
##    <int> <int>      <int>     <int>   <int>   <int> <chr>             <int>
##  1  2011    12         12         1     650     808 AA                1740
##  2  2011    11         19         6    1752    1910 AA                1903
##  3  2011    12         22         4    1728    1848 AA                1903
##  4  2011    10         23         7    2305       2 AA                 742
##  5  2011     9         27         2    1206    1300 AA                1948
##  6  2011     3         17         4    1647    1747 AA                1505
##  7  2011     6         21         2     955    1315 AA                 466
##  8  2011     5         20         5    2359     130 AA                 426
##  9  2011     4         19         2    2023    2142 AA                1925
## 10  2011     5         12         4    2133      53 AA                1294
## # ... with 227,486 more rows, and 15 more variables: TailNum <chr>,
## #   ActualElapsedTime <int>, AirTime <int>, ArrDelay <int>,
## #   DepDelay <int>, Origin <chr>, Dest <chr>, Distance <int>,
## #   TaxiIn <int>, TaxiOut <int>, Cancelled <int>, CancellationCode <chr>,
## #   Diverted <int>, Carrier <chr>, Code <chr>
```

mutate()

add columns from existing data.

```
# Add the new variable ActualGroundTime to a copy of hflights and save the result as g1.
g1 <- mutate(hflights, ActualGroundTime = ActualElapsedTime - AirTime)
```

summarize()

reduces each group to a single row by calculating aggregate measures.

```
# Print out a summary with variables min_dist and max_dist
summarize(hflights, min_dist = min(Distance), max_dist = max(Distance))
```

```
## # A tibble: 1 x 2
##   min_dist max_dist
##      <dbl>    <dbl>
## 1       79     3904
```

```
# Print out a summary with variable max_div
summarize(filter(hflights, Diverted == 1), max_div = max(Distance))
```

```
## # A tibble: 1 x 1
##   max_div
##     <dbl>
## 1    3904
```

Aggregate functions

- **min(x)** - minimum value of vector x.
- **max(x)** - maximum value of vector x.
- **mean(x)** - mean value of vector x.
- **median(x)** - median value of vector x.
- **quantile(x, p)** - pth quantile of vector x.
- **sd(x)** - standard deviation of vector x.
- **var(x)** - variance of vector x.
- **IQR(x)** - Inter Quartile Range (IQR) of vector x.
- **diff(range(x))** - total range of vector x.

dplyr aggregate functions

- **first(x)** - The first element of vector x.
- **last(x)** - The last element of vector x.
- **nth(x, n)** - The nth element of vector x.
- **n()** - The number of rows in the data.frame or group of observations that summarize() describes.
- **n_distinct(x)** - The number of unique values in vector x.

```
# Generate summarizing statistics for hflights
summarize(hflights,
          n_obs = n(),
          n_carrier = n_distinct(UniqueCarrier),
          n_dest = n_distinct(Dest))
```

```
## # A tibble: 1 x 3
##    n_obs n_carrier n_dest
##    <int>     <int>  <int>
## 1 227496        15    116
```

```
# All American Airline flights
aa <- filter(hflights, UniqueCarrier == "American")

# Generate summarizing statistics for aa
```

```
summarize(aa,
          n_flights = n(),
          n_canc = sum(Cancelled == 1),
          avg_delay = mean(ArrDelay, na.rm=TRUE))
```

```
## # A tibble: 1 x 3
##   n_flights n_canc avg_delay
##       <int>  <int>     <dbl>
## 1         0      0       NaN
```

pipe operator %>%

```
Take the hflights data set and then ...
Add a variable named diff that is the result of subtracting TaxiIn from TaxiOut, and then ...
Pick all of the rows whose diff value does not equal NA, and then ...
Summarize the data set with a value named avg that is the mean diff value.
```

```
hflights %>%
  mutate(diff = TaxiOut - TaxiIn) %>%
  filter(!is.na(diff)) %>%
  summarize(avg = mean(diff))
```

```
## # A tibble: 1 x 1
##     avg
##   <dbl>
## 1  8.99
```

Count the number of ovrnight flights

```
hflights %>%
  filter(!is.na(DepTime), !is.na(ArrTime), DepTime > ArrTime) %>%
  summarize(num = n())
```

```
## # A tibble: 1 x 1
##     num
##   <int>
## 1  2718
```

group_by()

rank()

```
# Ordered overview of average arrival delays per carrier
hflights %>%
  filter(!is.na(ArrDelay) & ArrDelay > 0) %>%
  group_by(UniqueCarrier) %>%
  summarize(avg = mean(ArrDelay)) %>%
  mutate(rank = rank(avg)) %>%
  arrange(rank)
```

```
## # A tibble: 15 x 3
##    UniqueCarrier   avg  rank
##    <chr>         <dbl> <dbl>
##  1 YV             18.7     1
```

```
##  2 F9             18.7     2
##  3 US             20.7     3
##  4 CO             22.1     4
##  5 AS             22.9     5
##  6 OO             24.1     6
##  7 XE             24.2     7
##  8 WN             25.3     8
##  9 FL             27.9     9
## 10 AA             28.5    10
## 11 DL             32.1    11
## 12 UA             32.5    12
## 13 MQ             38.8    13
## 14 EV             40.2    14
## 15 B6             45.5    15
```

dplyr through mySQL database

```
# library(RMySQL)
# library(SQ)
# # Set up a connection to the mysql database
# my_db <- src_mysql(dbname = "dplyr",
#                    host = "courses.csrrinzqubik.us-east-1.rds.amazonaws.com",
#                    port = 3306,
#                    user = "student",
#                    password = "datacamp")
#
# # Reference a table within that source: nycflights
# nycflights <- tbl(my_db, "dplyr")
#
# # glimpse at nycflights
# glimpse(nycflights)
#
# # Ordered, grouped summary of nycflights
# nycflights %>%
#   group_by(carrier) %>%
#   summarize(n_flights = n(),
#             avg_delay = mean(arr_delay, na.rm = TRUE)) %>%
#   arrange(avg_delay)
```