bike_share_monthly_data

SM

2025-07-06

– Load packages

```
library(readr)
```

— Read each CSV file into a separate variable

```
trip_202004 <- read_csv("202004-divvy-tripdata.csv")
```

```
## Rows: 84776 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): ride_id, rideable_type, start_station_name, end_station_name, memb...
## dbl  (6): start_station_id, end_station_id, start_lat, start_lng, end_lat, e...
## dttm (2): started_at, ended_at
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
trip_202005 <- read_csv("202005-divvy-tripdata.csv")
```

```
## Rows: 200274 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): ride_id, rideable_type, start_station_name, end_station_name, memb...
## dbl  (6): start_station_id, end_station_id, start_lat, start_lng, end_lat, e...
## dttm (2): started_at, ended_at
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

—Check column names for each file

```
names(trip_202004)
```

```
##  [1] "ride_id"            "rideable_type"      "started_at"
##  [4] "ended_at"           "start_station_name" "start_station_id"
##  [7] "end_station_name"   "end_station_id"     "start_lat"
## [10] "start_lng"          "end_lat"            "end_lng"
## [13] "member_casual"
```

```
names(trip_202005)
```

```
##  [1] "ride_id"            "rideable_type"      "started_at"
##  [4] "ended_at"           "start_station_name" "start_station_id"
##  [7] "end_station_name"   "end_station_id"     "start_lat"
## [10] "start_lng"          "end_lat"            "end_lng"
## [13] "member_casual"
```

– Store column names as lists

```
cols_202004 <- names(trip_202004)
cols_202005 <- names(trip_202005)
```

– Compare to the file

```
identical(cols_202004, cols_202005)
```

```
## [1] TRUE
```

– Combine the two data frames

```
install.packages("dplyr")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.4'
## (as 'lib' is unspecified)
```

```
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
```

```
combined_trips <- bind_rows(trip_202004, trip_202005)
```

– Peek at the combined data

```
glimpse(combined_trips)
```

```
## Rows: 285,050
## Columns: 13
## $ ride_id            <chr> "A847FADBBC638E45", "5405B80E996FF60D", "5DD24A79A4~
## $ rideable_type      <chr> "docked_bike", "docked_bike", "docked_bike", "docke~
## $ started_at         <dttm> 2020-04-26 17:45:14, 2020-04-17 17:08:54, 2020-04-~
## $ ended_at           <dttm> 2020-04-26 18:12:03, 2020-04-17 17:17:03, 2020-04-~
## $ start_station_name <chr> "Eckhart Park", "Drake Ave & Fullerton Ave", "McClu~
## $ start_station_id   <dbl> 86, 503, 142, 216, 125, 173, 35, 434, 627, 377, 508~
## $ end_station_name   <chr> "Lincoln Ave & Diversey Pkwy", "Kosciuszko Park", "~
## $ end_station_id     <dbl> 152, 499, 255, 657, 323, 35, 635, 382, 359, 508, 37~
## $ start_lat          <dbl> 41.8964, 41.9244, 41.8945, 41.9030, 41.8902, 41.896~
## $ start_lng          <dbl> -87.6610, -87.7154, -87.6179, -87.6975, -87.6262, -~
## $ end_lat            <dbl> 41.9322, 41.9306, 41.8679, 41.8992, 41.9695, 41.892~
## $ end_lng            <dbl> -87.6586, -87.7238, -87.6230, -87.6722, -87.6547, -~
## $ member_casual      <chr> "member", "member", "member", "member", "casual", "~
```

– STEP 2: Check the structure

```
str(combined_trips)
```

```
## spc_tbl_ [285,050 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:285050] "A847FADBBC638E45" "5405B80E996FF60D" "5DD24A79A4E006F4" "2A59
##  $ rideable_type     : chr [1:285050] "docked_bike" "docked_bike" "docked_bike" "docked_bike" ...
##  $ started_at        : POSIXct[1:285050], format: "2020-04-26 17:45:14" "2020-04-17 17:08:54" ...
```

```
##  $ ended_at          : POSIXct[1:285050], format: "2020-04-26 18:12:03" "2020-04-17 17:17:03" ...
##  $ start_station_name: chr [1:285050] "Eckhart Park" "Drake Ave & Fullerton Ave" "McClurg Ct & Erie S
##  $ start_station_id  : num [1:285050] 86 503 142 216 125 173 35 434 627 377 ...
##  $ end_station_name  : chr [1:285050] "Lincoln Ave & Diversey Pkwy" "Kosciuszko Park" "Indiana Ave &
##  $ end_station_id    : num [1:285050] 152 499 255 657 323 35 635 382 359 508 ...
##  $ start_lat         : num [1:285050] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num [1:285050] -87.7 -87.7 -87.6 -87.7 -87.6 ...
##  $ end_lat           : num [1:285050] 41.9 41.9 41.9 41.9 42 ...
##  $ end_lng           : num [1:285050] -87.7 -87.7 -87.6 -87.7 -87.7 ...
##  $ member_casual     : chr [1:285050] "member" "member" "member" "member" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ride_id = col_character(),
##   ..   rideable_type = col_character(),
##   ..   started_at = col_datetime(format = ""),
##   ..   ended_at = col_datetime(format = ""),
##   ..   start_station_name = col_character(),
##   ..   start_station_id = col_double(),
##   ..   end_station_name = col_character(),
##   ..   end_station_id = col_double(),
##   ..   start_lat = col_double(),
##   ..   start_lng = col_double(),
##   ..   end_lat = col_double(),
##   ..   end_lng = col_double(),
##   ..   member_casual = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
summary(combined_trips)
```

```
##    ride_id          rideable_type        started_at
##  Length:285050      Length:285050      Min.   :2020-04-01 00:00:30.00
##  Class :character   Class :character   1st Qu.:2020-04-26 13:25:41.00
##  Mode  :character   Mode  :character   Median :2020-05-12 15:26:15.00
##                                        Mean   :2020-05-08 21:33:40.18
##                                        3rd Qu.:2020-05-24 15:45:42.00
##                                        Max.   :2020-05-31 02:58:45.00
##
##     ended_at                      start_station_name start_station_id
##  Min.   :2020-04-01 00:10:45.00   Length:285050      Min.   :  2.0
##  1st Qu.:2020-04-26 13:57:11.50   Class :character   1st Qu.:112.0
##  Median :2020-05-12 15:51:49.50   Mode  :character   Median :211.0
##  Mean   :2020-05-08 22:07:47.50                      Mean   :235.8
##  3rd Qu.:2020-05-24 16:26:30.75                      3rd Qu.:322.0
##  Max.   :2020-05-31 03:03:04.00                      Max.   :673.0
##
##  end_station_name   end_station_id    start_lat       start_lng
##  Length:285050      Min.   :  2.0   Min.   :41.74   Min.   :-87.77
##  Class :character   1st Qu.:113.0   1st Qu.:41.88   1st Qu.:-87.66
##  Mode  :character   Median :212.0   Median :41.90   Median :-87.65
##                     Mean   :237.5   Mean   :41.91   Mean   :-87.65
##                     3rd Qu.:324.0   3rd Qu.:41.93   3rd Qu.:-87.63
##                     Max.   :673.0   Max.   :42.06   Max.   :-87.55
##                     NA's   :420
##     end_lat         end_lng       member_casual
```

```
##  Min.   :41.74   Min.   :-87.77   Length:285050
##  1st Qu.:41.88   1st Qu.:-87.66   Class :character
##  Median :41.90   Median :-87.65   Mode  :character
##  Mean   :41.91   Mean   :-87.65
##  3rd Qu.:41.93   3rd Qu.:-87.63
##  Max.   :42.06   Max.   :-87.55
##  NA's   :420     NA's   :420
```

```
head(combined_trips)
```

```
## # A tibble: 6 x 13
##   ride_id          rideable_type started_at          ended_at
##   <chr>            <chr>         <dttm>              <dttm>
## 1 A847FADBBC638E45 docked_bike   2020-04-26 17:45:14 2020-04-26 18:12:03
## 2 5405B80E996FF60D docked_bike   2020-04-17 17:08:54 2020-04-17 17:17:03
## 3 5DD24A79A4E006F4 docked_bike   2020-04-01 17:54:13 2020-04-01 18:08:36
## 4 2A59BBDF5CDBA725 docked_bike   2020-04-07 12:50:19 2020-04-07 13:02:31
## 5 27AD306C119C6158 docked_bike   2020-04-18 10:22:59 2020-04-18 11:15:54
## 6 356216E875132F61 docked_bike   2020-04-30 17:55:47 2020-04-30 18:01:11
## # i 9 more variables: start_station_name <chr>, start_station_id <dbl>,
## #   end_station_name <chr>, end_station_id <dbl>, start_lat <dbl>,
## #   start_lng <dbl>, end_lat <dbl>, end_lng <dbl>, member_casual <chr>
```

– STEP 3: Convert dates (if needed)

```
combined_trips <- combined_trips %>%
  mutate(
    started_at = as.POSIXct(started_at),
    ended_at = as.POSIXct(ended_at)
  )
```

– STEP 4: Add calculated columns

```
combined_trips <- combined_trips %>%
  mutate(
    ride_length_mins = as.numeric(difftime(ended_at, started_at, units = "mins")),
    day_of_week = weekdays(started_at)
  )
```

– STEP 5: Basic checks on new columns

```
summary(combined_trips$ride_length_mins)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##    -2.93     9.93    18.52    34.12    31.08 58720.03
```

```
table(combined_trips$day_of_week)
```

```
##
##    Friday    Monday  Saturday    Sunday  Thursday   Tuesday Wednesday
##     42696     34403     67587     44443     32123     30074     33724
```

– Tip: Remove negative ride lengths:

```
combined_trips <- combined_trips %>%
  dplyr::filter(ride_length_mins > 0)
```

– STEP 6: Start simple analysis – Example 1: Average ride length

```r
combined_trips %>%
  summarise(
    mean_ride_length = mean(ride_length_mins, na.rm = TRUE)
  )
```

```
## # A tibble: 1 x 1
##   mean_ride_length
##              <dbl>
## 1             34.2
```

– Example 2: Average by user type

```r
combined_trips %>%
  group_by(member_casual) %>%
  summarise(
    mean_ride_length = mean(ride_length_mins, na.rm = TRUE)
  )
```

```
## # A tibble: 2 x 2
##   member_casual mean_ride_length
##   <chr>                    <dbl>
## 1 casual                    55.9
## 2 member                    20.4
```

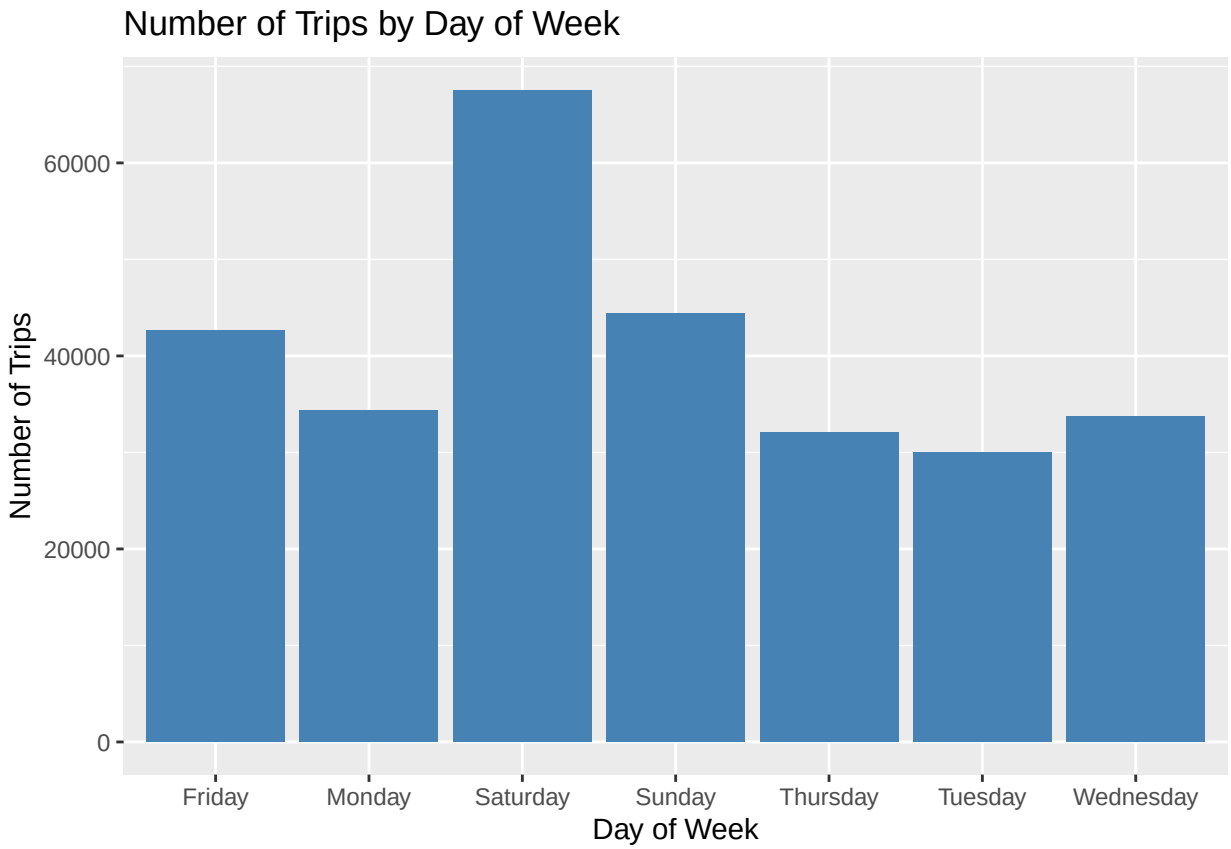
– Example 3: Trips by day

```r
combined_trips %>%
  group_by(day_of_week) %>%
  summarise(
    num_trips = n()
  ) %>%
  arrange(match(day_of_week, c("Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturd
```

```
## # A tibble: 7 x 2
##   day_of_week num_trips
##   <chr>           <int>
## 1 Sunday          44414
## 2 Monday          34360
## 3 Tuesday         30047
## 4 Wednesday       33688
## 5 Thursday        32095
## 6 Friday          42648
## 7 Saturday        67555
```

– STEP 7: Visualize- title = "Number of Trips by Day of Week"

```r
library(ggplot2)
combined_trips %>%
  group_by(day_of_week) %>%
  summarise(num_trips = n()) %>%
  ggplot(aes(x = day_of_week, y = num_trips)) +
  geom_col(fill = "steelblue") +
  labs(title = "Number of Trips by Day of Week",
       x = "Day of Week", y = "Number of Trips")
```



– Ride length stats

```
combined_trips %>%
  summarise(
    min_ride = min(ride_length_mins, na.rm = TRUE),
    mean_ride = mean(ride_length_mins, na.rm = TRUE),
    median_ride = median(ride_length_mins, na.rm = TRUE),
    max_ride = max(ride_length_mins, na.rm = TRUE)
  )
```

```
## # A tibble: 1 x 4
##   min_ride mean_ride median_ride max_ride
##      <dbl>     <dbl>       <dbl>    <dbl>
## 1   0.0167      34.2        18.5   58720.
```

– By member vs casual:

```
combined_trips %>%
  group_by(member_casual) %>%
  summarise(
    mean_ride = mean(ride_length_mins, na.rm = TRUE),
    median_ride = median(ride_length_mins, na.rm = TRUE),
    max_ride = max(ride_length_mins, na.rm = TRUE)
  )
```

```
## # A tibble: 2 x 4
##   member_casual mean_ride median_ride max_ride
##   <chr>             <dbl>       <dbl>    <dbl>
## 1 casual             55.9        26.3   55684.
```

```
## 2 member              20.4        14.7   58720.
```

–Popular start & end stations —Top 10 start stations

```
combined_trips %>%
  group_by(start_station_name) %>%
  summarise(num_trips = n()) %>%
  arrange(desc(num_trips)) %>%
  head(10)
```

```
## # A tibble: 10 x 2
##    start_station_name           num_trips
##    <chr>                            <int>
##  1 Clark St & Elm St                 2791
##  2 Dearborn St & Erie St             2209
##  3 Larrabee St & Webster Ave         2125
##  4 Indiana Ave & Roosevelt Rd        2100
##  5 Desplaines St & Kinzie St         2092
##  6 Clark St & Armitage Ave           2029
##  7 Stockton Dr & Wrightwood Ave      2024
##  8 Clark St & Lincoln Ave            2016
##  9 Broadway & Barry Ave              1972
## 10 Wabash Ave & Grand Ave            1965
```

—Top 10 end stations

```
combined_trips %>%
  group_by(end_station_name) %>%
  summarise(num_trips = n()) %>%
  arrange(desc(num_trips)) %>%
  head(10)
```

```
## # A tibble: 10 x 2
##    end_station_name            num_trips
##    <chr>                           <int>
##  1 Clark St & Elm St                2843
##  2 Dearborn St & Erie St            2275
##  3 Larrabee St & Webster Ave        2213
##  4 Broadway & Barry Ave             2160
##  5 Wabash Ave & Roosevelt Rd        2139
##  6 Indiana Ave & Roosevelt Rd       2092
##  7 Dearborn Pkwy & Delaware Pl      2018
##  8 Clark St & Armitage Ave          1979
##  9 Wabash Ave & Grand Ave           1971
## 10 St. Clair St & Erie St           1967
```

–Trips by day of week and user type — Most popular days for each rider type

```
combined_trips %>%
  group_by(member_casual, day_of_week) %>%
  summarise(num_trips = n()) %>%
  arrange(member_casual, day_of_week)
```

```
## `summarise()` has grouped output by 'member_casual'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 14 x 3
## # Groups:   member_casual [2]
```

```
##    member_casual day_of_week num_trips
##    <chr>         <chr>           <int>
##  1 casual        Friday          15255
##  2 casual        Monday          13200
##  3 casual        Saturday        31665
##  4 casual        Sunday          19672
##  5 casual        Thursday         9835
##  6 casual        Tuesday          9825
##  7 casual        Wednesday       10991
##  8 member        Friday          27393
##  9 member        Monday          21160
## 10 member        Saturday        35890
## 11 member        Sunday          24742
## 12 member        Thursday        22260
## 13 member        Tuesday         20222
## 14 member        Wednesday       22697
```

– Average ride time by day of week and user type

```
combined_trips %>%
  group_by(member_casual, day_of_week) %>%
  summarise(
    avg_ride_length = mean(ride_length_mins, na.rm = TRUE),
    num_trips = n()
  ) %>%
  arrange(member_casual, day_of_week)
```

```
## `summarise()` has grouped output by 'member_casual'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 14 x 4
## # Groups:   member_casual [2]
##    member_casual day_of_week avg_ride_length num_trips
##    <chr>         <chr>                 <dbl>     <int>
##  1 casual        Friday                 60.6     15255
##  2 casual        Monday                 56.0     13200
##  3 casual        Saturday               52.0     31665
##  4 casual        Sunday                 57.1     19672
##  5 casual        Thursday               55.4      9835
##  6 casual        Tuesday                63.6      9825
##  7 casual        Wednesday              51.9     10991
##  8 member        Friday                 19.0     27393
##  9 member        Monday                 18.7     21160
## 10 member        Saturday               22.8     35890
## 11 member        Sunday                 23.8     24742
## 12 member        Thursday               19.0     22260
## 13 member        Tuesday                19.5     20222
## 14 member        Wednesday              18.1     22697
```

– Trips by bike type —Total rides by bike type & user:

```
combined_trips %>%
  group_by(rideable_type, member_casual) %>%
  summarise(num_trips = n()) %>%
  arrange(desc(num_trips))
```

```
## `summarise()` has grouped output by 'rideable_type'. You can override using the
```

```
## `.groups` argument.
```

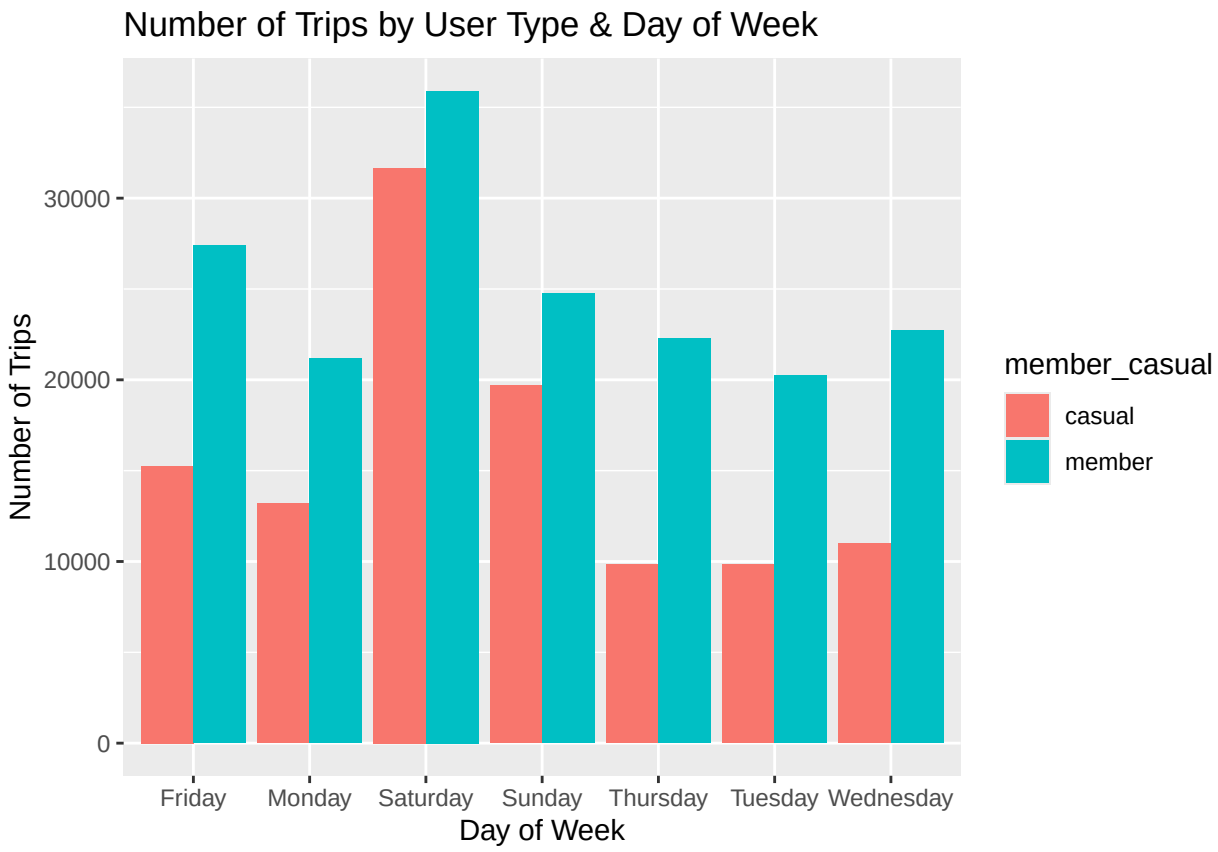
```
## # A tibble: 2 x 3
## # Groups:   rideable_type [1]
##   rideable_type member_casual num_trips
##   <chr>         <chr>             <int>
## 1 docked_bike   member           174364
## 2 docked_bike   casual           110443
```

– Visualize: Trips by weekday & user

```r
library(ggplot2)

combined_trips %>%
  group_by(member_casual, day_of_week) %>%
  summarise(num_trips = n()) %>%
  ggplot(aes(x = day_of_week, y = num_trips, fill = member_casual)) +
  geom_col(position = "dodge") +
  labs(
    title = "Number of Trips by User Type & Day of Week",
    x = "Day of Week", y = "Number of Trips"
  )
```

```
## `summarise()` has grouped output by 'member_casual'. You can override using the
## `.groups` argument.
```



– Visualize: Average ride length by weekday & user

```r
combined_trips %>%
  group_by(member_casual, day_of_week) %>%
  summarise(avg_ride_length = mean(ride_length_mins, na.rm = TRUE)) %>%
  ggplot(aes(x = day_of_week, y = avg_ride_length, fill = member_casual)) +
  geom_col(position = "dodge") +
  labs(
    title = "Average Ride Length by User Type & Day of Week",
    x = "Day of Week", y = "Average Ride Length (mins)"
  )
```

```
## `summarise()` has grouped output by 'member_casual'. You can override using the
## `.groups` argument.
```