# Case Study: How Does a Bike-Share Navigate Speedy Success? ¶

**Note**: This is a Google data analytics certificate case study.

## Scenario

According to the scenario, I'm a junior data analyst in the marketing team at Cyclistic, a bike-share company from Chicago.
Cyclistics has a flexible pricing plans: single-ride passes, full-day passes and annual memberships. Customers who purchase single-ride or full-day passes are referred to as casual riders. Customers who purchase annual memberships are Cyclistic members. Also, we know that casual riders are already aware of the Cyclistic program and have chosen Cyclistic for their mobility needs.
The finance team have concluded that annual members are much more profitable than casual riders. The management believes the company can increase profits by maximizing the number of annual memberships.
Therefore, I have to understand how casual riders and annual members use Cyclistic bikes differently and how we can convert casual riders into annual members by a new marketing strategy. To do this, I'll analyze the Cyclistic historical bike trip data.

# PART 1 - ASK

### 1. What is the main task?

- To identify how casual customers and members use bikes differently.
- To figure out why would casual riders buy Cyclistic annual memberships.
- To suggest recommendations on how to convert casual customers to members.

### 2. Who are the key stakeholders?

- Lily Moreno: The director of marketing and my manager.
- Cyclistic executive team: The notoriously detail-oriented executive team will decide whether to approve the recommended marketing program.
- Cyclistic marketing analytics team.

# PART 2, 3 - PREPARE, PROCESS

### 1. Where is the data?

- I will use the previous 12 months (**from october 2020 to september 2021**) of Cyclistic trip data from here (https://divvy-tripdata.s3.amazonaws.com/index.html).

  (Note: The datasets have a different name because Cyclistic is a fictional company.)

### 2. How is the data organized?

- 12 .csv files with 13 columns each.

### 3. Is the data ROCCC?

- **R**eliable - yes (without bias).
- **O**riginal - yes. The data has been made available by Motivate International Inc. under this license (https://www.divvybikes.com/data-license-agreement).
- **C**omprehensive - not exactly (no info about customers, financial info; some empty and NA values, duplicates).
- **C**urrent - yes, updated monthly.
- **C**ited - yes.

### 4. What tools to choose and why?

- I'll choose R because it's possible to clean, transform, analyze and visualize large datasets right in RStudio.

In [1]:

```
# Download all the needed packages
library(tidyverse)
library(lubridate) # for work with time data
library(skimr) # for describing statistics of data
library(scales) # for adjusting display of values on plot axis
```

```
── Attaching packages ─────────────────────────────────────── tidyverse
1.3.1 ──

✓ ggplot2 3.3.5     ✓ purrr   0.3.4
✓ tibble  3.1.5     ✓ dplyr   1.0.7
✓ tidyr   1.1.4     ✓ stringr 1.4.0
✓ readr   2.0.2     ✓ forcats 0.5.1

── Conflicts ────────────────────────────────────────── tidyverse_confl
icts() ──
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()    masks stats::lag()


Attaching package: ‘lubridate’


The following objects are masked from ‘package:base’:

    date, intersect, setdiff, union



Attaching package: ‘scales’


The following object is masked from ‘package:purrr’:

    discard


The following object is masked from ‘package:readr’:

    col_factor
```

In [2]:

```
# Uploading the trips data into R
trips_202010 <- read.csv("../input/cyclistic-tripdata-202010-202109/202010-divvy
-tripdata.csv")
trips_202011 <- read.csv("../input/cyclistic-tripdata-202010-202109/202011-divvy
-tripdata.csv")
trips_202012 <- read.csv("../input/cyclistic-tripdata-202010-202109/202012-divvy
-tripdata.csv")
trips_202101 <- read.csv("../input/cyclistic-tripdata-202010-202109/202101-divvy
-tripdata.csv")
trips_202102 <- read.csv("../input/cyclistic-tripdata-202010-202109/202102-divvy
-tripdata.csv")
trips_202103 <- read.csv("../input/cyclistic-tripdata-202010-202109/202103-divvy
-tripdata.csv")
trips_202104 <- read.csv("../input/cyclistic-tripdata-202010-202109/202104-divvy
-tripdata.csv")
trips_202105 <- read.csv("../input/cyclistic-tripdata-202010-202109/202105-divvy
-tripdata.csv")
trips_202106 <- read.csv("../input/cyclistic-tripdata-202010-202109/202106-divvy
-tripdata.csv")
trips_202107 <- read.csv("../input/cyclistic-tripdata-202010-202109/202107-divvy
-tripdata.csv")
trips_202108 <- read.csv("../input/cyclistic-tripdata-202010-202109/202108-divvy
-tripdata.csv")
trips_202109 <- read.csv("../input/cyclistic-tripdata-202010-202109/202109-divvy
-tripdata.csv")
```

## 5. What problems does the data have?

- Let's inspect our data and fix possible errors.

In [3]:

```
# Quick inspection of the data
print("October, 2020")
glimpse(trips_202010)
print("November, 2020")
glimpse(trips_202011)
print("December, 2020")
glimpse(trips_202012)
print("January, 2021")
glimpse(trips_202101)
print("February, 2021")
glimpse(trips_202102)
print("March, 2021")
glimpse(trips_202103)
print("April, 2021")
glimpse(trips_202104)
print("May, 2021")
glimpse(trips_202105)
print("June, 2021")
glimpse(trips_202106)
print("July, 2021")
glimpse(trips_202107)
print("August, 2021")
glimpse(trips_202108)
print("September, 2021")
glimpse(trips_202109)
```

```
[1] "October, 2020"
Rows: 388,653
Columns: 13
$ ride_id            <chr> "ACB6B40CF5B9044C", "DF450C72FD109C01",
 "B6396B54A1…
$ rideable_type      <chr> "electric_bike", "electric_bike", "elect
ric_bike", …
$ started_at         <chr> "2020-10-31 19:39:43", "2020-10-31 23:5
0:08", "2020…
$ ended_at           <chr> "2020-10-31 19:57:12", "2020-11-01 00:0
4:16", "2020…
$ start_station_name <chr> "Lakeview Ave & Fullerton Pkwy", "Southp
ort Ave & W…
$ start_station_id   <int> 313, 227, 102, 165, 190, 359, 313, 125,
 NA, 174, 11…
$ end_station_name   <chr> "Rush St & Hubbard St", "Kedzie Ave & Mi
lwaukee Ave…
$ end_station_id     <int> 125, 260, 423, 256, 185, 53, 125, 313, 1
99, 635, 30…
$ start_lat          <dbl> 41.92610, 41.94817, 41.77346, 41.95085,
 41.92886, 4…
$ start_lng          <dbl> -87.63898, -87.66391, -87.58537, -87.659
24, -87.663…
$ end_lat            <dbl> 41.89035, 41.92953, 41.79145, 41.95281,
 41.91778, 4…
$ end_lng            <dbl> -87.62607, -87.70782, -87.60005, -87.650
10, -87.691…
$ member_casual      <chr> "casual", "casual", "casual", "casual",
 "casual", "…
[1] "November, 2020"
Rows: 259,716
Columns: 13
$ ride_id            <chr> "BD0A6FF6FFF9B921", "96A7A7A4BDE4F82D",
 "C61526D065…
$ rideable_type      <chr> "electric_bike", "electric_bike", "elect
ric_bike", …
$ started_at         <chr> "2020-11-01 13:36:00", "2020-11-01 10:0
3:26", "2020…
$ ended_at           <chr> "2020-11-01 13:45:40", "2020-11-01 10:1
4:45", "2020…
$ start_station_name <chr> "Dearborn St & Erie St", "Franklin St &
 Illinois St…
$ start_station_id   <int> 110, 672, 76, 659, 2, 72, 76, NA, 58, 39
```

```
4, 623, NA,…
$ end_station_name   <chr> "St. Clair St & Erie St", "Noble St & Mi
lwaukee Ave…
$ end_station_id     <int> 211, 29, 41, 185, 2, 76, 72, NA, 288, 27
3, 2, 506, …
$ start_lat          <dbl> 41.89418, 41.89096, 41.88098, 41.89550,
 41.87650, 4…
$ start_lng          <dbl> -87.62913, -87.63534, -87.61675, -87.682
01, -87.620…
$ end_lat            <dbl> 41.89443, 41.90067, 41.87205, 41.91774,
 41.87645, 4…
$ end_lng            <dbl> -87.62338, -87.66248, -87.62955, -87.691
39, -87.620…
$ member_casual      <chr> "casual", "casual", "casual", "casual",
 "casual", "…
[1] "December, 2020"
Rows: 131,573
Columns: 13
$ ride_id            <chr> "70B6A9A437D4C30D", "158A465D4E74C54A",
 "5262016E0F…
$ rideable_type      <chr> "classic_bike", "electric_bike", "electr
ic_bike", "…
$ started_at         <chr> "2020-12-27 12:44:29", "2020-12-18 17:3
7:15", "2020…
$ ended_at           <chr> "2020-12-27 12:55:06", "2020-12-18 17:4
4:19", "2020…
$ start_station_name <chr> "Aberdeen St & Jackson Blvd", "", "",
 "", "", "", "…
$ start_station_id   <chr> "13157", "", "", "", "", "", "", "", "",
"", "", ""…
$ end_station_name   <chr> "Desplaines St & Kinzie St", "", "", "",
"", "", ""…
$ end_station_id     <chr> "TA1306000003", "", "", "", "", "", "",
 "", "", "",…
$ start_lat          <dbl> 41.87773, 41.93000, 41.91000, 41.92000,
 41.80000, 4…
$ start_lng          <dbl> -87.65479, -87.70000, -87.69000, -87.700
00, -87.590…
$ end_lat            <dbl> 41.88872, 41.91000, 41.93000, 41.91000,
 41.80000, 4…
$ end_lng            <dbl> -87.64445, -87.70000, -87.70000, -87.700
00, -87.590…
$ member_casual      <chr> "member", "member", "member", "member",
```

```
 "member", "…
[1] "January, 2021"
Rows: 96,834
Columns: 13
$ ride_id            <chr> "E19E6F1B8D4C42ED", "DC88F20C2C55F27F",
 "EC45C94683…
$ rideable_type      <chr> "electric_bike", "electric_bike", "elect
ric_bike", …
$ started_at         <chr> "2021-01-23 16:14:19", "2021-01-27 18:4
3:08", "2021…
$ ended_at           <chr> "2021-01-23 16:24:44", "2021-01-27 18:4
7:12", "2021…
$ start_station_name <chr> "California Ave & Cortez St", "Californi
a Ave & Cor…
$ start_station_id   <chr> "17660", "17660", "17660", "17660", "176
60", "17660…
$ end_station_name   <chr> "", "", "", "", "", "", "", "", "", "Woo
d St & Augu…
$ end_station_id     <chr> "", "", "", "", "", "", "", "", "", "65
7", "13258",…
$ start_lat          <dbl> 41.90034, 41.90033, 41.90031, 41.90040,
 41.90033, 4…
$ start_lng          <dbl> -87.69674, -87.69671, -87.69664, -87.696
66, -87.696…
$ end_lat            <dbl> 41.89000, 41.90000, 41.90000, 41.92000,
 41.90000, 4…
$ end_lng            <dbl> -87.72000, -87.69000, -87.70000, -87.690
00, -87.700…
$ member_casual      <chr> "member", "member", "member", "member",
 "casual", "…
[1] "February, 2021"
Rows: 49,622
Columns: 13
$ ride_id            <chr> "89E7AA6C29227EFF", "0FEFDE2603568365",
 "E6159D746B…
$ rideable_type      <chr> "classic_bike", "classic_bike", "electri
c_bike", "c…
$ started_at         <chr> "2021-02-12 16:14:56", "2021-02-14 17:5
2:38", "2021…
$ ended_at           <chr> "2021-02-12 16:21:43", "2021-02-14 18:1
2:09", "2021…
$ start_station_name <chr> "Glenwood Ave & Touhy Ave", "Glenwood Av
e & Touhy A…
```

```
$ start_station_id   <chr> "525", "525", "KA1503000012", "637", "13
216", "1800…
$ end_station_name   <chr> "Sheridan Rd & Columbia Ave", "Bosworth
 Ave & Howar…
$ end_station_id     <chr> "660", "16806", "TA1305000029", "TA13050
00034", "TA…
$ start_lat          <dbl> 42.01270, 42.01270, 41.88579, 41.89563,
 41.83473, 4…
$ start_lng          <dbl> -87.66606, -87.66606, -87.63110, -87.672
07, -87.625…
$ end_lat            <dbl> 42.00458, 42.01954, 41.88487, 41.90312,
 41.83816, 4…
$ end_lng            <dbl> -87.66141, -87.66956, -87.62750, -87.673
94, -87.645…
$ member_casual      <chr> "member", "casual", "member", "member",
 "member", "…
[1] "March, 2021"
Rows: 228,496
Columns: 13
$ ride_id            <chr> "CFA86D4455AA1030", "30D9DC61227D1AF3",
 "846D87A156…
$ rideable_type      <chr> "classic_bike", "classic_bike", "classic
_bike", "cl…
$ started_at         <chr> "2021-03-16 08:32:30", "2021-03-28 01:2
6:28", "2021…
$ ended_at           <chr> "2021-03-16 08:36:34", "2021-03-28 01:3
6:55", "2021…
$ start_station_name <chr> "Humboldt Blvd & Armitage Ave", "Humbold
t Blvd & Ar…
$ start_station_id   <chr> "15651", "15651", "15443", "TA130800002
1", "525", "…
$ end_station_name   <chr> "Stave St & Armitage Ave", "Central Park
Ave & Bloo…
$ end_station_id     <chr> "13266", "18017", "TA1308000043", "1332
3", "E008", …
$ start_lat          <dbl> 41.91751, 41.91751, 41.84273, 41.96881,
 42.01270, 4…
$ start_lng          <dbl> -87.70181, -87.70181, -87.63549, -87.657
66, -87.666…
$ end_lat            <dbl> 41.91774, 41.91417, 41.83066, 41.95283,
 42.05049, 4…
$ end_lng            <dbl> -87.69139, -87.71676, -87.64717, -87.649
99, -87.677…
```

```
$ member_casual      <chr> "casual", "casual", "casual", "casual",
 "casual", "…
[1] "April, 2021"
Rows: 337,230
Columns: 13
$ ride_id            <chr> "6C992BD37A98A63F", "1E0145613A209000",
 "E498E15508…
$ rideable_type      <chr> "classic_bike", "docked_bike", "docked_b
ike", "clas…
$ started_at         <chr> "2021-04-12 18:25:36", "2021-04-27 17:2
7:11", "2021…
$ ended_at           <chr> "2021-04-12 18:56:55", "2021-04-27 18:3
1:29", "2021…
$ start_station_name <chr> "State St & Pearson St", "Dorchester Ave
& 49th St"…
$ start_station_id   <chr> "TA1307000061", "KA1503000069", "20121",
"TA1305000…
$ end_station_name   <chr> "Southport Ave & Waveland Ave", "Dorches
ter Ave & 4…
$ end_station_id     <chr> "13235", "KA1503000069", "20121", "1323
5", "20121",…
$ start_lat          <dbl> 41.89745, 41.80577, 41.74149, 41.90312,
 41.74149, 4…
$ start_lng          <dbl> -87.62872, -87.59246, -87.65841, -87.673
94, -87.658…
$ end_lat            <dbl> 41.94815, 41.80577, 41.74149, 41.94815,
 41.74149, 4…
$ end_lng            <dbl> -87.66394, -87.59246, -87.65841, -87.663
94, -87.658…
$ member_casual      <chr> "member", "casual", "casual", "member",
 "casual", "…
[1] "May, 2021"
Rows: 531,633
Columns: 13
$ ride_id            <chr> "C809ED75D6160B2A", "DD59FDCE0ACACAF3",
 "0AB83CB88C…
$ rideable_type      <chr> "electric_bike", "electric_bike", "elect
ric_bike", …
$ started_at         <chr> "2021-05-30 11:58:15", "2021-05-30 11:2
9:14", "2021…
$ ended_at           <chr> "2021-05-30 12:10:39", "2021-05-30 12:1
4:09", "2021…
$ start_station_name <chr> "", "", "", "", "", "", "", "", "", "",
```

```
 "", "", "",…
$ start_station_id   <chr> "", "", "", "", "", "", "", "", "", "",
 "", "", "",…
$ end_station_name   <chr> "", "", "", "", "", "", "", "", "", "",
 "", "", "",…
$ end_station_id     <chr> "", "", "", "", "", "", "", "", "", "",
 "", "", "",…
$ start_lat          <dbl> 41.90000, 41.88000, 41.92000, 41.92000,
 41.94000, 4…
$ start_lng          <dbl> -87.63000, -87.62000, -87.70000, -87.700
00, -87.690…
$ end_lat            <dbl> 41.89000, 41.79000, 41.92000, 41.94000,
 41.94000, 4…
$ end_lng            <dbl> -87.61000, -87.58000, -87.70000, -87.690
00, -87.700…
$ member_casual      <chr> "casual", "casual", "casual", "casual",
 "casual", "…
[1] "June, 2021"
Rows: 729,595
Columns: 13
$ ride_id            <chr> "99FEC93BA843FB20", "06048DCFC8520CAF",
 "9598066F68…
$ rideable_type      <chr> "electric_bike", "electric_bike", "elect
ric_bike", …
$ started_at         <chr> "2021-06-13 14:31:28", "2021-06-04 11:1
8:02", "2021…
$ ended_at           <chr> "2021-06-13 14:34:11", "2021-06-04 11:2
4:19", "2021…
$ start_station_name <chr> "", "", "", "", "", "", "", "", "", "",
 "", "", "",…
$ start_station_id   <chr> "", "", "", "", "", "", "", "", "", "",
 "", "", "",…
$ end_station_name   <chr> "", "", "", "", "", "", "", "", "", "Mic
higan Ave &…
$ end_station_id     <chr> "", "", "", "", "", "", "", "", "", "130
42", "", ""…
$ start_lat          <dbl> 41.80, 41.79, 41.80, 41.78, 41.80, 41.7
8, 41.79, 41…
$ start_lng          <dbl> -87.59, -87.59, -87.60, -87.58, -87.59,
 -87.58, -87…
$ end_lat            <dbl> 41.80000, 41.80000, 41.79000, 41.80000,
 41.79000, 4…
$ end_lng            <dbl> -87.6000, -87.6000, -87.5900, -87.6000,
```

```
 -87.5900, -…
$ member_casual      <chr> "member", "member", "member", "member",
 "member", "…
[1] "July, 2021"
Rows: 822,410
Columns: 13
$ ride_id           <chr> "0A1B623926EF4E16", "B2D5583A5A5E76EE",
 "6F264597DD…
$ rideable_type     <chr> "docked_bike", "classic_bike", "classic_
bike", "cla…
$ started_at        <chr> "2021-07-02 14:44:36", "2021-07-07 16:5
7:42", "2021…
$ ended_at          <chr> "2021-07-02 15:19:58", "2021-07-07 17:1
6:09", "2021…
$ start_station_name <chr> "Michigan Ave & Washington St", "Califor
nia Ave & C…
$ start_station_id  <chr> "13001", "17660", "SL-012", "17660", "17
660", "1766…
$ end_station_name  <chr> "Halsted St & North Branch St", "Wood St
& Hubbard …
$ end_station_id    <chr> "KA1504000117", "13432", "KA1503000044",
"13196", "…
$ start_lat         <dbl> 41.88398, 41.90036, 41.86038, 41.90036,
 41.90035, 4…
$ start_lng         <dbl> -87.62468, -87.69670, -87.62581, -87.696
70, -87.696…
$ end_lat           <dbl> 41.89937, 41.88990, 41.89017, 41.89456,
 41.88659, 4…
$ end_lng           <dbl> -87.64848, -87.67147, -87.62619, -87.653
45, -87.658…
$ member_casual     <chr> "casual", "casual", "member", "member",
 "casual", "…
[1] "August, 2021"
Rows: 804,352
Columns: 13
$ ride_id           <chr> "99103BB87CC6C1BB", "EAFCCCFB0A3FC5A1",
 "9EF4F46C57…
$ rideable_type     <chr> "electric_bike", "electric_bike", "elect
ric_bike", …
$ started_at        <chr> "2021-08-10 17:15:49", "2021-08-10 17:2
3:14", "2021…
$ ended_at          <chr> "2021-08-10 17:22:44", "2021-08-10 17:3
9:24", "2021…
```

```
$ start_station_name <chr> "", "", "", "", "", "", "", "", "", "",
 "", "", "",…
$ start_station_id   <chr> "", "", "", "", "", "", "", "", "", "",
 "", "", "",…
$ end_station_name   <chr> "", "", "", "", "", "", "", "Clark St &
 Grace St", …
$ end_station_id     <chr> "", "", "", "", "", "", "", "TA130700012
7", "", "",…
$ start_lat          <dbl> 41.77000, 41.77000, 41.95000, 41.97000,
 41.79000, 4…
$ start_lng          <dbl> -87.68000, -87.68000, -87.65000, -87.670
00, -87.600…
$ end_lat            <dbl> 41.77000, 41.77000, 41.97000, 41.95000,
 41.77000, 4…
$ end_lng            <dbl> -87.68000, -87.63000, -87.66000, -87.650
00, -87.620…
$ member_casual      <chr> "member", "member", "member", "member",
 "member", "…
[1] "September, 2021"
Rows: 756,147
Columns: 13
$ ride_id            <chr> "9DC7B962304CBFD8", "F930E2C6872D6B32",
 "6EF7213790…
$ rideable_type      <chr> "electric_bike", "electric_bike", "elect
ric_bike", …
$ started_at         <chr> "2021-09-28 16:07:10", "2021-09-28 14:2
4:51", "2021…
$ ended_at           <chr> "2021-09-28 16:09:54", "2021-09-28 14:4
0:05", "2021…
$ start_station_name <chr> "", "", "", "", "", "", "", "", "", "",
 "Clark St &…
$ start_station_id   <chr> "", "", "", "", "", "", "", "", "", "",
 "TA13070001…
$ end_station_name   <chr> "", "", "", "", "", "", "", "", "", "",
 "", "", "",…
$ end_station_id     <chr> "", "", "", "", "", "", "", "", "", "",
 "", "", "",…
$ start_lat          <dbl> 41.89000, 41.94000, 41.81000, 41.80000,
 41.88000, 4…
$ start_lng          <dbl> -87.68000, -87.64000, -87.72000, -87.720
00, -87.740…
$ end_lat            <dbl> 41.89, 41.98, 41.80, 41.81, 41.88, 41.8
8, 41.74, 41…
```

```
$ end_lng           <dbl> -87.67, -87.67, -87.72, -87.72, -87.71,
 -87.74, -87…
$ member_casual     <chr> "casual", "casual", "casual", "casual",
 "casual", "…
```

**Note:**

We need to unite our monthly data into one data frame, but *start_station_id* and *end_station_id* columns in October and November of 2020 are **integer**, though in other tables they are **characters**. Let' fix this for correct joining.

In [4]:

```
# Coverting int to chr
print("October, 2020")
trips_202010 %>%
  as_tibble() %>%
  mutate(start_station_id = as.character(start_station_id),
         end_station_id = as.character(end_station_id)) %>%
  select(start_station_id, end_station_id) %>%
  head(2)

print("November, 2020")
trips_202011 %>%
  as_tibble() %>%
  mutate(start_station_id = as.character(start_station_id),
         end_station_id = as.character(end_station_id)) %>%
  select(start_station_id, end_station_id) %>%
  head(2)
```

```
[1] "October, 2020"
```

A tibble: 2 × 2

| start_station_id | end_station_id |
|---|---|
| <chr> | <chr> |
| 313 | 125 |
| 227 | 260 |

```
[1] "November, 2020"
```

A tibble: 2 × 2

| start_station_id | end_station_id |
|---|---|
| <chr> | <chr> |
| 110 | 211 |
| 672 | 29 |

In [5]:

```
# Unite all the df's into one
trips_total_raw <- rbind(trips_202010, trips_202011, trips_202012, trips_202101,
trips_202102,
                     trips_202103, trips_202104, trips_202105, trips_202106, tri
ps_202107,
                     trips_202108, trips_202109)
```

In [6]:

```
# Explore the structure of the combined table
str(trips_total_raw)
summary(trips_total_raw)
```

```
'data.frame':   5136261 obs. of  13 variables:
 $ ride_id           : chr  "ACB6B40CF5B9044C" "DF450C72FD109C01"
"B6396B54A15AC0DF" "44A4AEE261B9E854" ...
 $ rideable_type     : chr  "electric_bike" "electric_bike" "electr
ic_bike" "electric_bike" ...
 $ started_at        : chr  "2020-10-31 19:39:43" "2020-10-31 23:5
0:08" "2020-10-31 23:00:01" "2020-10-31 22:16:43" ...
 $ ended_at          : chr  "2020-10-31 19:57:12" "2020-11-01 00:0
4:16" "2020-10-31 23:08:22" "2020-10-31 22:19:35" ...
 $ start_station_name: chr  "Lakeview Ave & Fullerton Pkwy" "Southp
ort Ave & Waveland Ave" "Stony Island Ave & 67th St" "Clark St & Gr
ace St" ...
 $ start_station_id  : chr  "313" "227" "102" "165" ...
 $ end_station_name  : chr  "Rush St & Hubbard St" "Kedzie Ave & Mi
lwaukee Ave" "University Ave & 57th St" "Broadway & Sheridan Rd"
...
 $ end_station_id    : chr  "125" "260" "423" "256" ...
 $ start_lat         : num  41.9 41.9 41.8 42 41.9 ...
 $ start_lng         : num  -87.6 -87.7 -87.6 -87.7 -87.7 ...
 $ end_lat           : num  41.9 41.9 41.8 42 41.9 ...
 $ end_lng           : num  -87.6 -87.7 -87.6 -87.7 -87.7 ...
 $ member_casual     : chr  "casual" "casual" "casual" "casual" ...
```

```
   ride_id          rideable_type       started_at          ended_a
t
 Length:5136261     Length:5136261     Length:5136261     Length:51
36261
 Class :character   Class :character   Class :character   Class :ch
aracter
 Mode  :character   Mode  :character   Mode  :character   Mode  :ch
aracter




 start_station_name start_station_id   end_station_name   end_stati
on_id
 Length:5136261     Length:5136261     Length:5136261     Length:51
36261
 Class :character   Class :character   Class :character   Class :ch
aracter
 Mode  :character   Mode  :character   Mode  :character   Mode  :ch
aracter



   start_lat       start_lng         end_lat         end_lng
 Min.   :41.64   Min.   :-87.84   Min.   :41.51   Min.   :-88.07
 1st Qu.:41.88   1st Qu.:-87.66   1st Qu.:41.88   1st Qu.:-87.66
 Median :41.90   Median :-87.64   Median :41.90   Median :-87.64
 Mean   :41.90   Mean   :-87.65   Mean   :41.90   Mean   :-87.65
 3rd Qu.:41.93   3rd Qu.:-87.63   3rd Qu.:41.93   3rd Qu.:-87.63
 Max.   :42.08   Max.   :-87.52   Max.   :42.17   Max.   :-87.44
                                  NA's   :4821    NA's   :4821
 member_casual
 Length:5136261
 Class :character
 Mode  :character
```

**Note:**
There are some NA's in *end_lat* and *end_lng*. Also, the *started_at* and *ended_at* values are character, though they should be in date_time format. We'll fix this.
But first, let's check all the columns for empty or NA values and for duplicates.

### Duplicates

In [7]:

```
# Duplicates
print("Duplicates in ride_id:")
sum(duplicated(trips_total_raw$ride_id))
# Recheck
length(unique(trips_total_raw$ride_id)) == nrow(trips_total_raw)
```

```
[1] "Duplicates in ride_id:"
```

209

FALSE

In [8]:

```
# Deleting duplicates
non_duplicated_trips <- trips_total_raw[!duplicated(trips_total_raw$ride_id), ]
```

In [9]:

```
# Check for duplicates in a non-duplicated data frame
print("Duplicates in ride_id:")
sum(duplicated(non_duplicated_trips$ride_id))
length(unique(non_duplicated_trips$ride_id)) == nrow(non_duplicated_trips)
glimpse(non_duplicated_trips)
```

```
[1] "Duplicates in ride_id:"
```

```
0
```

```
TRUE
```

```
Rows: 5,136,052
Columns: 13
$ ride_id            <chr> "ACB6B40CF5B9044C", "DF450C72FD109C01",
 "B6396B54A1…
$ rideable_type      <chr> "electric_bike", "electric_bike", "elect
ric_bike", …
$ started_at         <chr> "2020-10-31 19:39:43", "2020-10-31 23:5
0:08", "2020…
$ ended_at           <chr> "2020-10-31 19:57:12", "2020-11-01 00:0
4:16", "2020…
$ start_station_name <chr> "Lakeview Ave & Fullerton Pkwy", "Southp
ort Ave & W…
$ start_station_id   <chr> "313", "227", "102", "165", "190", "35
9", "313", "1…
$ end_station_name   <chr> "Rush St & Hubbard St", "Kedzie Ave & Mi
lwaukee Ave…
$ end_station_id     <chr> "125", "260", "423", "256", "185", "53",
"125", "31…
$ start_lat          <dbl> 41.92610, 41.94817, 41.77346, 41.95085,
 41.92886, 4…
$ start_lng          <dbl> -87.63898, -87.66391, -87.58537, -87.659
24, -87.663…
$ end_lat            <dbl> 41.89035, 41.92953, 41.79145, 41.95281,
 41.91778, 4…
$ end_lng            <dbl> -87.62607, -87.70782, -87.60005, -87.650
10, -87.691…
$ member_casual      <chr> "casual", "casual", "casual", "casual",
 "casual", "…
```

**NA values**

In [10]:

```
# Checking the NA values in the variables
sum(is.na(non_duplicated_trips))
colSums(is.na(non_duplicated_trips))
```

128092

**ride_id:** 0 **rideable_type:** 0 **started_at:** 0 **ended_at:** 0 **start_station_name:** 0
**start_station_id:** 55839 **end_station_name:** 0 **end_station_id:** 62613 **start_lat:** 0
**start_lng:** 0 **end_lat:** 4820 **end_lng:** 4820 **member_casual:** 0

In [11]:

```
# Let's remove all NA values
trips_cleaned <- drop_na(non_duplicated_trips)
```

In [12]:

```
# Checking the NA values in the variables
sum(is.na(trips_cleaned))
colSums(is.na(trips_cleaned))
```

0

**ride_id:** 0 **rideable_type:** 0 **started_at:** 0 **ended_at:** 0 **start_station_name:** 0
**start_station_id:** 0 **end_station_name:** 0 **end_station_id:** 0 **start_lat:** 0 **start_lng:** 0
**end_lat:** 0 **end_lng:** 0 **member_casual:** 0

In [13]:

```
# What if we lost too much data?
data_loss_percent <- (nrow(non_duplicated_trips)-nrow(trips_cleaned)) / nrow(non
_duplicated_trips) * 100
data_loss_percent # No. The lost data is less than 2%, so we could delete it
```

1.75891910751682

### Get date-time values

- We'll have to work with time data. So let's convert our time columns (started_at and ended_at) from character to date-time.

In [14]:

```
# Use the lubridate package
trips_cleaned$started_at <- ymd_hms(trips_cleaned$started_at)
trips_cleaned$ended_at <- ymd_hms(trips_cleaned$ended_at)
```

In [15]:

```
# Check the new time columns class
class(trips_cleaned$started_at)
class(trips_cleaned$ended_at)
# It's date-time class now
```

'POSIXct' · 'POSIXt'

'POSIXct' · 'POSIXt'

## 6. Adding some valuable columns into our data frame

In [16]:

```
# Counting a ride length in a column
trips_cleaned$ride_length <- (trips_cleaned$ended_at) - (trips_cleaned$started_at)

# Let's calculate day of week, month, year that each ride started - to provide additional opportunities to aggregate the data
# Change language of weekday output
# Sys.setlocale("LC_TIME", "English")

trips_cleaned$day_of_week <- wday(trips_cleaned$started_at, label = TRUE, abbr = FALSE) # day of week
trips_cleaned$month <- month(trips_cleaned$started_at, label = TRUE, abbr = FALSE) # month
trips_cleaned$year <- year(trips_cleaned$started_at) # year

head(trips_cleaned)
```

A data.frame: 6 × 17

| | ride_id | rideable_type | started_at | ended_at | start_station_name | start_stat |
|---|---|---|---|---|---|---|
| | <chr> | <chr> | <dttm> | <dttm> | <chr> | <chr> |
| 1 | ACB6B40CF5B9044C | electric_bike | 2020-10-31 19:39:43 | 2020-10-31 19:57:12 | Lakeview Ave & Fullerton Pkwy | 313 |
| 2 | DF450C72FD109C01 | electric_bike | 2020-10-31 23:50:08 | 2020-11-01 00:04:16 | Southport Ave & Waveland Ave | 227 |
| 3 | B6396B54A15AC0DF | electric_bike | 2020-10-31 23:00:01 | 2020-10-31 23:08:22 | Stony Island Ave & 67th St | 102 |
| 4 | 44A4AEE261B9E854 | electric_bike | 2020-10-31 22:16:43 | 2020-10-31 22:19:35 | Clark St & Grace St | 165 |
| 5 | 10B7DD76A6A2EB95 | electric_bike | 2020-10-31 19:38:19 | 2020-10-31 19:54:32 | Southport Ave & Wrightwood Ave | 190 |
| 6 | DA6C3759660133DA | electric_bike | 2020-10-29 17:38:04 | 2020-10-29 17:45:43 | Larrabee St & Division St | 359 |

In [17]:

```
summary(trips_cleaned)
```

```
   ride_id          rideable_type        started_at
 Length:5045713     Length:5045713     Min.   :2020-10-01 00:00:0
6
 Class :character   Class :character   1st Qu.:2021-04-17 22:57:4
8
 Mode  :character   Mode  :character   Median :2021-06-23 16:21:1
2
                                       Mean   :2021-05-29 12:21:4
1
                                       3rd Qu.:2021-08-12 19:44:2
8
                                       Max.   :2021-09-30 23:59:4
8

    ended_at                  start_station_name start_station_i
d
 Min.   :2020-10-01 00:05:09   Length:5045713     Length:5045713
 1st Qu.:2021-04-17 23:22:20   Class :character   Class :characte
r
 Median :2021-06-23 16:42:02   Mode  :character   Mode  :characte
r
 Mean   :2021-05-29 12:42:48
 3rd Qu.:2021-08-12 20:05:22
 Max.   :2021-10-01 18:41:34

 end_station_name   end_station_id       start_lat       start_ln
g
 Length:5045713     Length:5045713     Min.   :41.64   Min.   :-8
7.84
 Class :character   Class :character   1st Qu.:41.88   1st Qu.:-8
7.66
 Mode  :character   Mode  :character   Median :41.90   Median :-8
7.64
                                       Mean   :41.90   Mean   :-8
7.65
                                       3rd Qu.:41.93   3rd Qu.:-8
7.63
                                       Max.   :42.07   Max.   :-8
7.52

    end_lat         end_lng       member_casual      ride_length
 Min.   :41.51   Min.   :-88.07   Length:5045713     Length:50457
13
```

```
 1st Qu.:41.88   1st Qu.:-87.66   Class :character   Class :difft
ime
 Median :41.90   Median :-87.64   Mode  :character   Mode  :numer
ic
 Mean   :41.90   Mean   :-87.65
 3rd Qu.:41.93   3rd Qu.:-87.63
 Max.   :42.17   Max.   :-87.49

   day_of_week           month              year
 Sunday   :778107   July     : 821679   Min.   :2020
 Monday   :631486   August   : 803646   1st Qu.:2021
 Tuesday  :647881   September: 755552   Median :2021
 Wednesday:668945   June     : 728878   Mean   :2021
 Thursday :681901   May      : 531181   3rd Qu.:2021
 Friday   :730670   October  : 339303   Max.   :2021
 Saturday :906723   (Other)  :1065474
```

## 7. Organizing the data

In [18]:

```
# First, let's convert our data frame into a tibble to simplify it's visualization
trips <- as_tibble(trips_cleaned)
# trips # oops, it doesn't work in Kaggle
head(trips, 5)
```

A tibble: 5 × 17

| ride_id | rideable_type | started_at | ended_at | start_station_name | start_station_i |
|---|---|---|---|---|---|
| <chr> | <chr> | <dttm> | <dttm> | <chr> | <chr> |
| ACB6B40CF5B9044C | electric_bike | 2020-10-31 19:39:43 | 2020-10-31 19:57:12 | Lakeview Ave & Fullerton Pkwy | 313 |
| DF450C72FD109C01 | electric_bike | 2020-10-31 23:50:08 | 2020-11-01 00:04:16 | Southport Ave & Waveland Ave | 227 |
| B6396B54A15AC0DF | electric_bike | 2020-10-31 23:00:01 | 2020-10-31 23:08:22 | Stony Island Ave & 67th St | 102 |
| 44A4AEE261B9E854 | electric_bike | 2020-10-31 22:16:43 | 2020-10-31 22:19:35 | Clark St & Grace St | 165 |
| 10B7DD76A6A2EB95 | electric_bike | 2020-10-31 19:38:19 | 2020-10-31 19:54:32 | Southport Ave & Wrightwood Ave | 190 |

In [19]:

```
# Let's arrange our data to find some outliers
trips %>%
  arrange(ride_length) %>%
  head(3)

# Count negative values in ride_length column
sum(trips$ride_length < 0)
```

A tibble: 3 × 17

| ride_id | rideable_type | started_at | ended_at | start_station_name | start_station_i |
|---|---|---|---|---|---|
| <chr> | <chr> | <dttm> | <dttm> | <chr> | <chr> |
| 3ED2B8BCE6A914EF | electric_bike | 2020-12-15 12:10:14 | 2020-11-25 08:00:16 | Wells St & Concord Ln | TA13080000 |
| 6DF34F98F5DB335F | electric_bike | 2020-12-15 11:35:39 | 2020-11-25 07:40:56 | Sheffield Ave & Willow St | TA13060000 |
| 20D609100DF8A71C | electric_bike | 2020-12-15 12:12:09 | 2020-11-25 09:02:39 | | |

3075

In [20]:

```
# There are a lot of negative values. These are errors. Also, I suppose to delete
 too short rides - let's say less than 5 secs, inclusive.
trips_filtered <- trips %>%
                    filter(ride_length > 5)

# Count wrong values in ride_length column
sum(trips_filtered$ride_length < 5)

head(trips_filtered, 5)
```

0

A tibble: 5 × 17

| ride_id | rideable_type | started_at | ended_at | start_station_name | start_station_i |
|---|---|---|---|---|---|
| <chr> | <chr> | <dttm> | <dttm> | <chr> | <chr> |
| ACB6B40CF5B9044C | electric_bike | 2020-10-31 19:39:43 | 2020-10-31 19:57:12 | Lakeview Ave & Fullerton Pkwy | 313 |
| DF450C72FD109C01 | electric_bike | 2020-10-31 23:50:08 | 2020-11-01 00:04:16 | Southport Ave & Waveland Ave | 227 |
| B6396B54A15AC0DF | electric_bike | 2020-10-31 23:00:01 | 2020-10-31 23:08:22 | Stony Island Ave & 67th St | 102 |
| 44A4AEE261B9E854 | electric_bike | 2020-10-31 22:16:43 | 2020-10-31 22:19:35 | Clark St & Grace St | 165 |
| 10B7DD76A6A2EB95 | electric_bike | 2020-10-31 19:38:19 | 2020-10-31 19:54:32 | Southport Ave & Wrightwood Ave | 190 |

### Simplifiying my typing work by renaming the data frame

In [21]:

```
t_f <- trips_filtered
```

**Continue organizing the table**

I've noticed some suspicious station names contained "TESTING". I assume, these are test rides by Cyclistic's specialists. We won't include this in our analysis.

In [22]:

```
# Finding stations with "TEST" in their names
trips_tested <- t_f %>%
  filter(grepl("TEST", start_station_name) | grepl("TEST", end_station_name))

glimpse(trips_tested) # we have 288 test trips. Let's remove them from our table
```

```
Rows: 288
Columns: 17
$ ride_id            <chr> "BB40C8C2F2D52384", "3D835403DB92934E",
 "B828DA986F…
$ rideable_type      <chr> "electric_bike", "electric_bike", "elect
ric_bike", …
$ started_at         <dttm> 2020-10-23 14:48:56, 2020-10-06 14:41:5
4, 2020-10-…
$ ended_at           <dttm> 2020-10-23 14:49:23, 2020-10-06 14:42:0
0, 2020-10-…
$ start_station_name <chr> "WATSON TESTING - DIVVY", "WATSON TESTIN
G - DIVVY",…
$ start_station_id   <chr> "676", "676", "676", "676", "676", "67
6", "676", "6…
$ end_station_name   <chr> "WATSON TESTING - DIVVY", "WATSON TESTIN
G - DIVVY",…
$ end_station_id     <chr> "676", "676", "676", "676", "676", "67
6", "676", "6…
$ start_lat          <dbl> 41.89476, 41.89475, 41.89477, 41.89475,
 41.89474, 4…
$ start_lng          <dbl> -87.73085, -87.73084, -87.73084, -87.730
88, -87.730…
$ end_lat            <dbl> 41.89476, 41.89475, 41.89476, 41.89474,
 41.89473, 4…
$ end_lng            <dbl> -87.73086, -87.73085, -87.73087, -87.730
88, -87.730…
$ member_casual      <chr> "casual", "casual", "casual", "casual",
 "casual", "…
$ ride_length        <drtn> 27 secs, 6 secs, 6 secs, 6 secs, 8 sec
s, 8 secs, 6…
$ day_of_week        <ord> Friday, Tuesday, Tuesday, Tuesday, Wedne
sday, Wedne…
$ month              <ord> October, October, October, October, Octo
ber, Octobe…
$ year               <dbl> 2020, 2020, 2020, 2020, 2020, 2020, 202
0, 2020, 202…
```

In [23]:

```
# Remove test trips from our table
t_f_v2 <- t_f %>%
  filter(!(grepl("TEST", start_station_name) | grepl("TEST", end_station_name)))

# Check for test stations
trips_tested_v2 <- t_f_v2 %>%
  filter(grepl("TEST", start_station_name) | grepl("TEST", end_station_name))

glimpse(trips_tested_v2) # 0
```

```
Rows: 0
Columns: 17
$ ride_id            <chr>
$ rideable_type      <chr>
$ started_at         <dttm>
$ ended_at           <dttm>
$ start_station_name <chr>
$ start_station_id   <chr>
$ end_station_name   <chr>
$ end_station_id     <chr>
$ start_lat          <dbl>
$ start_lng          <dbl>
$ end_lat            <dbl>
$ end_lng            <dbl>
$ member_casual      <chr>
$ ride_length        <drtn>  secs
$ day_of_week        <ord>
$ month              <ord>
$ year               <dbl>
```

### Removing errors

Also, I suppose to remove trips that are longer than one day (24×60×60), because they're probably an error and not representative.

In [24]:

```
# Deleting too long trips
t_f_v2 <- t_f_v2[!(t_f_v2$ride_length > (24*60*60)), ]

head(sort(t_f_v2$ride_length, decreasing = TRUE), n=50) # check. (A day equals t
o 86400 seconds)
```

```
Time differences in secs
 [1] 86394 86391 86383 86362 86313 86279 86274 86247 86238 86225 86
168 86145
[13] 86121 86103 86097 86086 86086 86043 86024 86018 86010 85979 85
979 85959
[25] 85958 85942 85932 85925 85879 85850 85672 85658 85654 85627 85
609 85601
[37] 85596 85545 85500 85486 85482 85474 85459 85448 85439 85408 85
406 85406
[49] 85393 85335
```

# PART 4, 5 - ANALYZE, SHARE

Now we have a summary file with clean data. My goal is to identify any surprises, trends or relationships in the data and get some valuable insights that will help the stakeholders' to make decisions.

**Descriptive analysis of ride_length**

In [25]:

```
print("Average ride duration:")
mean(t_f_v2$ride_length) # straight average (total ride length / rides) - 1220.71
5 (s)
print("Median of ride duration:")
median(t_f_v2$ride_length) # midpoint number in the ascending array of ride lengt
hs - 759 (s)
print("Shortest ride duration:")
min(t_f_v2$ride_length)  #shortest ride - 6 (s)
print("Longest ride duration:")
max(t_f_v2$ride_length) #longest ride - 86394 (s)
```

```
[1] "Average ride duration:"


Time difference of 1220.715 secs

[1] "Median of ride duration:"

Time difference of 759 secs

[1] "Shortest ride duration:"

Time difference of 6 secs

[1] "Longest ride duration:"

Time difference of 86394 secs
```

### Identifying how casual customers and members use bikes differently

In [26]:

```
# Setting my plot theme
plot_theme = theme(
    plot.title = element_text(size=20, face = 'bold'),
    plot.subtitle = element_text(size=10, color = 'gray', face = 'bold'),
    plot.caption = element_text(size=12, color = 'darkgray', face = 'bold'),
    axis.text.x = element_text(size=15),
    axis.text.y = element_text(size=15),
    axis.title.x = element_text(size=18),
    axis.title.y = element_text(size=18),
    strip.text.x = element_text(size=10),
    strip.text.y = element_text(size=10),
    legend.title = element_text(size=18),
    legend.text = element_text(size=16)
)

options(repr.plot.width = 12, repr.plot.height = 10)
```

- **Number of trips by type of customer**

In [27]:

```
t_f_v2 %>%
  group_by(member_casual) %>%
  summarise(number_of_trips = n()) %>%
  ggplot(aes(x=member_casual, y=number_of_trips, fill=member_casual)) +
  geom_col(position = "dodge") +
  labs(title = "Total Trips: Members vs. Casual Riders",
       x = "Type of Rider", y = "Number", fill = "Rider Type",
       caption = "source: Motivate International Inc.") +
  scale_y_continuous(label=comma) +
  geom_text(aes(label=comma(number_of_trips)), position = position_stack(vjust =
0.95), size = 5) +
  plot_theme
```

**Analyze:**

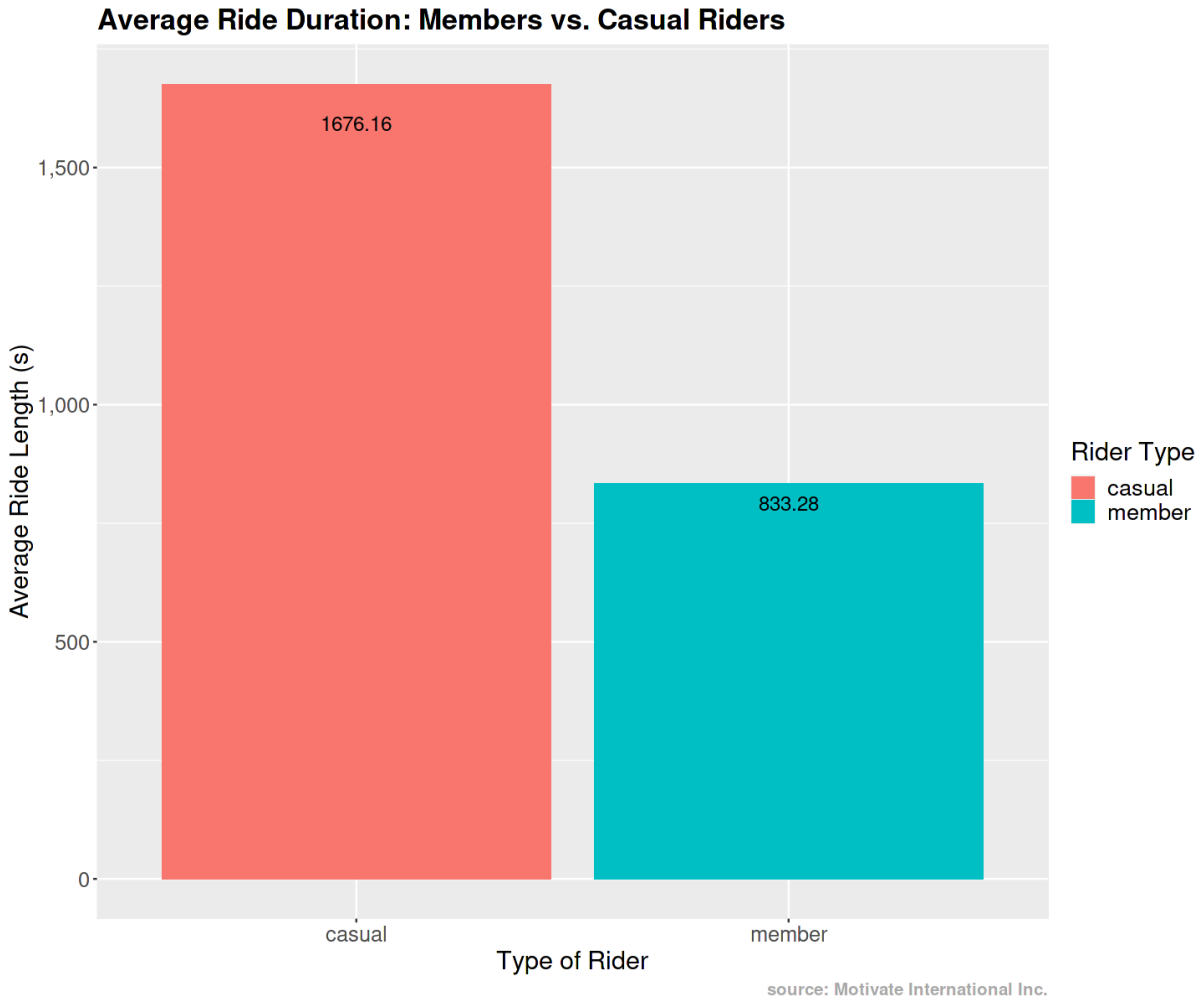
Annual members do more rides (in **1.2** times more) in total.

- **Average trips duration by type of client**

In [28]:

```
t_f_v2 %>%
  group_by(member_casual) %>%
  summarise(average_ride_length = mean(ride_length)) %>%
  ggplot(aes(x=member_casual, y=average_ride_length, fill=member_casual)) +
  geom_col(position = "dodge") +
  labs(title = "Average Ride Duration: Members vs. Casual Riders",
       x = "Type of Rider", y = "Average Ride Length (s)", fill = "Rider Type",
       caption = "source: Motivate International Inc.") +
  scale_y_continuous(label=comma) +
  geom_text(aes(label=round(average_ride_length,2)), position = position_stack(v
just = 0.95), size = 5) +
  plot_theme
```

```
Don't know how to automatically pick scale for object of type difft
ime. Defaulting to continuous.
```



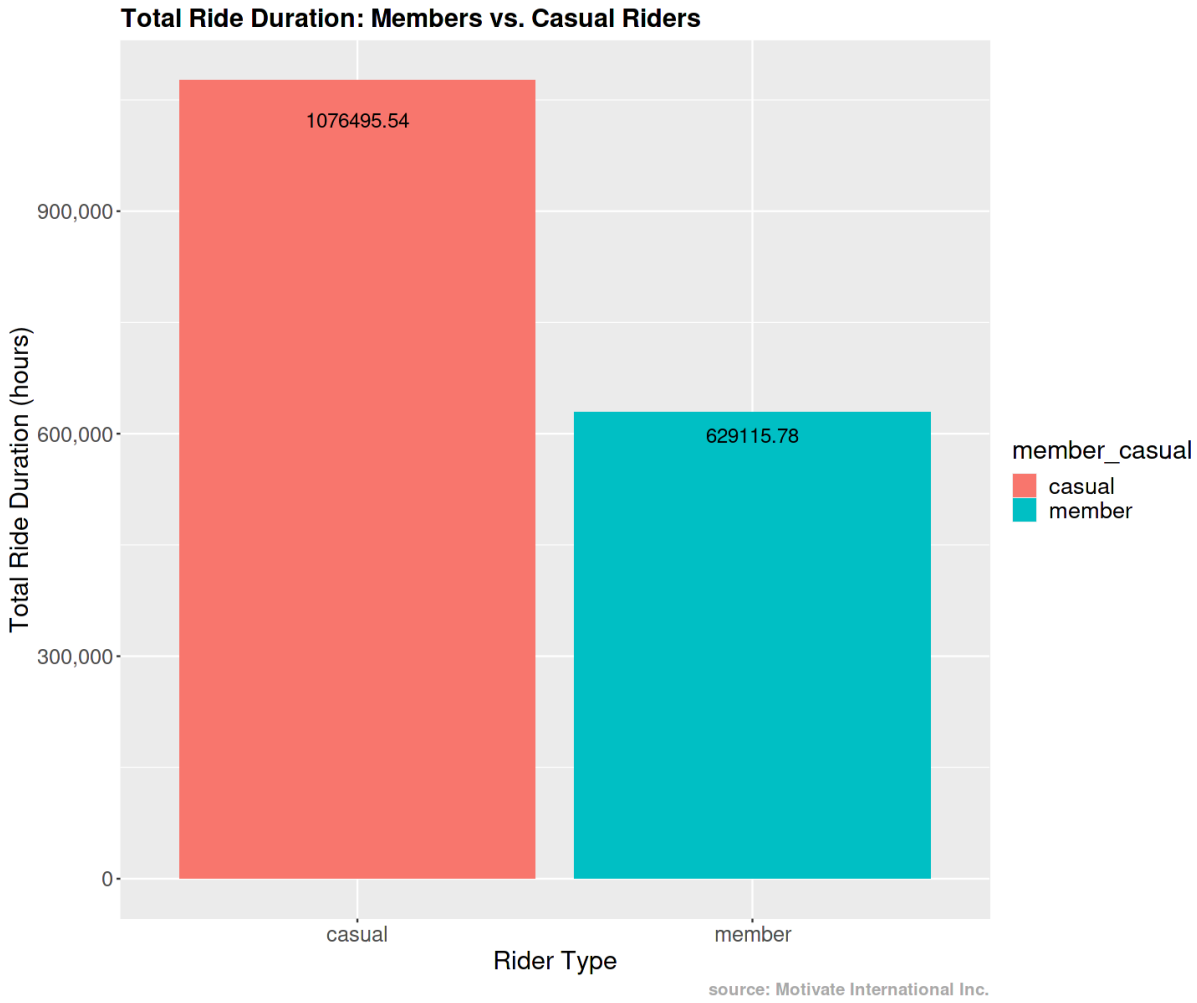
**Analyze:**
Casual users ride duration is in **2** more times (1676 / 833) than members' duration in average, though members do more rides in total.

- **Total ride duration by type of riders**

In [29]:

```
t_f_v2 %>%
  group_by(member_casual) %>%
  summarise(sum_ride_duration = sum(ride_length)/60/60) %>%
  ggplot(aes(x = member_casual, y = sum_ride_duration, fill=member_casual)) +
  geom_col(position = "dodge") +
  scale_y_continuous(labels = comma) +
  labs(x = "Rider Type", y = "Total Ride Duration (hours)",
       title = "Total Ride Duration: Members vs. Casual Riders",
       caption = "source: Motivate International Inc.") +
  geom_text(aes(label=round(sum_ride_duration,2)), position = position_stack(vju
st = 0.95), size = 5) +
  theme(
    plot.title = element_text(size=18, face = 'bold'),
    plot.caption = element_text(size=12, color = 'darkgray', face = 'bold'),
    axis.text.x = element_text(size=15),
    axis.text.y = element_text(size=15),
    axis.title.x = element_text(size=18),
    axis.title.y = element_text(size=18),
    legend.title = element_text(size=18),
    legend.text = element_text(size=16),
  )
```

```
Don't know how to automatically pick scale for object of type difft
ime. Defaulting to continuous.
```



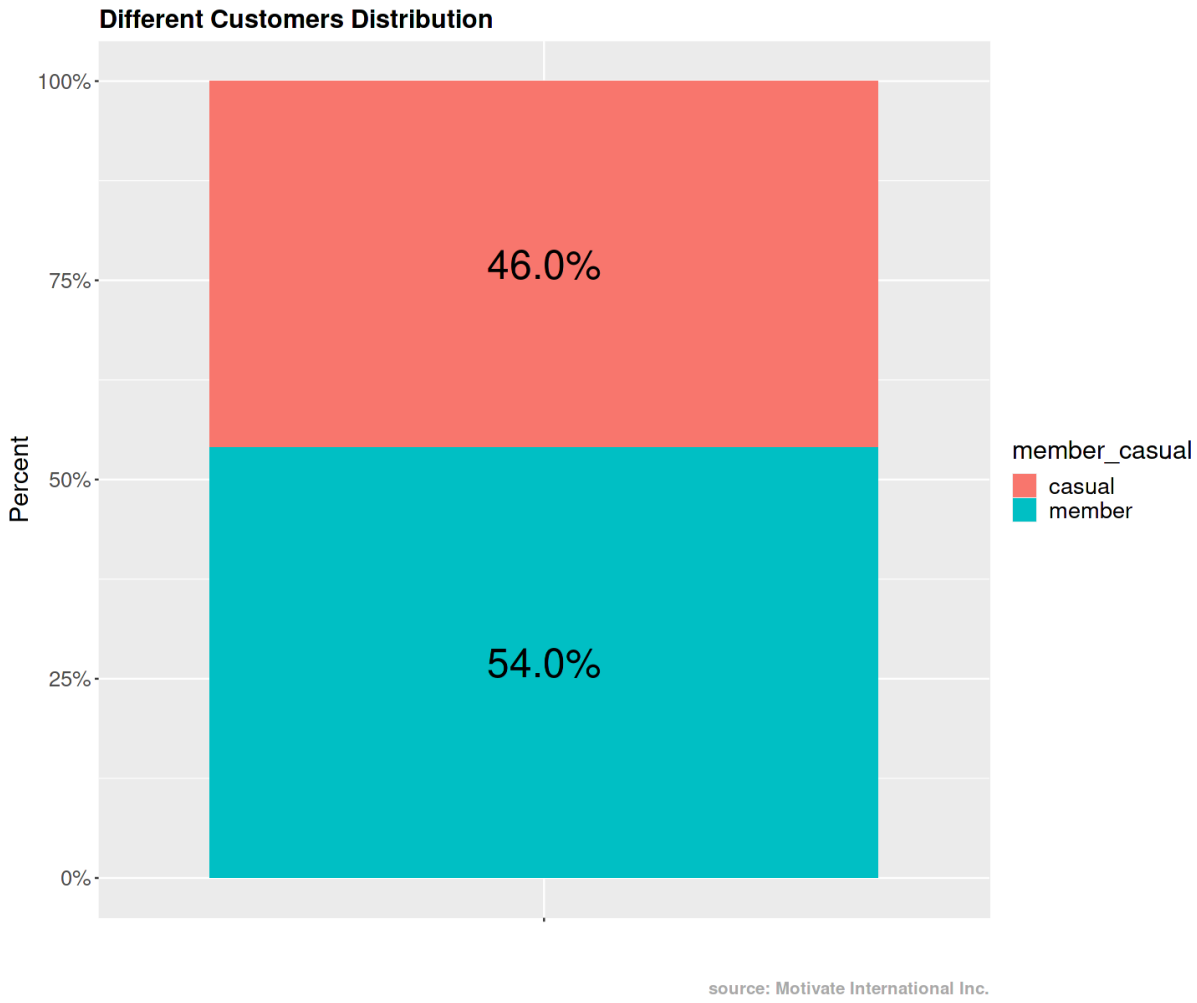

**Analyze:**

We can see that casual riders have a way bigger (in **1.71** times) total ride duration than members.
It's interesting, considering that annual members are much more profitable (according to the financial team) than casual riders.

- **How many different customers do we have?**

In [30]:

```
t_f_v2 %>%
  group_by(member_casual) %>%
  summarize(total_by_type = n()) %>%
  mutate(overall_total = sum(total_by_type)) %>%
  group_by(member_casual) %>%
  summarize(percent_total = total_by_type/overall_total) %>%
  ggplot(aes(fill=member_casual, y=percent_total, x="")) +
  geom_bar(position="fill", stat="identity") +
  geom_text(aes(label = percent(percent_total)),
            position = position_stack(vjust = 0.5), size = 10) +
  labs(x = "", y = "Percent",
       title = "Different Customers Distribution",
       caption = "source: Motivate International Inc.") +
  theme(
    plot.title = element_text(size=18, face = 'bold'),
    plot.caption = element_text(size=12, color = 'darkgray', face = 'bold'),
    axis.text.x = element_text(size=15),
    axis.text.y = element_text(size=15),
    axis.title.x = element_text(size=18),
    axis.title.y = element_text(size=18),
    legend.title = element_text(size=18),
    legend.text = element_text(size=16),
  ) +
  scale_y_continuous(labels = percent)
```


Different Customers Distribution
100%
75%
50%
25%
0%
Percent
46.0%
54.0%
member_casual
casual
member
source: Motivate International Inc.

**Analyze:**

There are more members than casual customers.

We can see that **54%** of customers have an annual subscription and **46%** - use single-ride or full-day passes.

Though there's almost 50% of casual customers in total, they don't generate enough revenue.

## Analyze by day of week

In [31]:

```
# First, let's order our weekdays in a normal order
t_f_v2$day_of_week <- ordered(t_f_v2$day_of_week, levels=c("Monday", "Tuesday",
"Wednesday", "Thursday", "Friday", "Saturday", "Sunday"))
```

- **Average ride time and number of rides by client type during a week**

In [32]:

```
t_f_v2 %>%
  group_by(member_casual, day_of_week) %>%
  summarise(num_of_rides = n(),
            average_ride_time = format(round(mean(ride_length), 4), nsmall = 4),
.groups = 'drop') %>%
  arrange(day_of_week)
```

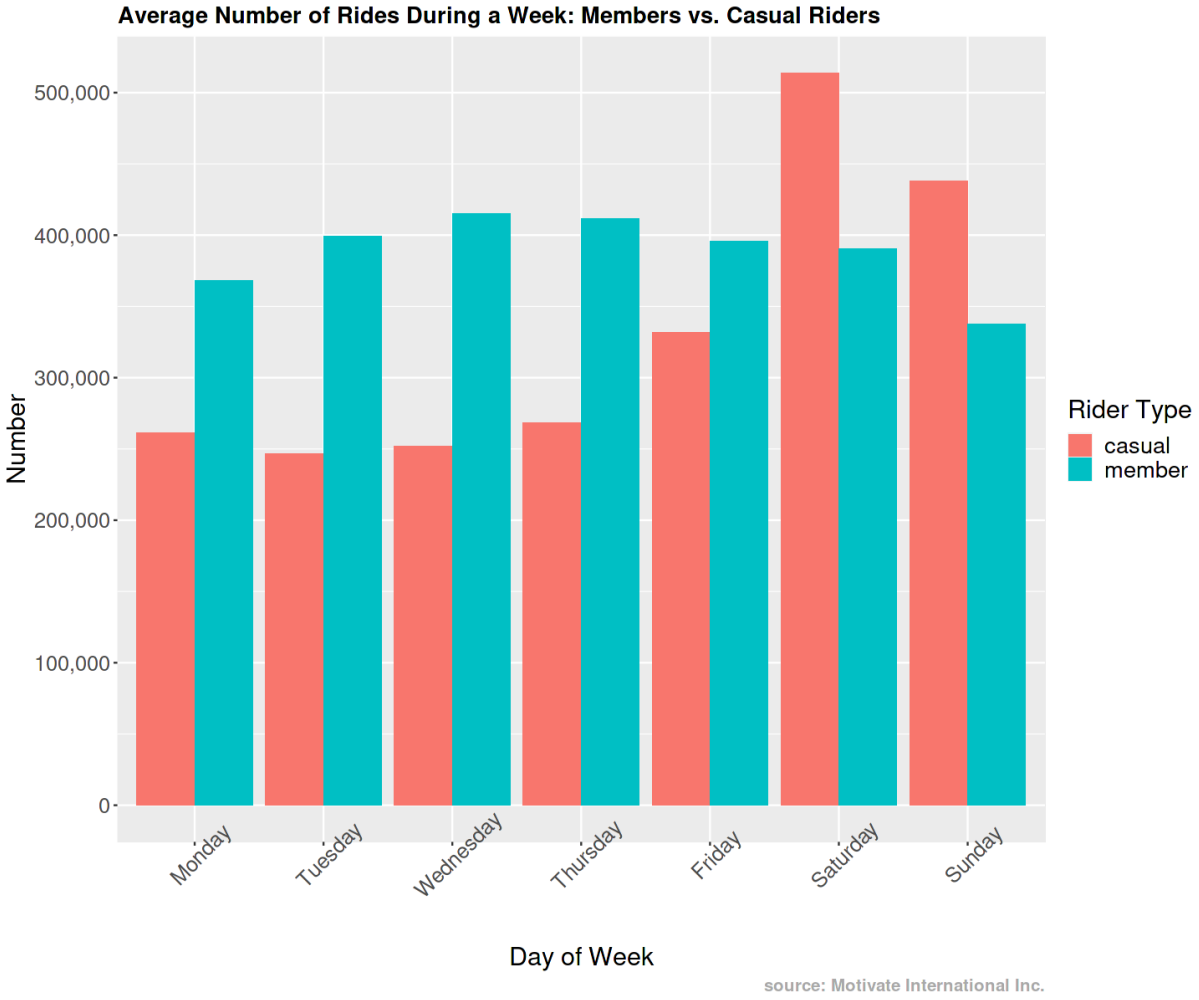
A tibble: 14 × 4

| member_casual | day_of_week | num_of_rides | average_ride_time |
|---|---|---|---|
| <chr> | <ord> | <int> | <chr> |
| casual | Monday | 261597 | 1691.4258 secs |
| member | Monday | 368058 | 804.4365 secs |
| casual | Tuesday | 246544 | 1538.8177 secs |
| member | Tuesday | 399309 | 785.0875 secs |
| casual | Wednesday | 252014 | 1461.4784 secs |
| member | Wednesday | 414919 | 790.5410 secs |
| casual | Thursday | 268387 | 1443.1196 secs |
| member | Thursday | 411493 | 783.3304 secs |
| casual | Friday | 331810 | 1564.8179 secs |
| member | Friday | 396013 | 818.7206 secs |
| casual | Saturday | 513648 | 1825.2452 secs |
| member | Saturday | 390333 | 926.0501 secs |
| casual | Sunday | 438056 | 1920.1710 secs |
| member | Sunday | 337821 | 944.9018 secs |

- **Average Number of Rides During a Week**

In [33]:

```
# Visualize the number of rides by day and type of rider
t_f_v2 %>%
  group_by(member_casual, day_of_week) %>%
  summarise(num_of_rides = n(), .groups = 'drop') %>%
  arrange(day_of_week)  %>%
  ggplot(aes(x = day_of_week, y = num_of_rides, fill = member_casual)) +
  geom_col(position = "dodge") +
  scale_y_continuous(labels = comma) + # changing y_axis numbers format
  labs(title = "Average Number of Rides During a Week: Members vs. Casual Rider
s",
       x = "Day of Week", y = "Number", fill = "Rider Type",
       caption = "source: Motivate International Inc.") +
  theme(
    plot.title = element_text(size=16, face = 'bold'),
    plot.caption = element_text(size=12, color = 'darkgray', face = 'bold'),
    axis.text.x = element_text(size=15),
    axis.text.y = element_text(size=15),
    axis.title.x = element_text(size=18),
    axis.title.y = element_text(size=18),
    legend.title = element_text(size=18),
    legend.text = element_text(size=16)
) +
  theme(axis.text.x = element_text(angle=45))
```




**Analyze:**

We can see that casual riders use bikes mostly on weekends - what supports a theory that they use the Cyclistic service for leisure or observation trips (or exercising).

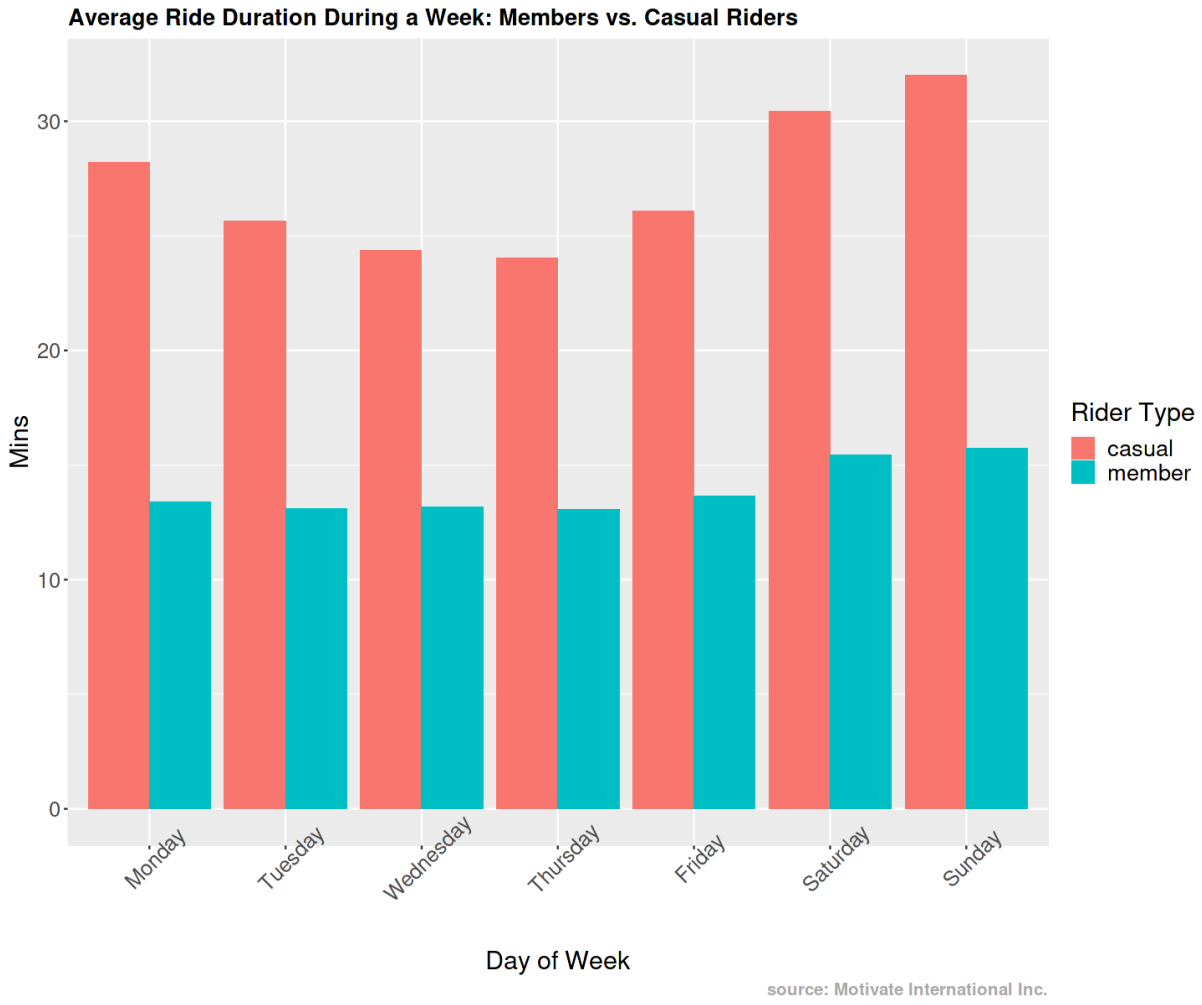
And members use bikes almost equally throughout a week, but with peaks on workdays.

- **Visualize average ride duration by day and type of rider**

In [34]:

```
t_f_v2 %>%
  group_by(member_casual, day_of_week) %>%
  summarise(avg_ride_time = (mean(ride_length)/60), .groups = 'drop') %>%
  arrange(day_of_week)  %>%
  ggplot(aes(x = day_of_week, y = avg_ride_time, fill = member_casual)) +
  geom_col(position = "dodge") +
  labs(title = "Average Ride Duration During a Week: Members vs. Casual Riders",
       x = "Day of Week", y = "Mins", fill = "Rider Type",
       caption = "source: Motivate International Inc.") +
  theme(
    plot.title = element_text(size=16, face = 'bold'),
    plot.caption = element_text(size=12, color = 'darkgray', face = 'bold'),
    axis.text.x = element_text(size=15),
    axis.text.y = element_text(size=15),
    axis.title.x = element_text(size=18),
    axis.title.y = element_text(size=18),
    legend.title = element_text(size=18),
    legend.text = element_text(size=16)
    ) +
  theme(axis.text.x = element_text(angle=45))
```

```
Don't know how to automatically pick scale for object of type difft
ime. Defaulting to continuous.
```



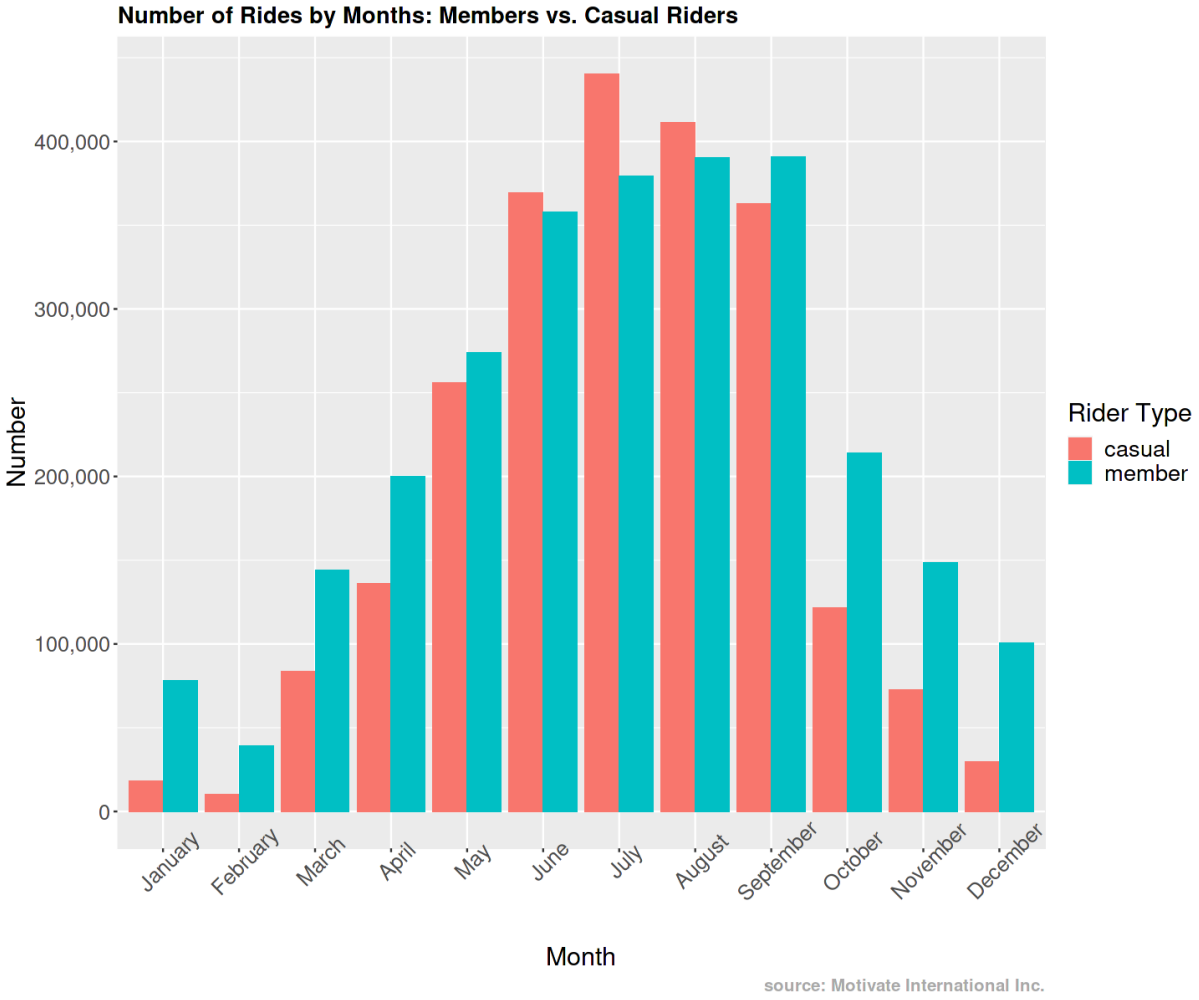

**Analyze:**
We can see that average ride duration of casual riders is much higher than members'. This reinforces hypothesis that casual riders use bikes mostly for **leisure trips** (or exercising / tourism) and members use bikes mostly for **practical purposes** (e.g. get to work).

## Analyze by month. Seasonal trends

- **Number of rides by month**

In [35]:

```
t_f_v2 %>%
  group_by(member_casual, month) %>%
  summarise(num_of_rides = n(), .groups = 'drop') %>%
  arrange(month) %>%
  ggplot(aes(x = month, y = num_of_rides, fill = member_casual)) +
  geom_col(position = "dodge") +
  labs(title = "Number of Rides by Months: Members vs. Casual Riders",
       x = "Month", y = "Number", fill = "Rider Type",
       caption = "source: Motivate International Inc.") +
  scale_y_continuous(labels = comma) +
  theme(axis.text.x = element_text(angle=45)) +
  theme(
    plot.title = element_text(size=16, face = 'bold'),
    plot.caption = element_text(size=12, color = 'darkgray', face = 'bold'),
    axis.text.x = element_text(size=15),
    axis.text.y = element_text(size=15),
    axis.title.x = element_text(size=18),
    axis.title.y = element_text(size=18),
    legend.title = element_text(size=18),
    legend.text = element_text(size=16)
  )
```




**Analyze:**

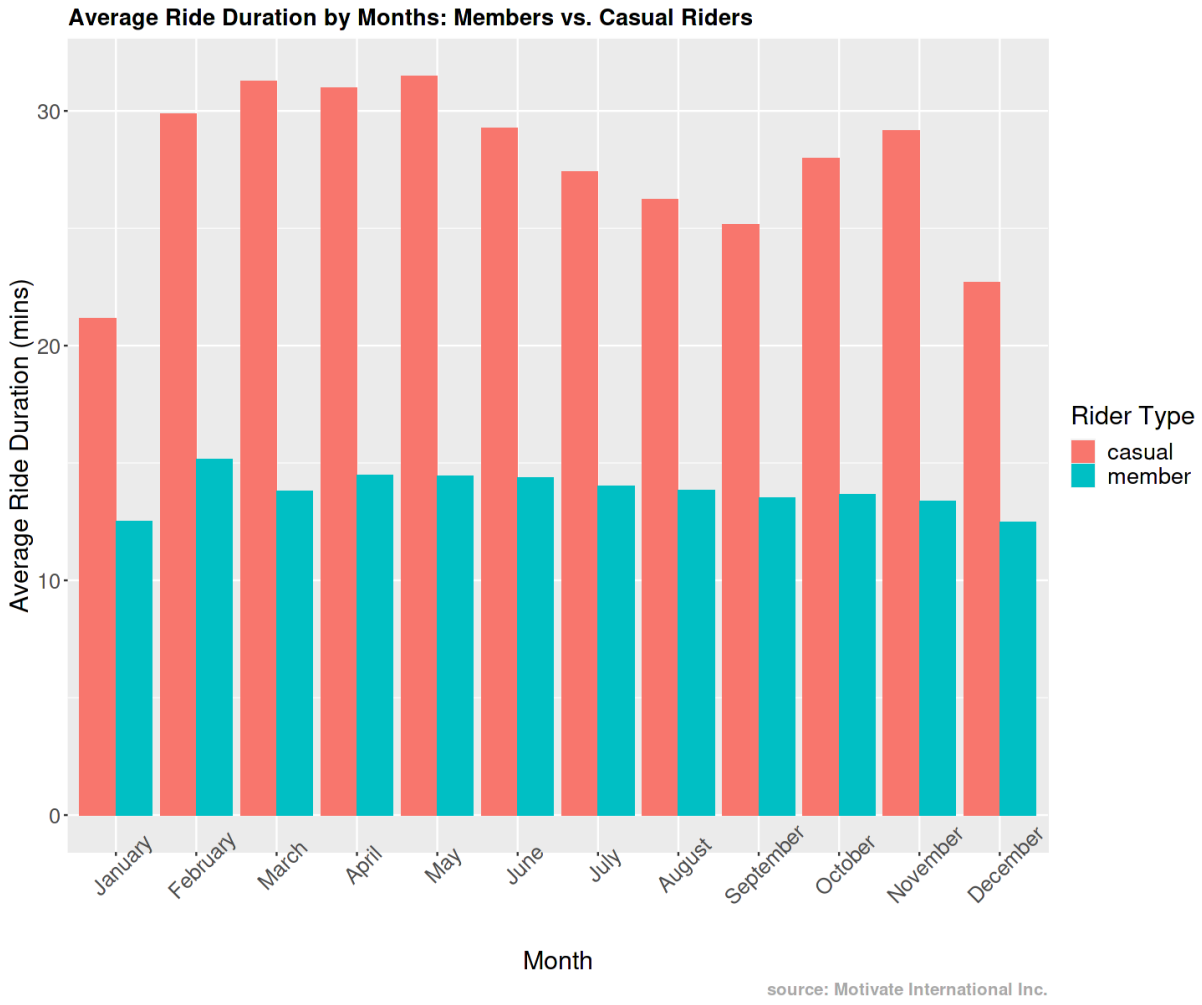
We see the biggest values in **summer** (the peak is in July). Thus, it's the best time to start our marketing campaign.

- **Average ride duration by month and type of rider**

In [36]:

```
t_f_v2 %>%
  group_by(member_casual, month) %>%
  summarise(avg_ride_time = (mean(ride_length)/60), .groups = 'drop') %>%
  arrange(month) %>%
  ggplot(aes(x = month, y = avg_ride_time, fill = member_casual)) +
  geom_col(position = "dodge") +
  labs(title = "Average Ride Duration by Months: Members vs. Casual Riders",
       x = "Month", y = "Average Ride Duration (mins)", fill = "Rider Type",
       caption = "source: Motivate International Inc.") +
  theme(axis.text.x = element_text(angle=45)) +
  theme(
    plot.title = element_text(size=16, face = 'bold'),
    plot.caption = element_text(size=12, color = 'darkgray', face = 'bold'),
    axis.text.x = element_text(size=15),
    axis.text.y = element_text(size=15),
    axis.title.x = element_text(size=18),
    axis.title.y = element_text(size=18),
    legend.title = element_text(size=18),
    legend.text = element_text(size=16)
  )
```

```
Don't know how to automatically pick scale for object of type difft
ime. Defaulting to continuous.
```




**Analyze:**
Members' ride duration stays almost equal throughout a year with a slight decline in December and January. Interesting, that the peak month by ride duration for members is **February**.

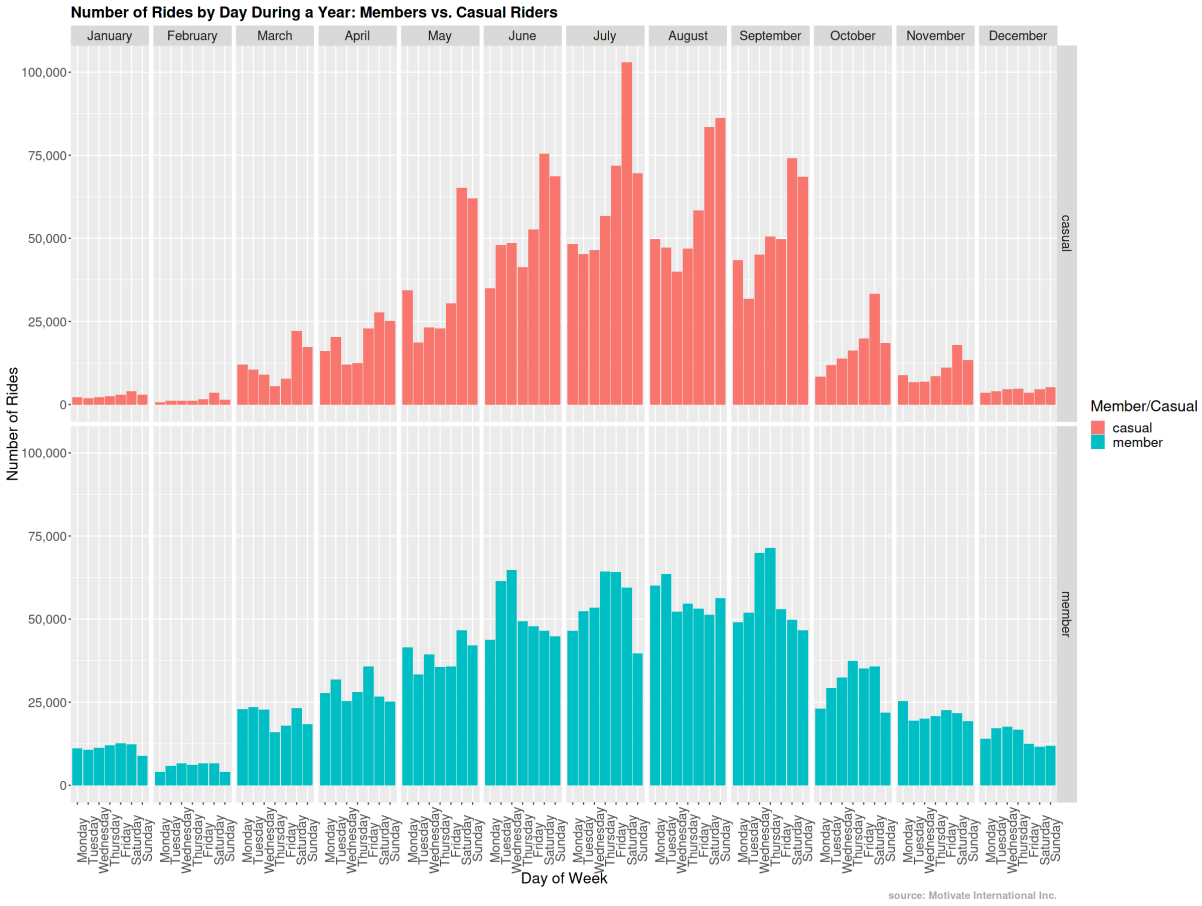
The peaks for casual customers is in **springtime** (longer than 30 mins). Summer weather is too hot for long trips. Later duration growth in autumn confirms it.

- **Number of rides by day of week during a year**

In [37]:

```
options(repr.plot.width = 20, repr.plot.height = 15)

t_f_v2 %>%
  group_by(month, day_of_week, member_casual) %>%
  summarise(num_of_rides = n(), .groups = 'drop') %>%
  drop_na() %>%
  ggplot(aes(x = day_of_week, y = num_of_rides, fill = member_casual)) +
  geom_col(position = "dodge") +
  scale_y_continuous(labels = comma) +
  facet_grid(member_casual~month) +
  labs(x = "Day of Week", y = "Number of Rides", fill = "Member/Casual",
       title = "Number of Rides by Day During a Year: Members vs. Casual Riders"
, fill = 'Member/Casual',
       caption = "source: Motivate International Inc.") +
  theme(axis.text.x = element_text(angle = 90)) +
  theme(
    plot.title = element_text(size=18, face = 'bold'),
    plot.caption = element_text(size=12, color = 'darkgray', face = 'bold'),
    axis.text.x = element_text(size=15),
    axis.text.y = element_text(size=15),
    axis.title.x = element_text(size=18),
    axis.title.y = element_text(size=18),
    legend.title = element_text(size=18),
    legend.text = element_text(size=16),
    strip.text.x = element_text(size=15),
    strip.text.y = element_text(size=15)
  )
```




**Analyze:**

Though total number of rides by annual members is bigger than casual riders, random users need more bikes on **weekends** from **may to september**.

## Analyze by hour

- **Average number of rides by hour**

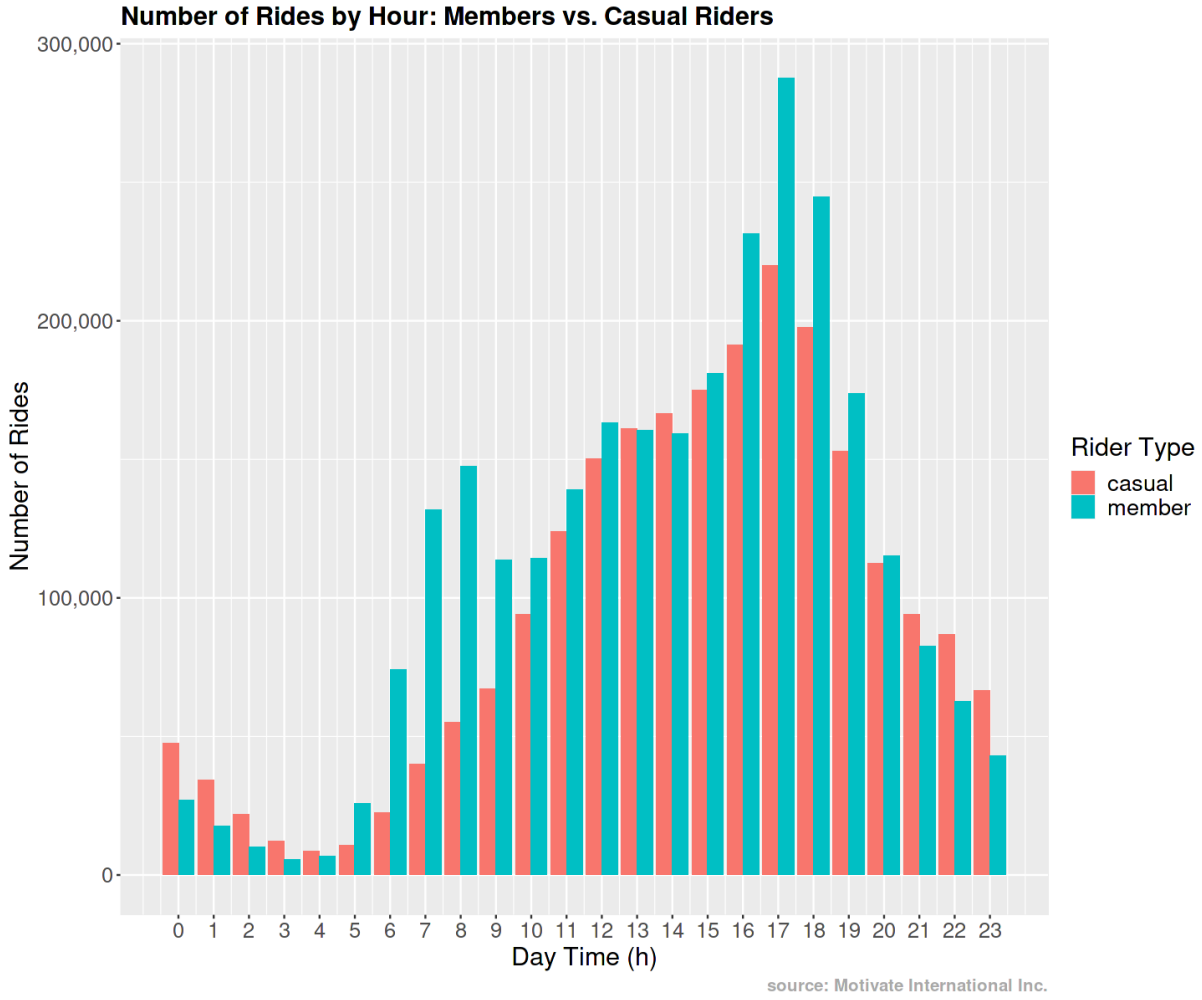
In [38]:

```
# Create hour column in our df
t_f_v2$start_hour <- hour(t_f_v2$started_at)
t_f_v2$end_hour <- hour(t_f_v2$ended_at)
```

In [39]:

```
options(repr.plot.width = 12, repr.plot.height = 10)

t_f_v2 %>%
  group_by(member_casual, start_hour) %>%
  summarise(num_of_rides = n(), .groups = 'drop') %>%
  arrange(start_hour) %>%
  ggplot(aes(x=start_hour, y=num_of_rides, fill=member_casual)) +
  geom_col(position = "dodge") +
  labs(title = "Number of Rides by Hour: Members vs. Casual Riders",
       x = "Day Time (h)", y = "Number of Rides", fill = "Rider Type",
       caption = "source: Motivate International Inc.") +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(breaks = c(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23)) +
  theme(
    plot.title = element_text(size=18, face = 'bold'),
    plot.caption = element_text(size=12, color = 'darkgray', face = 'bold'),
    axis.text.x = element_text(size=15),
    axis.text.y = element_text(size=15),
    axis.title.x = element_text(size=18),
    axis.title.y = element_text(size=18),
    legend.title = element_text(size=18),
    legend.text = element_text(size=16),
  )
```




**Analyze:**

Bike usage rises near 06:00-07:00 for members (but not for casual riders). Both of them have peaks between **15:00-18:00**.

Casual riders peaks (more than **150,000**) start after 12:00 and least until 19:00. At evening and night time casual riders use bikes more than members.

- **Average ride duration by hour**

In [40]:

```
t_f_v2 %>%
  group_by(member_casual, start_hour) %>%
  summarise(avg_ride_time = (mean(ride_length)/60), .groups = 'drop') %>%
  arrange(start_hour) %>%
  ggplot(aes(x=start_hour, y=avg_ride_time, fill=member_casual)) +
  geom_col(position = "dodge") +
  labs(x = "Day Time (h)", y = "Average Ride Duration (mins)", fill = "Rider Typ
e",
       title = "Average Ride Duration by Hour: Members vs. Casual Riders",
       caption = "source: Motivate International Inc.") +
  scale_x_continuous(breaks = c(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,1
9,20,21,22,23)) +
  theme(
    plot.title = element_text(size=18, face = 'bold'),
    plot.caption = element_text(size=12, color = 'darkgray', face = 'bold'),
    axis.text.x = element_text(size=15),
    axis.text.y = element_text(size=15),
    axis.title.x = element_text(size=18),
    axis.title.y = element_text(size=18),
    legend.title = element_text(size=18),
    legend.text = element_text(size=16),
  )
```

```
Don't know how to automatically pick scale for object of type difft
ime. Defaulting to continuous.
```


Average Ride Duration by Hour: Members vs. Casual Riders
Average Ride Duration (mins)
30
20
10
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Day Time (h)
Rider Type
casual
member
source: Motivate International Inc.

**Analyze:**

The average casual users ride duration is higher than members' throughout a day.

Members average ride duration stays nearly the same (less than 15 minutes) during all day.

Average casuals ride duration peaks (more than **30 minutes**) is nearly **11:00-15:00**.

So, best time for ads is between 11:00 and 18:00.

## Most popular stations

- **Top most popular stations**

In [41]:

```
t_f_v2 %>%
  group_by(start_station_name, member_casual) %>%
  summarise(num_of_usage = n(), .groups = 'drop') %>%
  filter(start_station_name != "") %>%
  arrange(-num_of_usage) %>%
  head(n=10)
```

A tibble: 10 × 3

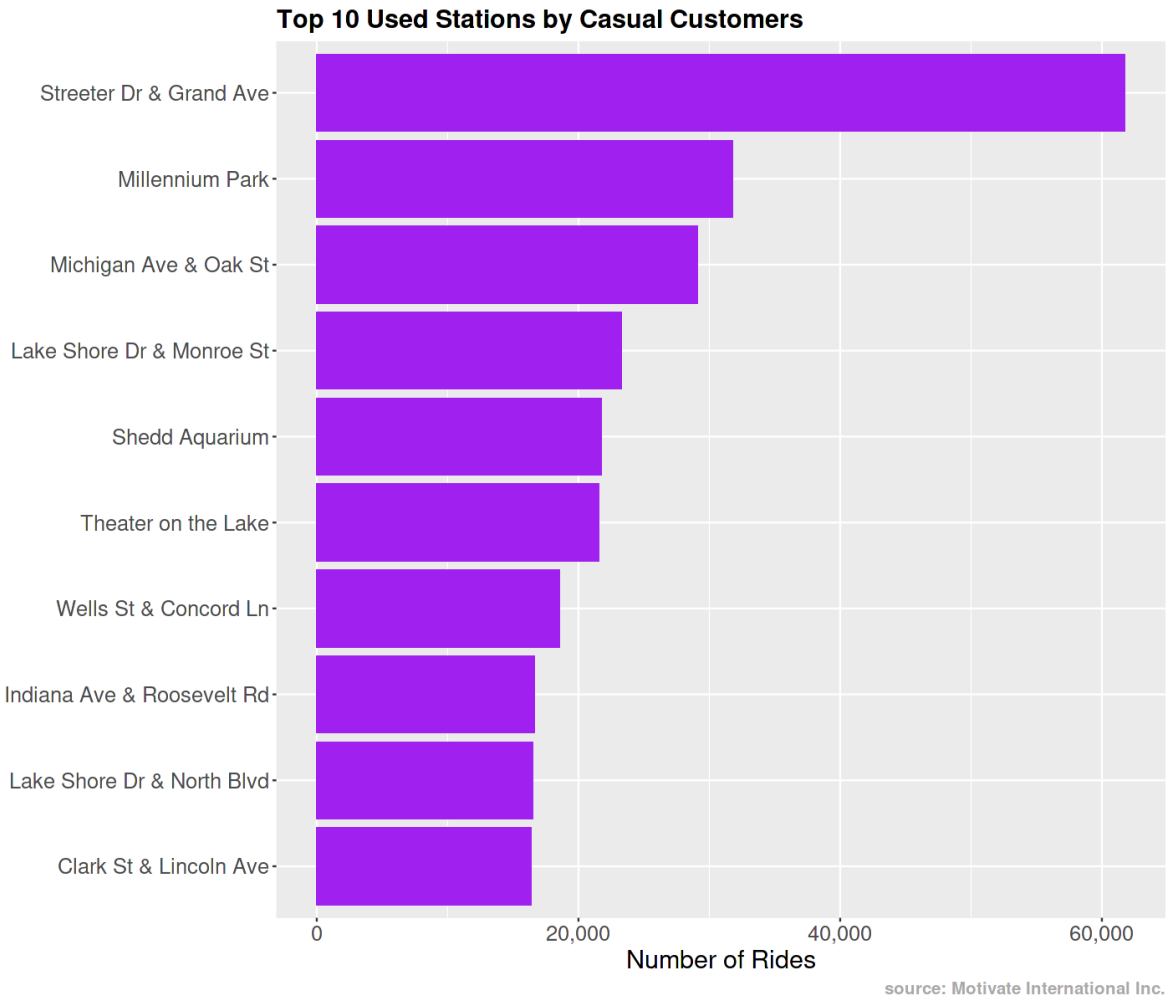
| start_station_name | member_casual | num_of_usage |
|---|---|---|
| <chr> | <chr> | <int> |
| Streeter Dr & Grand Ave | casual | 61762 |
| Millennium Park | casual | 31778 |
| Michigan Ave & Oak St | casual | 29105 |
| Clark St & Elm St | member | 23687 |
| Lake Shore Dr & Monroe St | casual | 23313 |
| Wells St & Concord Ln | member | 21841 |
| Shedd Aquarium | casual | 21781 |
| Theater on the Lake | casual | 21593 |
| Kingsbury St & Kinzie St | member | 20835 |
| Wells St & Elm St | member | 19648 |

**Analyze:**
Top 5 most popular stations are: 1) "Streeter Dr & Grand Ave"; 2) "Millennium Park"; 3) "Michigan Ave & Oak St"; 4) "Clark St & Elm St"; 5) "Lake Shore Dr & Monroe St".
We can use them for geo targeting our ad campaign.

- **Visualize the most popular stations for different customers**

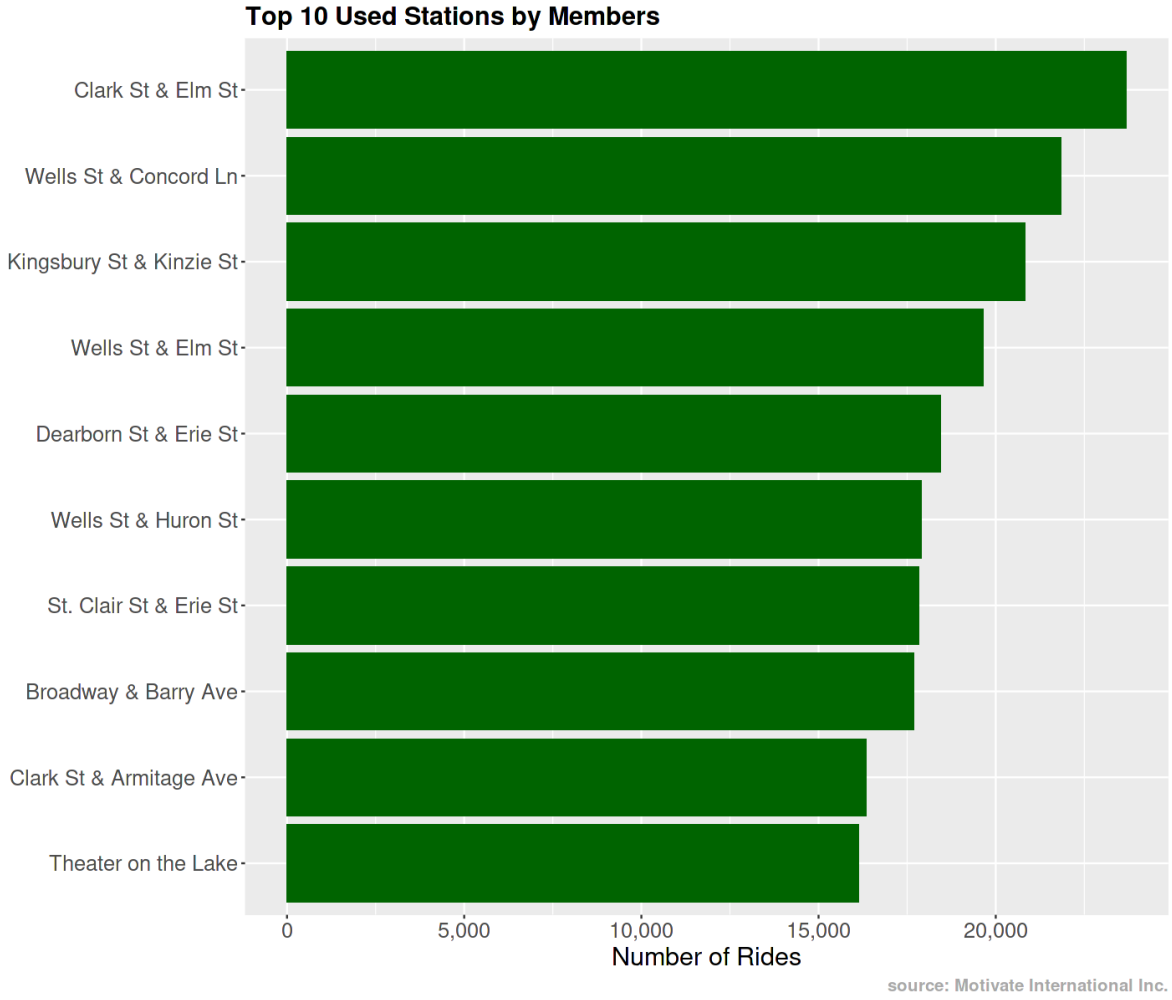
In [42]:

```
# Casual users
t_f_v2 %>%
  group_by(start_station_name, member_casual) %>%
  summarise(num_of_usage = n(), .groups = 'drop') %>%
  filter(start_station_name != "") %>%
  filter(member_casual == "casual") %>%
  arrange(-num_of_usage) %>%
  head(n=10) %>%
  ggplot() +
  geom_col(aes(x = reorder(start_station_name, num_of_usage), y = num_of_usage),
fill = "purple") +
  labs(title = "Top 10 Used Stations by Casual Customers", y = "Number of Rides"
, x = "",
       caption = "source: Motivate International Inc.") +
  coord_flip() +
  scale_y_continuous(labels = comma) +
  theme(
    plot.title = element_text(size=18, face = 'bold'),
    plot.caption = element_text(size=12, color = 'darkgray', face = 'bold'),
    axis.text.x = element_text(size=15),
    axis.text.y = element_text(size=15),
    axis.title.x = element_text(size=18),
    axis.title.y = element_text(size=18),
    legend.title = element_text(size=18),
    legend.text = element_text(size=16),
  )
```


Top 10 Used Stations by Casual Customers
Streeter Dr & Grand Ave
Millennium Park
Michigan Ave & Oak St
Lake Shore Dr & Monroe St
Shedd Aquarium
Theater on the Lake
Wells St & Concord Ln
Indiana Ave & Roosevelt Rd
Lake Shore Dr & North Blvd
Clark St & Lincoln Ave
0
20,000
40,000
60,000
Number of Rides
source: Motivate International Inc.

In [43]:

```
# Members
t_f_v2 %>%
  group_by(start_station_name, member_casual) %>%
  summarise(num_of_usage = n(), .groups = 'drop') %>%
  filter(start_station_name != "") %>%
  filter(member_casual == "member") %>%
  arrange(-num_of_usage) %>%
  head(n=10) %>%
  ggplot() +
  geom_col(aes(x = reorder(start_station_name, num_of_usage), y = num_of_usage),
fill = "darkgreen") +
  labs(title = "Top 10 Used Stations by Members", y = "Number of Rides", x = "",
       caption = "source: Motivate International Inc.") +
  coord_flip() +
  scale_y_continuous(labels = comma) +
  theme(
    plot.title = element_text(size=18, face = 'bold'),
    plot.caption = element_text(size=12, color = 'darkgray', face = 'bold'),
    axis.text.x = element_text(size=15),
    axis.text.y = element_text(size=15),
    axis.title.x = element_text(size=18),
    axis.title.y = element_text(size=18),
    legend.title = element_text(size=18),
    legend.text = element_text(size=16),
  )
```

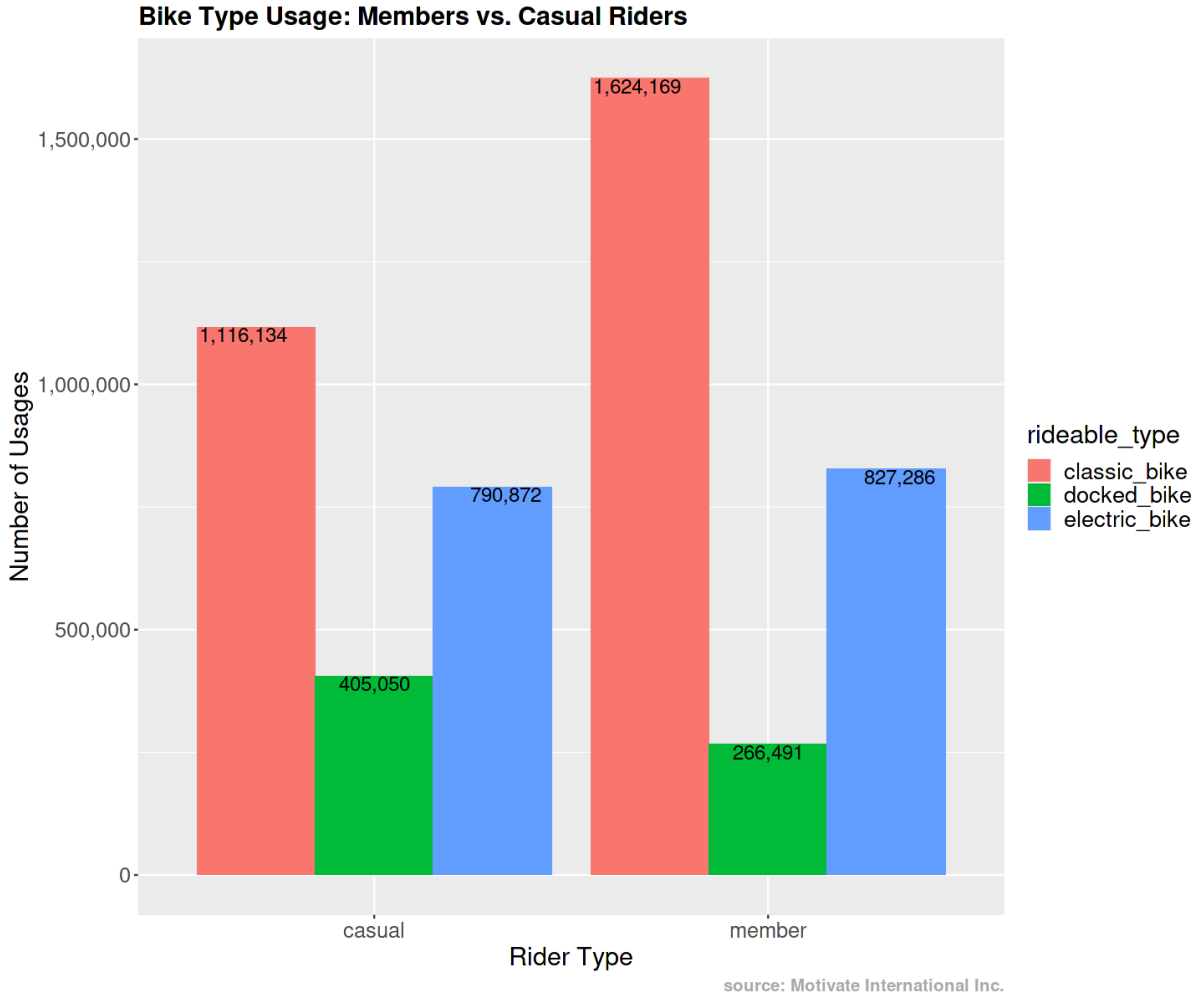



## Different types of bike usage

- **Check the customers preferences of bikes type**

In [44]:

```
t_f_v2 %>%
  group_by(rideable_type, member_casual) %>%
  summarise(num_of_usages = n(), .groups = 'drop') %>%
  ggplot(aes(x = member_casual, y = num_of_usages, fill = rideable_type)) +
  geom_col(position = "dodge") +
  labs(x = "Rider Type", y = "Number of Usages",
       title = "Bike Type Usage: Members vs. Casual Riders",
       caption = "source: Motivate International Inc.") +
  scale_y_continuous(labels = comma) +
  theme(
    plot.title = element_text(size=18, face = 'bold'),
    plot.caption = element_text(size=12, color = 'darkgray', face = 'bold'),
    axis.text.x = element_text(size=15),
    axis.text.y = element_text(size=15),
    axis.title.x = element_text(size=18),
    axis.title.y = element_text(size=18),
    legend.title = element_text(size=18),
    legend.text = element_text(size=16),
  ) +
  geom_text(aes(x = member_casual, y = num_of_usages, label = comma(num_of_usage
s), group = rideable_type),
            position = position_dodge(width = 1), vjust = 1, size = 5)
```




**Analyze:**

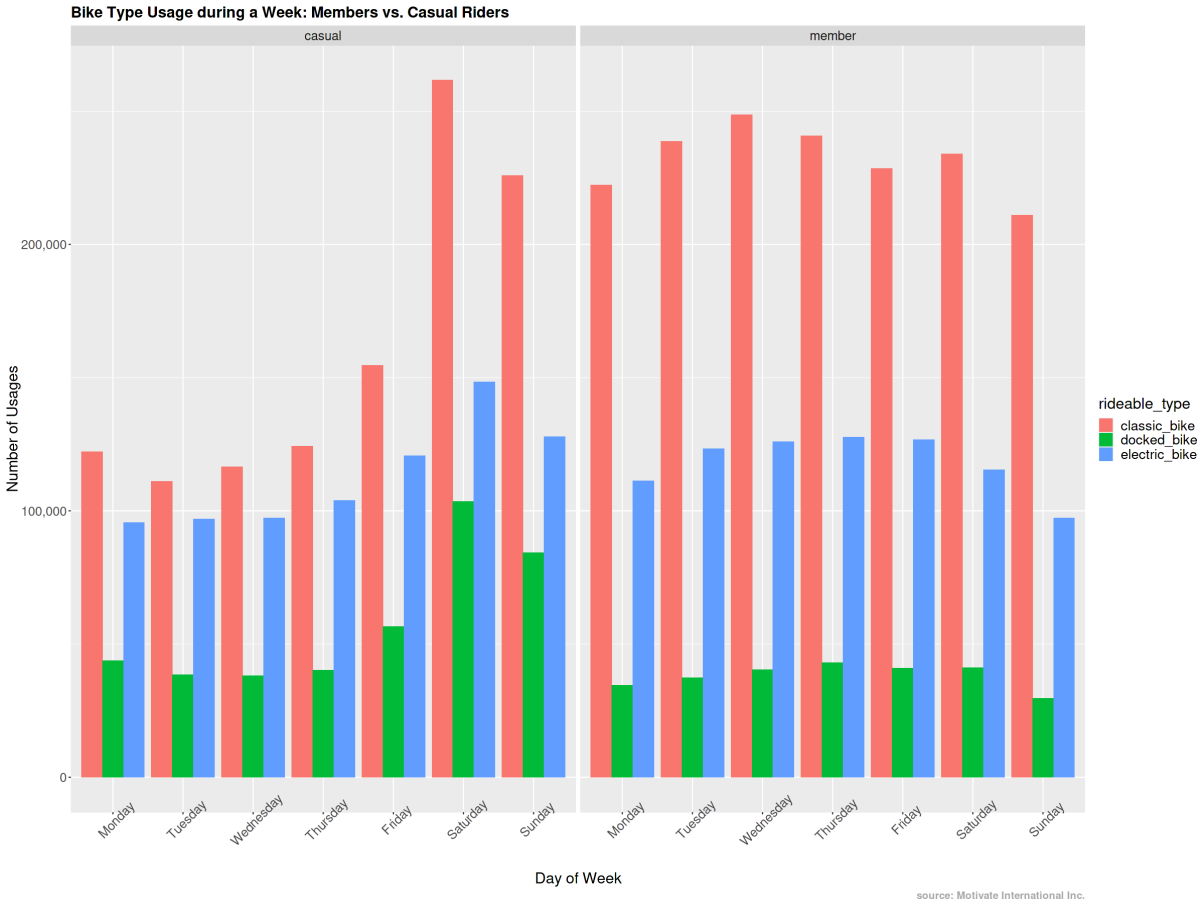
Both type of users prefer **classic** bikes.

- **Bike type usage throughout a week**

In [45]:

```
options(repr.plot.width = 20, repr.plot.height = 15)

t_f_v2 %>%
  group_by(rideable_type, member_casual, day_of_week) %>%
  summarise(num_of_usages = n(), .groups = 'drop') %>%
  ggplot(aes(x = day_of_week, y = num_of_usages, fill = rideable_type)) +
  geom_col(position = "dodge") +
  facet_wrap(~member_casual) +
  labs(x = "Day of Week", y = "Number of Usages",
       title = "Bike Type Usage during a Week: Members vs. Casual Riders",
       caption = "source: Motivate International Inc.") +
  scale_y_continuous(labels = comma) +
  theme(axis.text.x = element_text(angle=45)) +
  theme(
    plot.title = element_text(size=18, face = 'bold'),
    plot.caption = element_text(size=12, color = 'darkgray', face = 'bold'),
    axis.text.x = element_text(size=15),
    axis.text.y = element_text(size=15),
    axis.title.x = element_text(size=18),
    axis.title.y = element_text(size=18),
    legend.title = element_text(size=18),
    legend.text = element_text(size=16),
    strip.text.x = element_text(size=15),
    strip.text.y = element_text(size=15)
  )
```




**Analyze:**

We can see that casual riders tend to use more electric bikes on **weekends** that supports our hypothesis that they use Cyclistics for leisure longer trips and members use it for more practical purposes.

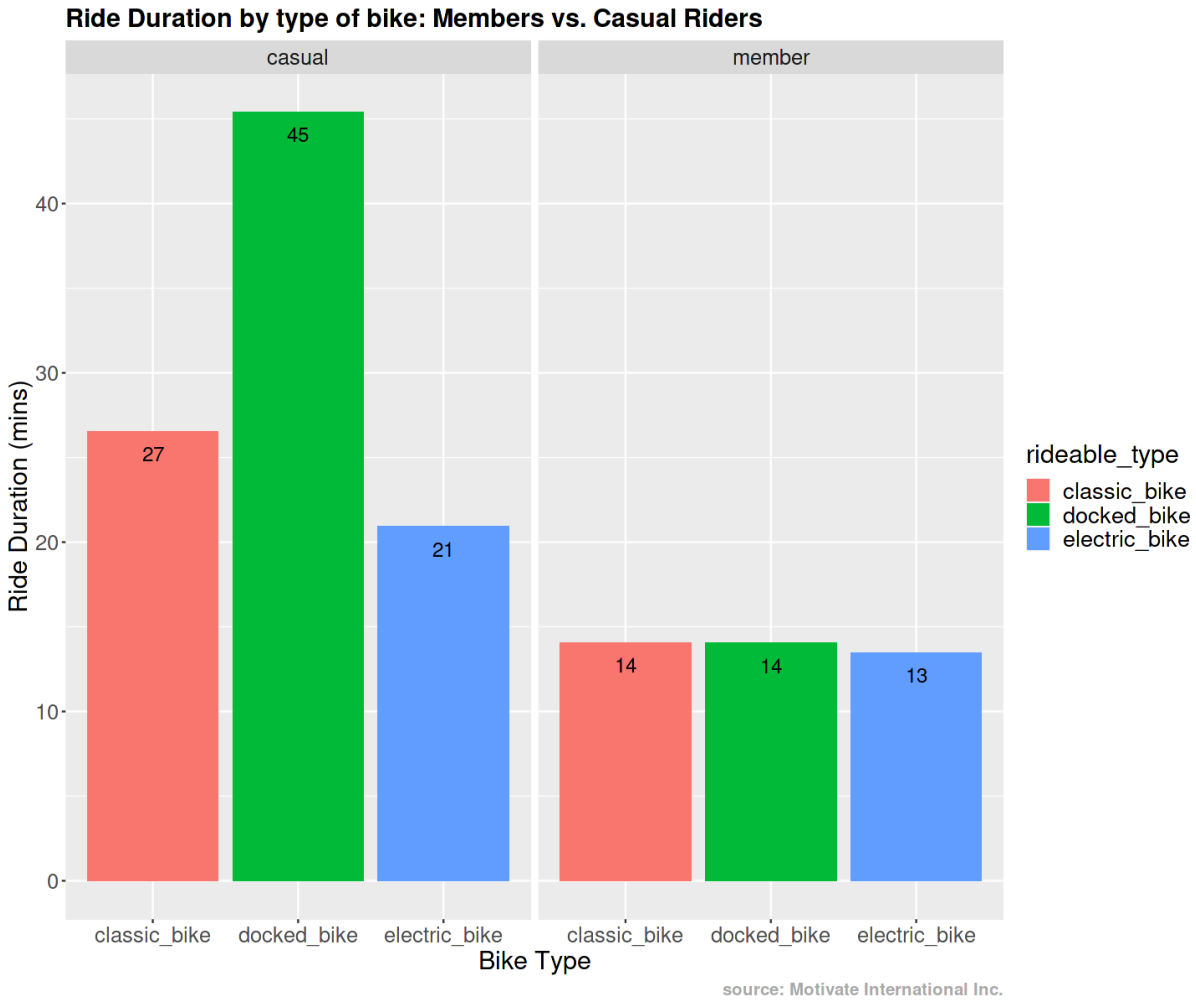
And again we see that all bike types usage of members fall on weekends.

In [46]:

```
options(repr.plot.width = 12, repr.plot.height = 10)

t_f_v2 %>%
  group_by(rideable_type, member_casual) %>%
  summarise(avg_time_spent = mean(ride_length)/60, .groups = 'drop') %>%
  ggplot(aes(x=rideable_type, y=avg_time_spent, fill = rideable_type)) +
  geom_col(position = "dodge") +
  facet_wrap(~member_casual) +
  labs(x = "Bike Type", y = "Ride Duration (mins)",
       title = "Ride Duration by type of bike: Members vs. Casual Riders",
       caption = "source: Motivate International Inc.") +
  scale_y_continuous(labels = comma) +
  theme(
    plot.title = element_text(size=18, face = 'bold'),
    plot.caption = element_text(size=12, color = 'darkgray', face = 'bold'),
    axis.text.x = element_text(size=15),
    axis.text.y = element_text(size=15),
    axis.title.x = element_text(size=18),
    axis.title.y = element_text(size=18),
    legend.title = element_text(size=18),
    legend.text = element_text(size=16),
    strip.text.x = element_text(size=15),
    strip.text.y = element_text(size=15)
  ) +
  geom_text(aes(label = round(avg_time_spent,0)), vjust = 2, size = 5)
```

```
Don't know how to automatically pick scale for object of type difft
ime. Defaulting to continuous.
```




**Analyze:**

Though the most popular type of bike is classic, "champions" by ride duration for casual customers are docked_bikes (**45** minutes).

Interesting, that members almost don't use any type of bike longer than others (near **14** mins).

Also, we can notice that, in general, casual users spend more

# PART 6 - ACT

Here are my recommendations on how to convert casual customers to members:

**1. A weekend offer.**

Most "popular days" are saturday and sunday. We can implement a new type of less expensive membership: for those who will use bikes only (or mostly) on weekends. Or give some bonuses for weekend members like extra minutes or discount.

**2. A seasonal offer.**

Extra minutes, discount or free water (we've seen a decrease of ride length during a heat) for summer or spring members.

**3. Special hours offer.**

The peak time of biking is from 3 PM to 6 PM. We can offer some bonuses, like a free hour (or 30 minutes), at this time for members.
Because of casual users tend to ride longer trips, we might suggest some bonuses for trips longer than 30 minutes.

**4. Geotargeting advertisement**

We can use digital media for geotargeting advertising near the most popular stations. Stations with over 20,000 total rides: "Streeter Dr & Grand Ave", "Millenium Park", "Michigan Ave & Oak St", "Lake Shore Dr & Monroe St", "Shedd Aquarium" and "Theater on the Lake".
Also, suggest some extra bonuses (like a coupon in a local store) for new members.

**5. Bike types offers**

Since electric bikes are less popular for casual customers, we can upgrade them with some extra features to emphasize it's merit. E.g. sightseeing guide (via the app, QR-code or built-in headphones) or travel routes navigator, etc.
Docked bikes are "champions" by ride duration. So we can offer some bonuses (lower price, discount, free minutes, a free water bottle, etc.) for members who ride long trips.
Classic bikes are already the most popular, but we can consider to implement all listed above options for classic bikes member users.

THANK YOU!