Cyclistic Bike-Share Case Study (Google Data Analytics Capstone)

Tools: R (`tidyverse`, `lubridate`, `janitor`, `ggplot2`), SQL (for sanity checks), Excel (spot checks).
Data: 12 months of Cyclistic trips (CSV, millions of rows).
Goal: Understand how **casual riders** vs **annual members** use bikes differently to inform a campaign that **converts casuals to members**.

ASK

Business task:
Identify behavioral differences between casual riders and members to recommend **data-driven tactics** that increase annual memberships.

Primary question:

How do annual members and casual riders use Cyclistic bikes differently?

Stakeholders:
Director of Marketing (Lily Moreno), Marketing Analytics Team, Executive Team.

PREPARE

Data source: Cyclistic historical trip data (12 months).
Format: CSV; typical columns include:

- `ride_id`, `rideable_type`, `started_at`, `ended_at`, `start_station_name`, `end_station_name`, `member_casual`, etc.

Storage: Local `data/` folder; sample data for repo, full data kept locally (files are large).

Quality & limitations:

- Missing station names/IDs for some rows.

- Outliers (negative or extremely long ride durations) must be removed.
- Time fields in local timezone; confirm and normalize.

PROCESS (Cleaning & feature engineering with R)

```
# packages
library(tidyverse)
library(lubridate)
library(janitor)

# load multiple months (example pattern)
files <- list.files("data", pattern = "\\.csv$", full.names = TRUE)
trips_raw <- files %>% map_df(read_csv)

# basic cleaning
trips <- trips_raw %>%
  clean_names() %>%
  mutate(
    started_at = ymd_hms(started_at),
    ended_at   = ymd_hms(ended_at),
    ride_length_min = as.numeric(difftime(ended_at, started_at, units
= "mins")),
    day_of_week = wday(started_at, label = TRUE, week_start = 1),
    month = floor_date(started_at, "month"),
    hour  = hour(started_at)
  ) %>%
  # keep valid rows only
  filter(
    !is.na(started_at), !is.na(ended_at),
    ride_length_min > 0,            # remove negative/zero durations
    ride_length_min <= 1440         # cap at 24h to remove extreme
outliers
  ) %>%
  # trim whitespace in station names
  mutate(
    start_station_name = str_squish(start_station_name),
```

```
    end_station_name   = str_squish(end_station_name)
  )
```

Sanity checks with SQL (optional):

```
-- avg ride length by member type
SELECT member_casual, AVG(TIMESTAMPDIFF(MINUTE, started_at, ended_at))
AS avg_mins
FROM trips
WHERE ended_at > started_at AND TIMESTAMPDIFF(MINUTE, started_at,
ended_at) <= 1440
GROUP BY member_casual;
```

ANALYZE (Key comparisons)

1) Ride duration & frequency

```
duration_summary <- trips %>%
  group_by(member_casual) %>%
  summarise(
    rides = n(),
    avg_mins = mean(ride_length_min),
    median_mins = median(ride_length_min)
  )
print(duration_summary)
```

2) Day-of-week & hour patterns

```
dow_pattern <- trips %>%
  count(member_casual, day_of_week, name = "rides")

hour_pattern <- trips %>%
  count(member_casual, hour, name = "rides")
```

3) Seasonality (monthly trend)

```
monthly <- trips %>%
  count(member_casual, month, name = "rides")
```

4) Start stations (top locations)

```
top_stations <- trips %>%
  filter(!is.na(start_station_name), start_station_name != "") %>%
  count(member_casual, start_station_name, sort = TRUE, name =
"rides") %>%
  group_by(member_casual) %>%
  slice_head(n = 10)
```

SHARE (Visuals for stakeholders)

Export plots to `images/` and embed them below.

Ride duration distribution (members vs casuals)

```
library(ggplot2)

ggplot(trips, aes(x = ride_length_min, fill = member_casual)) +
  geom_histogram(binwidth = 5, alpha = 0.6, position = "identity") +
  coord_cartesian(xlim = c(0, 120)) +
  labs(title = "Ride Length Distribution (0–120 mins)",
       x = "Minutes", y = "Count")
ggsave("images/duration_hist.png", width = 8, height = 5, dpi = 300)
```

Rides by day of week

```
ggplot(dow_pattern, aes(x = day_of_week, y = rides, fill =
member_casual)) +
  geom_col(position = "dodge") +
  labs(title = "Rides by Day of Week", x = "", y = "Rides")
ggsave("images/rides_by_dow.png", width = 8, height = 5, dpi = 300)
```

Hourly usage pattern

```
ggplot(hour_pattern, aes(x = hour, y = rides, color = member_casual))
+
  geom_line(linewidth = 1) +
  labs(title = "Hourly Usage Pattern", x = "Hour of Day", y = "Rides")
ggsave("images/hourly_usage.png", width = 8, height = 5, dpi = 300)
```

Monthly seasonality

```
ggplot(monthly, aes(x = month, y = rides, color = member_casual)) +
  geom_line(linewidth = 1) +
  labs(title = "Monthly Rides by Rider Type", x = "Month", y =
"Rides")
ggsave("images/monthly_trend.png", width = 8, height = 5, dpi = 300)
```

Top start stations (table preview)

```
top_stations %>% print(n = 20)
```

INSIGHTS (example findings—validate with your results)

- **Casual riders** tend to have **longer average ride durations** and higher **weekend usage** (leisure behavior).
- **Members** ride **shorter, more consistent trips** concentrated on **weekday peaks** (commute behavior).
- There's strong **seasonality** (summer spikes) for both, but casual riders show **bigger seasonal swings**.
- **Top casual stations** cluster near tourist/recreation areas; **member stations** cluster near transit/work hubs.

ACT (Recommendations)

1. **Convert casuals with targeted offers:**
   - **Weekend → Weekday trial**: “Ride to work free for 2 weeks if you rode this weekend.”
   - **Bundle**: multi-month discounted membership during peak season.
2. **Geo-targeting & creatives:**
   - Promote at **top casual stations** with QR codes and **membership benefits**.
   - Messaging focused on **cost savings + convenience** for frequent riders.
3. **Product nudges in app:**
   - If a casual rider takes ≥3 rides/month, show **in-app calculator** comparing per-ride vs membership.
4. **Measure & iterate:**
   - A/B test landing pages and in-app banners.
   - Track conversion rate, CAC, 30/90-day retention.

Success metrics:
Membership conversion rate, churn rate, average rides per member, revenue per user.

Repo Structure (suggested)

```
cyclistic-bike-share-case-study/
├─ README.md
├─ data/
│  ├─ sample_trip_2022-01.csv
│  └─ ...
├─ notebooks/
│  ├─ cyclistic_analysis.Rmd
│  └─ cyclistic_analysis.ipynb   # if you also use Python
├─ scripts/
│  ├─ clean_transform.R
│  ├─ sanity_checks.sql
```

```
│   └─ viz.R
├─ images/
│   ├─ duration_hist.png
│   ├─ rides_by_dow.png
│   ├─ hourly_usage.png
│   └─ monthly_trend.png
└─ .gitignore
```