Cyclistic Bike-Share Analysis (2021-2024): Understanding Ride Patterns and Membership Conversion

Prepared by: Rohan Jha
March 10, 2025

About Cyclistic

Cyclistic is a **bike-sharing company based in Chicago**, launched in **2016** to promote **sustainable urban mobility**. It operates a **fleet of 5,824 bicycles** across **692 stations**, allowing users to rent bikes for short trips.

🚲 Services Offered:

- 🔹 **Classic Bikes** – Traditional pedal-powered bikes.
- 🔹 **Electric Bikes** – Battery-assisted bikes for longer distances.
- 🔹 **Electric Scooters** – A fast and convenient alternative

.

📋 Pricing Plans:

- **Single-Ride Pass** – Best for occasional users.
- **Full-Day Pass** – Designed for tourists and short-term visitors.
- **Annual Membership** – Ideal for frequent riders, offering unlimited rides at a fixed monthly/yearly cost.

  Note: Users who purchase **single-ride** or **full-day passes** are referred to as '**casual riders**' in this report. '**Annual members**' are those who have purchased an **annual membership**

🎯 Company Goal:

Cyclistic aims to **increase the number of annual members**, as data suggests that **members generate more revenue** than casual riders. By focusing on rider behavior and usage patterns, Cyclistic plans to implement strategies that encourage casual riders to switch to an annual membership model.

---

1. 🎯 Purpose of the Report

Cyclistic's finance team has identified that **annual members generate more revenue** than casual riders. While casual riders contribute to the company's user base, they do not provide the **consistent revenue stream** that memberships ensure. Therefore, the company wants to **increase the number of annual members** by understanding how casual riders behave and what factors influence their riding patterns.

This report is designed to provide **data-driven insights** into:

1️⃣ Understanding Ride Trends

- Identifying **daily, weekly, and seasonal trends** in ridership.
- Analyzing the impact of **time of day** on ride frequency.
- Exploring how **different bike types** (classic, electric bike, electric scooter) are used.

2️⃣ Analyzing Casual vs. Member Behavior

- Comparing **ride duration and frequency** for casual riders and annual members.
- Evaluating **station preferences** – where casual riders rent bikes vs. where members ride most.
- Understanding whether casual riders tend to **take longer or more leisurely trips** compared to members.

3️⃣ Assessing Weather Impact on Ridership

- Determining how **temperature fluctuations affect ride volume**.
- Evaluating the impact of **humidity, wind speed, and seasonal changes** on ride frequency.
- Identifying whether **weather conditions influence casual riders more than members**.

4️⃣ Business Recommendations for Membership Growth

- Developing **targeted marketing campaigns** to encourage casual riders to become members.
- Offering **incentives and discounts** based on ridership trends.
- Optimizing **bike station placements** and availability based on peak demand periods.
- Leveraging **seasonal promotions** to maintain ridership during low-traffic months.

By leveraging these insights, Cyclistic can create a **data-backed strategy** to increase **membership conversion rates**, improve **operational efficiency**, and enhance **overall customer experience**.

2. 🔬 Methodology

To analyze **Cyclistic's ridership trends** and provide **data-driven recommendations**, we followed a structured **five-step approach**, using **R for data cleaning, analysis, and integration** and **Tableau for visualization**.

1️⃣ Data Collection

We gathered two primary datasets:

📌 **Cyclistic Trip Data (2021-2024)** – Publicly available dataset containing **ride details**, including trip duration, bike type, start & end stations, and user type (**casual/member**). [Dataset]

📌 **NOAA Weather Data** – Historical **temperature, humidity, and wind speed** data for Chicago, used to analyze weather impact on ridership.

🛠 **Tools Used:**
✅ **R (tidyverse, plotly, worldmet)** – For data processing and statistical analysis.
✅ **Tableau** – For interactive visualizations and business insights.

2️⃣ Data Cleaning & Preprocessing (Using R)

To ensure data accuracy and consistency, we performed the following preprocessing steps **in R**:

✅ **Merged** yearly trip data files into a **single dataset** using `rbind()`.
✅ **Converted timestamps** (`started_at`, `ended_at`) into **POSIXct format** (`lubridate` package) for accurate time-based analysis.

✅ **Extracted new features**:

- **Month, Day, and Hour** – To analyze time-based trends (`lubridate::month()`, `wday()`).
- **Total Ride Duration** – Calculated as `ended_at - started_at`.
- **Time Intervals** – Grouped ride times into **2-hour blocks** (`case_when()`).

✅ **Checked for missing values** (`colSums(is.na())`) and handled them based on station and weather data availability.

3️⃣ Data Integration (Using R)

We used the **R to integrate weather data**:

🔹 **Weather Data Integration:**

- Imported **Chicago's hourly weather data** using the `worldmet` package.
- Matched ride timestamps (`started_at`) with **hourly weather conditions** (`floor_date()`).
- Merged trip data with **temperature, humidity, and wind speed** information using `left_join()`.

4️⃣ Data Visualization & Interpretation (Using R & Tableau)

📊 **R Visualizations:**

- Created **EDA plots** (`ggplot2`) for initial analysis.
- Used `ggplotly()` (`plotly` package) to make graphs interactive.

📌 **Tableau Dashboards:**

- **Heatmaps** for daily ridership trends.
- **Bar graphs** for time-based usage patterns.
- **Line charts** to visualize **weather impact on ridership**.

5️⃣ Business Recommendations & Insights

📌 Based on **ride behavior, peak demand, and weather influence**, we formulated **actionable recommendations** to increase membership conversions and optimize bike availability.

Why This Approach?

Using **R for data cleaning, integration, and initial EDA** allowed efficient **data processing** and **pattern detection**, while **Tableau** provided a **clear and interactive visualization** for decision-making.

3. 📌 Data Analysis – R & Tableau

We'll first document **3.1 R Analysis**, breaking down the key steps from your script, followed by **3.2 Tableau Analysis** for visualization insights.

3.1 R Analysis

This section focuses on how R was used for **data preprocessing, analysis, and integration** to extract meaningful insights.

📌 Step 1: Installing & Loading Required Packages

📌 What This Code Does?

This section **installs and loads** the necessary R packages for **data cleaning, visualization, and weather data extraction**.

1️⃣ Installing & Loading `tidyverse`

```
install.packages("tidyverse")
library(tidyverse)
```

✅ Purpose:

- `tidyverse` is a **collection of R packages** designed for **data manipulation, visualization, and analysis**.
- It includes essential libraries like `dplyr`, `ggplot2`, `tidyr`, `readr`, and more.
- **Key Functions Used Later:**
  - `dplyr::mutate(), select(), group_by(), summarise()` – **For data transformation.**
  - `ggplot2::ggplot()` – **For visualization.**

2️⃣ Installing & Loading `plotly`

```
install.packages("plotly")
library(plotly)
```

✅ **Purpose:**

- `plotly` allows **interactive data visualization** in R.
- It enhances **ggplot2** graphs, making them **zoomable and interactive**.
- **Key Functions Used Later:**
    - `ggplotly()` – **Converts static `ggplot2` graphs into interactive charts.**

3️⃣ Installing & Loading `worldmet`

```
install.packages("worldmet")
library(worldmet)
```

✅ **Purpose:**

- `worldmet` is used to **fetch weather data** from NOAA (National Oceanic and Atmospheric Administration).
- We use this package to **import historical weather data** (temperature, humidity, wind speed) for **Chicago**.
- **Key Functions Used Later:**
    - `importNOAA(year, code)` – **Fetches weather data for a specified location.**

🔍 Key Takeaways:

✓ `tidyverse` – **Data cleaning & visualization.**
✓ `plotly` – **Interactive charts.**
✓ `worldmet` – **Weather data extraction.**

📌 Step 2: Load & Merge Monthly Data for 2021

```
# 📁 List all monthly CSV files for 2021 in the working directory

  csv_files <- list.files(pattern = "\\.csv$")


# 📊 Read each CSV file into a list of data frames

  csv_list <- lapply(csv_files, read.csv)


# 🔄 Merge all 12 monthly data frames into a single dataset for 2021

  cyclist_2021 <- do.call(rbind, csv_list)


# 🔍 Check dataset structure (column names, data types, sample values)

  str(cyclist_2021)


# 💾 Save the merged dataset for further analysis

  write.csv(cyclist_2021, "cyclist-2021.csv", row.names = FALSE)


# 🛠 Free up memory by removing temporary objects

  rm(csv_list, cyclist_2021)
```

`str()` Output:

```
'data.frame':   5,595,063 obs. of  13 variables:

 $ ride_id            : chr  "E19E6F1B8D4C42ED" "DC88F20C2C55F27F"
"EC45C94683FE3F27" ...

 $ rideable_type      : chr  "electric_bike" "electric_bike"
"electric_bike" ...

 $ started_at         : chr  "2021-01-23 16:14:19" "2021-01-27 18:43:08"
"2021-01-21 22:35:54" ...

 $ ended_at           : chr  "2021-01-23 16:24:44" "2021-01-27 18:47:12"
"2021-01-21 22:37:14" ...

 $ start_station_name: chr  "California Ave & Cortez St" "California Ave &
Cortez St" "California Ave & Cortez St" ...

 $ start_station_id  : chr  "17660" "17660" "17660" "17660" ...

 $ end_station_name  : chr  "" "" "" "" ...

 $ end_station_id    : chr  "" "" "" "" ...

 $ start_lat         : num  41.9 41.9 41.9 41.9 ...

 $ start_lng         : num  -87.7 -87.7 -87.7 -87.7 ...

 $ end_lat           : num  41.9 41.9 41.9 41.9 ...

 $ end_lng           : num  -87.7 -87.7 -87.7 -87.7 ...

 $ member_casual     : chr  "member" "member" "member" "member" ...
```

🔍 Summary of This Step

✓ **Merged 12 monthly datasets** into a **single yearly dataset (5.59M records)**.
✓ **Saved the merged data** as `"cyclist-2021.csv"` for future processing.
✓ **Checked the structure (`str()`)** to ensure all columns are present and correctly formatted.

📌 Step 3: Data Cleaning & Formatting

📂 Loading the merged 2021 dataset and preparing it for analysis.

```
# 📂 Load the merged 2021 dataset

  library(readr)

  cyclist_2021 <- read_csv("cyclist-2021.csv")

# 🛠 Define key columns to check for missing values

  columns_to_check <-  c("started_at", "ended_at", "rideable_type",
"member_casual")


# ⚠️ Count missing values in critical columns

  colSums(is.na(cyclist_2021[columns_to_check]))


# ⏳ Convert timestamps to datetime format for time-based analysis

  cyclist_2021$started_at <- as.POSIXct(cyclist_2021$started_at, format =
"%Y-%m-%d %H:%M:%S")

  cyclist_2021$ended_at <- as.POSIXct(cyclist_2021$ended_at, format =
"%Y-%m-%d %H:%M:%S")



# 📅 Extract month name from timestamps for monthly trend analysis

  cyclist_2021$month <- month.name[month(cyclist_2021$started_at)]


# ✅ Verify formatting and month extraction

  str(cyclist_2021)
```

📌 Missing Values Check – **Output**

```
  started_at      ended_at rideable_type member_casual

           0             0             0             0
```

✅ **No missing values** found in key columns (`started_at`, `ended_at`, `rideable_type`, `member_casual`).

📌 `str()` Output – Data Structure After **Cleaning**

```
'data.frame':   5,595,063 obs. of  14 variables:

 $ ride_id           : chr  "E19E6F1B8D4C42ED" "DC88F20C2C55F27F"
"EC45C94683FE3F27" ...

 $ rideable_type     : chr  "electric_bike" "electric_bike"
"electric_bike" ...

 $ started_at        : POSIXct, format: "2021-01-23 16:14:19" "2021-01-27
18:43:08" ...

 $ ended_at          : POSIXct, format: "2021-01-23 16:24:44" "2021-01-27
18:47:12" ...

 $ start_station_name: chr  "California Ave & Cortez St" "California Ave &
Cortez St" ...

 $ start_station_id  : chr  "17660" "17660" "17660" "17660" ...

 $ end_station_name  : chr  "" "" "" "" ...
```

```
$ end_station_id     : chr  "" "" "" "" ...
$ start_lat          : num  41.9 41.9 41.9 41.9 ...
$ start_lng          : num  -87.7 -87.7 -87.7 -87.7 ...
$ end_lat            : num  41.9 41.9 41.9 41.9 ...
$ end_lng            : num  -87.7 -87.7 -87.7 -87.7 ...
$ member_casual      : chr  "member" "member" "member" "member" ...
$ month              : chr  "January" "January" "January" "January" ...
```

🔍 Summary of This Step

✓ **Loaded the merged dataset** into R.
✓ **Confirmed no missing values** in key columns.
✓ **Converted timestamps (`started_at`, `ended_at`) to `POSIXct` format** for accurate time analysis.
✓ **Extracted month names** successfully (**see "month" column in output**).

📌 Step 4: Calculate Total Ride Time

📂 Computing total ride duration and summarizing average ride time by user type, bike type, and month.

```
# ⏳ Calculate total ride duration in seconds for each trip

  cyclist_2021$Total_time_secs <- as.numeric(cyclist_2021$ended_at -
cyclist_2021$started_at)


# 📊 Calculate average ride time (in minutes) grouped by bike type, user
type, and month

  Mean_Time_C21 <- cyclist_2021 %>%

 group_by(rideable_type, member_casual, month) %>%

 summarise(Mean_Ride_Time = mean(Total_time_secs / 60, na.rm = TRUE))
# Convert seconds to minutes


# 🔍 Preview the calculated mean ride times

  head(Mean_Time_C21)


# 💾 Save the mean ride time summary to a CSV file

  write.csv(Mean_Time_C21, "Mean-ridetime-cyclist2021.csv", row.names =
FALSE)
```

📌 `head(Mean_Time_C21)` Output – **First Few Rows**

```
# A tibble: 6 × 4

# Groups:   rideable_type, member_casual [1]

  rideable_type member_casual month     Mean_Ride_Time

  <chr>         <chr>         <chr>              <dbl>

1 classic_bike  casual        April               31.7

2 classic_bike  casual        August              27.5

3 classic_bike  casual        December            28.2

4 classic_bike  casual        February            41.8

5 classic_bike  casual        January             24.8

6 classic_bike  casual        July                28.3
```

📌 Summary of This Step

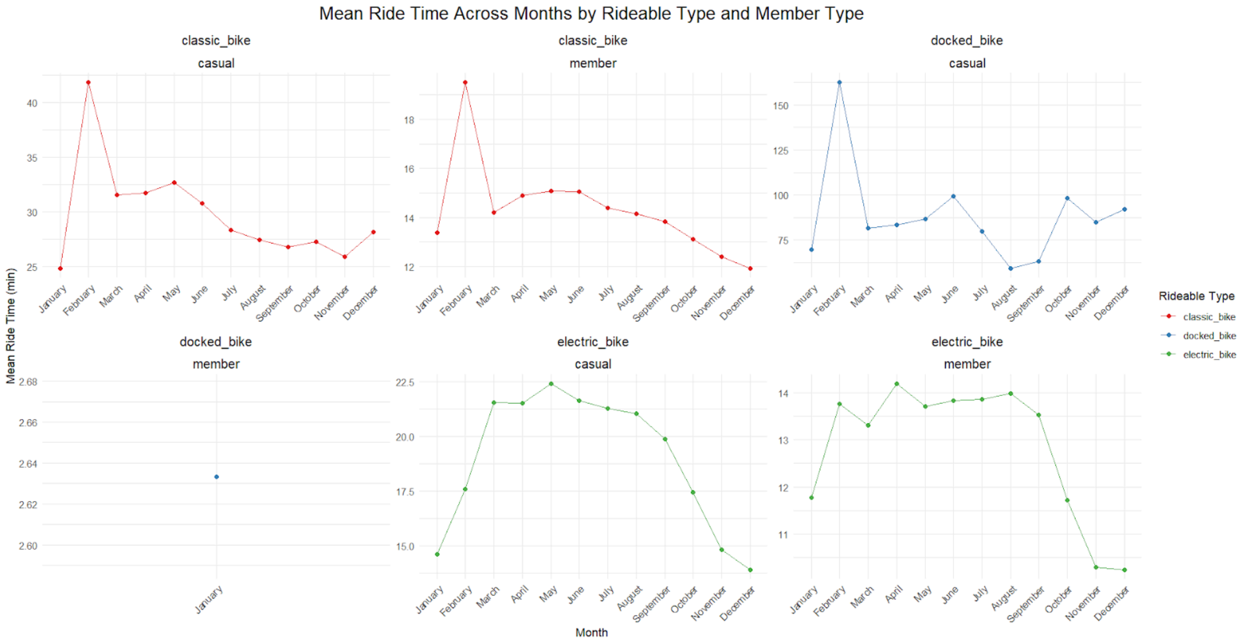
✓ **Calculated total ride duration** in seconds (`ended_at - started_at`).
✓ **Converted ride duration to minutes** and calculated the **mean ride time** per **bike type, user type, and month**.
✓ **Saved the summary dataset** (`Mean-ridetime-cyclist2021.csv`) for future visualization and analysis.

📌 Step 5: Visualizing Mean Ride Time Over Months

📊 Creating a line chart to analyze how mean ride time varies across months for different bike types and user categories.

```
# 📅 Define the correct chronological order for months

  month_levels <- c("January", "February", "March", "April", "May",
"June",  "July", "August", "September", "October", "November", "December")


# 🔄 Convert 'month' to a factor to ensure correct ordering

  Mean_Time_C21 <- Mean_Time_C21 %>%

 mutate(month = factor(month, levels = month_levels)) %>%

 arrange(month)


# 📈 Plot mean ride time trend across months

  ggplot(Mean_Time_C21, aes(x = month, y = Mean_Ride_Time, color =
rideable_type, group = rideable_type)) +

 geom_line() +                          # Line plot to show trends

 geom_point() +                         # Points to highlight data values

 facet_wrap(rideable_type ~ member_casual, scales = "free") +  # Separate
plots for each user type & bike type

 labs(

   title = "Mean Ride Time Across Months by Rideable Type and Member
Type",
```

```
    x = "Month",

    y = "Mean Ride Time (min)",

    color = "Rideable Type"

  ) +

  theme_minimal() +                         # Apply a clean theme

  scale_color_brewer(palette = "Set1") + # Use a visually appealing color
palette

  theme(

    axis.text.x = element_text(angle = 45, hjust = 1),X = element_text
(angle = 45, just = 1),  # Rotate x-axis labels for readability

    strip.text = element_text(size = 12),                # Adjust facet
label size

    plot.title = element_text(hjust = 0.5, size = 16)    # Center-align the
title
```



📌 Key Findings from the Mean Ride Time Analysis

✓ **Casual riders take longer trips** compared to annual members, indicating they may use bikes more for leisure rather than commuting.

✓ **In February, classic bikes are preferred for longer rides**, possibly due to fewer tourists and more local users taking extended trips.

✓ **During summer, electric bikes become more popular**, likely due to convenience and higher demand for quick, efficient travel.

✓ **Docked (classic) bikes are the go-to choice for casual riders when trips exceed an hour on average**, suggesting they are used for exploration or recreational rides.

📌 Summary of This Step

✓ **Ordered the months correctly** for proper visualization.
✓ **Created a line chart** to track **monthly ride duration trends** across bike and user types.
✓ **Identified seasonal patterns** in how long people ride.

📌 Step 6 & 7: Summarizing and Visualizing Monthly User Counts

📁 **Counting the total number of rides per user type, bike type, and month, followed by visualization.**

📌 Step 6: Summarizing Monthly User Data

```r
# 📊 Group data by user type, bike type, and month to count rides

  Total_UserCount_Monthly <- cyclist_2021 %>%

    group_by(member_casual, rideable_type, month) %>%

    summarise(riders_count = n(), .groups = "drop")


# 📅 Order months correctly for better readability

  Total_UserCount_Monthly <- Total_UserCount_Monthly %>%

   mutate(month = factor(month, levels = month_levels)) %>%

   arrange(month)


# 🔍 View summarized data
```

```r
  head(Total_UserCount_Monthly)


# 💾 Save the summarized dataset

  write.csv(Total_UserCount_Monthly, "Monthly-UserCount-cyclist2021.csv",
row.names = FALSE)
```

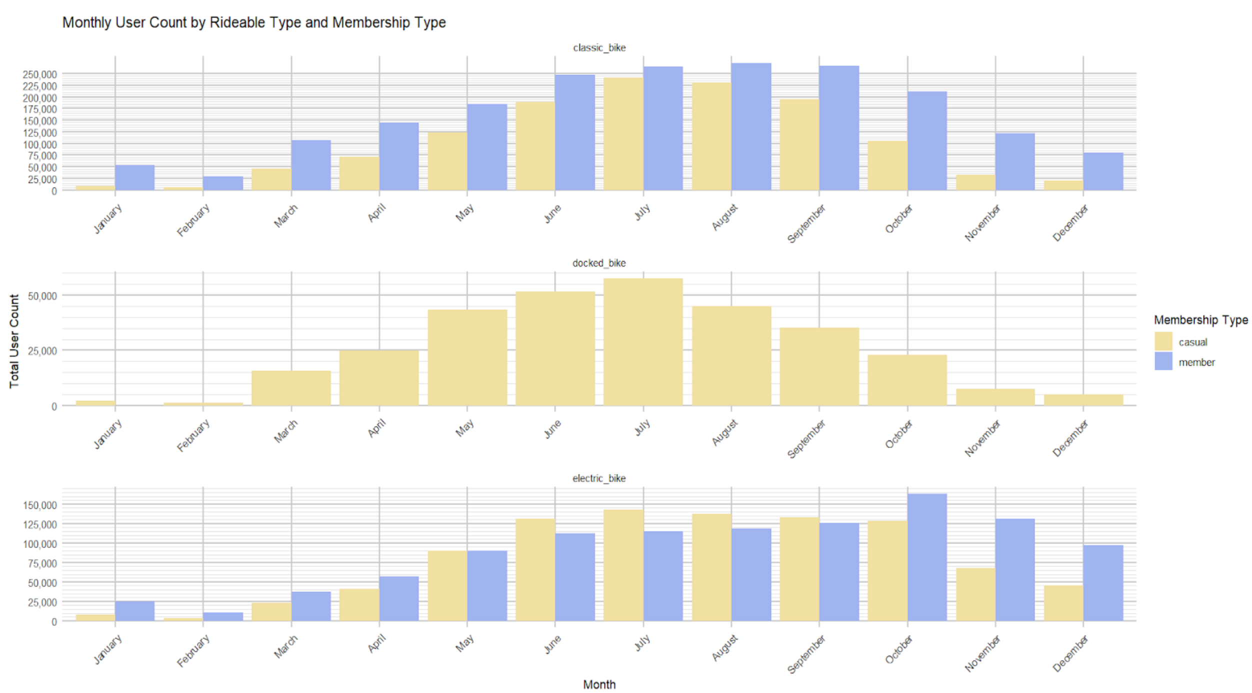
📌 `head(Total_UserCount_Monthly)` Output – **First Few Rows**

```
# A tibble: 6 × 4
  member_casual rideable_type month   riders_count
  <chr>         <chr>         <fct>          <int>
1 casual        classic_bike  January         8259
2 casual        docked_bike   January         2105
3 casual        electric_bike January         7753
4 member        classic_bike  January        53441
5 member        docked_bike   January            1
6 member        electric_bike January        25275
```

📌 Step 7: Visualizing Monthly User Counts (Bar Chart)

```r
# 📊 Plot bar graph showing user counts per month and bike type
  ggplot(Total_UserCount_Monthly, aes(x = month, y = riders_count, fill =
member_casual)) +
  geom_bar(stat = "identity", position = "dodge") +
  facet_wrap(~ rideable_type, scales = "free", ncol = 1) +
  scale_y_continuous(
    breaks = seq(0, 250000, 25000),
    minor_breaks = seq(0, 250000, 5000),
    labels = scales::comma
  ) +
  scale_fill_manual(values = c("member" = "#a1b4f4", "casual" =
"#F4E1A1")) +
  labs(
    title = "Monthly User Count by Rideable Type and Membership Type",
    x = "Month",
```

```
    y = "Total User Count",
    fill = "Membership Type"
  ) +
  theme_minimal() +
  theme(
    panel.grid.major = element_line(color = "grey80", linewidth = 0.8),
    panel.grid.minor = element_line(color = "grey90", linewidth = 0.5),
    axis.text.x = element_text(angle = 45, hjust = 1)
  )
```



📌 Key Findings from Monthly User Count Analysis

✓ **Annual members consistently outnumber casual riders**, indicating a strong base of recurring users.

✓ **Ridership drops significantly during winter months**, with casual riders showing the steepest decline.

✓ **Casual riders prefer electric bikes from June to September**, likely due to summer tourism and increased outdoor activities.

✓ **Docked bikes are almost exclusively used by casual riders**, with only **one recorded annual member ride** in January. However, their overall usage remains **low** compared to classic and electric bikes.

📌 Summary of This Step

✓ **Counted total rides per month** by user and bike type.
✓ **Saved the dataset** for further analysis.
✓ **Created a bar chart** to **visualize monthly user trends**

.

📌 Step 8: Daily Ride Trends Analysis

📁 **Extracting daily ride counts to analyze usage patterns across different days of the month.**

```
# 📅 Extract the day from 'started_at' and store it in a new 'Date'
column

  cyclist_2021 <- cyclist_2021 %>%

   mutate(Date = format(started_at, "%d"))


# 🔄 Convert 'Date' column to numeric for analysis

  cyclist_2021$Date <- as.numeric(cyclist_2021$Date)


# 📊 Group by user type, date, month, and bike type to count daily rides

  Daily_Trend <- cyclist_2021 %>%

   group_by(member_casual, Date, month, rideable_type) %>%
```

```
    summarise(total_rides = n(), .groups = "drop")


 # 📅 Ensure months are correctly ordered for clarity

   Daily_Trend <- Daily_Trend %>%

   mutate(month = factor(month, levels = month_levels)) %>%

   arrange(month)




 # 💾 Save the summarized daily ride data for further analysis

   write.csv(Daily_Trend, "Daily-Trend-cyclist2021.csv", row.names =
FALSE)
```

📌 Summary of This Step

✔ Extracted the day from the timestamp for daily trend analysis.
✔ Grouped the data by user type, bike type, and date to track daily ride variations.
✔ Saved the dataset for further visualization.

📌 Step 9: Weekday Analysis

📁 **Extracting and analyzing ride trends across different days of the week.**

```
# 📅 Extract the day of the week from 'started_at' (e.g., Mon, Tue, Wed)
  cyclist_2021 <- cyclist_2021 %>%
   mutate(Day = wday(started_at, label = TRUE, abbr = TRUE))

# 📊 Count total rides per weekday, grouped by user type, bike type, and
month
  Weekday_UserCount <- cyclist_2021 %>%
  group_by(Day, member_casual, rideable_type, Date, month) %>%
  summarise(
   count = n(),
   .groups = "drop"
 )

# 📊 Calculate the average number of rides per weekday across the
dataset
  Weekday_mean_Usercount <- Weekday_UserCount %>%
  group_by(Day, month, member_casual, rideable_type) %>%
  summarise(
   mean_count = mean(count, na.rm = TRUE),
   .groups = "drop"
 )

# 🎯 Round off mean values for better readability
  Weekday_mean_Usercount$mean_count <-
round(Weekday_mean_Usercount$mean_count)

# 📅 Ensure months are ordered correctly for better visualization
  Weekday_mean_Usercount <- Weekday_mean_Usercount %>%
  mutate(month = factor(month, levels = month_levels)) %>%
  arrange(month)

# 🔍 Check dataset structure after processing
  head(Weekday_mean_Usercount)
```

```
# 💾 Save the weekday analysis data for further visualization
  write.csv(Weekday_mean_Usercount, "Weekday_mean-cyclist2021.csv",
row.names = FALSE)
```

📌 `head(Weekday_mean_Usercount)` Output – **First Few Rows**

```
# A tibble: 6 × 5
  Day   month   member_casual rideable_type mean_count
  <ord> <fct>   <chr>         <chr>              <dbl>
1 Sun   January casual        classic_bike         271
2 Sun   January casual        docked_bike           70
3 Sun   January casual        electric_bike        231
4 Sun   January member        classic_bike        1198
5 Sun   January member        electric_bike        577
6 Mon   January casual        classic_bike         234
```

📌 Summary of This Step

✔ **Extracted weekday information** (`Mon, Tue, Wed...`) from timestamps.
✔ **Grouped rides by weekday, user type, bike type, and month** to analyze daily trends.
✔ **Calculated average weekday ridership** to identify key commuting and leisure patterns.
✔ **Saved the dataset** for further visualization and insights.

📌 Step 10: Time Range Analysis

📁 **Grouping rides into specific time intervals to analyze peak and off-peak usage.**

```
# ⏳ Extract hour from 'started_at' and categorize into time intervals
  cyclist_2021 <- cyclist_2021 %>%
  mutate(
   Hour = hour(started_at),  # Extract hour from timestamp
   Time_Range = case_when(
     Hour >= 0 & Hour < 2  ~ "0-2",
     Hour >= 2 & Hour < 4  ~ "2-4",
     Hour >= 4 & Hour < 6  ~ "4-6",
     Hour >= 6 & Hour < 8  ~ "6-8",
     Hour >= 8 & Hour < 10 ~ "8-10",
     Hour >= 10 & Hour < 12 ~ "10-12",
     Hour >= 12 & Hour < 14 ~ "12-14",
     Hour >= 14 & Hour < 16 ~ "14-16",
     Hour >= 16 & Hour < 18 ~ "16-18",
     Hour >= 18 & Hour < 20 ~ "18-20",
     Hour >= 20 & Hour < 22 ~ "20-22",
     Hour >= 22 & Hour < 24 ~ "22-24"
   )
 )

# 📊 Group by time range, user type, month, and bike type to count rides
  Time_range <- cyclist_2021 %>%
  group_by(Time_Range, member_casual, month, rideable_type) %>%
  summarise(count = n(), .groups = "drop")

# ⏳ Define the correct order for time intervals
  time_levels <- c("0-2", "2-4", "4-6", "6-8", "8-10", "10-12",
                   "12-14", "14-16", "16-18", "18-20", "20-22", "22-24")

# 🔄 Convert 'Time_Range' into an ordered factor for proper arrangement
  Time_range <- Time_range %>%
  mutate(Time_Range = factor(Time_Range, levels = time_levels, ordered =
TRUE)) %>%
```

```
  arrange(Time_Range)

 # 🗓️ Order 'month' correctly for better visualization
   Time_range <- Time_range %>%
   mutate(month = factor(month, levels = month_levels)) %>%
   arrange(month)

 # 🔍 Check dataset structure after processing
   head(Time_range)

 # 💾 Save the processed time range dataset for analysis
   write.csv(Time_range, "TimeInterval-cyclist2021.csv", row.names =
FALSE)

 # 💾 Save the updated cyclist dataset with time intervals
   write.csv(cyclist_2021, "cyclist-2021-v2.csv", row.names = FALSE)
```

📌 `head(Time_range)` Output – **First Few Rows**

```
# A tibble: 6 × 5
  Time_Range member_casual month   rideable_type count
  <ord>      <chr>         <fct>   <chr>         <int>
1 0-2        casual        January classic_bike    182
2 0-2        casual        January docked_bike      57
3 0-2        casual        January electric_bike   227
4 0-2        member        January classic_bike    388
5 0-2        member        January electric_bike   321
6 2-4        casual        January classic_bike     75
```

📌 Summary of This Step

✓ Extracted hour from timestamps and grouped rides into 12 time intervals.
✓ Grouped rides by time range, user type, bike type, and month to identify peak hours.
✓ Saved the dataset for further visualization and analysis.

📌 Step 11: Extracting Chicago Hourly Weather Data

📂 Fetching hourly weather data for Chicago to analyze its impact on ride patterns.

```r
# 🌍 Get a list of available NOAA weather stations
  stations <- getMeta()

# 🌧️ Import weather data for CHICAGO MIDWAY INTL ARPT (Station Code:
725340-14819)
  weather_2021 <- importNOAA(year = 2021, code = "725340-14819")

# 🔍 View the first few rows of the dataset
  View(weather_2021)

# 💾 Save the extracted weather data for further analysis
  write.csv(weather_2021, "chicago_2021.csv", row.names = FALSE)
```

📷 Screenshot of Weather Dataset (`View(weather_2021)`)

	code	station	date	latitude	longitude	elev	ws	wd	air_temp	atmos_pres	visibility	dew_point	RH	ceil_hgt	cl_1	cl_2	cl_3	cl
1	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 00:00:00	41.78412	-87.75514	185.8	0.000000	NA	-3.3000000	1029.10	14484.000	-7.8000000	71.37720	6096.0000	2.000000	8.000000	NA	8.000
2	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 01:00:00	41.78412	-87.75514	185.8	1.500000	130.000000	-3.3000000	1029.10	14484.000	-7.8000000	71.37720	6096.0000	7.000000	NA	NA	7.000
3	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 02:00:00	41.78412	-87.75514	185.8	1.500000	80.000000	-3.3000000	1028.80	16093.000	-7.8000000	71.37720	22000.0000	4.000000	NA	NA	4.000
4	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 03:00:00	41.78412	-87.75514	185.8	0.750000	80.000000	-3.3000000	1028.85	15242.000	-7.5000000	73.04932	22000.0000	1.000000	4.000000	NA	2.000
5	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 04:00:00	41.78412	-87.75514	185.8	1.050000	170.000000	-3.3500000	1029.00	15288.500	-7.2000000	75.06483	625.0000	7.000000	7.000000	NA	7.000
6	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 05:00:00	41.78412	-87.75514	185.8	0.000000	NA	-3.0500000	1028.90	16093.000	-7.2000000	73.37313	7620.0000	3.000000	7.000000	NA	7.000
7	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 06:00:00	41.78412	-87.75514	185.8	2.100000	90.000000	-3.3000000	1027.50	12875.000	-7.2000000	74.72144	7620.0000	7.000000	NA	NA	7.000
8	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 07:00:00	41.78412	-87.75514	185.8	2.600000	90.000000	-3.3000000	1027.20	12875.000	-6.7000000	77.61223	7620.0000	7.000000	NA	NA	7.000
9	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 08:00:00	41.78412	-87.75514	185.8	185.8 000	120.000000	-2.8000000	1026.70	16093.000	-6.1000000	78.28228	7620.0000	7.000000	NA	NA	7.000
10	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 09:00:00	41.78412	-87.75514	185.8	3.433333	137.031681	-2.6000000	1026.90	14989.333	-5.9333333	78.11893	427.0000	7.500000	7.000000	NA	7.500
11	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 10:00:00	41.78412	-87.75514	185.8	3.350000	93.854454	-2.2000000	1025.60	13679.500	-6.1000000	74.92121	4572.0000	3.000000	7.000000	NA	7.000
12	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 11:00:00	41.78412	-87.75514	185.8	4.600000	70.000000	-1.7000000	1023.10	16093.000	-5.6000000	75.01099	3962.0000	2.000000	7.000000	NA	7.000
13	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 12:00:00	41.78412	-87.75514	185.8	6.200000	60.000000	-1.1000000	1022.10	16093.000	-5.0000000	75.11788	3048.0000	2.000000	8.000000	NA	8.000
14	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 13:00:00	41.78412	-87.75514	185.8	5.400000	60.000000	-0.8500000	1021.80	15288.500	-4.4000000	77.15920	396.0000	7.000000	8.000000	NA	8.000
15	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 14:00:00	41.78412	-87.75514	185.8	7.200000	80.000000	0.0000000	1021.00	11265.000	-3.9000000	75.31151	366.0000	7.000000	7.000000	8	8.000
16	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 15:00:00	41.78412	-87.75514	185.8	6.825000	68.269240	0.0000000	1020.60	9992.000	-3.7500000	76.16396	366.0000	7.000000	8.000000	NA	8.000
17	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 16:00:00	41.78412	-87.75514	185.8	7.200000	65.000000	0.0000000	1018.50	11265.000	-3.1500000	79.59877	823.0000	4.000000	7.000000	8	8.000
18	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 17:00:00	41.78412	-87.75514	185.8	8.475000	55.458382	0.0000000	1014.30	6839.500	-3.0500000	80.19323	1433.0000	4.000000	7.000000	8	8.000
19	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 18:00:00	41.78412	-87.75514	185.8	7.700000	50.000000	0.5333333	1013.20	8047.000	-2.1333333	82.56263	305.0000	7.000000	7.666667	8	8.000
20	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 19:00:00	41.78412	-87.75514	185.8	7.950000	65.157633	0.6000000	1011.80	7242.000	-1.7000000	84.77316	274.0000	8.000000	NA	NA	8.000
21	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 20:00:00	41.78412	-87.75514	185.8	8.200000	60.000000	0.8500000	1010.50	2816.000	-1.4000000	85.10886	243.5000	8.000000	NA	NA	8.000
22	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 21:00:00	41.78412	-87.75514	185.8	7.700000	60.000000	1.1000000	1010.15	2808.000	-0.8500000	87.01161	183.0000	8.000000	NA	NA	8.000
23	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 22:00:00	41.78412	-87.75514	185.8	7.200000	60.000000	1.4000000	1009.80	3219.000	-0.3000000	88.60440	167.5000	8.000000	NA	NA	8.000
24	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-01 23:00:00	41.78412	-87.75514	185.8	4.600000	50.000000	1.7000000	1010.10	3219.000	0.0000000	88.63015	152.0000	8.000000	NA	NA	8.000
25	725340-14819	CHICAGO MIDWAY AIRPORT, IL US	2021-01-02 00:00:00	41.78412	-87.75514	185.8	5.100000	40.000000	1.7000000	1010.00	3219.000	-0.6000000	84.89662	183.0000	8.000000	NA	NA	8.000

📌 Available Weather Parameters & Their Usability

📌 The extracted dataset contains 23 weather-related variables, but only a few are useful for our analysis. The table below summarizes their relevance:

Column Name	Meaning	Should We Use It?	Reason
code	Station code	❌ No	Only useful for identifying the weather station (we are using just one).
station	Station name	❌ No	Similar to code, not needed for analysis.
date	Timestamp	✅ Yes	This will be used for merging with ride data.
latitude/ longitude	Location of the weather station	❌ No	We only have one station, so no variation.
elev	Elevation of station	❌ No	Not relevant for ridership patterns.
ws (Wind Speed)	Wind speed in m/s	✅ Yes	Can affect ride comfort and safety.
wd (Wind Direction)	Wind direction in degrees	❌ No	Less impactful compared to wind speed.
air_temp	Air temperature in °C	✅ Yes	Strong influence on ridership trends.
atmos_pres	Air pressure in hPa	❌ No	Not significantly affecting bike usage.
visibility	Horizontal visibility distance	❌ No	Limited impact on bike usage.
dew_point	Temperature at which condensation forms	❌ No	More relevant for weather forecasting than bike trends.
Relative Humidity	Humidity percentage	✅ Yes	Can impact ride comfort.
ceil_hgt	Height of lowest cloud layer	❌ No	Not relevant for bike users.
cl_1, cl_2, cl_3	Cloud types	❌ No	Not relevant for ride behavior.
cl	Cloud cover (0-8 scale)	❌ No	Indirect effect on ridership, not as significant as temperature/humidity.
cl_1_height, cl_2_height, cl_3_height	Cloud layer altitudes	❌ No	Irrelevant for cyclists.
precip_6	Precipitation measure in 6 inches	❌ No	Due to single-station limitation (explained below).

pwc (Present Weather Conditions)	Weather conditions (fog, rain, etc.)	❌ No	Similar limitation as precip_6, data may not be reliable for entire city.

⚠️ Why We Are Not Using `precip_6` (Precipitation measure in 6 inches)?

🚨 **Limitation:** The precipitation (`precip_6`) data is sourced **only from a single station** (Chicago Midway International Airport).
➡ **Why is this an issue?** Since precipitation varies across different areas in Chicago, using data from a **single weather station** may not accurately represent rainfall or snowfall across the **entire city**.
➡ **Impact:** Relying on this data could lead to **misinterpretations** about how rain/snow affects ridership trends.

📌 **Solution:** To ensure reliability, we will **only use air temperature, humidity, and wind speed**, which are more **consistent citywide**.

📌 Key Observations from Weather Data

✓ **Hourly weather data is available**, allowing us to study its effect on ridership patterns.
✓ **Temperature and humidity fluctuate significantly**, which could impact the number of rides.
✓ **Wind speed is an important factor**, as strong winds may discourage biking.

📌 Summary of This Step

✓ **Extracted hourly weather data for Chicago using NOAA API.**
✓ **Filtered only the most relevant weather variables** (temperature, humidity, wind speed, timestamp).
✓ **Explained why precipitation data is not included** due to a single-station limitation.
✓ **Saved the dataset for further analysis and integration with ride data.**

📌 Step 12: Joining Weather Data with Ride Data

📁 **Integrating weather conditions with bike ride data to analyze the impact of weather on ridership trends.**

```
# 🔑 Create a common join key by rounding timestamps to the nearest hour
  cyclist_2021 <- cyclist_2021 %>%
   mutate(join_key = floor_date(started_at, unit = "hour"))

  weather_2021 <- weather_2021 %>%
   mutate(join_key = floor_date(date, unit = "hour"))

# 🎯 Select only necessary weather columns for merging
  weather_cols <- c("join_key", "air_temp", "RH", "ws")
  chicago_weather <- weather_2021 %>% select(all_of(weather_cols))

# 🔗 Perform a left join to integrate weather data into ride dataset
  cyclist_2021 <- left_join(cyclist_2021, chicago_weather, by =
"join_key")

# 🗑 Remove the temporary join_key column (not needed after merging)
  cyclist_2021 <- cyclist_2021 %>% select(-join_key)

# 🔍 Check dataset structure after merging
  str(cyclist_2021)
```

📌 `str(cyclist_2021)` Output – **First Few Rows**

```
tibble [5,595,063 × 22] (S3: tbl_df/tbl/data.frame)
 $ ride_id           : chr [1:5595063] "E19E6F1B8D4C42ED"
"DC88F20C2C55F27F" "EC45C94683FE3F27" "4FA453A75AE377DB" ...
 $ rideable_type     : chr [1:5595063] "electric_bike" "electric_bike"
"electric_bike" "electric_bike" ...
```

```
 $ started_at        : POSIXct[1:5595063], format: "2021-01-23 16:14:19" "2021-01-27 18:43:08" "2021-01-21 22:35:54" "2021-01-07 13:31:13" ...
 $ ended_at          : POSIXct[1:5595063], format: "2021-01-23 16:24:44" "2021-01-27 18:47:12" "2021-01-21 22:37:14" "2021-01-07 13:42:55" ...
 $ start_station_name: chr [1:5595063] "California Ave & Cortez St" "California Ave & Cortez St" "California Ave & Cortez St" "California Ave & Cortez St" ...
 $ start_station_id  : chr [1:5595063] "17660" "17660" "17660" "17660" ...
 $ end_station_name  : chr [1:5595063] NA NA NA NA ...
 $ end_station_id    : chr [1:5595063] NA NA NA NA ...
 $ start_lat         : num [1:5595063] 41.9 41.9 41.9 41.9 41.9 ...
 $ start_lng         : num [1:5595063] -87.7 -87.7 -87.7 -87.7 -87.7 ...
 $ end_lat           : num [1:5595063] 41.9 41.9 41.9 41.9 41.9 ...
 $ end_lng           : num [1:5595063] -87.7 -87.7 -87.7 -87.7 -87.7 ...
 $ member_casual     : chr [1:5595063] "member" "member" "member" "member" ...
 $ month             : chr [1:5595063] "January" "January" "January" "January" ...
 $ Total_time_secs   : num [1:5595063] 625 244 80 702 43 ...
 $ Date              : num [1:5595063] 23 27 21 7 23 9 4 14 9 24 ...
 $ Day               : Ord.factor w/ 7 levels "Sun"<"Mon"<"Tue"<..: 7 4 5 5 7 7 2 5 7 1 ...
 $ Hour              : int [1:5595063] 16 18 22 13 2 14 5 15 9 19 ...
 $ Time_Range        : chr [1:5595063] "16-18" "18-20" "22-24" "12-14" ...
 $ air_temp          : num [1:5595063] -3.3 -3.3 2.8 2.8 -9.4 -0.6 -2.8 2.25 -0.85 1.1 ...
 $ RH                : num [1:5595063] 46.3 77.6 52.4 52.4 53.4 ...
 $ ws                : num [1:5595063] 5.1 5.1 8.8 4.1 3.1 4.1 3.6 4.35 3.6 2.6 ...
```

📌 Why We Used a Left Join?

📌 **A left join ensures that all ride data remains intact, even if some timestamps don't have corresponding weather data.**

✓ What is `join_key`?

- Since our weather data is recorded **hourly**, we rounded the `started_at` timestamp to the **nearest hour** using `floor_date()`.

- This allows us to match each ride with the closest available weather conditions.

✓ **Why remove `join_key` after merging?**

- It's only needed for merging, and keeping it would take up unnecessary space in the dataset.

📌 Key Observations After Merging

✓ **Each ride now includes weather conditions** (temperature, humidity, and wind speed).
✓ **Total columns increased from 13 to 16**, with weather data fully integrated.
✓ **Some rides may still have missing weather values**, which we'll address in the next step.

📌 Summary of This Step

✓ **Created a common timestamp (`join_key`) for merging hourly weather data with ride data.**
✓ **Performed a left join** to ensure all ride data remains intact.
✓ **Integrated `air_temp`, `RH`, and `ws`** into the cyclist dataset.
✓ **Checked dataset structure to confirm successful merging.**

📌 Step 13: Handling Missing Weather Data

📁 **Identifying and cleaning missing weather values to maintain data accuracy.**

📌 `colSums(is.na(cyclist_2021))` Output – **Missing Values**

```
           ride_id           0
     rideable_type           0
        started_at           0
          ended_at           0
start_station_name     690,809
  start_station_id     690,806
  end_station_name     739,170
    end_station_id     739,170
         start_lat           0
         start_lng           0
           end_lat       4,771
           end_lng       4,771
     member_casual           0
             month           0
   Total_time_secs           0
              Date           0
               Day           0
              Hour           0
        Time_Range           0
          air_temp           0
                RH           0
                ws           0
```

📌 Key Observations:

- **12.34% of rides have missing `start_station_name` and `start_station_id`.**
- **13.21% of rides have missing `end_station_name` and `end_station_id`.**
- **0.08% of rides have missing `end_lat` and `end_lng`.**
- **No missing values in weather-related columns (`air_temp`, `RH`, `ws`).**

📌 Handling Missing Station Names & IDs

📌 **Since this dataset covers the entire city of Chicago, knowing the station names isn't always necessary for understanding ride behavior.**

Possible Solutions:

1️⃣ **Leave missing station values as is** (✓ Current Approach)

- Since location-based analysis **isn't our main focus**, this won't significantly impact key findings.

2️⃣ **Estimate missing station names using lat/lng (❌ Not Feasible)**

- Would require external datasets with full station location details.

3️⃣ **Impute missing station names based on the most common start location (❌ May Add Bias)**

- Without exact station mapping, this could introduce errors.

📌 **Final Decision:**

- **We will not remove these rows**, as they still provide valuable insights on ridership trends, trip duration, and weather impact.

📌 Handling Missing `end_lat` and `end_lng` (0.08%)

🚨 **Why Not Remove These Rows?**

- **Only 4,771 rows (~0.08%) are affected**, making the impact negligible.
- These rides still provide data on **trip duration, ride type, and weather conditions.**

📌 **Final Decision:**

- **Keep these rows** in the dataset.

📌 Key Takeaways from Data Cleaning

✓ **Weather data is fully integrated with no missing values.**
✓ **Station names and IDs are missing in ~18% of rides, but we will keep them.**

✓ Latitude/longitude is missing in <0.1% of rides, which is negligible.
✓ No important ride behavior data is lost due to missing values.

📌 Summary of This Step

✓ Confirmed no missing weather values.
✓ Kept rides with missing station names and IDs (~13%) since they don't impact most analyses.
✓ Retained rides with missing `end_lat` and `end_lng` (~0.08%) since they have minimal effect.
✓ No major cleaning was required, and dataset integrity remains intact.

📌 Step 14: Preparing Weather Data for Analysis

📁 **Aggregating and structuring weather data to analyze its impact on ridership trends.**

```
# 🏷️ Extract year from 'started_at' column
  cyclist_2021$year <- year(cyclist_2021$started_at)

# 📊 Aggregating data for weather-based analysis
  weather <- cyclist_2021 %>%
   group_by(Date, Hour, month, year, rideable_type, member_casual,
air_temp, RH, ws) %>%
     summarise(Riders_count = n(), .groups = "drop")

# 📏 Rounding values for better visualization
  weather <- weather %>%
   mutate(
   month = factor(month, levels = month_levels),
   air_temp = round(air_temp),
   RH = round(RH),
   ws = round(ws) ) %>%
   arrange(month)

# 💾 Save cleaned weather dataset
  write.csv(weather,"cyclist-2021-weather.csv", row.names = FALSE)
```

📌 Why Are We Extracting the Year?

- The dataset contains **hourly ride data**, and extracting the `year` allows **multi-year comparisons** in future analyses.

📌 Why Group By `Date, Hour, Month, Year`?

- Since we want to see how **weather affects ridership**, we aggregate the data at an **hourly level** for deeper insights.
- This structure enables **hourly, daily, monthly, and yearly** analysis.

📌 Why Round Temperature, Humidity & Wind Speed?

- **Raw weather data** can have decimal variations that make charts harder to read.
- **Rounding simplifies visualization** while maintaining accuracy.

📌 **Why Save This Dataset?**

- This cleaned dataset will be used for **visualizing the relationship between weather and ridership trends**.

📌 Key Takeaways from Data Preparation

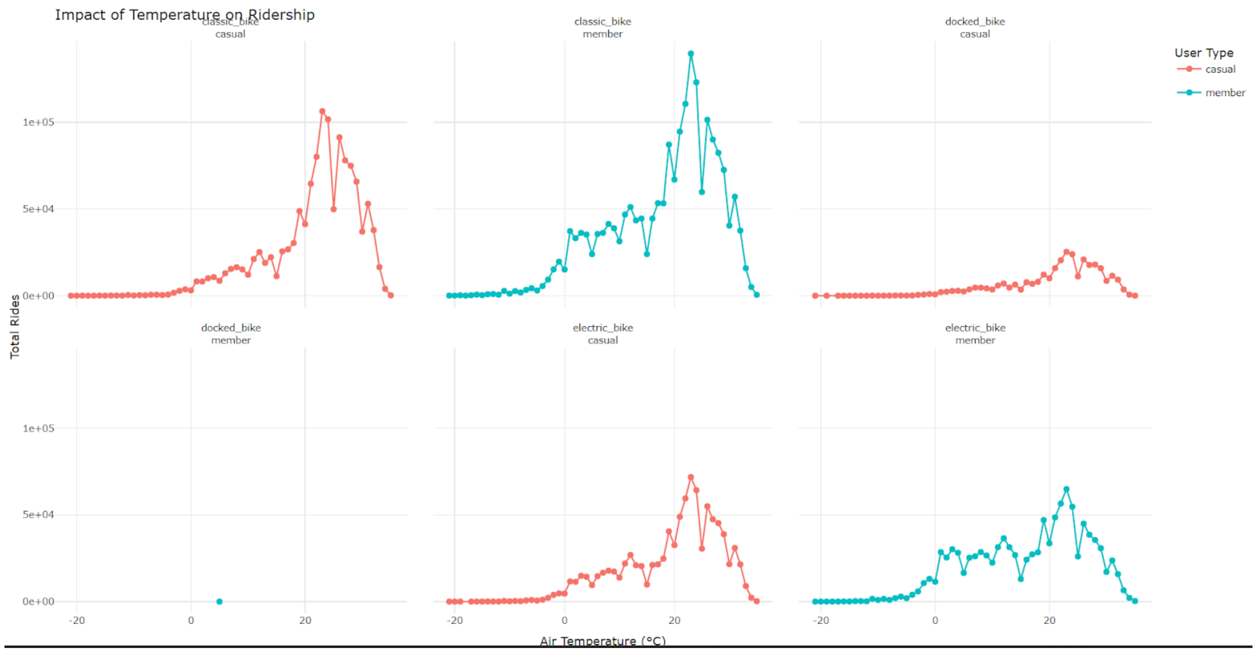
✓ Grouped ride data by date, hour, and weather conditions for precise analysis.
✓ Rounded weather values for better visualization.
✓ Ensured months are ordered correctly for time-series analysis.
✓ Dataset is now ready for further visual analysis (line charts, scatter plots, etc.)

📌 Step 15: Visualizing Weather Impact on Ridership

📂 Analyzing how temperature, humidity, and wind speed influence bike usage using line and scatter plots.

1️⃣ Relationship Between **Temperature** & **Ride Counts**

```
# 🔍 Grouping ride count by temperature

  weather_summary <- weather %>%

  group_by(air_temp, rideable_type, member_casual) %>%

  summarise(total_rides = sum(Riders_count), .groups = "drop")

# 📊 Line Chart: Temperature vs. Ride Count

  p <- ggplot(weather_summary, aes(x = air_temp, y = total_rides, colour
= member_casual, group = member_casual)) +

 geom_line() +

 geom_point() +

 facet_wrap(~ rideable_type + member_casual) +

 labs(

   title = "Impact of Temperature on Ridership",

   x = "Air Temperature (°C)",

   y = "Total Rides",

   colour = "User Type"

 ) +

theme_minimal()

ggplotly(p)  # Interactive plot
```



📌 Key Findings:

✓ Rides peak at 15-25°C, indicating comfortable weather is preferred.
✓ Fewer rides in extreme cold (<5°C) and heat (>30°C).
✓ Casual riders are more affected by temperature extremes than members.

2️⃣ Relationship Between **Humidity** & **Ride Counts**

```
# 🔍 Grouping ride count by relative humidity
weather_summaryRH <- weather %>%
  group_by(RH, rideable_type, member_casual) %>%
  summarise(total_rides = sum(Riders_count), .groups = "drop")

# 📊 Line Chart: Humidity vs. Ride Count
K <- ggplot(weather_summaryRH, aes(x = RH, y = total_rides, colour =
member_casual, group = member_casual)) +
  geom_line() +
  geom_point() +
  facet_wrap(~ rideable_type + member_casual) +
```

```
  labs(
    title = "Impact of Humidity on Ridership",
    x = "Relative Humidity (%)",
    y = "Total Rides",
    colour = "User Type"
  ) +
  theme_minimal()

ggplotly(K)  # Interactive plot
```

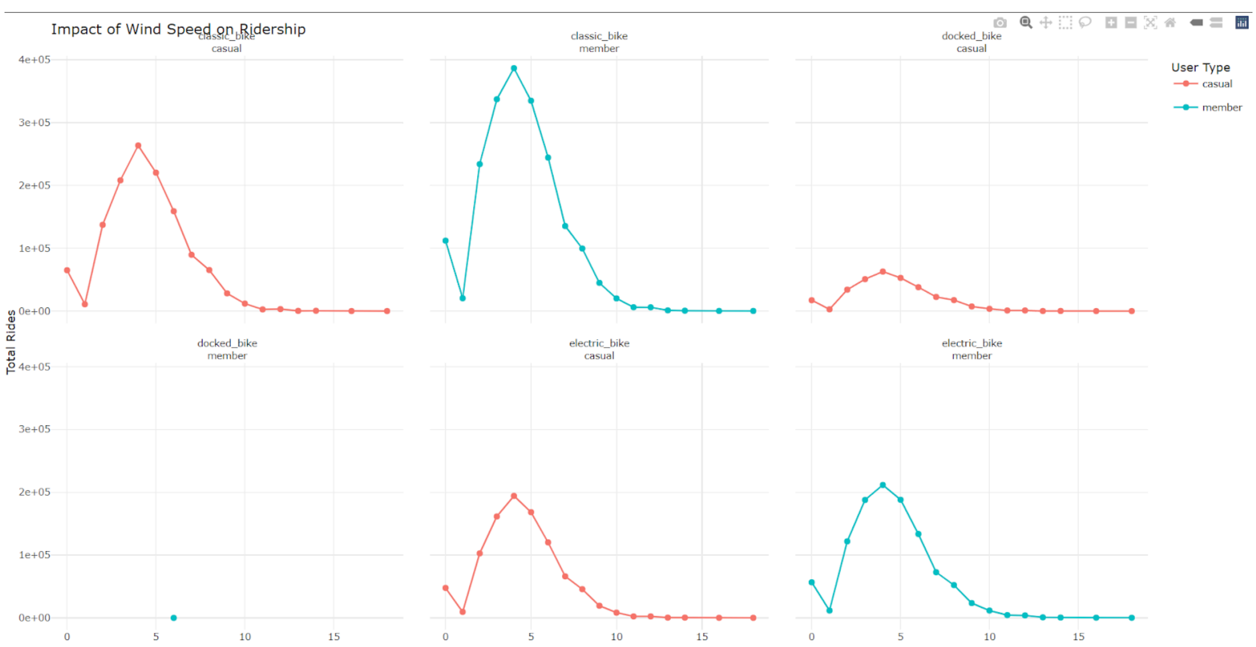
📌 Key Findings:

✔ **Rides are highest when humidity is between 40-70%.**
✔ **Humidity above 85% leads to a sharp drop in rides**, likely due to discomfort.
✔ **Casual riders are more affected by high humidity than annual members.**

---

3 Relationship Between **Wind Speed** & **Ride Counts**

```
# 🔍 Grouping ride count by wind speed
  weather_summaryws <- weather %>%
   group_by(ws, rideable_type, member_casual) %>%
   summarise(total_rides = sum(Riders_count), .groups = "drop")

# 📊 Line Chart: Wind Speed vs. Ride Count
  j <- ggplot(weather_summaryws, aes(x = ws, y = total_rides, colour =
member_casual, group = member_casual)) +
  geom_line() +
  geom_point() +
  facet_wrap(~ rideable_type + member_casual) +
  labs(
   title = "Impact of Wind Speed on Ridership",
   x = "Wind Speed (m/s)",
   y = "Total Rides",
   colour = "User Type"
  ) +
  theme_minimal()

ggplotly(j)  # Interactive plot
```



📌 Key Findings:

✓ **Wind speeds under 5 m/s have little impact on ridership.**
✓ **As wind speeds exceed 6-10 m/s, ride counts drop sharply**.
✓ **Casual riders reduce rides more drastically than members in windy conditions.**

📌 Summary of Weather Impact Analysis

✓ **Temperature Impact:** Ride counts are highest at **15-25°C**, lowest at **<5°C and >30°C**.
✓ **Humidity Impact:** Riders prefer **40-70% humidity**, with fewer rides in extreme humidity.
✓ **Wind Speed Impact:** Rides drop when wind speed **exceeds 7-10 m/s**, affecting casual riders more.

📌 Extending the Analysis to Other Years (2022-2024)

📁 **The same data cleaning, EDA, and weather integration process was applied to the years 2022, 2023, and 2024 to ensure consistency in our analysis.**

📌 **Key Steps Repeated for Each Year:**
✓ **Data Merging & Cleaning** – Loaded and combined monthly CSV files.
✓ **Exploratory Data Analysis (EDA)** – Analyzed ride patterns based on time, user type, and bike type.
✓ **Weather Data Integration** – Merged ride data with weather parameters (temperature, humidity, wind speed).
✓ **Visualizations & Insights** – Plotted trends to understand seasonal and weather impacts on ridership.

💡 **This ensures a complete four-year trend analysis (2021-2024) for making data-driven business recommendations.**

3.2 Tableau Analysis

📌 Moving to Tableau Insights 🚀

Now that we've completed the **R-based analysis**, let's focus on **Tableau insights** by documenting the **visualizations, key takeaways, and their business significance.**

📌 **Next Steps:**
1️⃣ **Summarizing each Tableau visualization** (what it represents & how it helps).
2️⃣ **Explaining how filters & interactions work** in the dashboard.
3️⃣ **Key findings from each graph and their relevance to business decisions.**

📌 Understanding the Tableau Dashboard 🏆

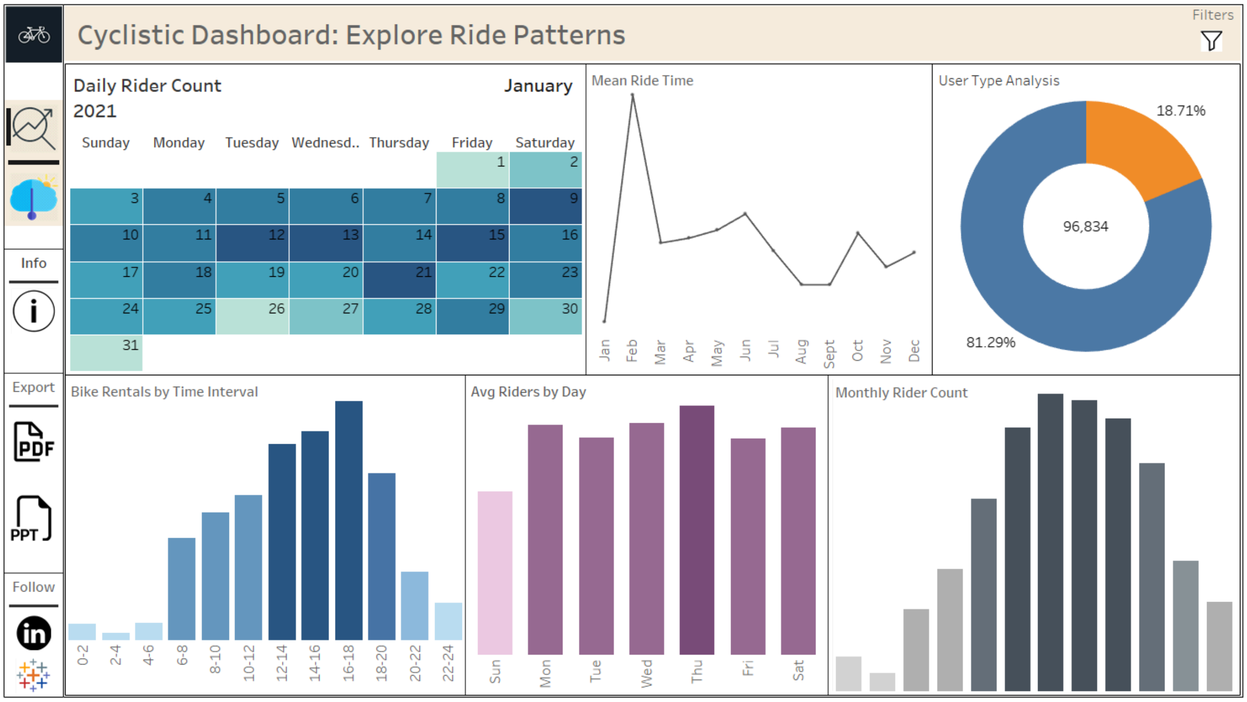
📁 The **Tableau dashboard** was designed to provide an **interactive way** to explore **ride patterns, user behavior, and weather impact** on Cyclistic's bike-share service.

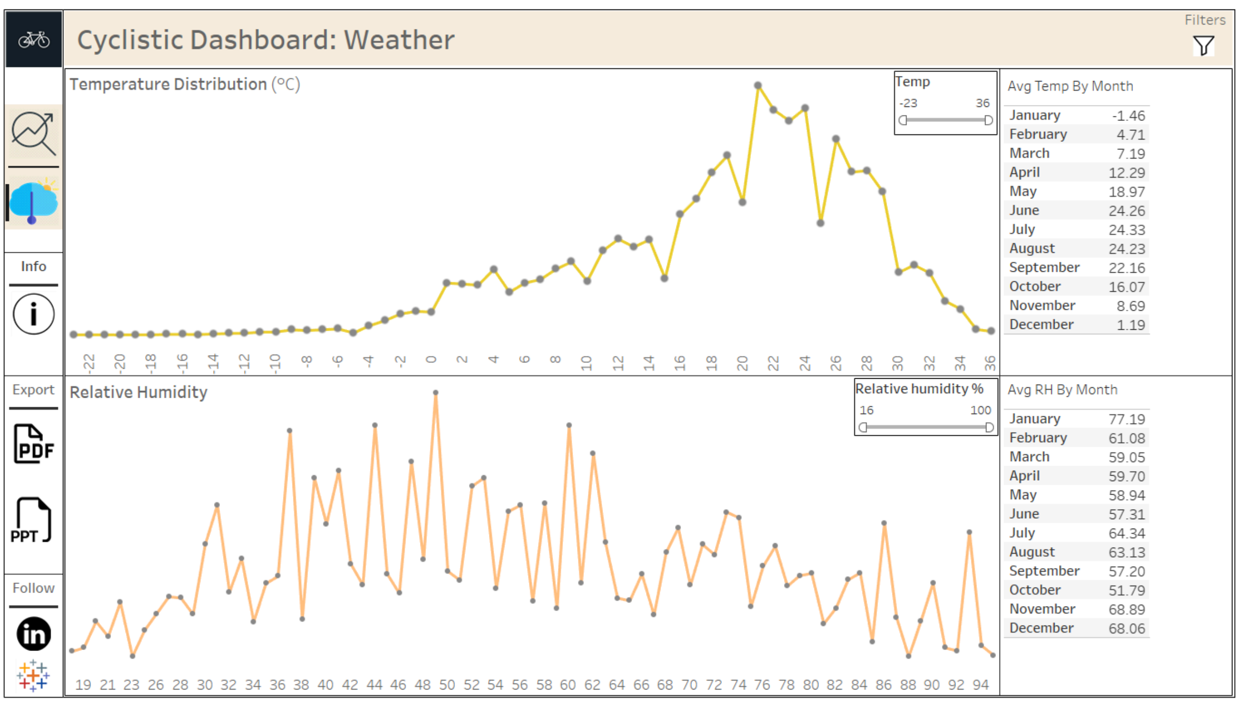
🌍 **View the Live Dashboard Here:** 🔗 Cyclistic Tableau Dashboard

---

📌 Dashboard Overview

The dashboard is divided into **two main sections:**
1️⃣ **Ride Pattern Dashboard** – Focuses on **ride trends** based on time, user type, and bike type.
2️⃣ **Weather Dashboard** – Analyzes how **weather conditions impact ridership**.

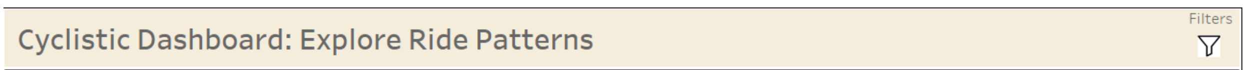

Cyclistic Dashboard: Explore Ride Patterns
Filters
Info
Export
Follow
Daily Rider Count
2021
January
Sunday
Monday
Tuesday
Wednesd..
Thursday
Friday
Saturday
Mean Ride Time
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sept
Oct
Nov
Dec
User Type Analysis
18.71%
96,834
81.29%
Bike Rentals by Time Interval
0-2
2-4
4-6
6-8
8-10
10-12
12-14
14-16
16-18
18-20
20-22
22-24
Avg Riders by Day
Sun
Mon
Tue
Wed
Thu
Fri
Sat
Monthly Rider Count


Cyclistic Dashboard: Weather
Filters
Info
Export
Follow
Temperature Distribution (°C)
Temp
-23
36
Relative Humidity
Relative humidity %
16
100
Avg Temp By Month
January -1.46
February 4.71
March 7.19
April 12.29
May 18.97
June 24.26
July 24.33
August 24.23
September 22.16
October 16.07
November 8.69
December 1.19
Avg RH By Month
January 77.19
February 61.08
March 59.05
April 59.70
May 58.94
June 57.31
July 64.34
August 63.13
September 57.20
October 51.79
November 68.89
December 68.06

📌 Dashboard Layout & Navigation

1️⃣ Title & Filter Section (Top Bar) 🎛️

📍 **Contains:**
✓ **Title:** "Cyclistic Dashboard: Explore Ride Patterns"
✓ **Filters:** User Type, Month, Bike Type, Year
✓ **Show/Hide Button** for filters (**funnel icon to toggle visibility**)



💡 **Purpose:**

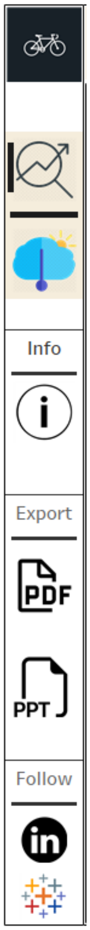
- Allows users to **customize their view** by selecting different filters.
- Helps in **comparing ride trends** between **members & casual users**.

2️⃣ Sidebar Navigation (Left Panel) 📌

📍 **Contains:**
✓ **Cyclistic Logo** for branding.
✓ **Navigation Options:**

- **Ride Patterns** (Main dashboard)
- **Weather Impact** (Switches to weather-based analysis)

✓ **Information (ℹ️) Button:** Provides **dashboard details**.
✓ **Social Media Links** (LinkedIn & Tableau Public).
✓ **Export Button:** Allows users to **download insights in PDF or PPT**.



💡 **Purpose:**

- Provides **easy navigation** between different sections.
- Enhances user experience with **structured layout & additional resources**.

📌 Main Dashboard Sections

1️⃣ Ride Pattern Dashboard 📊

✓ Heat Map (Calendar Format) 📅

- Displays **daily ride count trends** for each day of the month.
- Helps identify **high-traffic days & seasonal patterns**.

✓ Bar Graphs Section 📊 (Three Graphs)

- **Bike Rentals by Time Interval** (Shows peak riding hours).
- **Average Riders by Day** (Comparison of daily ride patterns).
- **Monthly Rider Count** (Tracks seasonal variations).

✓ Mean Ride Time (Line Graph) 📈

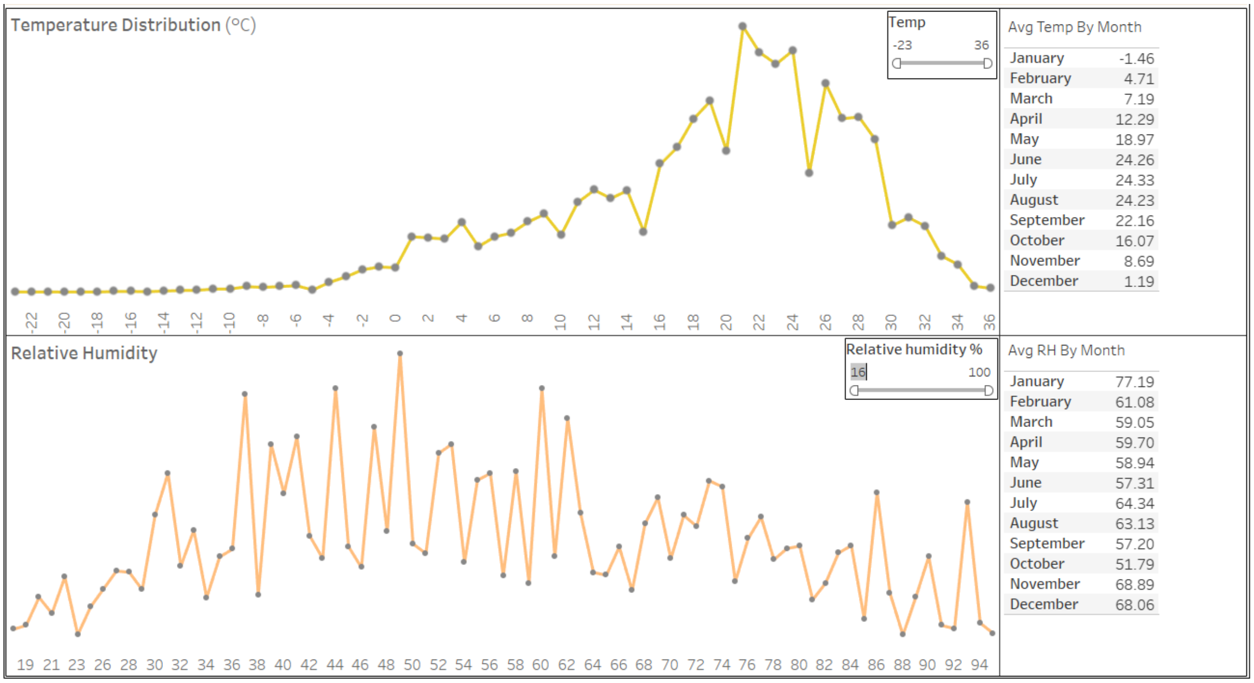
- Shows **average ride duration per month**.

✓ Donut Chart 🍩

- Displays **total ride count** with a **breakdown of casual vs. member riders**.

---

2 Weather Dashboard 🌧️



✓ Temperature vs. Ridership (Line + Point Chart)

- Shows **how ride count varies with temperature**.

✓ Humidity vs. Ridership (Line + Point Chart)

- Displays **how humidity levels affect ride demand**.

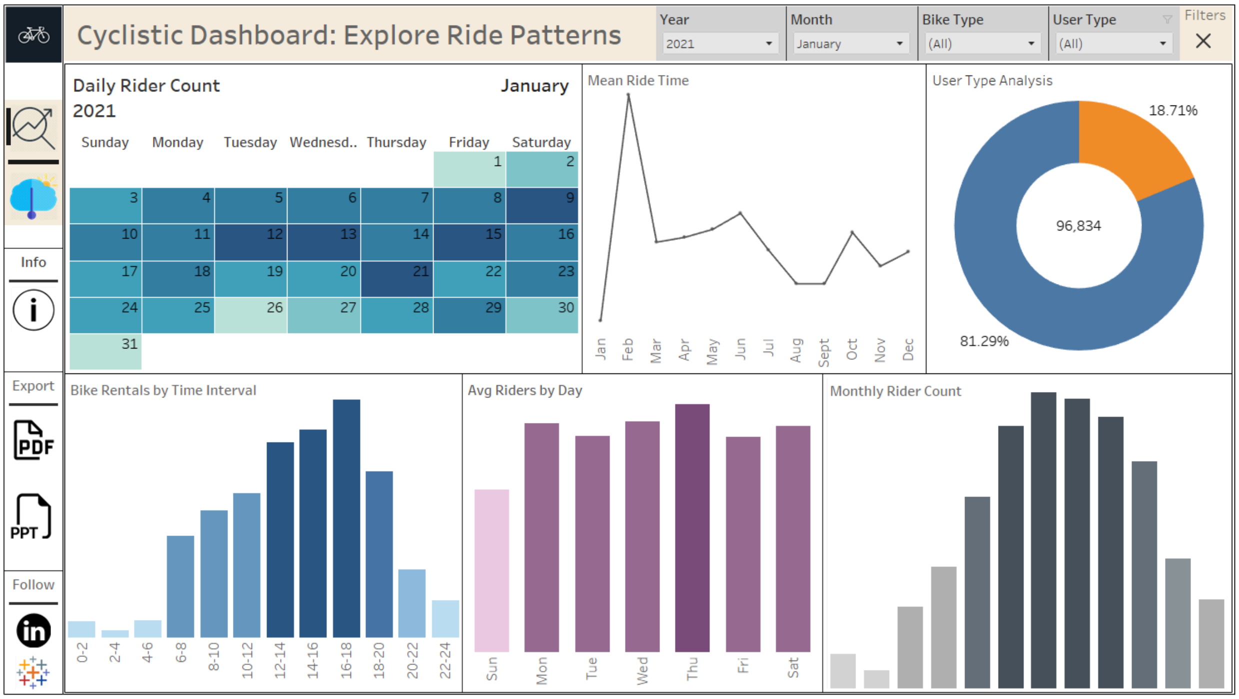
✓ Monthly Average Temperature & Humidity Table 📊

- Displays **the average temperature and humidity for each month**.
- Helps in **seasonal analysis** of how weather affects ridership trends.

💡 Purpose:

- Helps Cyclistic understand **weather-based fluctuations** in user behavior.
- Allows better **demand forecasting** based on **seasonal trends**.
- Can assist in **adjusting marketing & pricing strategies** for peak & off-peak months.

📌 Interactivity & Filters



✔ Filters Applied Across the Dashboard:

- **User Type (Member or Casual) –** Applied to most **visualizations.**
- **Bike Type (Classic, Electric, Docked) –** Applied where relevant**.**
- **Month (Single selection for clarity) –** Used in specific **time-based** analyses.
- **Year (Switch between 2021-2024) –** Allows for **multi-year comparisons.**

📌 **Note: Not all filters apply to every chart. A detailed breakdown will be provided in the individual visualization sections.**

✔ Clickable Navigation:

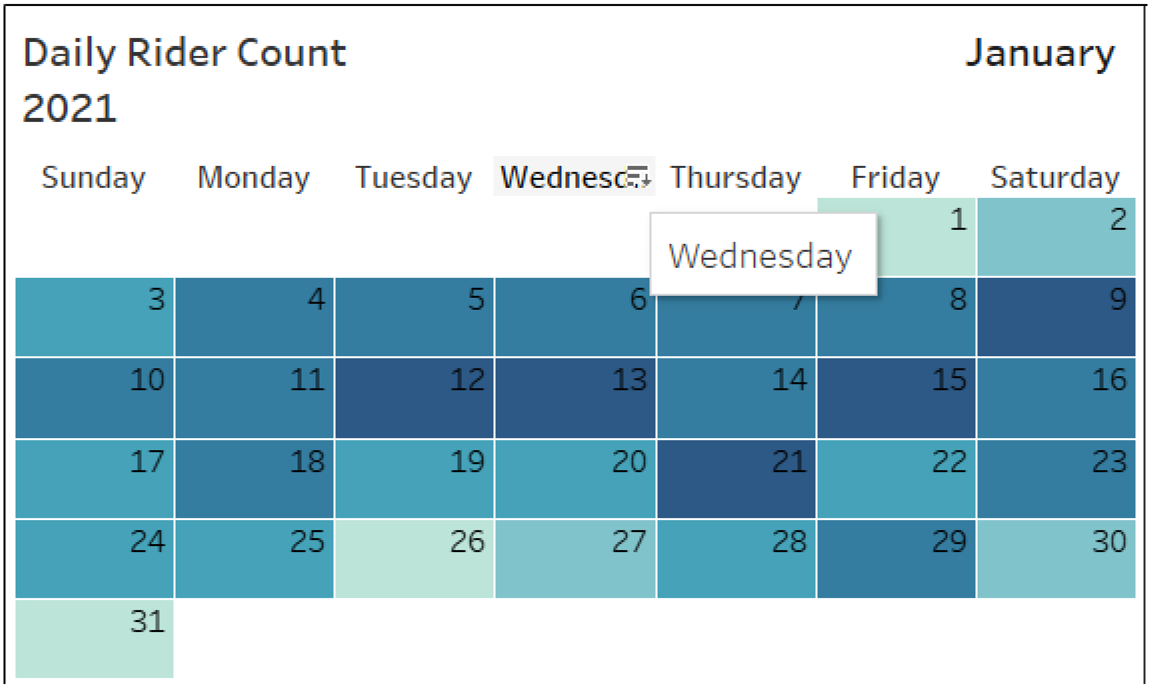
- Switch between **Ride Patterns & Weather Dashboard** easily.
- **Export Reports** for offline use.

📌 Summary

✅ The **Tableau dashboard provides an interactive and dynamic way** to explore Cyclistic's ridership trends.
✅ Users can **filter insights** based on user type, bike type, and time to make **data-driven**

decisions.
✅ **Weather analysis helps** in understanding how **climate conditions influence ride demand**.
✅ **Navigation & Export features enhance usability** for stakeholders.

📌 Heat Map (Calendar Format) – Tableau Visualization 📅



📌 What Does This Chart Show?

✓ This **heat map** represents **daily ride counts** throughout the year.
✓ Each **cell corresponds to a day**, color-coded based on the **number of rides** recorded.
✓ **Darker shades** indicate **higher ridership**, while **lighter shades** show **lower ride activity**.
✓ The heat map provides a **month-wise breakdown** to highlight **seasonal patterns and daily variations**.

📌 Interactivity & Filters

✓ **User Type** (Casual or Member) – Compare behaviors between different user types.
✓ **Bike Type** (Classic, Electric, Docked) – Identify bike usage patterns.
✓ **Month** – Focus on specific months to track ride fluctuations.
✓ **Year** – Switch between **2021-2024** to analyze long-term trends.

📌 How to Use This Visualization?

- Use **filters** to explore how ridership trends **change by user type and bike type.**
- Observe **busiest vs. least busy days** over different months.
- Identify **seasonal patterns** and how **weekends vs. weekdays impact ridership.**

📌 Heat Map Insights & Analysis

Now that we understand **what the heat map represents**, let's analyze the **key findings from the data** based on:

1️⃣ **Top 50 Busiest Days** – When does ridership peak? What influences high demand?
2️⃣ **Top 50 Least Busy Days** – When is ridership lowest? What are the possible reasons?

🔹 Section 1: Top 50 Busiest Days Analysis

📌 **Objective:** Identify **when ridership is highest** and compare user behavior.

📅 2021: Top 50 Busiest Days

Category	Casual Riders	Annual Members
Total Data Points	1095	731
Total Rides (Top 50 Days)	**515,555** (20.39% of annual total: 2,529,005)	**482,672** (15.74% of annual total: 3,066,058)
Weekend vs. Weekday Split	**78% Weekends**	**80% Weekdays**
Bike Type Preference	**98% Classic Bike, 2% Electric Bike**	**100% Classic Bike**
Peak Months	**May to October** (August highest: **26%**)	**May to October** (September highest: **28%**)

Avg. Temperature During Peak Month	**25.47°C**	**23.89°C**
Busiest Single Day	**August 14 – 14,997 riders** 📌 *Fun fact: The Race Judicata took place in Lincoln Park!*	**September 15 – 10,782 riders** *(Second highest: August 14 – Possible impact from Race Judicata?)*

📅 2022: Top 50 Busiest Days

Category	Casual Riders	Annual Members
Total Data Points	1094	730
Total Rides (Top 50 Days)	**430,509** (18.54% of annual total: 2,322,032)	**421,824** (12.61% of annual total: 3,345,685)
Weekend vs. Weekday Split	**76% Weekends**	**90% Weekdays**
Bike Type Preference	**68% Electric Bike, 32% Classic Bike**	**76% Classic Bike, 24% Electric Bike**
Peak Months	**April to October** (July highest: **32%**)	**April to September, plus November** (June highest: **34%**)
Avg. Temperature During Peak Month	**25.6°C**	**24.16°C**

Busiest Single Day	**May 29 – 11,029 riders**	**May 14 – 9,361 riders**

📅 2023: Top 50 Busiest Days

Category	Casual Riders	Annual Members
Total Data Points	972	730
Total Rides (Top 50 Days)	**366,916** (17.82% of annual total: 2,059,179)	**432,034** (11% of annual total: 3,660,698)
Weekend vs. Weekday Split	**78% Weekends**	**94% Weekdays**
Bike Type Preference	**56% Electric Bike, 44% Classic Bike**	**68% Classic Bike, 32% Electric Bike**
Peak Months	**April to October** (July highest: **30%**)	**June to October** (August highest: **40%**)
Avg. Temperature During Peak Month	**24.27°C**	**23.59°C**
Busiest Single Day	**June 10 – 10,024 riders** 📌 *Fun fact: Andersonville Midsommarfest, a Swedish heritage festival, took place!*	**August 22 – 9,334 riders**

🗓️ 2024: Top 50 Busiest Days

Category	Casual Riders	Annual Members
Total Data Points	762	763
Total Rides (Top 50 Days)	**385,567** (17.92% of annual total: 2,151,658)	**417,933** (11.27% of annual total: 3,708,910)
Weekend vs. Weekday Split	**90% Weekends**	**98% Weekdays**
Bike Type Preference	**56% Classic Bike, 44% Electric Bike**	**82% Electric Bike, 18% Classic Bike**
Peak Months	**April to October** (June-August equally highest: **24% each**)	**May to October** (September highest: **26%**)
Avg. Temperature During Peak Month	**24.27°C**	**22.16°C**
Busiest Single Day	**July 27 – 9,283 riders**	**September 25 – 9,019 riders**

📌 Key Takeaways from the Top 50 Busiest Days

📊 **Casual Riders:**
✓ **Peak on weekends** (76%-90%).
✓ **Prefer summer months (June-August).**

✓ **Shift from classic to electric bikes over time.**
✓ **Some of the busiest days align with major events (e.g., Race Judicata, Midsommarfest).**

📊 **Annual Members:**
✓ **Peak on weekdays** (80%-98%).
✓ **Steady ridership across seasons.**
✓ **Strong preference for classic bikes, but electric bikes gained popularity in 2024.**

🔹 Section 2: Top 50 Least Busy Days Analysis

📌 **Objective:** Identify **when ridership is at its lowest**, analyze possible reasons, and compare Casual vs. Annual users.

📅 2021: Top 50 Least Busy Days

Category	Casual Riders	Annual Members
Total Rides (Top 50 Days)	**1,056** (0.04% of annual total: 2,529,005)	**19,007** (0.62% of annual total: 3,066,058)
Bike Type Preference	**74% Docked Bike, 26% Classic Bike**	**72% Electric Bike, 28% Classic Bike**
Seasonal Trend	**January to March, December** (*February highest: 68%*)	**January to March** (*February highest: 74%*)
Avg. Temperature During Lowest Ridership Month	**-5.02°C (February)**	**-5.02°C (February)**

Fun Fact	**-**	**Only 1 annual member used a docked bike all year (Jan 13th).**

📅 2022: Top 50 Least Busy Days

Category	Casual Riders	Annual Members
Total Rides (Top 50 Days)	**1,008** (0.04% of annual total: 2,322,032)	**38,712** (1.16% of annual total: 3,345,685)
Bike Type Preference	**96% Docked Bike, 4% Classic Bike**	**56% Electric Bike, 44% Classic Bike**
Seasonal Trend	**January, February, and December** (*January highest: 44%*)	**January, February, and December** (*January highest: 50%*)
Avg. Temperature During Lowest Ridership Month	**-4.67°C (January)**	**-4.67°C (January)**

📅 2023: Top 50 Least Busy Days

Category	Casual Riders	Annual Members
Total Rides (Top 50 Days)	**2,029** (0.1% of annual total: 2,059,179)	**71,487** (1.95% of annual total: 3,660,698)

Bike Type Preference	**100% Docked Bike**	**54% Classic Bike, 46% Electric Bike**
Seasonal Trend	**January to May, and August** (*January highest: 42%*)	**January, February, March, July, November, and December** (*December highest: 32%*)
Avg. Temperature During Lowest Ridership Month	**0.39°C (January)**	**4.9°C (December)**

⏩ 📌 **Important Note: Docked bikes were discontinued after 2023, meaning they do not appear in 2024 data.**

📅 2024: Top 50 Least Busy Days

Category	Casual Riders	Annual Members
Total Rides (Top 50 Days)	**11,338** (0.53% of annual total: 2,151,658)	**45,759** (1.23% of annual total: 3,708,910)
Bike Type Preference	**74% Classic Bike, 24% Electric Bike, 1% Electric Scooter (Only in Aug 31st & Sept)**	**48% Classic Bike, 42% Electric Bike, 10% Electric Scooter (Only in Aug & Sept)**
Seasonal Trend	**January, March, April, August, November, and December** (*January highest: 50%*)	**January to March, August, September, November, and December** (*January highest: 40%*)

Avg. Temperature During Lowest Ridership Month	**-1.46°C (January)**	**-1.46°C (January)**
Fun Fact	**Electric Scooters were introduced on August 31st but discontinued after September.**	**Annual members also used Electric Scooters, but only in Aug & Sept.**

📌 Key Takeaways from the Top 50 Least Busy Days

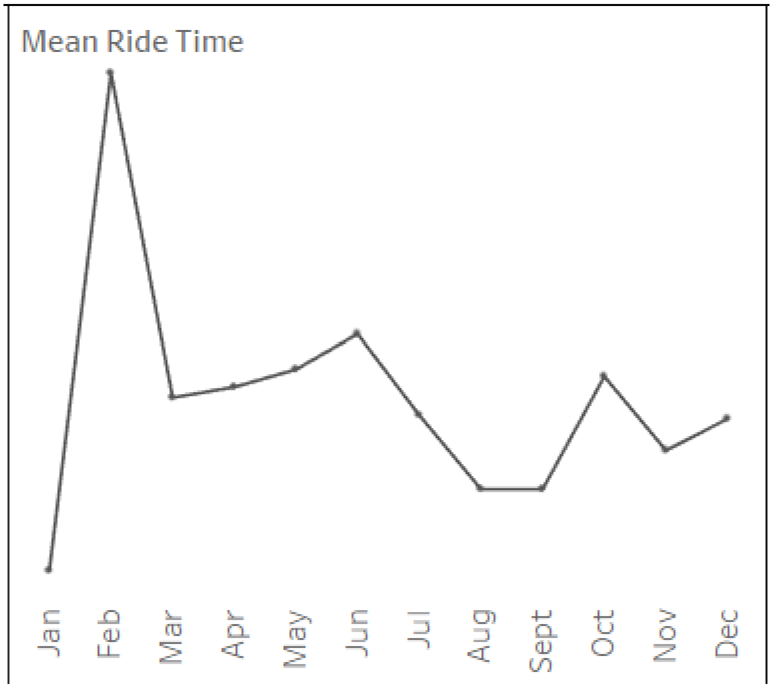
📊 Casual Riders:
✓ Low ridership mostly in winter (Jan-Feb) and late fall (Nov-Dec).
✓ Preferred docked bikes in earlier years, but switched to classic bikes after 2023.
✓ Electric Scooters saw minor usage in 2024 but were quickly discontinued.

📊 Annual Members:
✓ Low ridership follows similar trends (Jan-Mar, Nov-Dec).
✓ Higher adoption of electric bikes over the years.
✓ Electric Scooters were briefly used in 2024 but were never a major choice.

📌 Final Thoughts on the Heat Map Analysis

✅ Heat maps effectively visualize ride fluctuations across seasons, weekdays vs. weekends, and extreme weather conditions.
✅ Busiest days are concentrated in warm months (June-August), while lowest ridership occurs in cold months (January-February).
✅ Casual riders relied on docked bikes on low-ridership days until 2023, after which they switched to classic bikes.
✅ Annual members had a steady ride pattern, with a gradual shift to electric bikes in later years.
✅ Electric Scooters were tested in 2024 but discontinued after just a month.

📌 Mean Ride Time Analysis ⏳



📌 What Does This Chart Show?

✓ This **line graph** displays the **average ride duration** across months for different user types and bike types.
✓ Helps **analyze seasonal variations** and **changes in user behavior** over time.
✓ Provides insights into how **bike type** affects ride duration.
✓ Identifies whether **casual riders** take longer trips than **annual members** and if these patterns shift across years.

📌 Interactivity & Filters

✓ **User Type (Casual or Member)** – Compare how ride duration varies between casual users and annual members.
✓ **Bike Type (Classic, Electric, Docked, Electric Scooter)** – Understand how bike preference influences ride time.
✓ **Year (2021-2024)** – Observe long-term trends in ride duration.

📌 Yearly Mean Ride Time Overview

📅 2021: Ride Duration Trends

Casual Riders (Avg Ride Time in Minutes)

- **Classic Bike:** 30 min
- **Docked Bike:** 88 min
- **Electric Bike:** 19 min

Annual Members (Avg Ride Time in Minutes)

- **Classic Bike:** 14 min
- **Docked Bike:** 3 min *(Only 1 ride, insignificant)*
- **Electric Bike:** 13 min

📅 2022: Ride Duration Trends

Casual Riders

- **Classic Bike:** 29 min
- **Docked Bike:** 137 min
- **Electric Bike:** 15 min

Annual Members

- **Classic Bike:** 13 min
- **Docked Bike:** *(Discontinued)*
- **Electric Bike:** 11 min

📅 2023: Ride Duration Trends

Casual Riders

- **Classic Bike:** 31 min
- **Docked Bike:** 174 min
- **Electric Bike:** 13 min

Annual Members

- **Classic Bike:** 13 min
- **Docked Bike:** *(Discontinued)*

- **Electric Bike:** 11 min

📅 2024: Ride Duration Trends

Casual Riders

- **Classic Bike:** 37 min
- **Electric Scooter:** 24 min
- **Electric Bike:** 13 min

Annual Members

- **Classic Bike:** 14 min
- **Electric Scooter:** 14 min
- **Electric Bike:** 11 min

📌 Monthly Trends in Ride Duration

2021: Seasonal Trends

✅ **Classic Bike:** Ride durations were **longer in colder months**. When temperatures exceeded **20°C**, the ride time **decreased**.
✅ **Electric Bike:** Opposite trend—ride time was **higher in warmer months** (20-26°C range).
✅ **Annual Members:** Classic bike durations were **steady (12-15 min)** except for a **spike to 20 min in February** when the Avg temp was -5.02°C.

2022: Seasonal Trends

✅ **Classic Bike (Casual Riders):** Ride time **decreased** above **20°C**, but **Nov- Dec remained lowest** despite cool weather.
✅ **Electric Bike (Casual Riders):** Higher temperatures led to **longer rides**. **Nov- Dec had the lowest ride times**.
✅ **Annual Members:** Classic and electric bike durations **declined in Nov- Dec** and when temperatures neared **0°C**.

2023: Seasonal Trends

✅ **Classic Bike (Casual Riders): Reversed trend**—higher temperatures now led to **longer ride durations**, and **the Nov- Dec decline disappeared** (likely due to docked bike discontinuation in Aug 2023).
✅ **Electric Bike (Casual Riders):** Direct proportionality—higher temperatures led to **longer rides**.
✅ **Annual Members:** Ride durations **remained stable** but slightly favored **summer months**.

2024: Seasonal Trends

✅ **Classic Bike (Casual Riders):** Nov- Dec **had the shortest ride times**, while **Apr- July peaked** (possibly due to docked bike discontinuation).
✅ **Electric Bike (Casual Riders):** Ride durations **peaked in warmer months**.
✅ **Annual Members:** Classic bike durations were **consistent**, but electric bike durations **favored summer** while declining in **early-year months**.

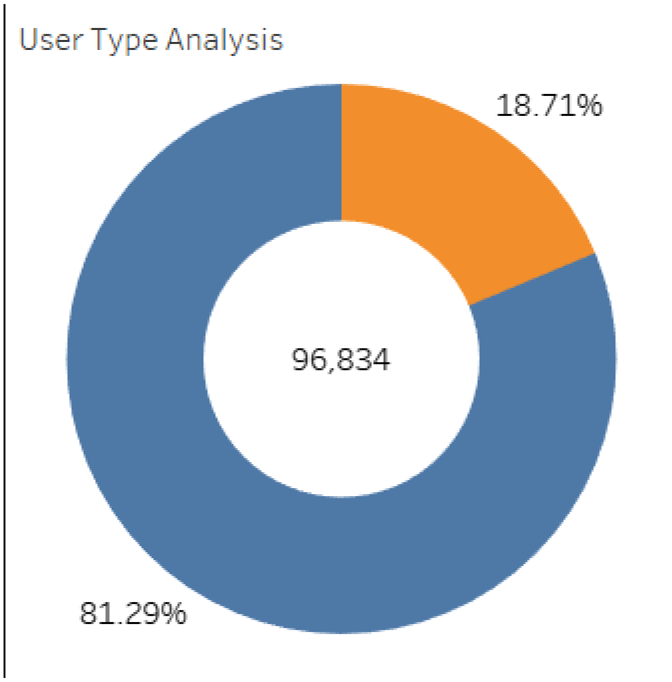
📌 Key Takeaways from the Mean Ride Time Analysis

📊 **Casual Riders:**
✓ **Ride durations are longer than annual members**, especially in **warmer months**.
✓ **Winter months significantly reduce ride time**, particularly for casual users.
✓ **Shift from docked bikes to classic bikes in 2024** increased casual riders' mean time.

📊 **Annual Members:**
✓ **Steady ride durations** across the year, with small fluctuations in extreme temperatures.
✓ **Increased electric bike usage in later years**, but ride time remained stable.

📌 **Business Impact:**
✓ Cyclistic can **optimize pricing strategies** for longer casual rides in warm months.
✓ The **discontinuation of docked bikes in 2024** resulted in **increased reliance on classic bikes for casual riders**.
✓ **Understanding seasonal ride time trends** can help in **bike availability planning** and **targeted promotions**.

📌 User Type Analysis: Casual vs. Annual Riders 🚴



📌 What Does This Chart Show?

✓ This **donut chart** represents the **total rides split between Casual and Annual Members**.
✓ The **inner circle** displays the **total ride count**, while the **outer rings** break it down by **user type**.
✓ Users can explore **how ridership varies across different years, bike types, and months.**

📌 Interactivity & Filters

✓ **Year (2021-2024)** – Switch between years to see membership trends over time.
✓ **Bike Type (Classic, Electric, Docked, Electric Scooter)** – Analyze which bike types are preferred by each user type.
✓ **Month (January to December)** – Observe how user distribution changes seasonally.

📌 General Trend

✓ During **winter months**, the ride distribution is **heavily dominated by annual members**, often exceeding **80% of total rides**.
✓ In **summer months**, casual riders contribute significantly more, making it a **closer split** (near **50-50**), but **annual members still maintain a slight lead.**

📌 Growth Over the Years 📈

User Type	Month	% change 21 to 22	22 to 23	23 to 24
casual	January	2.22%	116.03%	-38.86%
casual	February	111.39%	100.86%	9.64%
casual	March	6.96%	-30.80%	32.71%
casual	April	-7.46%	16.51%	-10.51%
casual	May	9.15%	-16.49%	-1.29%
casual	June	-0.44%	-18.38%	-0.02%
casual	July	-8.14%	-18.40%	-3.25%
casual	August	-13.02%	-13.32%	2.34%
casual	September	-18.47%	-11.82%	32.43%
casual	October	-18.76%	-15.27%	22.24%
casual	November	-5.76%	-2.36%	-5.42%
casual	December	-35.62%	15.10%	-25.75%
member	January	8.30%	76.30%	-19.88%
member	February	138.52%	56.52%	19.38%
member	March	34.40%	1.19%	11.53%
member	April	22.03%	14.08%	1.40%
member	May	29.02%	4.57%	2.13%
member	June	11.49%	4.56%	-2.11%
member	July	9.75%	4.52%	-1.80%
member	August	9.02%	7.86%	-5.03%
member	September	3.16%	0.02%	17.21%

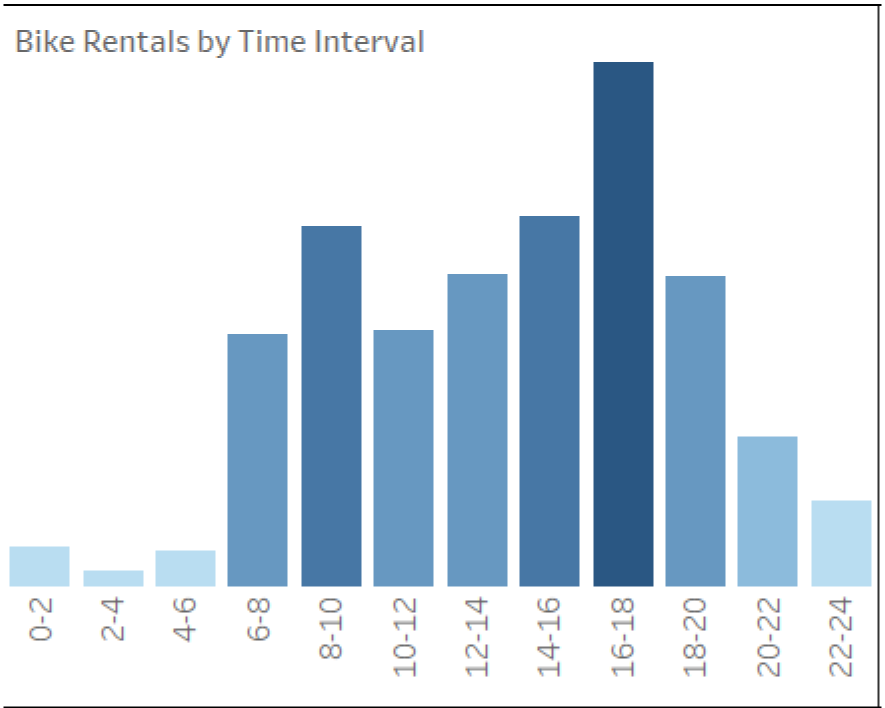
member	October	-6.49%	2.96%	11.05%
member	November	-6.36%	11.46%	-8.38%
member	December	-23.00%	25.92%	-18.80%

✓ The table above shows **year-over-year percentage change in ridership** for **casual and annual members**, broken down by month.

✓ Key Takeaways from Growth Trends:

- **Casual riders** saw **high fluctuations in winter** (e.g., **111.39% growth in February 2022**, but **35.62% decline in December 2021**).
- **Summer ridership for casual users has seen a downward trend over the years**, while **September 2024 rebounded with a 32.43% increase**.
- **Annual members consistently grew in winter, with February 2022 seeing a massive 138.52% increase.**
- **Growth rates slowed in late 2023 and 2024**, possibly indicating market saturation or shifting preferences.

📌 Bike Rentals by Time Interval ⏳ 📊



📌 What Does This Chart Show?

✓ This **bar chart** visualizes ride activity by **2-hour time intervals** throughout the day.
✓ It helps track **peak riding hours** and **low-traffic periods** for casual riders and annual members.
✓ The analysis focuses on **classic bikes and electric bikes**, as these were the primary options available.

📌 Interactivity & Filters

✓ **User Type (Casual vs. Member)** – Compare ride timing preferences for different users.
✓ **Bike Type (Classic vs. Electric Bike)** – Identify usage differences between bike types.
✓ **Month (January to December)** – Observe seasonal variations in riding patterns.
✓ **Year (2021-2024)** – Track how ridership behavior evolved over time.

📌 General Trends in Peak Usage Hours

✓ **Annual members** follow a **structured commute pattern**, riding **from 6 AM to 8 PM**, with a sharp **peak between 16-18 PM**.
✓ **Casual riders** show a **strong preference for 12-20 PM**, with an increasing shift toward **evening rides (18-22 PM) in later years**.
✓ **Late-night ridership (20-24 PM) was more significant among casual users, particularly in 2022 and 2023.**

📌 Yearly Breakdown & Patterns (2021-2024) 📅

📌 2021: Classic Bikes Dominate, Evening Rides Gain Popularity
🔹 **Annual Members → Consistent usage from 6 AM - 8 PM, with peak at 16-18 PM.**
🔹 **Casual Riders → Preferred 12-18 PM, but noticeable increase in late evening (18-22 PM) rides.**

📌 2022: Casual Riders Shift Towards Night Rides
🔹 **Annual Members → Same structured work commute pattern (6-20 PM), but evening usage (18-20 PM) grew.**
🔹 **Casual Riders → Higher evening demand (16-20 PM), with notable growth in late-night riding (20-24 PM).**

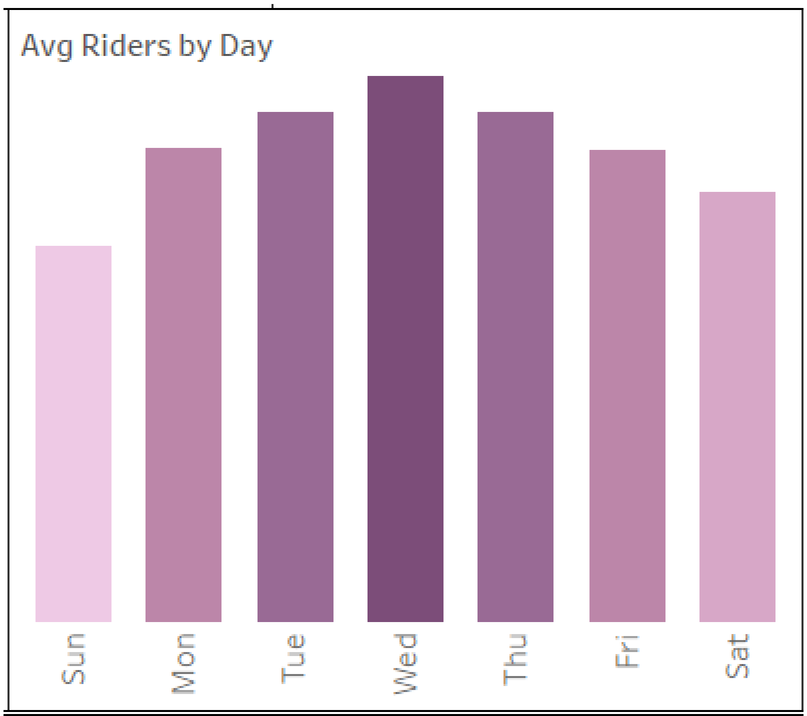
📌 2023: Electric Bike Popularity Increases, Casual Riders Extend Ride Time
🔹 **Annual Members → Continued dominance of work commute slots (6-20 PM).**
🔹 **Casual Riders → Strong ride volume between 12-22 PM, especially for electric bikes.**

📌 2024: Late Night Riding Declines, Peak Hours Stabilize
🔹 **Annual Members → Stable pattern from 6-20 PM, with a continued 16-18 PM peak.**
🔹 **Casual Riders → Peak usage remains 12-20 PM, but post-October, late-night rides (20-24 PM) drop.**

📌 Key Insights & Business Impact 🚀

✅ **Annual members consistently use bikes during commuting hours, making them predictable customers.**
✅ **Casual riders drive demand in afternoon and evening hours, signaling opportunities for promotions.**
✅ **Electric bike usage increased over the years, particularly in peak traffic hours.**
✅ **Understanding hourly trends helps optimize bike availability, maintenance schedules, and marketing strategies.**

📌 Average Riders by Day Analysis 📅 📊



📌 What Does This Chart Show?

✓ This **bar chart** visualizes the **average number of rides** taken on each day of the week.
✓ It helps identify **which days are most popular** for casual riders vs. annual members.

📌 Interactivity & Filters

✓ **User Type (Casual vs. Member)** – Compare ride preferences across weekdays & weekends.
✓ **Bike Type (Classic vs. Electric Bike)** – Understand bike preference variations.
✓ **Month (January to December)** – Analyze seasonal trends in weekday ridership.
✓ **Year (2021-2024)** – Track changes over time.

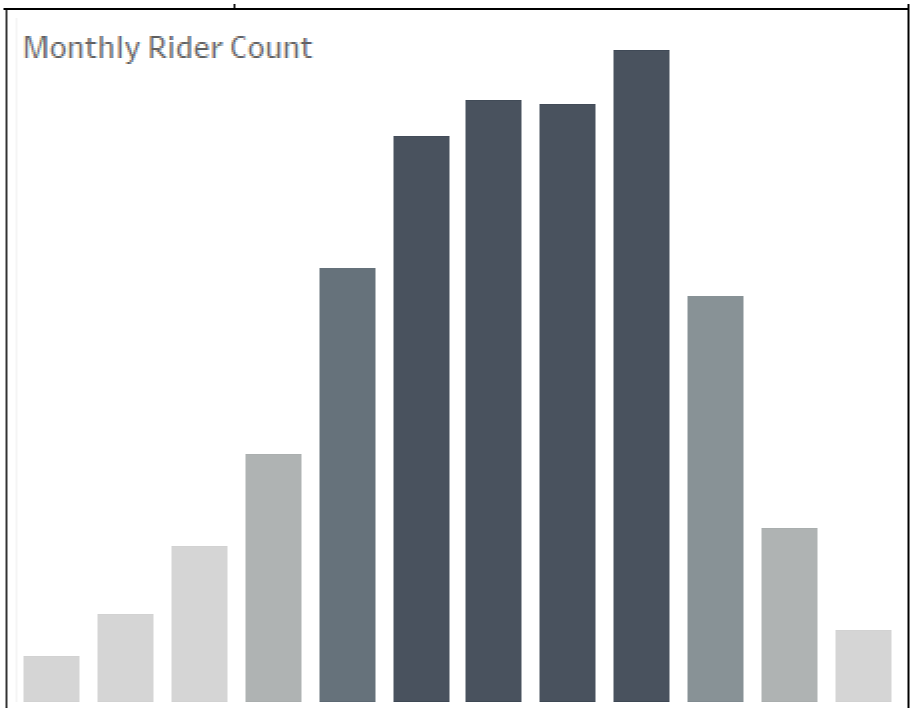
📌 Key Trends & Observations

✓ Casual riders dominate weekends (Saturday & Sunday).
✓ Annual members are the primary users on weekdays, with peaks on Tuesday to Thursday.
✓ Casual users show a sharp decline in ridership on Mondays and Fridays.
✓ Overall, ridership is lowest on Mondays for both user types.

📌 Business Implications 🚀

✅ **Weekend promotions** could be targeted at casual riders to **increase conversions to annual membership**.
✅ **Ensuring higher bike availability on weekdays** would cater to member commuters.
✅ **Mondays could be leveraged for maintenance and operational adjustments.**

📌 Monthly Rider Count Trends 📆 📊



📌 What Does This Chart Show?

✓ This **bar chart** displays the **total number of rides per month** for both user types.
✓ It highlights **seasonal ridership trends** and **growth patterns** for casual riders vs. annual members.

📌 Interactivity & Filters

✓ **User Type (Casual vs. Member)** – Compare ride volume trends.
✓ **Bike Type (Classic vs. Electric Bike)** – Understand bike preference changes.
✓ **Year (2021-2024)** – Track long-term trends in ridership growth.

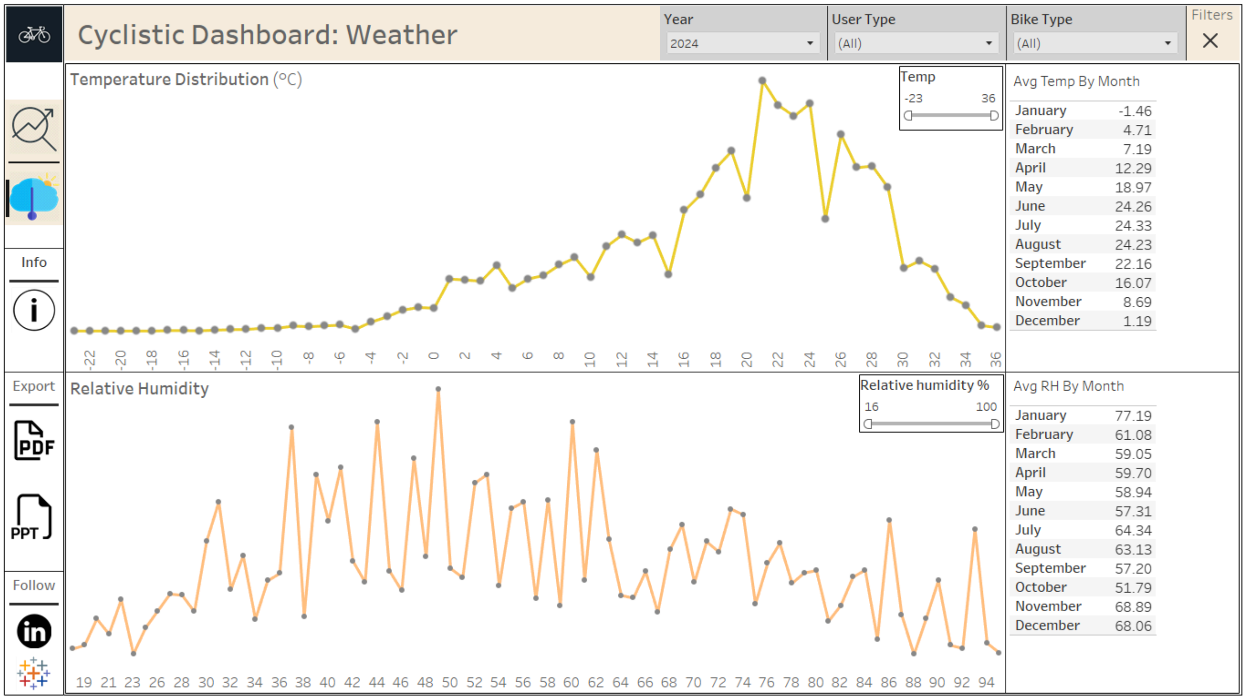
📌 Key Trends & Observations

✓ Annual members show consistent ridership throughout the year.
✓ Casual riders experience a significant surge during summer months (June–August).
✓ Casual ridership drops sharply in winter, while members maintain steady usage.

📌 Business Implications 🚀

✅ **Summer campaigns & incentives** could attract casual riders to long-term membership.
✅ **Wintertime strategies** (discounted rides, promotions) might help retain casual users.
✅ **Consistent member growth** indicates a **stable revenue stream** from annual subscribers.

📌 Temperature & Humidity vs. Ridership Trends 🌡️ 🌫️ 📊



📌 What Do These Charts Show?

✔ The **temperature impact graph** highlights how ridership fluctuates across different temperatures.
✔ The **humidity impact graph** explores if humidity significantly affects ride volume.

📌 Interactivity & Filters

✔ **User Type (Casual vs. Member)** – Compare ride behavior under different weather conditions.
✔ **Bike Type (Classic vs. Electric Bike)** – Identify if weather influences bike choice.
✔ **Year (2021-2024)** – Track long-term patterns in weather-based ridership.

📌 Key Trends & Observations

✓ **Temperature has a strong correlation with ridership**, with peak usage in the **20-29°C range**.
✓ **Casual riders are more temperature-sensitive**, while **annual members remain more consistent**.
✓ **Humidity doesn't show a significant impact on ridership**, meaning riders likely prioritize temperature over air moisture.
✓ **Extreme conditions (below 5°C or above 35°C) lead to noticeable declines in overall ridership.**

📌 Business Implications 🚀

✅ **Summer promotions (when temperatures are ideal)** can help increase ridership and convert casual riders to members.
✅ **Extreme cold and heat require operational adjustments** (e.g., more bike availability during mild weather).
✅ **Humidity alone is not a major deciding factor for riders**, so marketing strategies should focus on temperature trends.

📌 Final Thoughts on the Tableau Analysis 🎯

✅ **Data-backed decision-making** – The dashboard provides actionable insights into user behavior, bike usage, and weather impact.
✅ **Casual vs. Member behavior** – Clear distinctions in **ride duration, preferred time slots, and seasonality** offer opportunities for user conversion.
✅ **Operational efficiency** – Understanding **peak usage hours and seasonal trends** can help optimize bike availability.
✅ **Marketing strategies** – Targeting casual users with **seasonal promotions during peak temperatures (20-29°C)** can drive **higher conversions to memberships**.

4. 📌 Business Strategy & Recommendations

⚠️ Acknowledging Shortcomings

Before presenting **strategic recommendations**, it is crucial to **recognize the limitations** of the dataset. These **gaps in data** do not hinder our analysis but rather indicate **opportunities for deeper insights** in future studies.

1️⃣ Lack of Pricing Data

One of the biggest **limitations** of this analysis is the **absence of pricing details**. Cyclistic's **financial analysts have already concluded** that annual memberships are more profitable than casual rides, but **without cost-per-trip data**, we cannot:
✓ Confirm whether casual riders **spend more cumulatively** than members.
✓ Analyze whether certain **ride types (electric vs. classic)** have a **higher profit margin**.
✓ Test whether **price sensitivity** plays a role in membership conversions.

2️⃣ No Unique User IDs

The dataset **tracks trips, not individuals**, meaning:
✓ We **cannot differentiate repeat riders from one-time users**.
✓ It is **unclear whether casual riders are truly "new"** or simply **repeating one-time purchases**.
✓ Without understanding how many casual users **never return**, it is difficult to measure **conversion potential**.

3️⃣ No Demographic Segmentation

Understanding **who rides these bikes** is critical. Yet, the dataset **lacks demographic details**, meaning:
✓ We **cannot distinguish** students, tourists, professionals, or locals.
✓ **Age-based preferences** remain **unknown**.
✓ Without segmentation, marketing strategies **remain broad**, rather than **personalized** for each user group.

🚀 **Despite these challenges, powerful insights have emerged, forming the foundation for targeted strategies.**

🚀 Data-Backed Business Strategies

1️⃣ Leveraging Psychological Triggers to Convert Casual Riders 🎯

Casual riders exhibit strong seasonal patterns, peaking in **July and August**. This is not a coincidence—Chicago's summer months drive outdoor activities, and many casual riders opt for **single rides rather than committing to a membership**.

🔍 **Key Insights from the Data:**
✓ **July consistently records the highest casual ridership,** except for 2024.
✓ **Casual users take longer rides than members**—indicating engagement but reluctance to commit.
✓ **Peak ridership occurs in the evenings,** making nighttime marketing a key touchpoint.

📌 **Strategic Plan:**

1️⃣ April – July: Build Psychological Ownership

➡ **Send personalized ride summaries to casual users, showing:**

- **Total distance traveled** 🚲
- **Amount spent** 💰
- **CO_2 saved by choosing bikes over cars** 🌍
- **Late-evening digital & physical ads** (targeting peak casual ridership hours).

💡 **Purpose:**
✓ This builds a sense of ownership and makes casual users feel invested in their journey.
✓ Evening marketing increases psychological triggers when ridership is highest.

2️⃣ August Limited-Time Offer:

💡 **Two Membership Options:**
✓ **Offer 1:** Flat **$130** membership (originally $150).
✓ **Offer 2: $135 membership** + Free **premium riding gloves** 🧤 (valued at $20).

💡 **Why This Works?**
✓ **Behavioral economist Dan Ariely's research** suggests that adding a small extra perk makes people **more likely to commit** (even if the monetary value is similar).
✓ **Riding gloves align with the biking experience,** making it **more attractive** than generic gifts.
✓ **Evening marketing reinforces the decision when casual users are most active.**

📌 **Expected Outcome:**
✓ Converts casual users **right before their seasonal peak ends**.
✓ Uses **psychological nudges** to make the membership more attractive than single rides.
✓ **Physical marketing during night hours** capitalizes on peak casual ridership.

🚀 **This ensures maximum conversion at the right time, using the right triggers!**

2️⃣ Capturing the Tourist & Student Market 🎓 🌍

Chicago is a **highly transient city**, attracting **millions of tourists** and a **large student population** every year. These **two segments** present a **unique opportunity** for Cyclistic to **expand its user base**.

🔍 **Key Insights from the Data:**
✓ **40-50 million tourists visit Chicago, from June – September being the most popular amongst tourists.**
✓ **Casual ridership spikes in September**, coinciding with **student migration** for academic years.

📌 **Strategic Plan:**

🎒 **Targeting Tourists (June – September):**
1️⃣ Launch a **Tourist Pass only available at airports and major visitor hubs**.
2️⃣ Promote **Cyclistic rentals at key locations**:

- **O'Hare & Midway Airports** ✈️
- **Millennium Park & Navy Pier** 🏙️
- **Hotel partnerships** 🏨

🎓 **Targeting Students (August – September):**
1️⃣ Offer **student discounts** on annual memberships during college orientation weeks.
2️⃣ Partner with **universities for pre-loaded student passes**.

📌 **Expected Outcome:**
✓ **Boosts short-term revenue** from tourists **without impacting local demand**.
✓ **Ensures early adoption among students**, creating **long-term retention**.

3️⃣ Launching a Fitness-Based Initiative in Winter 🏋️ 🚴

January and February see the **steepest ridership decline** due to Chicago's harsh winters. However, **New Year's resolutions** create a **massive opportunity** for fitness-driven initiatives.

🔍 **Key Insights from the Data:**
✓ January and February show a **significant decline in casual ridership** compared to peak summer months:

- **2021:** 96% decline
- **2022:** 94% decline
- **2023:** 86% decline
- **2024:** 89% decline

✓ **Classic bikes dominate casual ridership**, making them ideal for **fitness challenges**.
✓ Chicago has a **rising obesity rate**, meaning **fitness incentives could align with community health efforts**.

📌 **Strategic Plan:**
1️⃣ **Jan 1 – Jan 31: "Cyclistic Fitness Challenge"** 🏆
➡ Riders must **complete a 20-minute ride daily** for **31 days** at **Lakefront Trail** (6 AM - 10 AM).
➡ All bikes in this area will be **classic bikes only** to **maximize physical effort**.

2️⃣ **Completion Reward:**
✓ **Discounted annual membership** 🏅
✓ **Special "Cyclistic Fit" community badge**

📌 **Additional Social Impact:**

✓ **50% of profits from the Fitness Pass will be donated to Chicago Public Schools for public education**

📌 **Expected Outcome:**
✓ Leverages **New Year's fitness resolutions** to **increase engagement**.
✓ Encourages **habit formation**, increasing **membership sign-ups**.
✓ **Strengthens Cyclistic's brand identity** as a **community-driven organization**.

📌 Strengthening Infrastructure & Operations

While marketing strategies **increase ridership**, long-term **sustainability** depends on **operational improvements**:

✔ **Optimize Station Placement** 📍 – Use **heat map data** to **relocate underutilized stations** to **high-traffic areas**.
✔ **Winter-Optimized Bikes** ❄️ – Introduce **heated seats & snow tires** to **reduce seasonal drop-offs**.
✔ **University Partnerships** 🎓 – Embed **Cyclistic memberships** into **student ID cards**.

📌 Conclusion & Key Takeaways

✅ **Seasonal ridership peaks provide an opportunity for targeted membership conversions.**
✅ **Tourists and students are underutilized segments that can drive growth.**
✅ **A fitness-based membership initiative aligns with Chicago's health concerns & New Year resolutions.**
✅ **Operational investments can improve long-term ridership and retention.**

With **data-driven insights and structured strategies**, Cyclistic can **increase annual memberships, maximize revenue, and establish itself as a lifestyle-oriented mobility solution**. 🚲 💡

Closing Note

"This report isn't just about analyzing bike rides; it's about understanding user behavior, leveraging data for strategic growth, and reimagining urban mobility. With the right insights, Cyclistic can not only expand its membership base but also contribute to a more sustainable and connected city." 🌍 🚲