Project Report

on

“Uber Data Analysis”

Submitted as Mini Project Report

FOR MINI PROJECT LAB(KCS-554)

Session 2022-23

in

Information Technology

By
Dharmendra Kumar
2000320130056

Under the guidance of
Ms. Shweta Kaushik

ABES ENGINEERING COLLEGE, GHAZIABAD

AFFILIATED TO
DR. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY, U.P., LUCKNOW
(Formerly UPTU)

STUDENT’S DECLARATION

I / We hereby declare that the work being presented in this report entitled **“Uber Data Analysis”** is an authentic record of my / our work carried out under the supervision of **“Ms. Shweta Kaushik”.**

The matter embodied in this report has not been submitted by me/us for the award of any other degree.

Dated:

Signature of students

Dharmendra Kumar

Roll no: 2000320130056

Information Technology

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Signature of HOD
Name: Prof. Amit Sinha
Information Technology
Date...........................

Signature of Supervisor
Ms. Shweta Kaushik
Assistant Professor
Information Technology

ACKNOWLEDGEMENT

It gives us great pleasure to present the report of the B. Tech Mini Project undertaken during B. Tech. Third Year. We owe special gratitude to **Ms. Shweta Kaushik** *for his constant support and guidance throughout our work. Her sincerity, thoroughness, and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavors have seen the light of day.*

We also take the opportunity to acknowledge the contribution of Professor **Dr. Amit Sinha***, Head, Department of* **Information Technology***, ABESEC Ghaziabad for his full support and assistance during the development of the project.*

We also do not like to miss the opportunity to acknowledge the contribution of all department faculty members for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution to the completion of the project.

Signature:

Name:

Roll No.:

Date :

TABLE OF CONTENTS

Abstract

Talking about our Uber data analysis project, data storytelling is an important component of Machine Learning through which companies can understand the background of various operations. With the help of visualization, companies can avail the benefit of understanding complex data and gaining insights that would help them craft decisions. You will learn how to implement the ggplot2 on the Uber Pickups dataset and at the end, master the art of data visualization in R.

Introduction

Data storytelling is an important component of Machine Learning through which companies can understand the background of various operations.

Companies may use data visualization to better understand complicated datasets and make better decisions.

With the help of visualization, companies can avail the benefit of understanding complex data and gaining insights that would help them craft decisions.

We’ll use R packages like ggplot2 to create data analysis in this project.

This is more of a data visualization project that will guide you towards using the ggplot2 library for understanding the data and for developing an intuition for understanding the customers who avail the trips

We utilize user data to extract insights and provide an accurate prediction of clients who will take Uber trips and rides.

The study will look at several criteria such as the number of journeys made in a day, the number of travels made in a month, and so on.

As a result of this study, we can determine the average number of passengers that Uber may have in a day, the peak hours when there are more consumers available, the number of trips identified at the highest on which day of the month, and so on.

Important: The goal of this project is to learn visualizations in R. I do not claim copyright over any of the content here.

REQUIREMENT SPECIFICATION

User and Data

- R Studio IDE
- R Language
- Inbuilt Libraries/Packages
- Uber Database

System Requirement

- An Intel-compatible platform running Windows 11, 10 /8.1/8 /7 /Vista /XP /2000 Windows Server 2022, 2019 /2016 /2012 /2008 /2003
- At least 256 MB of RAM, a mouse, and enough disk space for recovered files, image files, etc.
- Administrative privileges are required to install and run R Studio utilities.
- A network connection for data recovery over the network.

Tasks

- Plotting the trips by the hours in a day

- Plotting data by trips during every day of the month

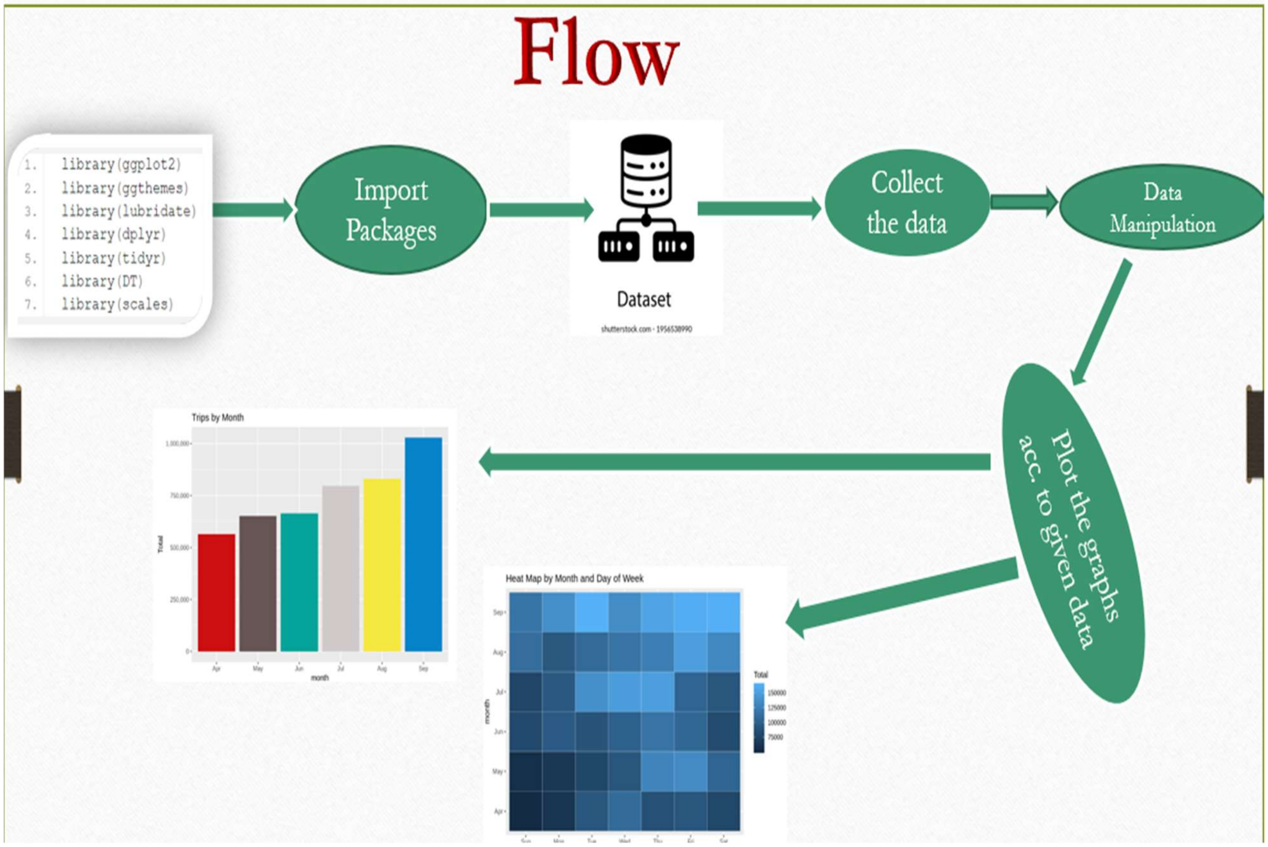
- Number of Trips taking place during months in a year

- Finding out the number of Trips by bases

- Creating a Heatmap visualization of day, hour, and month

- Creating a map visualization of rides in New York

Proposed work & Methodology



Flow

```
library(ggplot2)
library(ggthemes)
library(lubridate)
library(dplyr)
library(tidyr)
library(DT)
library(scales)
```

Import Packages

Dataset

shutterstock.com · 1956538990

Collect the data

Data Manipulation

Plot the graphs acc. to given data

Figure – 1 (To Explanation of project working)

IMPLEMENTATION & RESULTS

Coding:

1. Importing the Essential Packages:

In the first step of our R project, we will import the essential packages that we will use in this uber data analysis project. Some of the important libraries of R that we will use are:-

ggplot2

This is the backbone of this project. ggplot2 is the most popular data visualization library that is most widely used for creating aesthetic visualization plots.

ggthemes

This is more of an add-on to our main ggplot2 library. With this, we can create better create extra themes and scales with the mainstream ggplot2 package.

lubricate

Our dataset involves various time frames. To understand our data in separate time categories, we will make use of the lubridate package.

dplyr

This package is the lingua franca of data manipulation in R.

tidyr

This package will help you to tidy your data. The basic principle of tidyr is to tidy the columns where each variable is present in a column, each observation is represented by a row and each value depicts a cell.

DT

With the help of this package, we will be able to interface with the JavaScript Library called – Datatables.

scales

With the help of graphical scales, we can automatically map the data to the correct scales with well-placed axes and legends.

CODE:

```
library(ggplot2)
library(ggthemes)
library(lubridate)
library(dplyr)
library(tidyr)
library(DT)
library(scales)
```

2. Creating vector of colors to be implemented in our plots

In this step of data science project, we will create a vector of our colors that will be included in our plotting functions. You can also select your own set of colors.

CODE:

```
colors = c(""#CC1011", "#665555", "#05a399", "#cfcaca",
"#f5e840", "#0683c9", "#e075b0"")
```

3. Reading the Data into their designated variables

Now, we will read several csv files that contain the data from April 2014 to September 2014. We will store these in corresponding data frames like apr_data, may_data, etc. After we have read the files, we will combine all of this data into a single dataframe called ‘data_2014’.Then, in the next step, we will perform the appropriate formatting of Date.Time column. Then, we will proceed to create factors of time objects like day, month, year etc.

CODE:

```
 apr_data <- read.csv("uber-raw-data-apr14.csv")

 may_data <- read.csv("uber-raw-data-may14.csv")

 jun_data <- read.csv("uber-raw-data-jun14.csv")

 jul_data <- read.csv("uber-raw-data-jul14.csv")

 aug_data <- read.csv("uber-raw-data-aug14.csv")

 sep_data <- read.csv("uber-raw-data-sep14.csv")

 data_2014 <- rbind(apr_data,may_data, jun_data, jul_data,
aug_data, sep_data)

 data_2014$Date.Time   <-   as.POSIXct(data_2014$Date.Time,
format =  "%m/%d/%Y %H:%M:%S")

 data_2014$Time  <-  format(as.POSIXct(data_2014$Date.Time,
format  = "%m/%d/%Y %H:%M:%S"), format="%H:%M:%S")

data_2014$Date.Time <- ymd_hms(data_2014$Date.Time)

data_2014$day <- factor(day(data_2014$Date.Time))

data_2014$month  <-  factor(month(data_2014$Date.Time,  label
= TRUE))

data_2014$year <- factor(year(data_2014$Date.Time))
```

```
data_2014$dayofweek   <-   factor(wday(data_2014$Date.Time,
label = TRUE))

data_2014$hour <- factor(hour(hms(data_2014$Time)))

data_2014$minute <- factor(minute(hms(data_2014$Time)))

data_2014$second <- factor(second(hms(data_2014$Time)))
```

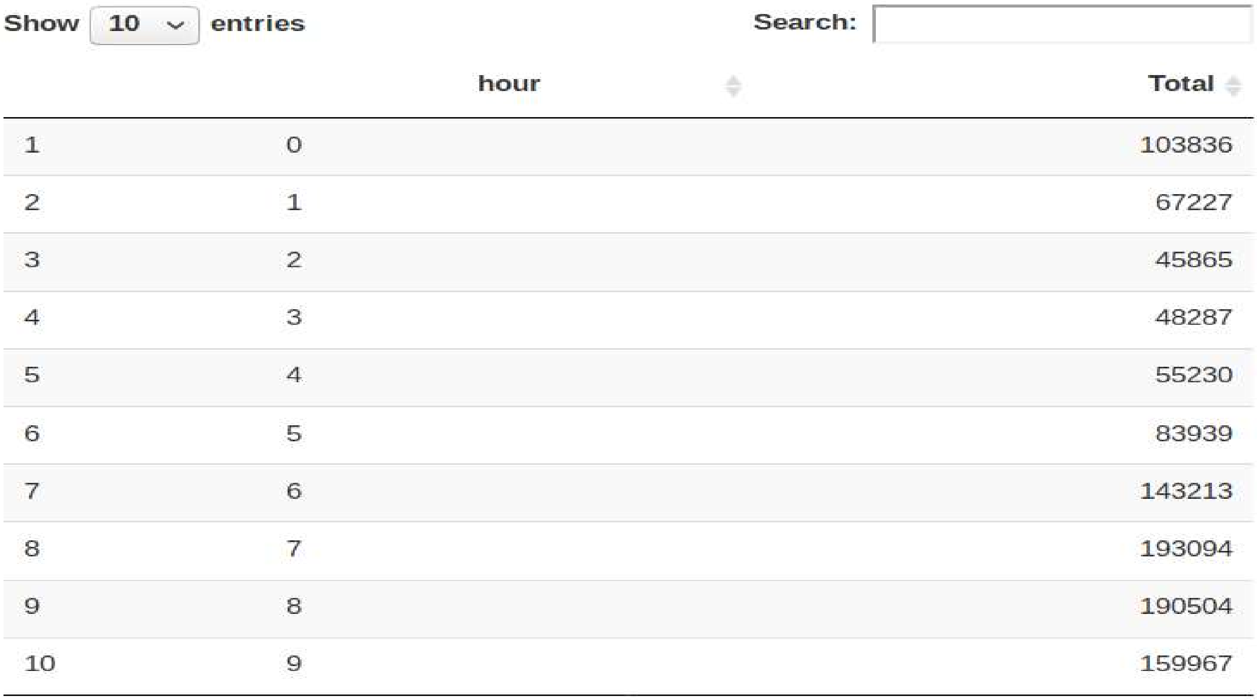
4. Plotting the trips by the hours in a day

In the next step or R project, we will use the ggplot function to plot the number of trips that the passengers had made in a day. We will also use dplyr to aggregate our data. In the resulting visualizations, we can understand how the number of passengers fares throughout the day. We observe that the number of trips are higher in the evening around 5:00 and 6:00 PM.

CODE:

```
hour_data <- data_2014 %>%

           group_by(hour) %>%

               dplyr::summarize(Total = n())

datatable(hour_data)
```

OUTPUT:

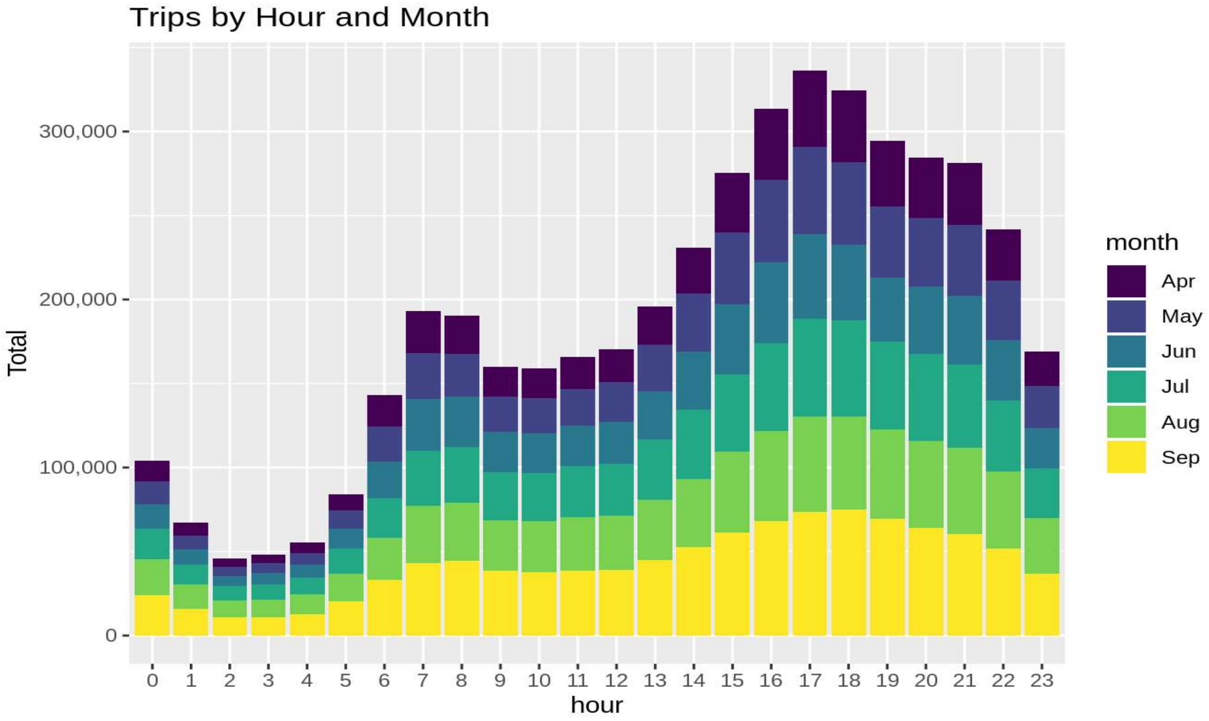


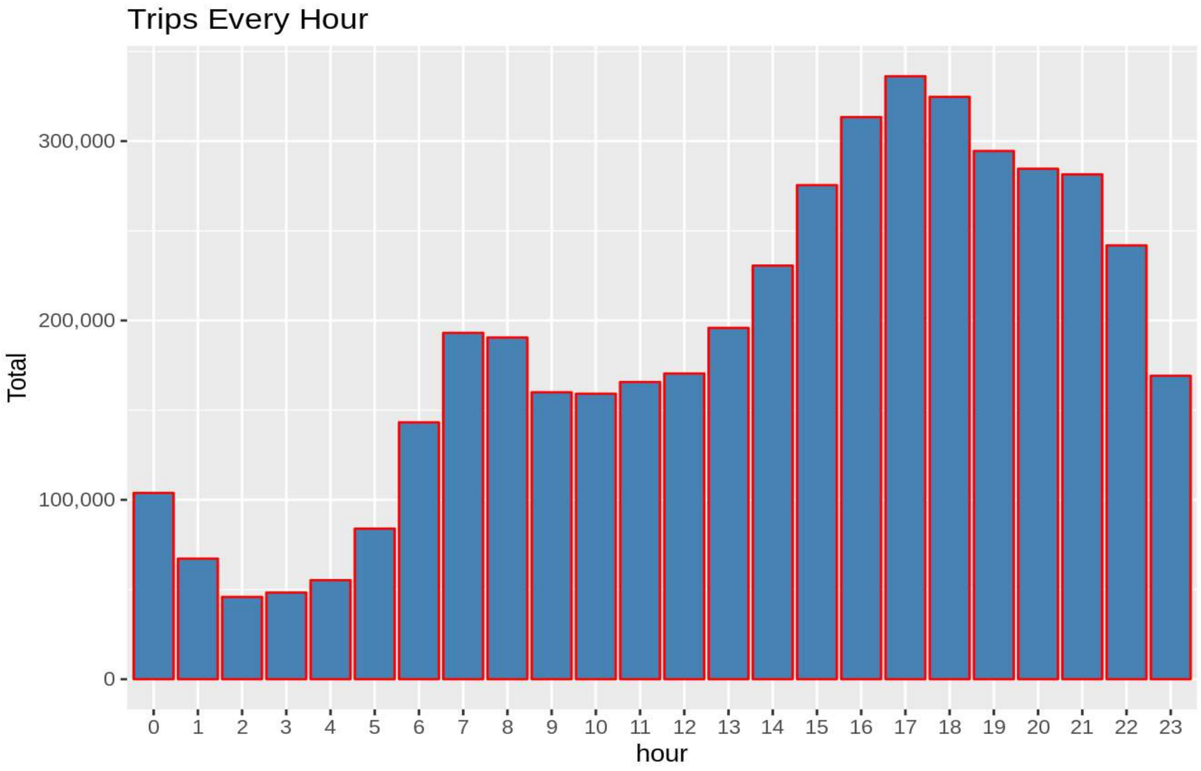
	hour	Total
1	0	103836
2	1	67227
3	2	45865
4	3	48287
5	4	55230
6	5	83939
7	6	143213
8	7	193094
9	8	190504
10	9	159967

CODE:

```
ggplot(hour_data, aes(hour, Total)) +

        geom_bar( stat = "identity", fill =
"steelblue", color = "red") +
           ggtitle("Trips Every Hour") +
            theme(legend.position = "none") +
            scale_y_continuous(labels = comma)

month_hour <- data_2014 %>%
          group_by(month, hour) %>%
             dplyr::summarize(Total = n())

ggplot(month_hour, aes(hour, Total, fill = month)) +
       geom_bar( stat = "identity") +
          ggtitle("Trips by Hour and Month") +
           scale_y_continuous(labels = comma)
```

OUTPUT:


Trips by Hour and Month
Total
300,000
200,000
100,000
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
hour
month
Apr
May
Jun
Jul
Aug
Sep


Trips Every Hour
Total
300,000
200,000
100,000
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
hour

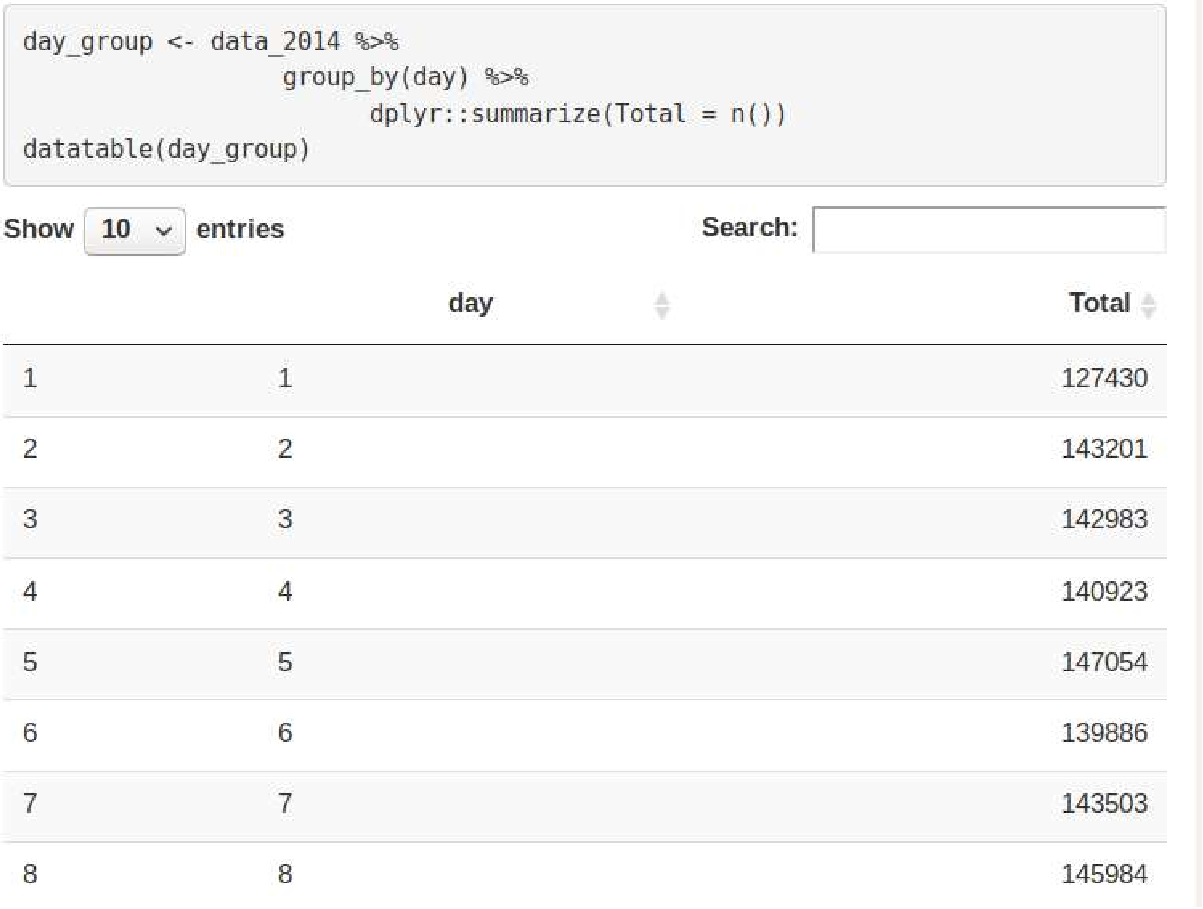
5. Plotting data by trips during every day of the month

In this section of DataFlair R project, we will learn how to plot our data based on every day of the month. We observe from the resulting visualization that 30th of the month had the highest trips in the year which is mostly contributed by the month of April.

CODE:

```
day_group <- data_2014 %>%
          group_by(day) %>%
             dplyr::summarize(Total = n())
datatable(day_group)
```

OUTPUT:



```
day_group <- data_2014 %>%
               group_by(day) %>%
                    dplyr::summarize(Total = n())
datatable(day_group)
```

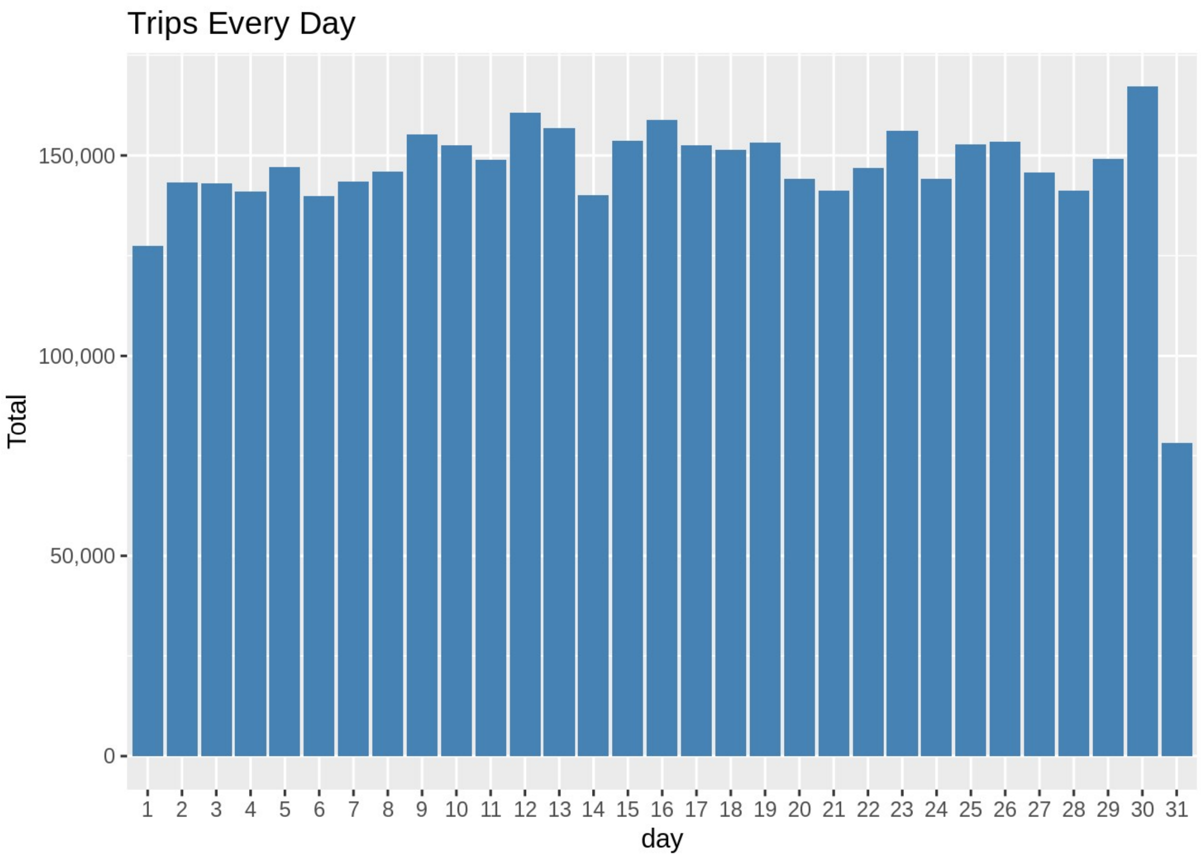
Show 10 entries   Search:

	day	Total
1	1	127430
2	2	143201
3	3	142983
4	4	140923
5	5	147054
6	6	139886
7	7	143503
8	8	145984

CODE:

```
    ggplot(day_group, aes(day, Total)) +
               geom_bar( stat = "identity", fill = "steelblue")
+
                      ggtitle("Trips Every Day") +
                       theme(legend.position = "none") +
                       scale_y_continuous(labels = comma)
```

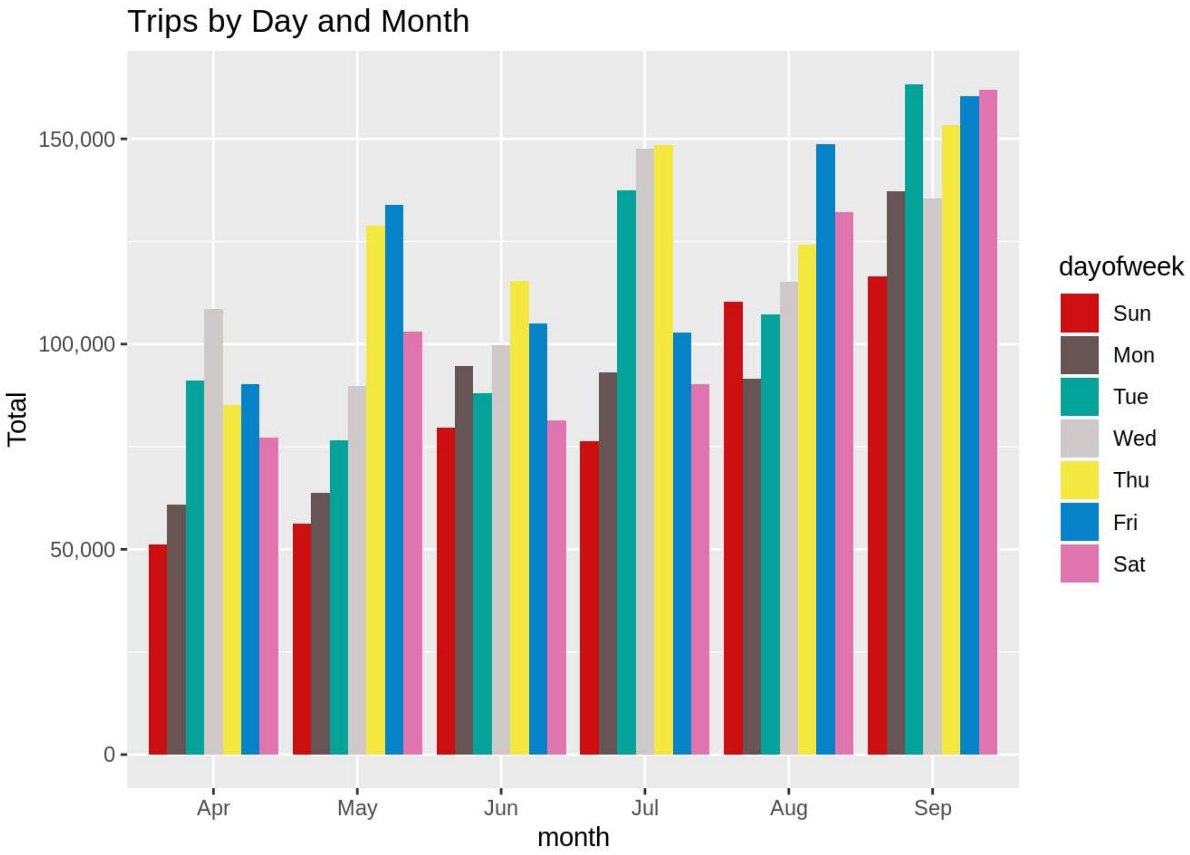
OUTPUT:



CODE:

```
     day_month_group <- data_2014 %>%
              group_by(month, day) %>%
                  dplyr::summarize(Total = n())

     ggplot(day_month_group, aes(day, Total, fill = month)) +
             geom_bar( stat = "identity") +
                ggtitle("Trips by Day and Month") +
                 scale_y_continuous(labels = comma) +
                 scale_fill_manual(values = colors)
```

OUTPUT:



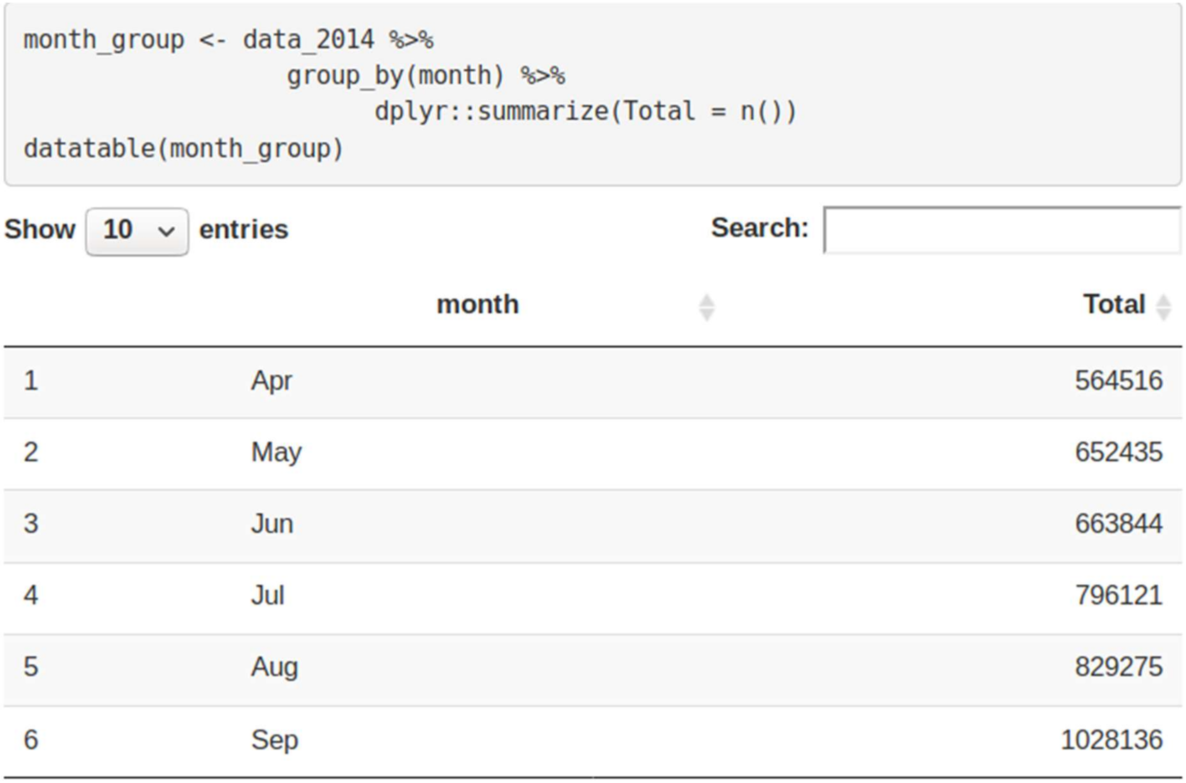
6. Number of Trips taking place during months in a year

In this section, we will visualize the number of trips that are taking place each month of the year. In the output visualization, we observe that most trips were made during the month of September. Furthermore, we also obtain visual reports of the number of trips that were made on every day of the week.

CODE:

```
month_group <- data_2014 %>%

          group_by(month) %>%

             dplyr::summarize(Total = n())

datatable(month_group)
```

OUTPUT:



```
month_group <- data_2014 %>%
                  group_by(month) %>%
                        dplyr::summarize(Total = n())
datatable(month_group)
```

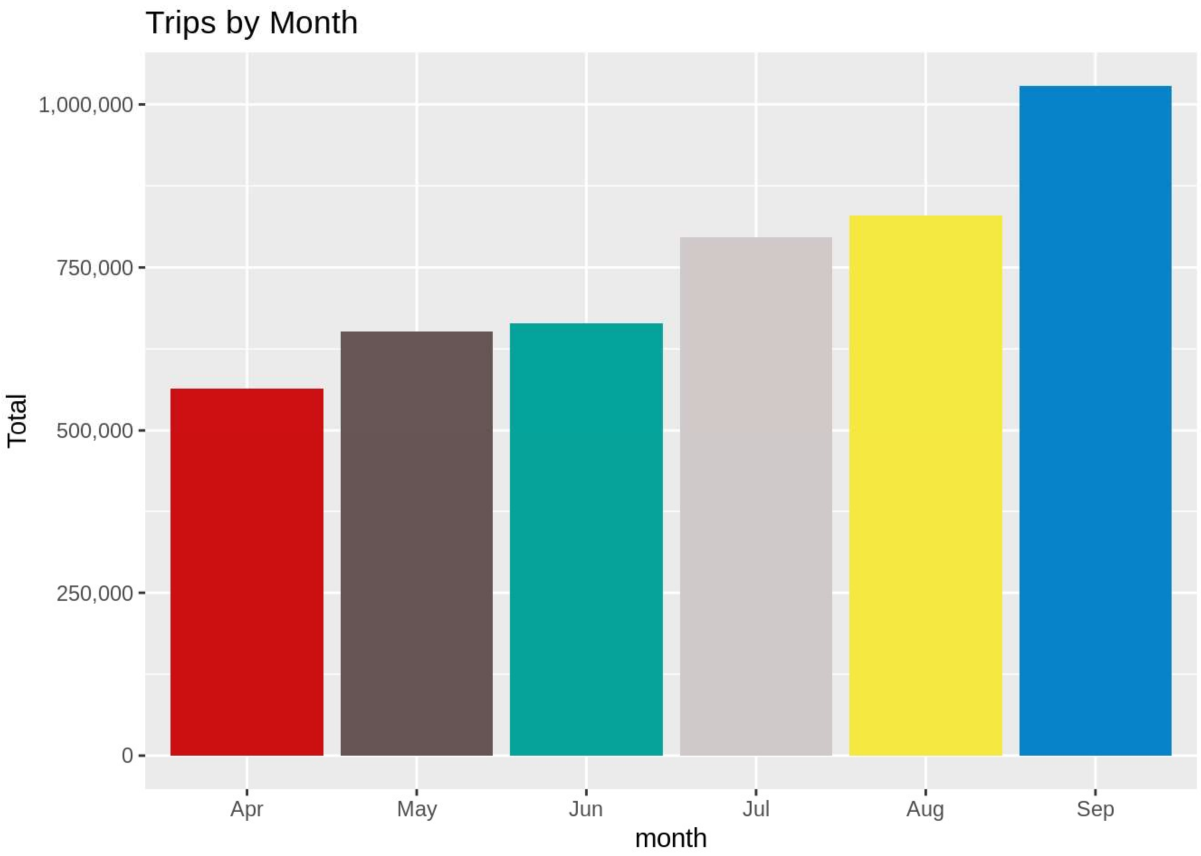
Show 10 entries   Search:

	month	Total
1	Apr	564516
2	May	652435
3	Jun	663844
4	Jul	796121
5	Aug	829275
6	Sep	1028136

CODE:

```
ggplot( , aes(month, Total, fill = month)) +
        geom_bar( stat = "identity") +
           ggtitle("Trips by Month") +
            theme(legend.position = "none") +
            scale_y_continuous(labels = comma) +
            scale_fill_manual(values = colors)
```

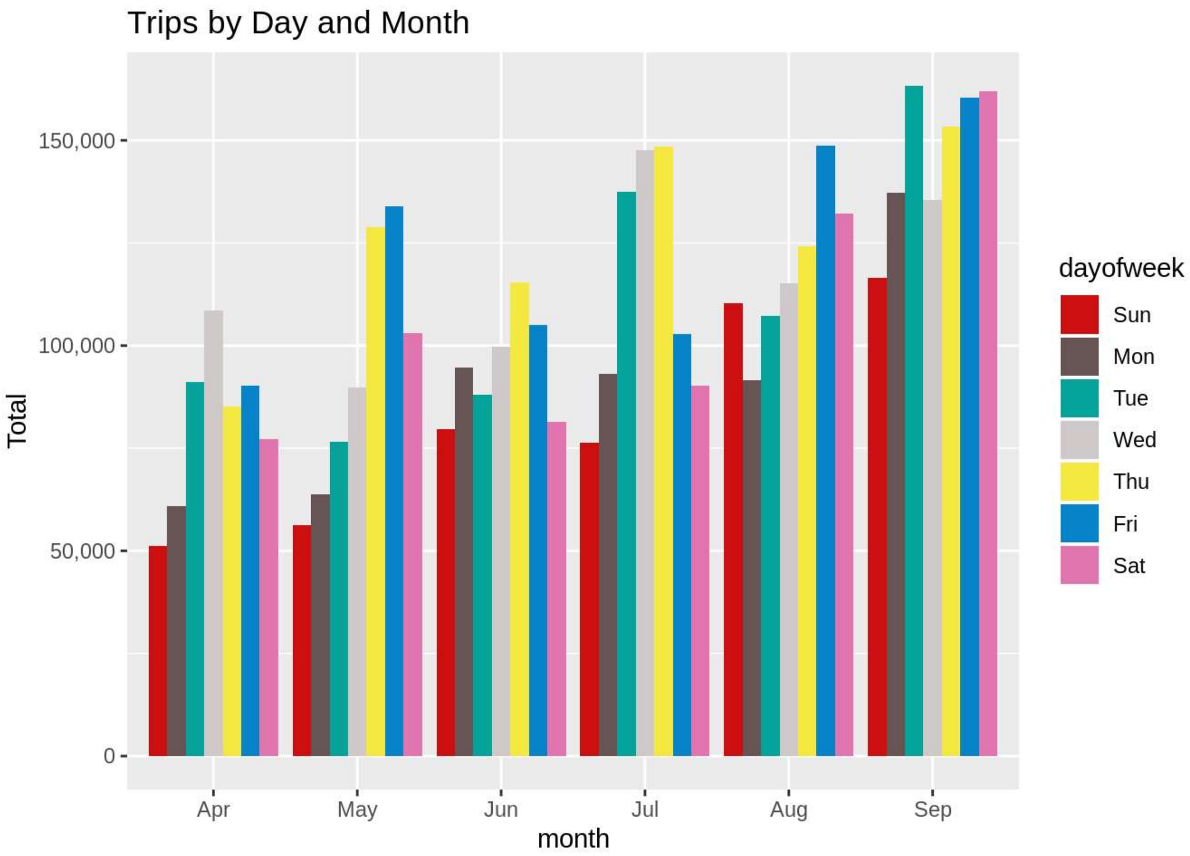
OUTPUT:



CODE:

```
month_weekday <- data_2014 %>%
         group_by(month, dayofweek) %>%
             dplyr::summarize(Total = n())

ggplot(month_weekday, aes(month, Total, fill = dayofweek))
+
       geom_bar( stat = "identity", position = "dodge") +
          ggtitle("Trips by Day and Month") +
           scale_y_continuous(labels = comma) +
           scale_fill_manual(values = colors)
```

OUTPUT:



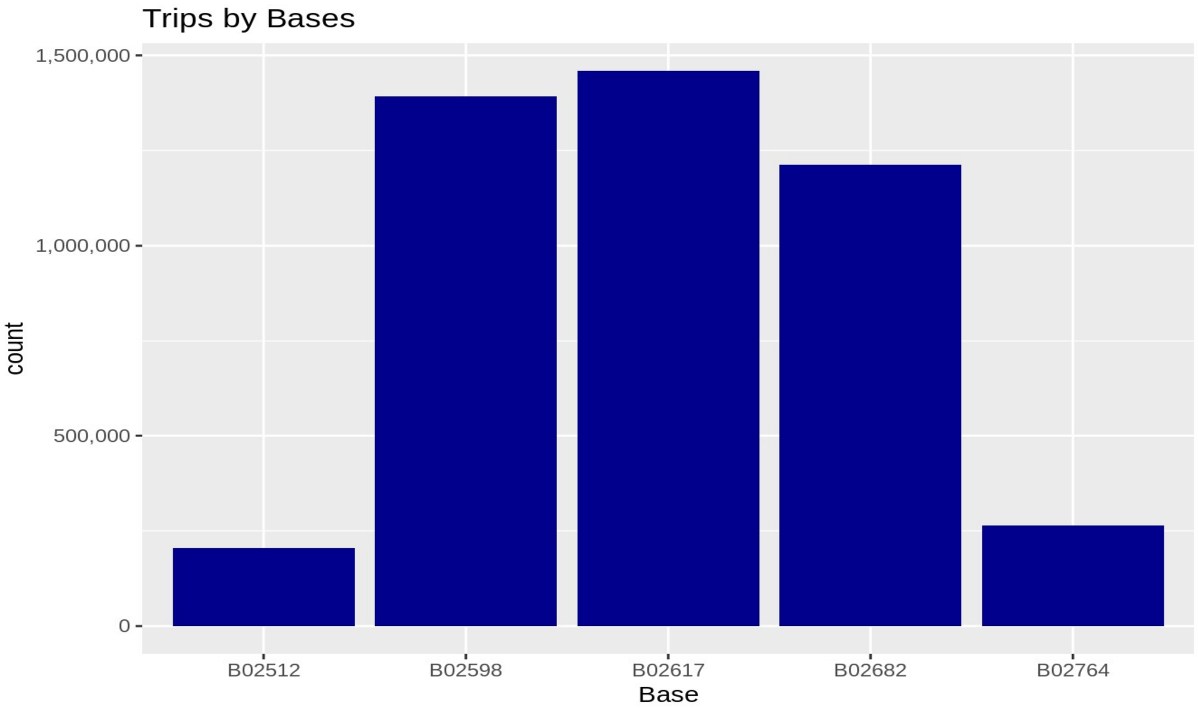
7. Finding out the number of Trips by bases

In the following visualization, we plot the number of trips that have been taken by the passengers from each of the bases. There are five bases in all out of which, we observe that B02617 had the highest number of trips. Furthermore, this base had the highest number of trips in the month B02617. Thursday observed highest trips in the three bases – B02598, B02617, B02682.

CODE:

```
ggplot(data_2014, aes(Base)) +
  geom_bar(fill = "darkred") +
  scale_y_continuous(labels = comma) +
  ggtitle("Trips by Bases")
```

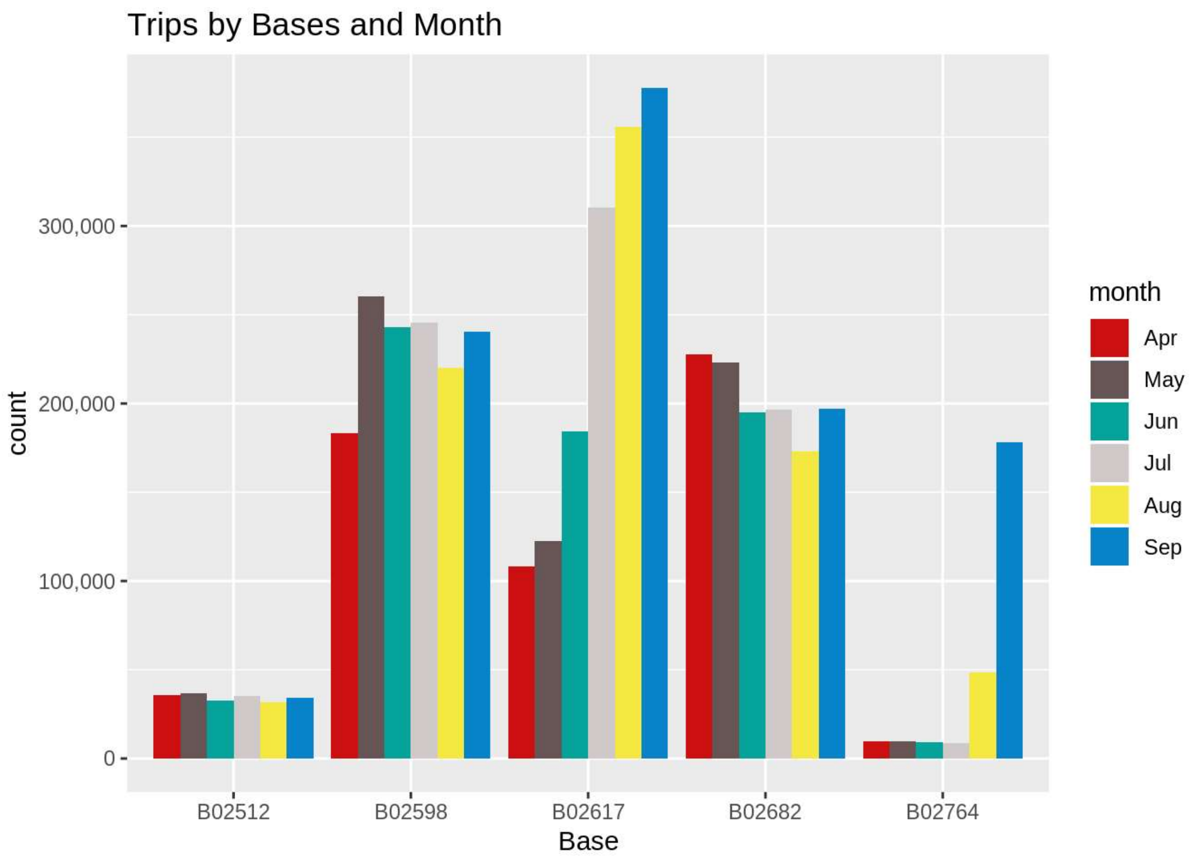
OUTPUT:



CODE:

```
ggplot(data_2014, aes(Base, fill = month)) +
  geom_bar(position = "dodge") +
  scale_y_continuous(labels = comma) +
  ggtitle("Trips by Bases and Month") +
  scale_fill_manual(values = colors)
```

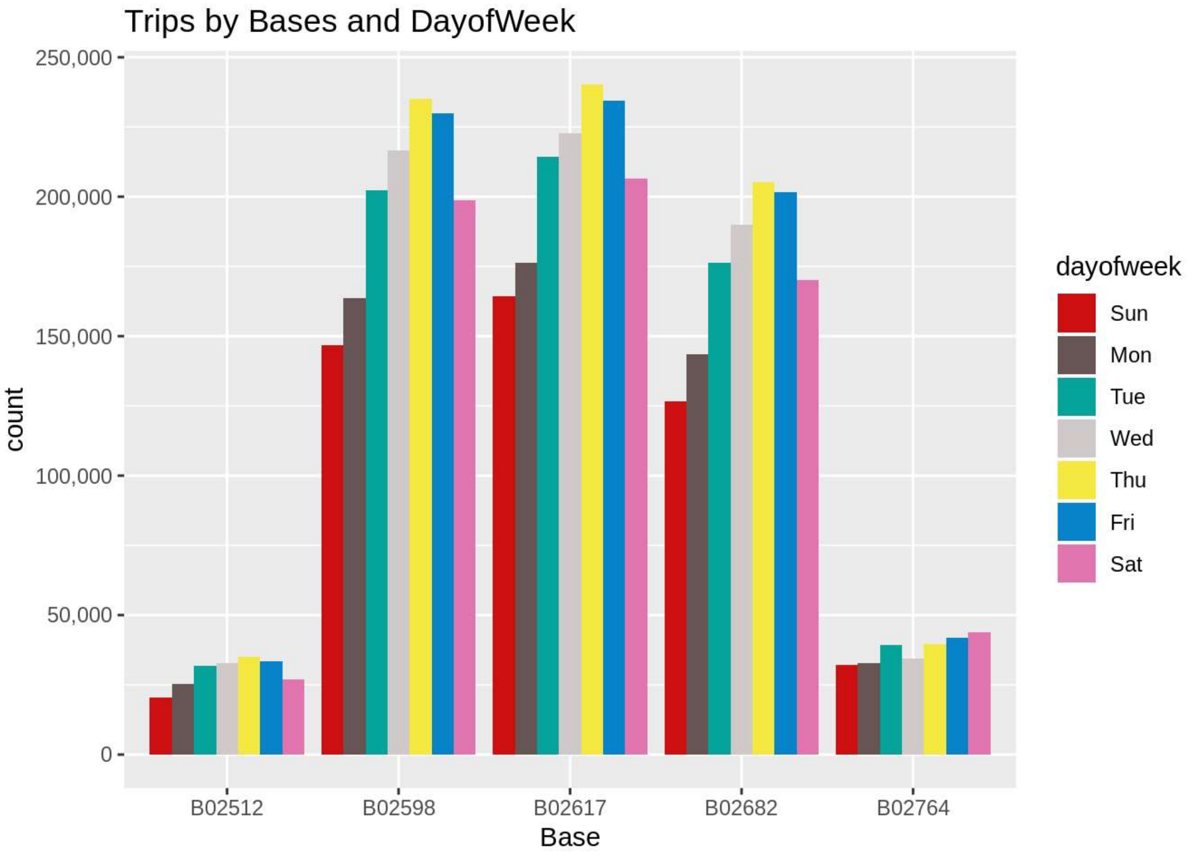
OUTPUT:



CODE:

```
ggplot(data_2014, aes(Base, fill = dayofweek)) +
  geom_bar(position = "dodge") +
  scale_y_continuous(labels = comma) +
  ggtitle("Trips by Bases and DayofWeek") +
  scale_fill_manual(values = colors)
```

OUTPUT:



8. Creating a Heatmap visualization of day, hour, and month

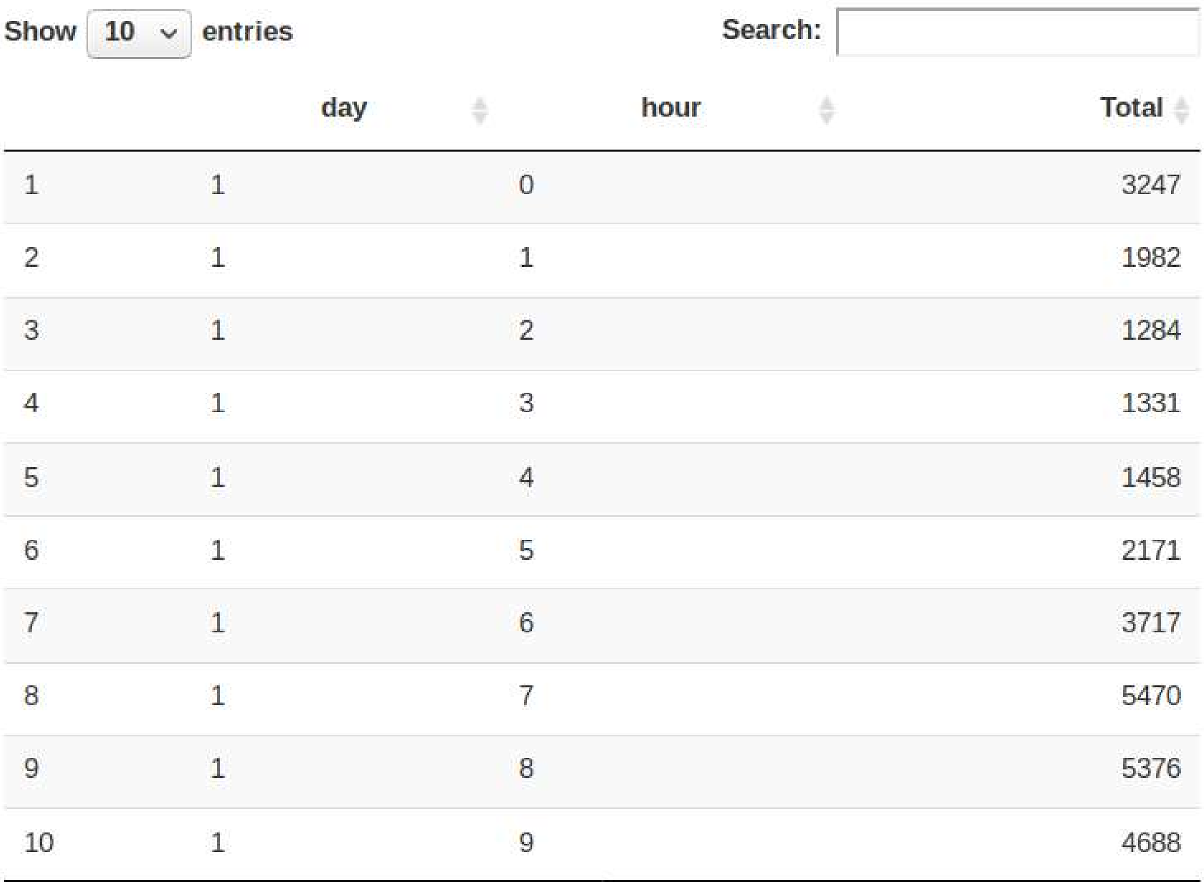
In this section, we will learn how to plot heatmaps using ggplot(). We will plot five heatmap plots –

- First, we will plot Heatmap by Hour and Day.
- Second, we will plot Heatmap by Month and Day.
- Third, a Heatmap by Month and Day of the Week.
- Fourth, a Heatmap that delineates Months and Bases.
- Finally, we will plot the heatmap, by base and day of the week.

CODE:

```
day_and_hour <- data_2014 %>%
         group_by(day, hour) %>%
            dplyr::summarize(Total = n())

datatable(day_and_hour)
```

OUTPUT:



Show 10 entries

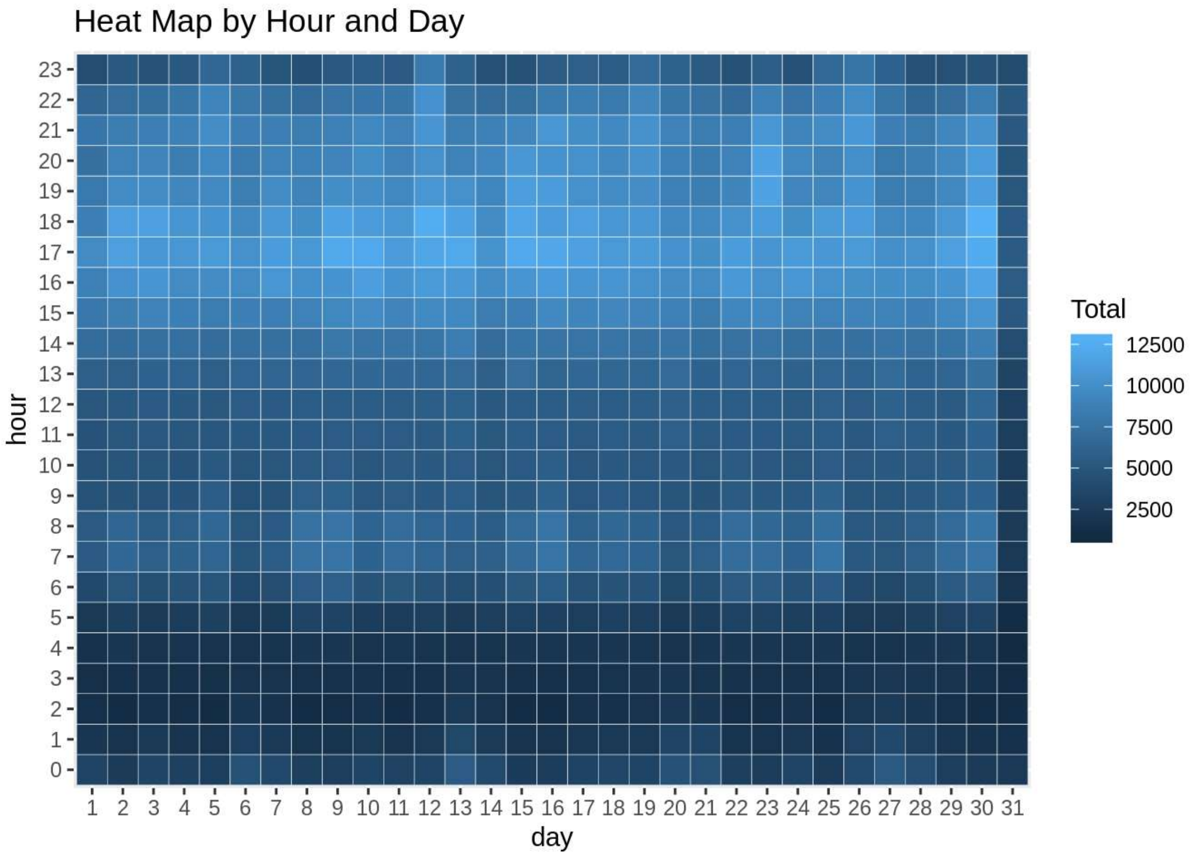
Search:

	day	hour	Total
1	1	0	3247
2	1	1	1982
3	1	2	1284
4	1	3	1331
5	1	4	1458
6	1	5	2171
7	1	6	3717
8	1	7	5470
9	1	8	5376
10	1	9	4688

CODE:

```
ggplot(day_and_hour, aes(day, hour, fill = Total)) +
            geom_tile(color = "white") +
              ggtitle("Heat Map by Hour and Day")
```

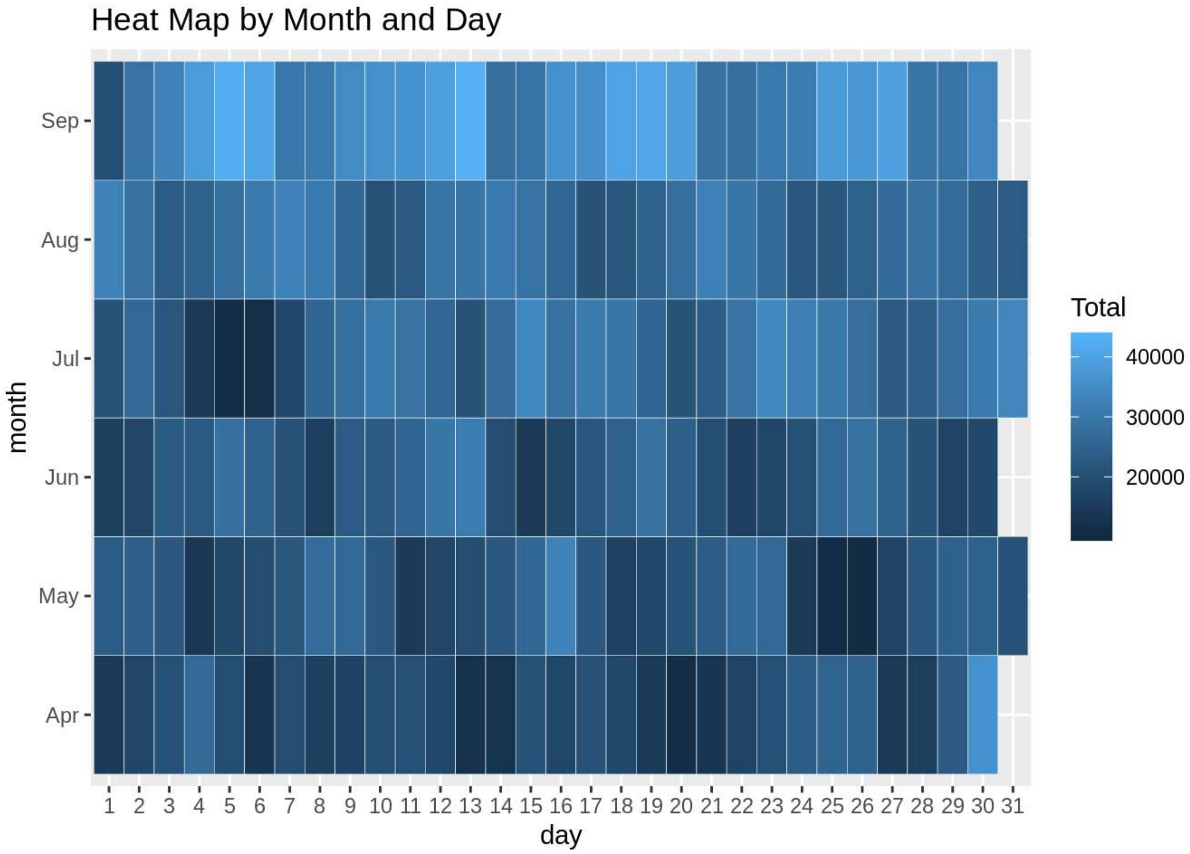
OUTPUT:



CODE:

```
ggplot(day_month_group, aes(day, month, fill = Total)) +
            geom_tile(color = "white") +
              ggtitle("Heat Map by Month and Day")
```

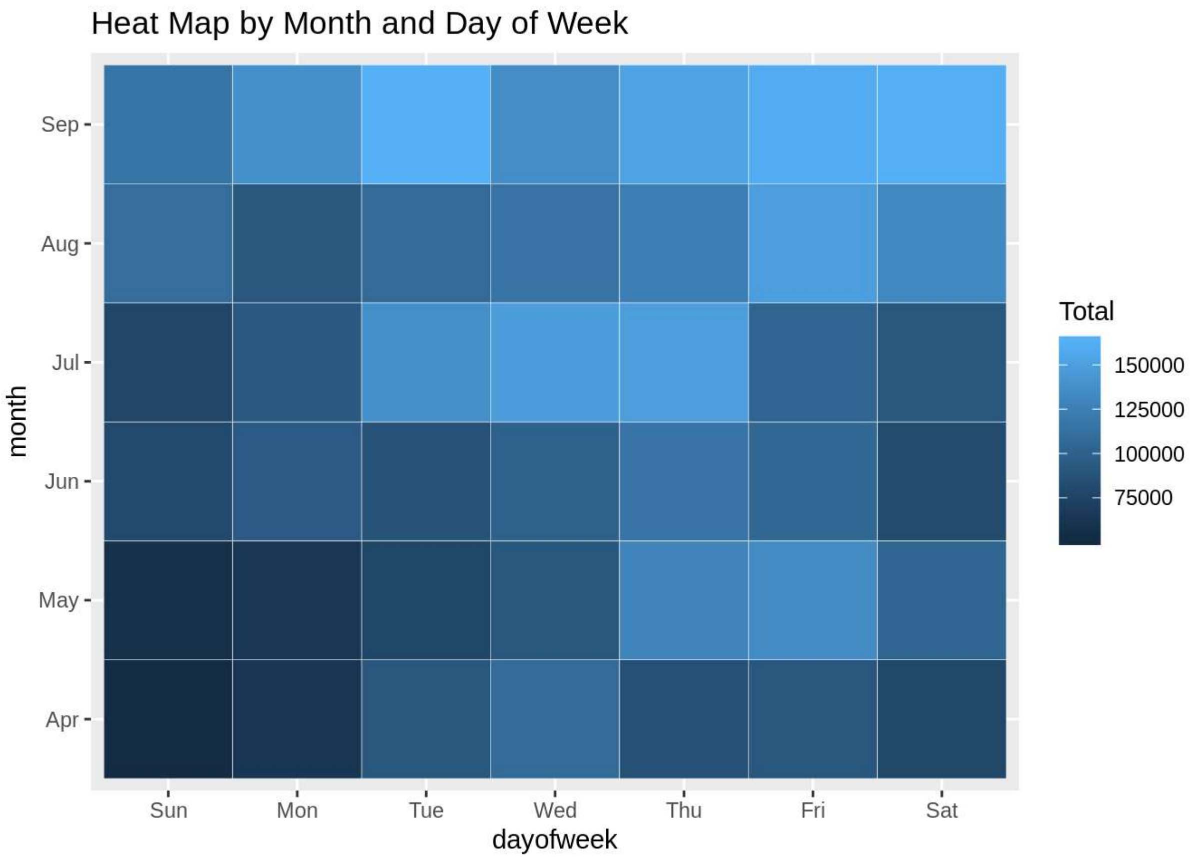
OUTPUT:



CODE:

```
ggplot(month_weekday, aes(dayofweek, month, fill = Total))
+
            geom_tile(color = "white") +
              ggtitle("Heat Map by Month and Day of Week")
```

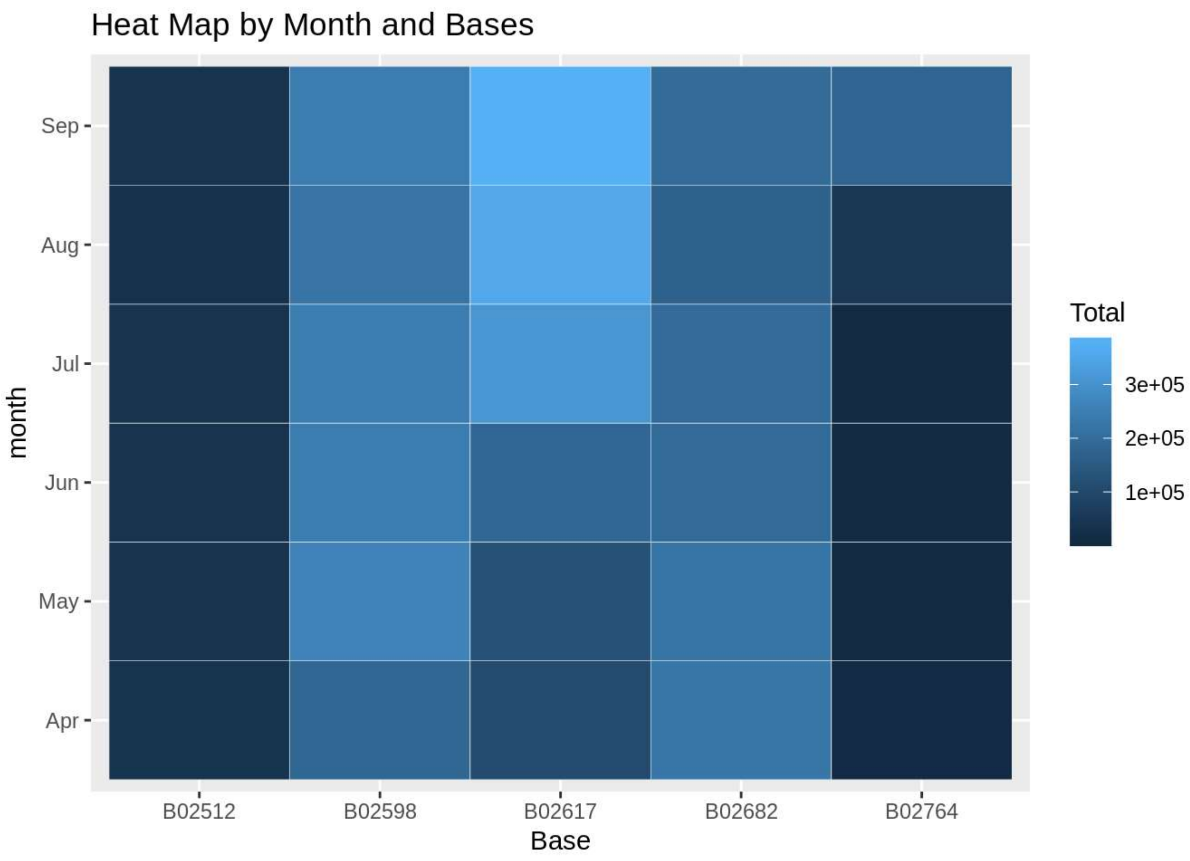
OUTPUT:



CODE:

```
month_base <-  data_2014 %>%
                    group_by(Base, month) %>%
                        dplyr::summarize(Total = n())

day0fweek_bases <-  data_2014 %>%
                    group_by(Base, dayofweek) %>%
                        dplyr::summarize(Total = n())

ggplot(month_base, aes(Base, month, fill = Total)) +
            geom_tile(color = "white") +
              ggtitle("Heat Map by Month and Bases")
```

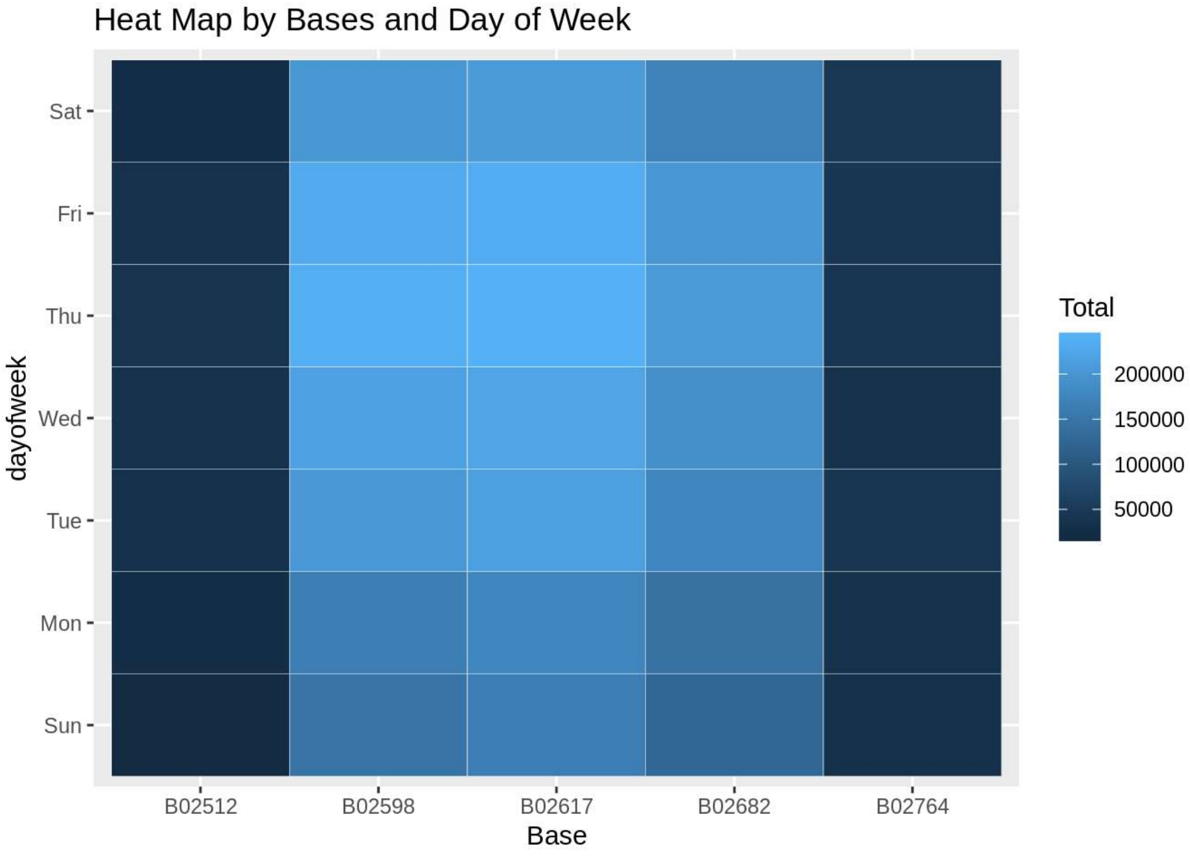
OUTPUT:



CODE:

```
ggplot(day0fweek_bases, aes(Base, dayofweek, fill = Total))
+
            geom_tile(color = "white") +
              ggtitle("Heat Map by Bases and Day of Week")
```

OUTPUT:



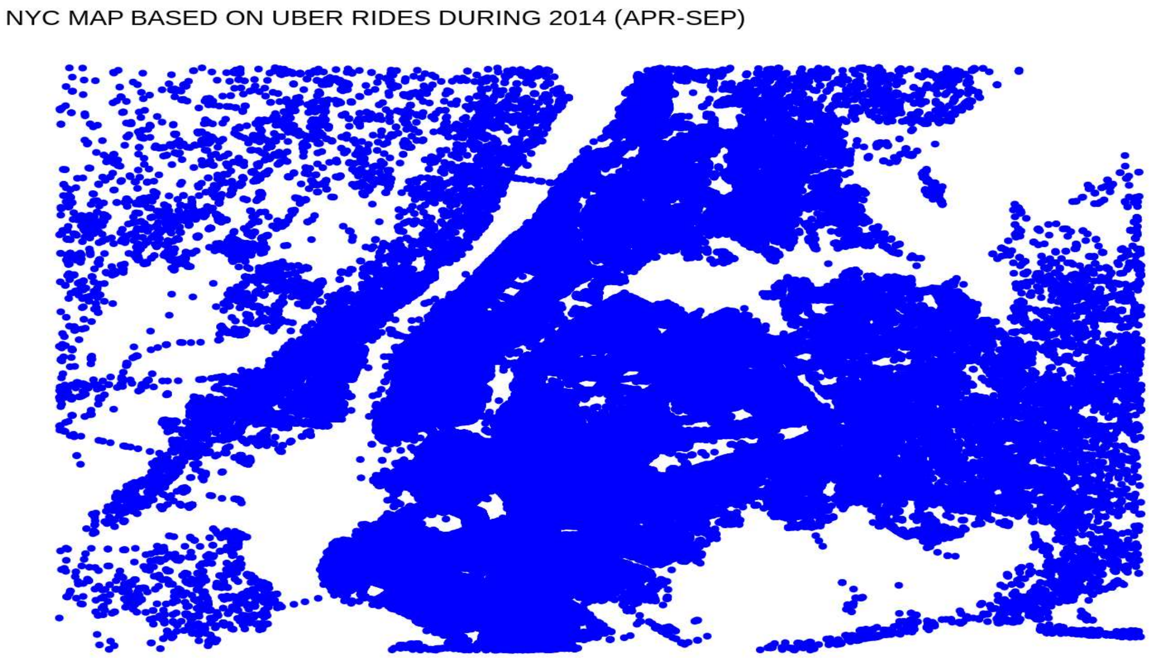
9. Creating a map visualization of rides in New York

In the final section, we will visualize the rides in New York City by creating a geo-plot that will help us to visualize the rides during 2014 (Apr-Sep) and by the bases in the same period.

CODE:

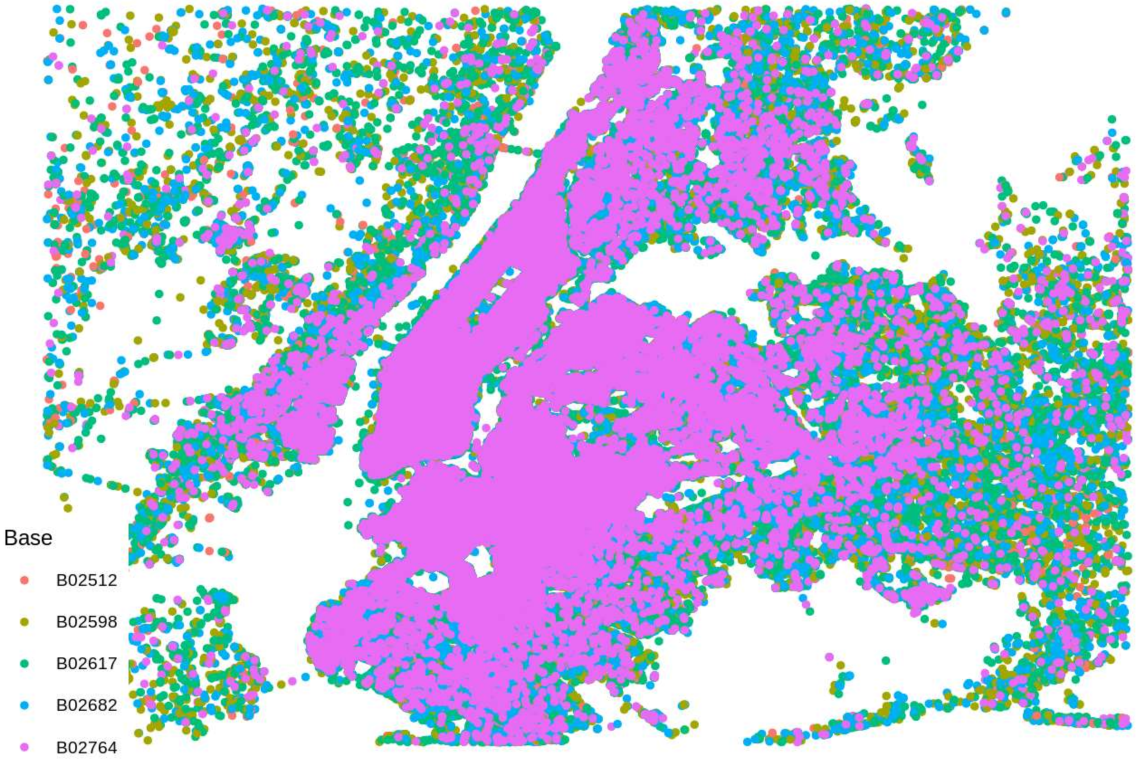
```
min_lat <- 40.5774
max_lat <- 40.9176
min_long <- -74.15
max_long <- -73.7004

ggplot(data_2014, aes(x=Lon, y=Lat)) +
  geom_point(size=1, color = "blue") +
     scale_x_continuous(limits=c(min_long, max_long)) +
      scale_y_continuous(limits=c(min_lat, max_lat)) +
        theme_map() +
           ggtitle("NYC MAP BASED ON UBER RIDES DURING 2014
(APR-SEP)")

ggplot(data_2014, aes(x=Lon, y=Lat, color = Base)) +
  geom_point(size=1) +
     scale_x_continuous(limits=c(min_long, max_long)) +
      scale_y_continuous(limits=c(min_lat, max_lat)) +
       theme_map() +
          ggtitle("NYC MAP BASED ON UBER RIDES DURING 2014
(APR-SEP) by BASE")
```

OUTPUT:



NYC MAP BASED ON UBER RIDES DURING 2014 (APR-SEP)

NYC MAP BASED ON UBER RIDES DURING 2014 (APR-SEP) by BASE



Limitation

- Low quality of data

- Inconsistency in data collection

Conclusion

- At the end of the Uber data analysis R project

- Data visualization helps us for making more understanding of the complex dataset.

- we observed how to create data visualizations.

- We made use of packages like ggplot2 that allowed us to plot various types of visualizations that pertained to several time-frames of the year.

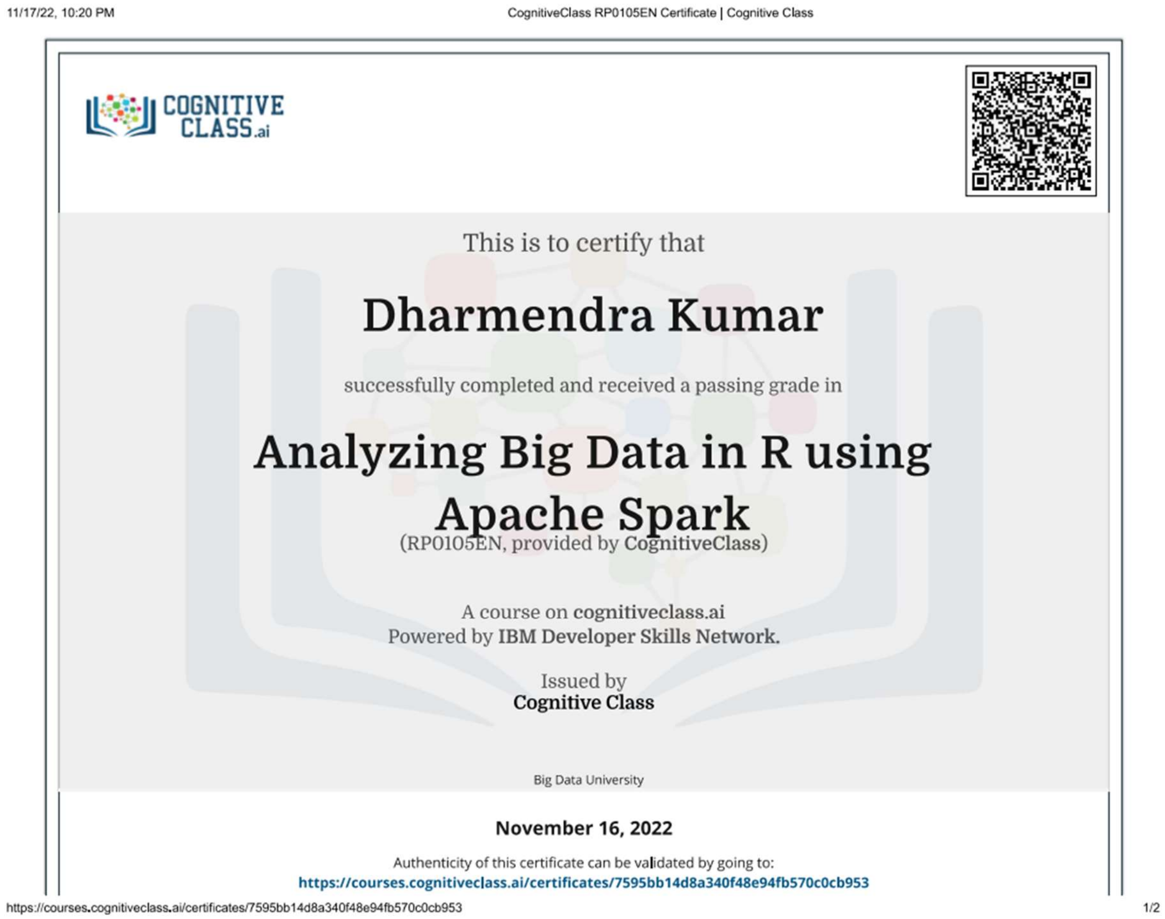
With this, we could conclude how time affected customer trips. Finally, we made a geo plot of New York that provided us with the details of how various users made trips from different bases.

Reference

1) Sabouri, S., Park, K., Smith, A., Tian, G., & Ewing, R. (2020). Exploring the influence of built environment on Uber demand. Transportation Research Part D: Transport and Environment, 81, 102296.
2) Jin, S. T., Kong, H., & Sui, D. Z. (2019). Uber, public transit, and urban transportation equity: A case study in new york city. The Professional Geographer, 71(2), 315-330.
3) Brodeur, A., & Nield, K. (2018). An empirical analysis of taxi, Lyft and Uber rides: Evidence from weather shocks in NYC. Journal of Economic Behavior & Organization, 152, 1-16.
4) Young, M., Allen, J., & Farber, S. (2020). Measuring when Uber behaves as a substitute or supplement to transit: An examination of travel-time differences in Toronto. Journal of Transport Geography, 82, 102629.

Certification

I have completed my certification in **“Analyzing Big Data in R Using Apache Spark”** from **“Cognitive Class”.**



11/17/22, 10:20 PM CognitiveClass RP0105EN Certificate | Cognitive Class

COGNITIVE CLASS.ai

This is to certify that

Dharmendra Kumar

successfully completed and received a passing grade in

Analyzing Big Data in R using Apache Spark

(RP0105EN, provided by CognitiveClass)

A course on **cognitiveclass.ai**
Powered by **IBM Developer Skills Network.**

Issued by
Cognitive Class

Big Data University

November 16, 2022

Authenticity of this certificate can be validated by going to:
https://courses.cognitiveclass.ai/certificates/7595bb14d8a340f48e94fb570c0cb953

https://courses.cognitiveclass.ai/certificates/7595bb14d8a340f48e94fb570c0cb953 1/2