Explore_bikeshare_data

December 13, 2023

0.0.1 Explore Bike Share Data

For this project, your goal is to ask and answer three questions about the available bikeshare data from Washington, Chicago, and New York. This notebook can be submitted directly through the workspace when you are confident in your results.

You will be graded against the project Rubric by a mentor after you have submitted. To get you started, you can use the template below, but feel free to be creative in your solutions!

```
In [8]: library(ggplot2)
        library(lubridate)
```

```
Attaching package: lubridate

The following object is masked from package:base:

    date
```

```
In [2]: ny = read.csv('new_york_city.csv')
        wash = read.csv('washington.csv')
        chi = read.csv('chicago.csv')
```

```
In [3]: head(ny)
```

X	Start.Time	End.Time	Trip.Duration	Start.Station	End.Statio
5688089	2017-06-11 14:55:05	2017-06-11 15:08:21	795	Suffolk St & Stanton St	W Broadwa
4096714	2017-05-11 15:30:11	2017-05-11 15:41:43	692	Lexington Ave & E 63 St	1 Ave & E 7
2173887	2017-03-29 13:26:26	2017-03-29 13:48:31	1325	1 Pl & Clinton St	Henry St &
3945638	2017-05-08 19:47:18	2017-05-08 19:59:01	703	Barrow St & Hudson St	W 20 St & 8
6208972	2017-06-21 07:49:16	2017-06-21 07:54:46	329	1 Ave & E 44 St	E 53 St & 3
1285652	2017-02-22 18:55:24	2017-02-22 19:12:03	998	State St & Smith St	Bond St &

```
In [4]: head(wash)
```

X	Start.Time	End.Time	Trip.Duration	Start.Station
1621326	2017-06-21 08:36:34	2017-06-21 08:44:43	489.066	14th & Belmont St NW
482740	2017-03-11 10:40:00	2017-03-11 10:46:00	402.549	Yuma St & Tenley Circle NW
1330037	2017-05-30 01:02:59	2017-05-30 01:13:37	637.251	17th St & Massachusetts Ave NW
665458	2017-04-02 07:48:35	2017-04-02 08:19:03	1827.341	Constitution Ave & 2nd St NW/DOL
1481135	2017-06-10 08:36:28	2017-06-10 09:02:17	1549.427	Henry Bacon Dr & Lincoln Memorial
1148202	2017-05-14 07:18:18	2017-05-14 07:24:56	398.000	1st & K St SE

```
In [5]: head(chi)
```

X	Start.Time	End.Time	Trip.Duration	Start.Station	Enc
1423854	2017-06-23 15:09:32	2017-06-23 15:14:53	321	Wood St & Hubbard St	Da
955915	2017-05-25 18:19:03	2017-05-25 18:45:53	1610	Theater on the Lake	She
9031	2017-01-04 08:27:49	2017-01-04 08:34:45	416	May St & Taylor St	Wo
304487	2017-03-06 13:49:38	2017-03-06 13:55:28	350	Christiana Ave & Lawrence Ave	St.
45207	2017-01-17 14:53:07	2017-01-17 15:02:01	534	Clark St & Randolph St	Des
1473887	2017-06-26 09:01:20	2017-06-26 09:11:06	586	Clinton St & Washington Blvd	Car

0.1 Data Wrangling/Data Cleaning

0.1.1 We have to wrangle some data first before we can start running out analysis.

```
In [18]: # Need to create an NA filled column in the wash dataframe titled "Gender" and "Birth.Y
         # We do this so when we use a concat feature, all of the dataframes have the same amoun

         wash$Gender <-"NA"
         wash$Birth.Year <- "NA"
         head(wash)
```

X	Start.Time	End.Time	Trip.Duration	Start.Station
1621326	2017-06-21 08:36:34	2017-06-21 08:44:43	489.066	14th & Belmont St NW
482740	2017-03-11 10:40:00	2017-03-11 10:46:00	402.549	Yuma St & Tenley Circle NW
1330037	2017-05-30 01:02:59	2017-05-30 01:13:37	637.251	17th St & Massachusetts Ave NW
665458	2017-04-02 07:48:35	2017-04-02 08:19:03	1827.341	Constitution Ave & 2nd St NW/DOL
1481135	2017-06-10 08:36:28	2017-06-10 09:02:17	1549.427	Henry Bacon Dr & Lincoln Memorial
1148202	2017-05-14 07:18:18	2017-05-14 07:24:56	398.000	1st & K St SE

```
In [20]: # Need to create a City column in each dataframe. We are doing this because once we
         # concat the three together, this will allow us to sort by city data.

         chi$City <- "Chicago"
         ny$City <- "New York City"
         wash$City <- "Washington"

         head(chi)
         head(ny)
         head(wash)
```

X	Start.Time	End.Time	Trip.Duration	Start.Station	End
1423854	2017-06-23 15:09:32	2017-06-23 15:14:53	321	Wood St & Hubbard St	Da
955915	2017-05-25 18:19:03	2017-05-25 18:45:53	1610	Theater on the Lake	She
9031	2017-01-04 08:27:49	2017-01-04 08:34:45	416	May St & Taylor St	Wo
304487	2017-03-06 13:49:38	2017-03-06 13:55:28	350	Christiana Ave & Lawrence Ave	St.
45207	2017-01-17 14:53:07	2017-01-17 15:02:01	534	Clark St & Randolph St	Des
1473887	2017-06-26 09:01:20	2017-06-26 09:11:06	586	Clinton St & Washington Blvd	Car

X	Start.Time	End.Time	Trip.Duration	Start.Station	End.Station
5688089	2017-06-11 14:55:05	2017-06-11 15:08:21	795	Suffolk St & Stanton St	W Broadwa
4096714	2017-05-11 15:30:11	2017-05-11 15:41:43	692	Lexington Ave & E 63 St	1 Ave & E 7
2173887	2017-03-29 13:26:26	2017-03-29 13:48:31	1325	1 Pl & Clinton St	Henry St &
3945638	2017-05-08 19:47:18	2017-05-08 19:59:01	703	Barrow St & Hudson St	W 20 St & 8
6208972	2017-06-21 07:49:16	2017-06-21 07:54:46	329	1 Ave & E 44 St	E 53 St & 3
1285652	2017-02-22 18:55:24	2017-02-22 19:12:03	998	State St & Smith St	Bond St &

X	Start.Time	End.Time	Trip.Duration	Start.Station
1621326	2017-06-21 08:36:34	2017-06-21 08:44:43	489.066	14th & Belmont St NW
482740	2017-03-11 10:40:00	2017-03-11 10:46:00	402.549	Yuma St & Tenley Circle NW
1330037	2017-05-30 01:02:59	2017-05-30 01:13:37	637.251	17th St & Massachusetts Ave NW
665458	2017-04-02 07:48:35	2017-04-02 08:19:03	1827.341	Constitution Ave & 2nd St NW/DOL
1481135	2017-06-10 08:36:28	2017-06-10 09:02:17	1549.427	Henry Bacon Dr & Lincoln Memorial
1148202	2017-05-14 07:18:18	2017-05-14 07:24:56	398.000	1st & K St SE

In [23]:
```
#Concat the three dataframes together in order to run our analysis over all three at on
#We can find the data we are looking for without doing this, but this will create clea

# The function takes two dataframes and uses rbind to concat and add the second datafra
city_concat <- function(df1, df2) {
    return(rbind(df1, df2))
}
```

In [27]:
```
#Adds the wash dataframe to the ny dataframe us rbind
city_df <- city_concat(ny, wash)

#Adds the chi dataframe to the city_df dataframe us rbind
city_df <- city_concat(city_df, chi)
head(city_df)
```

X	Start.Time	End.Time	Trip.Duration	Start.Station	End.Station
5688089	2017-06-11 14:55:05	2017-06-11 15:08:21	795	Suffolk St & Stanton St	W Broadwa
4096714	2017-05-11 15:30:11	2017-05-11 15:41:43	692	Lexington Ave & E 63 St	1 Ave & E 7
2173887	2017-03-29 13:26:26	2017-03-29 13:48:31	1325	1 Pl & Clinton St	Henry St &
3945638	2017-05-08 19:47:18	2017-05-08 19:59:01	703	Barrow St & Hudson St	W 20 St & 8
6208972	2017-06-21 07:49:16	2017-06-21 07:54:46	329	1 Ave & E 44 St	E 53 St & 3
1285652	2017-02-22 18:55:24	2017-02-22 19:12:03	998	State St & Smith St	Bond St &

0.1.2 Question 1

0.2 What does the data tell us about the monhtly usage of these rentals? Can we determine which months were the least and most popular per city?

```
In [29]: #Changing the start time and end time to a different format, so we can extract the mont
         city_df$Start.Time <-ymd_hms(city_df$Start.Time)
         city_df$End.Time <-ymd_hms(city_df$End.Time)

         #Extracting the month data from the start time and creating a new column with the data.
         city_df$Month <- month(city_df$Start.Time)

         head(city_df)
```

```
Warning message:
 1 failed to parse.
```

X	Start.Time	End.Time	Trip.Duration	Start.Station	End.Station
5688089	2017-06-11 14:55:05	2017-06-11 15:08:21	795	Suffolk St & Stanton St	W Broadwa
4096714	2017-05-11 15:30:11	2017-05-11 15:41:43	692	Lexington Ave & E 63 St	1 Ave & E 7
2173887	2017-03-29 13:26:26	2017-03-29 13:48:31	1325	1 Pl & Clinton St	Henry St &
3945638	2017-05-08 19:47:18	2017-05-08 19:59:01	703	Barrow St & Hudson St	W 20 St & 8
6208972	2017-06-21 07:49:16	2017-06-21 07:54:46	329	1 Ave & E 44 St	E 53 St & 3
1285652	2017-02-22 18:55:24	2017-02-22 19:12:03	998	State St & Smith St	Bond St &

```
In [34]: #The total count per month in each city.
         by(city_df$City, city_df$Month,summary)
```

```
city_df$Month: 1
      Chicago New York City    Washington
          650          5745          8946
------------------------------------------------------------
city_df$Month: 2
      Chicago New York City    Washington
          930          6364         11563
------------------------------------------------------------
city_df$Month: 3
      Chicago New York City    Washington
          803          5820         12612
------------------------------------------------------------
city_df$Month: 4
      Chicago New York City    Washington
         1526         10661         18522
------------------------------------------------------------
city_df$Month: 5
      Chicago New York City    Washington
         1905         12180         17072
------------------------------------------------------------
city_df$Month: 6
```

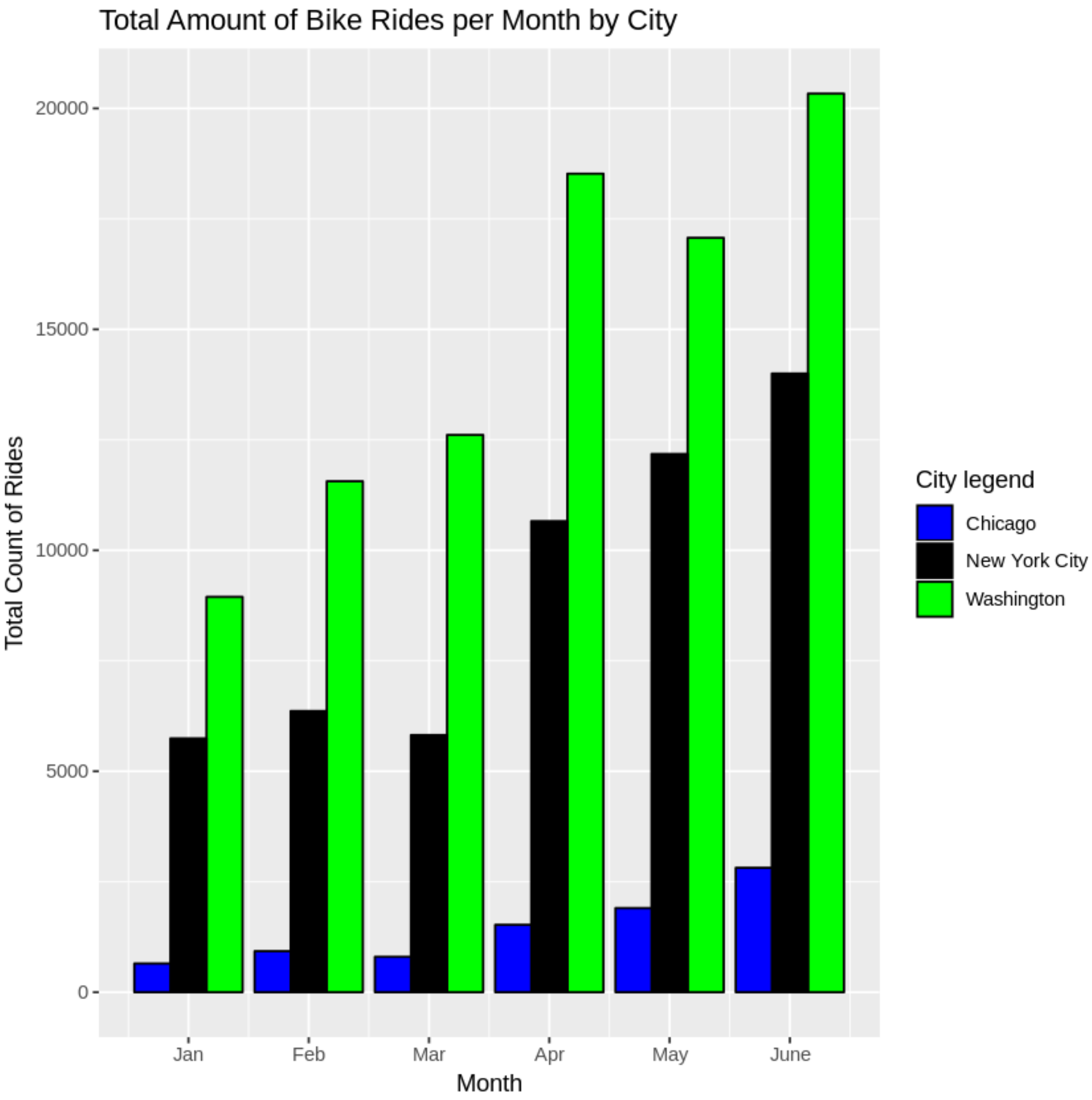
```
     Chicago New York City    Washington 
        2816         14000         20335 
```

```
In [36]: #The total count for all three cities by month.
         table(city_df$Month)
```

```
    1     2     3     4     5     6 
15341 18857 19235 30709 31157 37151 
```

```
In [44]: #Chart to show the visual side-by-side analysis.
         # Using fill to fill the data from the City column.
         # Position can be stacked or side-by-side. Dodge is used for side-by-side.
         # Scale_fill_manual to create the legend for the graph.

         ggplot(aes(x=Month, fill=City), data = city_df) +
               geom_bar(position= 'dodge', colour = "black") +
               scale_x_continuous(breaks = c(1, 2, 3, 4, 5, 6), labels = c('Jan', 'Feb', 'Mar',
                ggtitle('Total Amount of Bike Rides per Month by City')  +
                labs(y = 'Total Count of Rides', x = 'Month') +
                scale_fill_manual('City legend', values = c('Chicago'='blue', "New York City"="b
```

```
Warning message:
Removed 1 rows containing non-finite values (stat_count).
```



Summary of your question 1 results goes here.

+Based on the visuals we see on the graph, we see that the total count of riders in Washington lead the group each month, followed by NYC, and then Chicago. +Our numerical data shows us that the most popular month was June with 37151 riders, while the least popular was January with 15341. +The highest single month total came in June, with Washington seeing 20335 riders that month. +The lowest single month total came in January, with Chicago seeing only 650 riders.

It is hard to tell just from the data what makes these numbers stand out like this. Without putting much thought into it, we would say that Washington and NYC are doing well with the program and shoudl continue what they are doing. While in Chicgao, the program may not be beneficial.

When taking an outside viewpoint, we may be left to ask are these numbers based on population differences, SES differences, bicycle saftey and availability. It is hard to pinopint why the data relies this information, without having to dig up more information behind the data.

0.2.1 Question 2

0.3 Does average transit time differ between user type?

```
In [48]: #Seeing the total users in our dataframe.
         total_users = sort(table(city_df$User.Type))
         print(total_users)

         #Percentage of users.
         perc_total_users = (total_users/length(city_df$User.Type) * 100)
         print(perc_total_users)
```

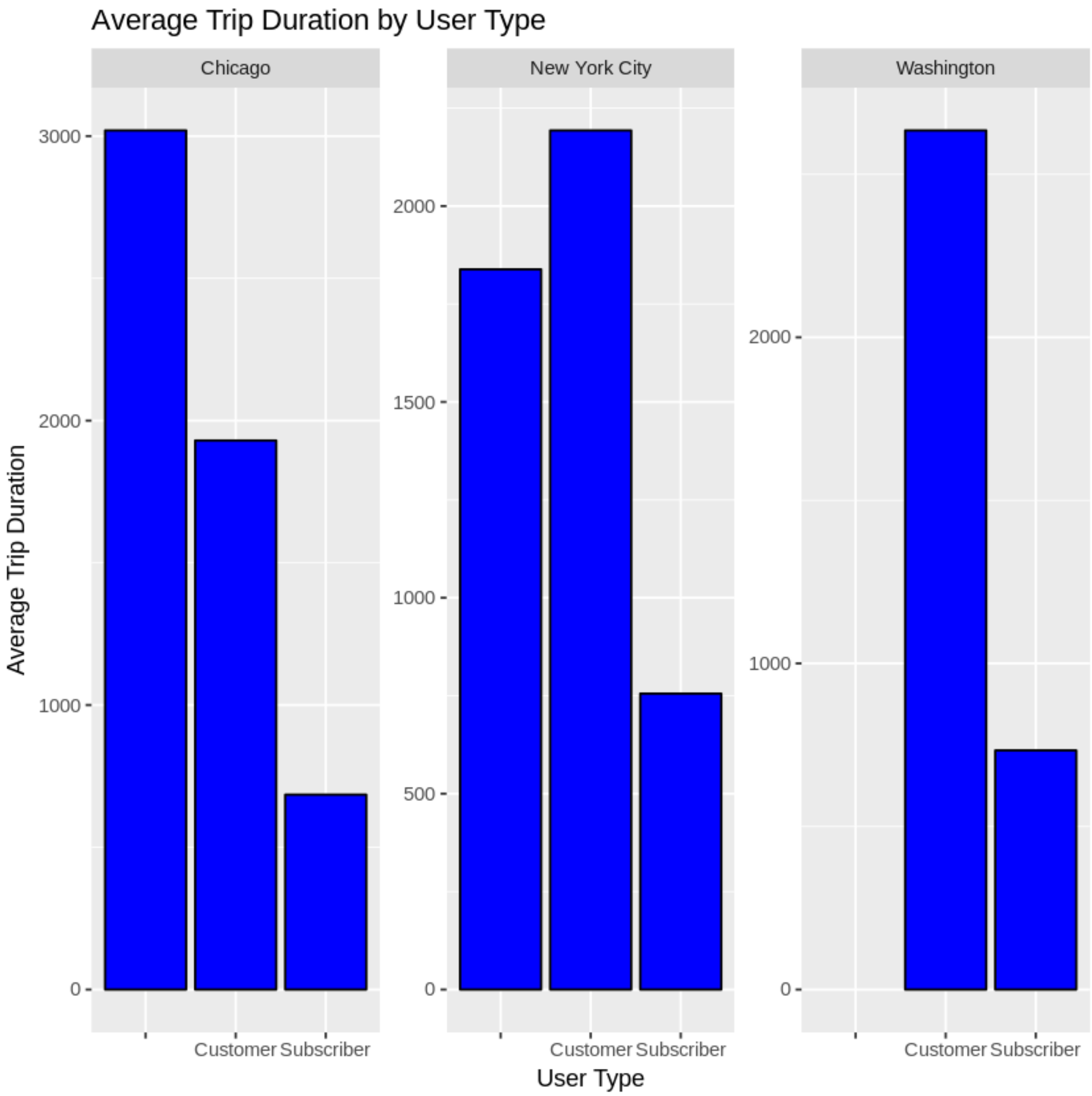
```
            Customer Subscriber
       121      30754     121576

               Customer  Subscriber
 0.07936976 20.17303921 79.74759103
```

```
In [51]: #Summary statistcis of the trip duration by user type.
         by(city_df$Trip.Duration, city_df$User.Type, summary)
```

```
city_df$User.Type:
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
    201     764    1112    1848    1536   51595       2
------------------------------------------------------------
city_df$User.Type: Customer
     Min.    1st Qu.    Median      Mean    3rd Qu.      Max.
     61.3     915.4    1450.0    2514.7     2404.5 1088634.0
------------------------------------------------------------
city_df$User.Type: Subscriber
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
    60.0    352.2    567.5    739.5    914.7 170032.9
```

```
In [64]: ggplot(aes(x=User.Type, y = Trip.Duration), data = city_df) +
             geom_histogram(stat="summary", fun.y="mean", color = "black", fill = "blue") +
             facet_wrap(~City, scales='free') +
             ggtitle('Average Trip Duration by User Type') +
             labs(y= 'Average Trip Duration', x = 'User Type')
```

```
Warning message:
Ignoring unknown parameters: binwidth, bins, padWarning message:
Removed 2 rows containing non-finite values (stat_summary).
```



Summary of your question 2 results goes here.

+The numerical data shows us that the average trip duration for a customer is 2514.7, while the average trip duration for a subscriber is 739.5. This is a significant difference, but we can also see that the subscribers make up almost 80% of the data aand the max customer average is 1088634. Both of these factors can signifcantly skew the means or dilute them down.

+The random fact that we find is that in Chicago, the average trip duration is highest among people who do not classify as customer or subscriber. And in NYC, the unkown user type is almost as high as the Customer user type in average trip duration. We would have to look at the data more to figure out who these people represent and why their data is so large in these two cities, but non-existent in Washington. This is a weird fact for a group that only makes up 7% of the demographics found.

+Based on face-value, this data can be helpful to the company behind the ride sharing because it shows them that customers, rather than subscribers, may lead to higher profits. But, corrleation does not equal causation, so there is much more we need to know.

0.3.1 Question 3

0.4 What do the ages of our riders look like for NY and Chicago?

new concat of the two df's. Compute age by subtracting age from year of start time. Run results.

```
In [66]: # Create a new dataframe using ny and chi only, since wash didn't contain Birth.Year

         age_df <- city_concat(ny, chi)
         head(age_df)
```

X	Start.Time	End.Time	Trip.Duration	Start.Station	End.Statio
5688089	2017-06-11 14:55:05	2017-06-11 15:08:21	795	Suffolk St & Stanton St	W Broadwa
4096714	2017-05-11 15:30:11	2017-05-11 15:41:43	692	Lexington Ave & E 63 St	1 Ave & E 7
2173887	2017-03-29 13:26:26	2017-03-29 13:48:31	1325	1 Pl & Clinton St	Henry St &
3945638	2017-05-08 19:47:18	2017-05-08 19:59:01	703	Barrow St & Hudson St	W 20 St & 8
6208972	2017-06-21 07:49:16	2017-06-21 07:54:46	329	1 Ave & E 44 St	E 53 St & 3
1285652	2017-02-22 18:55:24	2017-02-22 19:12:03	998	State St & Smith St	Bond St &

```
In [68]: #Changing the start time and end time to a different format, so we can extract the year
         age_df$Start.Time <-ymd_hms(age_df$Start.Time)
         age_df$End.Time <-ymd_hms(age_df$End.Time)

         #Extracting the year data from the start time and creating a new column with the data.
         age_df$Year <- year(age_df$Start.Time)

         head(age_df)
```

X	Start.Time	End.Time	Trip.Duration	Start.Station	End.Statio
5688089	2017-06-11 14:55:05	2017-06-11 15:08:21	795	Suffolk St & Stanton St	W Broadwa
4096714	2017-05-11 15:30:11	2017-05-11 15:41:43	692	Lexington Ave & E 63 St	1 Ave & E 7
2173887	2017-03-29 13:26:26	2017-03-29 13:48:31	1325	1 Pl & Clinton St	Henry St &
3945638	2017-05-08 19:47:18	2017-05-08 19:59:01	703	Barrow St & Hudson St	W 20 St & 8
6208972	2017-06-21 07:49:16	2017-06-21 07:54:46	329	1 Ave & E 44 St	E 53 St & 3
1285652	2017-02-22 18:55:24	2017-02-22 19:12:03	998	State St & Smith St	Bond St &

```
In [69]: # Creating an Age column.
         age_df$Age <- (age_df$Year - age_df$Birth.Year)
         head(age_df)
```

X	Start.Time	End.Time	Trip.Duration	Start.Station	End.Statio
5688089	2017-06-11 14:55:05	2017-06-11 15:08:21	795	Suffolk St & Stanton St	W Broadwa
4096714	2017-05-11 15:30:11	2017-05-11 15:41:43	692	Lexington Ave & E 63 St	1 Ave & E 7
2173887	2017-03-29 13:26:26	2017-03-29 13:48:31	1325	1 Pl & Clinton St	Henry St &
3945638	2017-05-08 19:47:18	2017-05-08 19:59:01	703	Barrow St & Hudson St	W 20 St & 8
6208972	2017-06-21 07:49:16	2017-06-21 07:54:46	329	1 Ave & E 44 St	E 53 St & 3
1285652	2017-02-22 18:55:24	2017-02-22 19:12:03	998	State St & Smith St	Bond St &

```
In [71]: by(age_df$Age, age_df$City, summary)
```

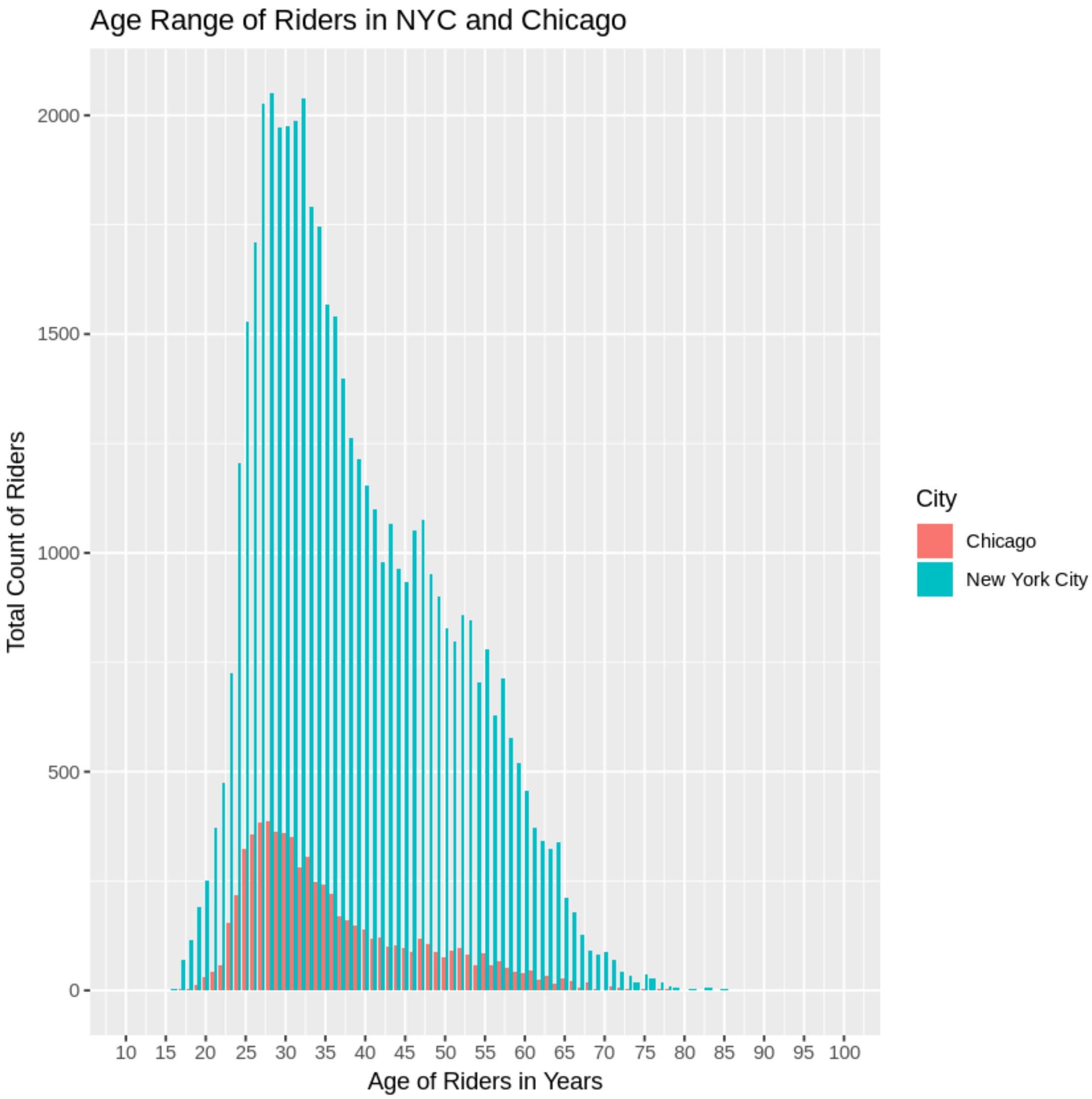
```
age_df$City: Chicago
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
   15.0    28.0    33.0    36.1    42.0   118.0    1747 
------------------------------------------------------------ 
age_df$City: New York City
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
  16.00   29.00   36.00   38.79   47.00  132.00    5218 
```

```
In [73]: #See the spread so we can tell where some of our outliers are and minimize our spread o
         table(age_df$Age)
```

```

  15   16   17   18   19   20   21   22   23   24   25   26   27   28   29   30 
   1    3   73  118  204  283  416  531  879 1423 1851 2066 2411 2438 2335 2336 
  31   32   33   34   35   36   37   38   39   40   41   42   43   44   45   46 
2337 2319 2094 1996 1809 1762 1568 1423 1364 1293 1220 1102 1165 1065 1030 1138 
  47   48   49   50   51   52   53   54   55   56   57   58   59   60   61   62 
1193 1056  988  905  887  955  929  761  863  687  782  629  565  496  418  365 
  63   64   65   66   67   68   69   70   71   72   73   74   75   76   77   78 
 359  355  240  202  133  109   85   88   80   47   37   19   41   28   20   10 
  79   80   81   82   83   85   87   90   91   94   99  100  107  116  117  118 
   5    1    2    1    7    3    2    1    1    1    1    1    3    3    9    4 
 124  131  132 
   1    1    3 
```

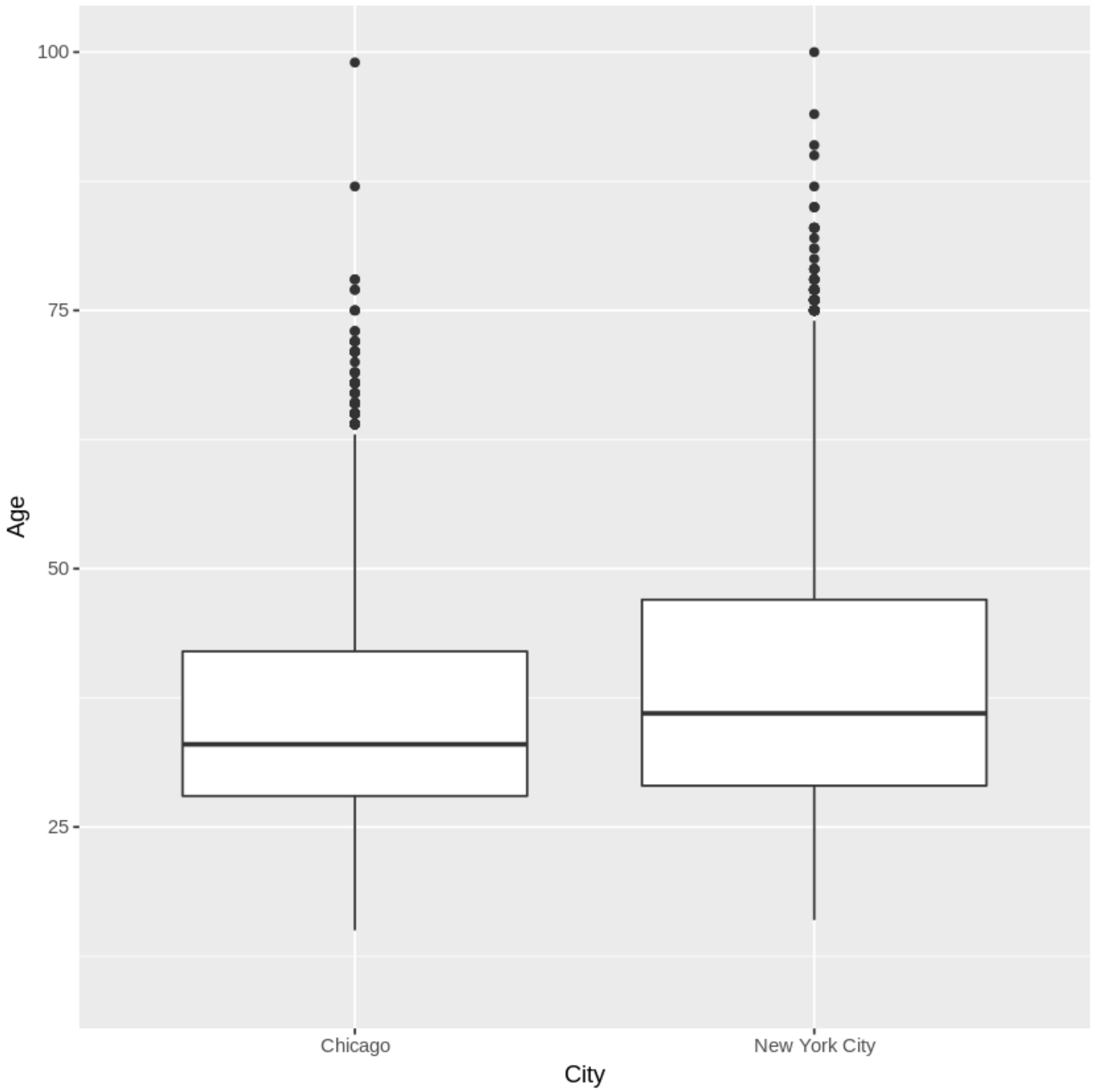
```
In [78]: ggplot(aes(x=Age, fill = City), data=age_df)+
             geom_bar(position='dodge')+
             ggtitle("Age Range of Riders in NYC and Chicago")+
             scale_x_continuous(breaks = seq(10, 100, by = 5))+
             labs(x = "Age of Riders in Years", y = "Total Count of Riders")+
             coord_cartesian(xlim=c(10, 100))
```

```
Warning message:
Removed 6965 rows containing non-finite values (stat_count).
```



```
In [92]: ggplot(data=age_df, mapping=aes(x=City, y = Age))+
             geom_boxplot()+
             coord_cartesian (ylim = c(10, 100))
```

```
Warning message:
Removed 6965 rows containing non-finite values (stat_boxplot).
```



Summary of your question 3 results goes here.

+The first thing that we should look at with this data set are the Min and Max numbers for each city. The Mins are 15 and 16, which woudl make sense for individuals riding bikes. The maxs are 118 and 132. This age seems a little high for me as far as people riding bikes. It could be possible, but I would set those up as outliers. For the visual, we set the max at 100, to not skew away focus from the main data.

+The interesting statline would be the 1st interquartile, mean, and 3rd interquartile levels. For New York, the number are 28, 36, and 42. For Chicago they are 29, 39, and 47. These numbers are pretty close to one another, which tells us that the age range per city are similar to one another. We can see this in our graph, as the trends in the graph tend to rise up around 25 and lower around 40. Even though the total counts vary, the average data tells us that these cities are alike.

0.5 Finishing Up

Congratulations! You have reached the end of the Explore Bikeshare Data Project. You should be very proud of all you have accomplished!

Tip: Once you are satisfied with your work here, check over your report to make sure that it is satisfies all the areas of the rubric.

0.6 Directions to Submit

Before you submit your project, you need to create a .html or .pdf version of this notebook in the workspace here. To do that, run the code cell below. If it worked correctly, you should get a return code of 0, and you should see the generated .html file in the workspace directory (click on the orange Jupyter icon in the upper left).

Alternatively, you can download this report as .html via the **File** > **Download as** submenu, and then manually upload it into the workspace directory by clicking on the orange Jupyter icon in the upper left, then using the Upload button.

Once you've done this, you can submit your project by clicking on the "Submit Project" button in the lower right here. This will create and submit a zip file with this .ipynb doc and the .html or .pdf version you created. Congratulations!

```
In [ ]: system('python -m nbconvert Explore_bikeshare_data.ipynb')
```