Part_I_notebook

July 30, 2022

1 Part I - (Ford GoBike System Data)

1.1 by (Rilwan Shittu)

1.2 Introduction

This dataset includes information about individual rides made in a bike-sharing system covering the greater San Francisco Bay area in February 2019.

1.3 Preliminary Wrangling

```
In [45]: # import all packages and set plots to be embedded inline
         import numpy as np
         import pandas as pd
         import matplotlib.pyplot as plt
         import seaborn as sb
         import math


         %matplotlib inline
```

```
In [46]: # load the csv file into the pandas dataframe
         df = pd.read_csv('201902-fordgobike-tripdata.csv')
         df_clean = df.copy()
```

```
In [47]: # Get an overhead view of the data
         df_clean.head()
```

```
Out[47]:    duration_sec                start_time                  end_time  \
         0         52185  2019-02-28 17:32:10.1450  2019-03-01 08:01:55.9750
         1         42521  2019-02-28 18:53:21.7890  2019-03-01 06:42:03.0560
         2         61854  2019-02-28 12:13:13.2180  2019-03-01 05:24:08.1460
         3         36490  2019-02-28 17:54:26.0100  2019-03-01 04:02:36.8420
         4          1585  2019-02-28 23:54:18.5490  2019-03-01 00:20:44.0740

            start_station_id                                start_station_name  \
         0              21.0  Montgomery St BART Station (Market St at 2nd St)
         1              23.0                     The Embarcadero at Steuart St
         2              86.0                           Market St at Dolores St
```

```
3           375.0                             Grove St at Masonic Ave
4             7.0                               Frank H Ogawa Plaza

   start_station_latitude  start_station_longitude  end_station_id  \
0               37.789625              -122.400811            13.0
1               37.791464              -122.391034            81.0
2               37.769305              -122.426826             3.0
3               37.774836              -122.446546            70.0
4               37.804562              -122.271738           222.0

                              end_station_name  end_station_latitude  \
0              Commercial St at Montgomery St             37.794231
1                         Berry St at 4th St             37.775880
2  Powell St BART Station (Market St at 4th St)             37.786375
3                       Central Ave at Fell St             37.773311
4                        10th Ave at E 15th St             37.792714

   end_station_longitude  bike_id   user_type  member_birth_year  \
0            -122.402923     4902    Customer             1984.0
1            -122.393170     2535    Customer                NaN
2            -122.404904     5905    Customer             1972.0
3            -122.444293     6638  Subscriber             1989.0
4            -122.248780     4898  Subscriber             1974.0

  member_gender bike_share_for_all_trip
0          Male                      No
1           NaN                      No
2          Male                      No
3         Other                      No
4          Male                     Yes
```

In [48]:
```
# Check the general information regarding all variables
df_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 183412 entries, 0 to 183411
Data columns (total 16 columns):
duration_sec               183412 non-null int64
start_time                 183412 non-null object
end_time                   183412 non-null object
start_station_id           183215 non-null float64
start_station_name         183215 non-null object
start_station_latitude     183412 non-null float64
start_station_longitude    183412 non-null float64
end_station_id             183215 non-null float64
end_station_name           183215 non-null object
end_station_latitude       183412 non-null float64
end_station_longitude      183412 non-null float64
```

```
bike_id                    183412 non-null int64
user_type                  183412 non-null object
member_birth_year          175147 non-null float64
member_gender              175147 non-null object
bike_share_for_all_trip    183412 non-null object
dtypes: float64(7), int64(2), object(7)
memory usage: 22.4+ MB
```

```
In [49]: # Drop all rows with missing values
         df_clean.dropna(inplace=True)

In [50]: # Convert start time and end time columns to datetime
         df_clean[['start_time','end_time']] = df_clean[['start_time','end_time']].apply(pd.to_d

In [51]: # convert the stations and bike id's to object types
         df_clean[['start_station_id','end_station_id','bike_id']] = df_clean[['start_station_id

In [52]: # Create a column for the duration of rides in minutes from their duration in seconds
         df_clean['duration_min'] = round(df_clean['duration_sec'].astype(float) / 60,2)

In [53]: # Changing the datatype of their year of birth from float to integer
         df_clean['member_birth_year'] = df_clean['member_birth_year'].astype(int)

In [54]: # Create a column for the members age from their year of birth
         df_clean['member_age'] = 2019 - df_clean['member_birth_year'].astype(int)

In [55]: # Engineer a feature that reveals the actual distance travelled in km
         def get_distance(row, r = 6371):
             """function to measure the distance between latitudinal and longitudinal degrees""
             dlon = row[1]['end_station_longitude'] - row[1]['start_station_longitude']
             dlat = row[1]['end_station_latitude'] - row[1]['start_station_latitude']
             a = ((math.sin(dlat/2))**2 + math.cos(row[1]['start_station_latitude']) * math.cos(
             c = 2 * math.atan2(math.sqrt(a), math.sqrt(1-a))
             return r * c

         df_clean['dist_bet_stations'] = [round(get_distance(row),2) for row in df_clean.iterrow

In [56]: # Check for effectiveness of changes made above
         df_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 174952 entries, 0 to 183411
Data columns (total 19 columns):
duration_sec               174952 non-null int64
start_time                 174952 non-null datetime64[ns]
end_time                   174952 non-null datetime64[ns]
start_station_id           174952 non-null object
start_station_name         174952 non-null object
```

```
start_station_latitude     174952 non-null float64
start_station_longitude    174952 non-null float64
end_station_id             174952 non-null object
end_station_name           174952 non-null object
end_station_latitude       174952 non-null float64
end_station_longitude      174952 non-null float64
bike_id                    174952 non-null object
user_type                  174952 non-null object
member_birth_year          174952 non-null int64
member_gender              174952 non-null object
bike_share_for_all_trip    174952 non-null object
duration_min               174952 non-null float64
member_age                 174952 non-null int64
dist_bet_stations          174952 non-null float64
dtypes: datetime64[ns](2), float64(6), int64(3), object(8)
memory usage: 26.7+ MB
```

1.3.1 What is the structure of your dataset?

The original dataset included 183,412 rows and 16 columns. However after wrangling the data and engineering some new features, the dataset to be used for analysis now includes 174,952 rows and 19 columns.

1.3.2 What is/are the main feature(s) of interest in your dataset?

I am most interested in determining the factors that affect the duration of the ride.

1.3.3 What features in the dataset do you think will help support your investigation into your feature(s) of interest?

I think the user type, age, gender and distance travelled will all be interesting to investigate in relation to the period the bike was being used.

1.4 Univariate Exploration

I will start with a quick descriptive summary of the numeric variables in the dataset.

In [57]:
```
# A descriptive summary of the numeric variables in the dataset
df_clean.describe()
```

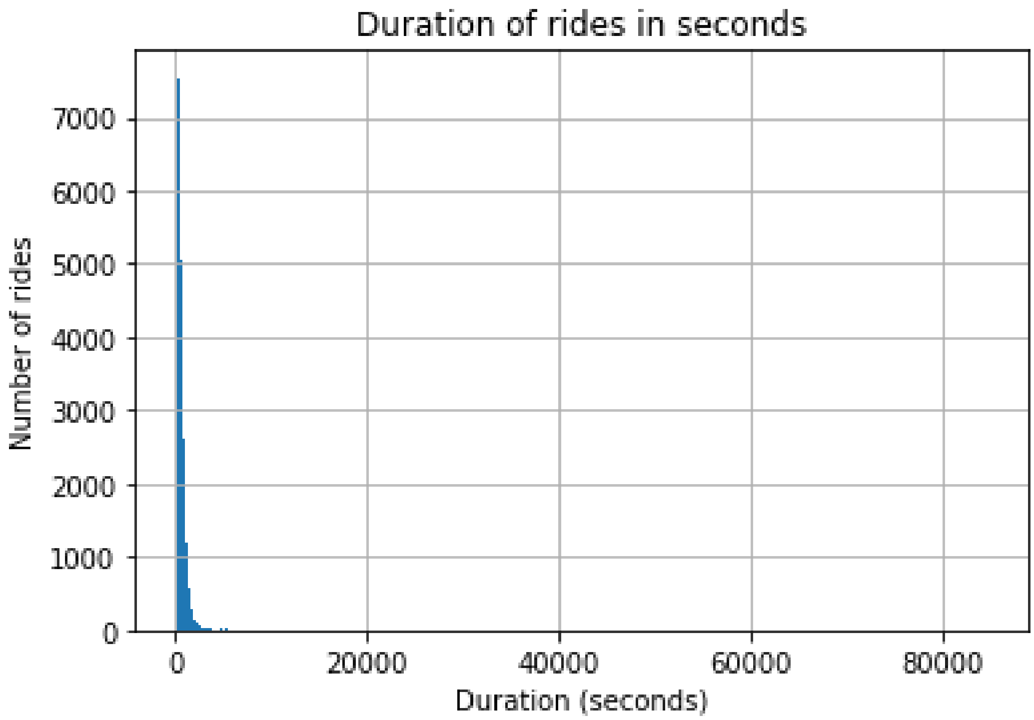
Out[57]:

	duration_sec	start_station_latitude	start_station_longitude \
count	174952.000000	174952.000000	174952.000000
mean	704.002744	37.771220	-122.351760
std	1642.204905	0.100391	0.117732
min	61.000000	37.317298	-122.453704
25%	323.000000	37.770407	-122.411901
50%	510.000000	37.780760	-122.398279
75%	789.000000	37.797320	-122.283093

```
max     84548.000000               37.880222              -121.874119

       end_station_latitude  end_station_longitude  member_birth_year  \
count         174952.000000          174952.000000      174952.000000
mean              37.771414            -122.351335        1984.803135
std                0.100295               0.117294          10.118731
min               37.317298            -122.453704        1878.000000
25%               37.770407            -122.411647        1980.000000
50%               37.781010            -122.397437        1987.000000
75%               37.797673            -122.286533        1992.000000
max               37.880222            -121.874119        2001.000000

        duration_min     member_age  dist_bet_stations
count  174952.000000  174952.000000      174952.000000
mean       11.733373      34.196865         107.532067
std        27.370085      10.118731          69.444751
min         1.020000      18.000000           0.000000
25%         5.380000      27.000000          57.930000
50%         8.500000      32.000000          90.360000
75%        13.150000      39.000000         141.430000
max      1409.130000     141.000000        4405.790000
```

Next, I will examine the main variables of interest which are the variables that give direct information on the duration of the rides. They include the durations in seconds and minutes.

In [58]:
```
# looking at the duration of rides in seconds
binsize = 30
bins = np.arange(61, df_clean['duration_sec'].max()+binsize ,binsize)
df_clean.duration_sec.hist(bins=bins)
plt.xlabel('Duration (seconds)')
plt.ylabel('Number of rides')
plt.title('Duration of rides in seconds');
```




```
In [59]: # Taking a look the visually observed outliers to determine if they should be dropped.
         # All the seconds figures are accurate as they are the outcome of the difference betwee
         # Therefore, they will not be dropped and a log transform will be performed on the vari
         outliers = (df_clean['duration_sec'] > 5000)

         print(outliers.sum())
         print(df_clean.loc[outliers,:])
```

```
884
        duration_sec              start_time                end_time  \
0              52185 2019-02-28 17:32:10.145 2019-03-01 08:01:55.975
2              61854 2019-02-28 12:13:13.218 2019-03-01 05:24:08.146
3              36490 2019-02-28 17:54:26.010 2019-03-01 04:02:36.842
91              5621 2019-02-28 21:41:16.900 2019-02-28 23:14:58.186
199            15123 2019-02-28 18:23:19.035 2019-02-28 22:35:22.294
297            13061 2019-02-28 18:28:18.728 2019-02-28 22:05:59.954
511             7421 2019-02-28 19:16:02.778 2019-02-28 21:19:44.144
524             6447 2019-02-28 19:30:09.314 2019-02-28 21:17:36.905
779            36190 2019-02-28 10:30:03.377 2019-02-28 20:33:14.228
790             5927 2019-02-28 18:52:30.715 2019-02-28 20:31:17.765
813             9994 2019-02-28 17:41:05.362 2019-02-28 20:27:39.511
926             5085 2019-02-28 18:48:01.975 2019-02-28 20:12:47.641
939            16804 2019-02-28 15:30:33.480 2019-02-28 20:10:38.106
945            16768 2019-02-28 15:30:32.430 2019-02-28 20:10:01.410
```

1023	5029	2019-02-28	18:35:38.200	2019-02-28	19:59:27.660
1826	9174	2019-02-28	16:14:40.336	2019-02-28	18:47:35.285
2188	9424	2019-02-28	15:50:03.285	2019-02-28	18:27:07.429
2298	20156	2019-02-28	12:44:12.902	2019-02-28	18:20:09.562
2544	31633	2019-02-28	09:20:43.087	2019-02-28	18:07:56.344
2588	30666	2019-02-28	09:35:13.840	2019-02-28	18:06:20.011
2693	21883	2019-02-28	11:56:08.063	2019-02-28	18:00:52.007
3401	62452	2019-02-28	00:04:01.344	2019-02-28	17:24:54.137
3530	13844	2019-02-28	13:27:32.213	2019-02-28	17:18:16.517
3985	8730	2019-02-28	14:21:39.135	2019-02-28	16:47:09.399
4040	8221	2019-02-28	14:25:20.970	2019-02-28	16:42:22.222
4223	9472	2019-02-28	13:45:40.233	2019-02-28	16:23:32.309
4233	26988	2019-02-28	08:52:26.557	2019-02-28	16:22:14.575
4658	9185	2019-02-28	12:53:52.772	2019-02-28	15:26:58.516
4828	6617	2019-02-28	13:07:44.244	2019-02-28	14:58:01.965
4855	6222	2019-02-28	13:10:37.343	2019-02-28	14:54:20.269
...	...		...		...
175322	13747	2019-02-02	11:02:47.927	2019-02-02	14:51:55.394
175831	10745	2019-02-02	10:27:39.392	2019-02-02	13:26:45.166
175857	7958	2019-02-02	11:09:28.910	2019-02-02	13:22:07.575
175860	8059	2019-02-02	11:07:39.715	2019-02-02	13:21:59.456
176130	59813	2019-02-01	19:54:49.848	2019-02-02	12:31:43.043
176188	5113	2019-02-02	10:56:49.725	2019-02-02	12:22:03.382
176490	5169	2019-02-02	09:59:54.842	2019-02-02	11:26:04.752
176639	11063	2019-02-02	07:55:46.804	2019-02-02	11:00:10.190
176707	5967	2019-02-02	09:10:34.954	2019-02-02	10:50:02.543
177144	51488	2019-02-01	17:22:49.870	2019-02-02	07:40:58.473
177219	6270	2019-02-01	23:48:08.946	2019-02-02	01:32:39.506
177251	11168	2019-02-01	21:40:04.717	2019-02-02	00:46:13.697
177279	30767	2019-02-01	15:44:06.638	2019-02-02	00:16:53.653
177337	8468	2019-02-01	20:53:53.210	2019-02-01	23:15:01.934
177484	13693	2019-02-01	17:45:35.738	2019-02-01	21:33:49.027
177662	12154	2019-02-01	16:30:44.854	2019-02-01	19:53:19.801
177908	15565	2019-02-01	14:22:38.724	2019-02-01	18:42:03.928
177912	15541	2019-02-01	14:22:14.557	2019-02-01	18:41:15.818
178099	9588	2019-02-01	15:19:32.158	2019-02-01	17:59:20.321
178416	31203	2019-02-01	08:40:28.487	2019-02-01	17:20:31.553
178649	14321	2019-02-01	12:50:10.691	2019-02-01	16:48:51.797
178656	12894	2019-02-01	13:13:12.725	2019-02-01	16:48:07.003
179375	6508	2019-02-01	12:21:19.721	2019-02-01	14:09:48.338
179529	6974	2019-02-01	11:44:08.980	2019-02-01	13:40:23.503
179535	6963	2019-02-01	11:43:20.012	2019-02-01	13:39:23.047
179732	10568	2019-02-01	10:08:37.189	2019-02-01	13:04:45.198
180303	6629	2019-02-01	09:45:11.581	2019-02-01	11:35:41.145
182133	6086	2019-02-01	07:00:02.042	2019-02-01	08:41:28.251
182411	13609	2019-02-01	04:38:43.601	2019-02-01	08:25:33.493
183326	5713	2019-02-01	01:02:55.168	2019-02-01	02:38:09.002

	start_station_id	start_station_name \
0	21	Montgomery St BART Station (Market St at 2nd St)
2	86	Market St at Dolores St
3	375	Grove St at Masonic Ave
91	252	Channing Way at Shattuck Ave
199	28	The Embarcadero at Bryant St
297	19	Post St at Kearny St
511	266	Parker St at Fulton St
524	66	3rd St at Townsend St
779	58	Market St at 10th St
790	5	Powell St BART Station (Market St at 5th St)
813	241	Ashby BART Station
926	29	O'Farrell St at Divisadero St
939	5	Powell St BART Station (Market St at 5th St)
945	5	Powell St BART Station (Market St at 5th St)
1023	53	Grove St at Divisadero
1826	284	Yerba Buena Center for the Arts (Howard St at ...
2188	187	Jack London Square
2298	358	Williams Ave at 3rd St
2544	81	Berry St at 4th St
2588	58	Market St at 10th St
2693	284	Yerba Buena Center for the Arts (Howard St at ...
3401	154	Doyle St at 59th St
3530	17	Embarcadero BART Station (Beale St at Market St)
3985	13	Commercial St at Montgomery St
4040	87	Folsom St at 13th St
4223	371	Lombard St at Columbus Ave
4233	139	Garfield Square (25th St at Harrison St)
4658	99	Folsom St at 15th St
4828	378	Empire St at 7th St
4855	257	Fifth St at Delaware St
...	...	...
175322	30	San Francisco Caltrain (Townsend St at 4th St)
175831	132	24th St at Chattanooga St
175857	53	Grove St at Divisadero
175860	53	Grove St at Divisadero
176130	60	8th St at Ringold St
176188	70	Central Ave at Fell St
176490	375	Grove St at Masonic Ave
176639	370	Jones St at Post St
176707	33	Golden Gate Ave at Hyde St
177144	207	Broadway at Coronado Ave
177219	377	Fell St at Stanyan St
177251	108	16th St Mission BART
177279	62	Victoria Manalo Draves Park
177337	375	Grove St at Masonic Ave
177484	223	16th St Mission BART Station 2
177662	112	Harrison St at 17th St

```
177908                15  San Francisco Ferry Building (Harry Bridges Pl...
177912                15  San Francisco Ferry Building (Harry Bridges Pl...
178099                99                            Folsom St at 15th St
178416                78                             Folsom St at 9th St
178649               263                    Channing Way at San Pablo Ave
178656                 5       Powell St BART Station (Market St at 5th St)
179375               370                            Jones St at Post St
179529                19                           Post St at Kearny St
179535                19                           Post St at Kearny St
179732               364                      China Basin St at 3rd St
180303               338                        13th St at Franklin St
182133               380                        Masonic Ave at Turk St
182411               106                         Sanchez St at 17th St
183326                31                     Raymond Kimbell Playground

        start_station_latitude  start_station_longitude end_station_id  \
0                    37.789625              -122.400811             13
2                    37.769305              -122.426826              3
3                    37.774836              -122.446546             70
91                   37.865847              -122.267443            244
199                  37.787168              -122.388098            368
297                  37.788975              -122.403452             19
511                  37.862464              -122.264791            201
524                  37.778742              -122.392741              3
779                  37.776619              -122.417385            375
790                  37.783899              -122.408445            126
813                  37.852477              -122.270213            248
926                  37.782405              -122.439446             71
939                  37.783899              -122.408445             70
945                  37.783899              -122.408445             70
1023                 37.775946              -122.437777            381
1826                 37.784872              -122.400876             22
2188                 37.796248              -122.279352            187
2298                 37.729279              -122.392896            362
2544                 37.775880              -122.393170             93
2588                 37.776619              -122.417385             11
2693                 37.784872              -122.400876              3
3401                 37.841924              -122.288045            213
3530                 37.792251              -122.397086              6
3985                 37.794231              -122.402923              3
4040                 37.769757              -122.415674            145
4223                 37.802746              -122.413579            377
4233                 37.751017              -122.411901            112
4658                 37.767037              -122.415443             88
4828                 37.347745              -121.890800            314
4855                 37.870407              -122.299676            239
...                        ...                      ...            ...
175322               37.776598              -122.395282             24
```

175831	37.751819	-122.426614	134
175857	37.775946	-122.437777	53
175860	37.775946	-122.437777	53
176130	37.774520	-122.409449	43
176188	37.773311	-122.444293	72
176490	37.774836	-122.446546	377
176639	37.787327	-122.413278	6
176707	37.781650	-122.415408	77
177144	37.835788	-122.251621	253
177219	37.771917	-122.453704	377
177251	37.764710	-122.419957	223
177279	37.777791	-122.406432	63
177337	37.774836	-122.446546	380
177484	37.764765	-122.420091	223
177662	37.763847	-122.413004	145
177908	37.795392	-122.394203	15
177912	37.795392	-122.394203	15
178099	37.767037	-122.415443	99
178416	37.773717	-122.411647	11
178649	37.862827	-122.290230	254
178656	37.783899	-122.408445	5
179375	37.787327	-122.413278	6
179529	37.788975	-122.403452	16
179535	37.788975	-122.403452	16
179732	37.772000	-122.389970	20
180303	37.803189	-122.270579	187
182133	37.779047	-122.447291	377
182411	37.763242	-122.430675	79
183326	37.783813	-122.434559	31

	end_station_name \
0	Commercial St at Montgomery St
2	Powell St BART Station (Market St at 4th St)
3	Central Ave at Fell St
91	Shattuck Ave at Hearst Ave
199	Myrtle St at Polk St
297	Post St at Kearny St
511	10th St at Fallon St
524	Powell St BART Station (Market St at 4th St)
779	Grove St at Masonic Ave
790	Esprit Park
813	Telegraph Ave at Ashby Ave
926	Broderick St at Oak St
939	Central Ave at Fell St
945	Central Ave at Fell St
1023	20th St at Dolores St
1826	Howard St at Beale St
2188	Jack London Square

```
2298                              Lane St at Revere Ave
2544                       4th St at Mission Bay Blvd S
2588                              Davis St at Jackson St
2693        Powell St BART Station (Market St at 4th St)
3401                              32nd St at Adeline St
3530                     The Embarcadero at Sansome St
3985        Powell St BART Station (Market St at 4th St)
4040                               29th St at Church St
4223                              Fell St at Stanyan St
4233                             Harrison St at 17th St
4658                               11th St at Bryant St
4828                    Santa Clara St at Almaden Blvd
4855                     Bancroft Way at Telegraph Ave
...                                                 ...
175322                            Spear St at Folsom St
175831                           Valencia St at 24th St
175857                           Grove St at Divisadero
175860                           Grove St at Divisadero
176130  San Francisco Public Library (Grove St at Hyde...
176188                             Page St at Scott St
176490                            Fell St at Stanyan St
176639                   The Embarcadero at Sansome St
176707                             11th St at Natoma St
177144                            Haste St at College Ave
177219                            Fell St at Stanyan St
177251                   16th St Mission BART Station 2
177279                             Bryant St at 6th St
177337                           Masonic Ave at Turk St
177484                   16th St Mission BART Station 2
177662                             29th St at Church St
177908  San Francisco Ferry Building (Harry Bridges Pl...
177912  San Francisco Ferry Building (Harry Bridges Pl...
178099                             Folsom St at 15th St
178416                           Davis St at Jackson St
178649                          Vine St at Shattuck Ave
178656      Powell St BART Station (Market St at 5th St)
179375                   The Embarcadero at Sansome St
179529                          Steuart St at Market St
179535                          Steuart St at Market St
179732    Mechanics Monument Plaza (Market St at Bush St)
180303                               Jack London Square
182133                            Fell St at Stanyan St
182411                             7th St at Brannan St
183326                      Raymond Kimbell Playground

        end_station_latitude  end_station_longitude bike_id   user_type  \
0                  37.794231            -122.402923    4902    Customer
2                  37.786375            -122.404904    5905    Customer
```

3	37.773311	-122.444293	6638	Subscriber
91	37.873676	-122.268487	5244	Subscriber
199	37.785434	-122.419622	5380	Subscriber
297	37.788975	-122.403452	5830	Subscriber
511	37.797673	-122.262997	6001	Subscriber
524	37.786375	-122.404904	733	Subscriber
779	37.774836	-122.446546	5465	Subscriber
790	37.761634	-122.390648	6438	Subscriber
813	37.855956	-122.259795	6411	Subscriber
926	37.773063	-122.439078	5515	Subscriber
939	37.773311	-122.444293	6501	Customer
945	37.773311	-122.444293	2464	Customer
1023	37.758238	-122.426094	6138	Subscriber
1826	37.789756	-122.394643	6610	Customer
2188	37.796248	-122.279352	932	Customer
2298	37.731727	-122.390056	6357	Customer
2544	37.770407	-122.391198	2270	Customer
2588	37.797280	-122.398436	5522	Customer
2693	37.786375	-122.404904	4635	Subscriber
3401	37.823847	-122.281193	4683	Subscriber
3530	37.804770	-122.403234	6591	Customer
3985	37.786375	-122.404904	5857	Subscriber
4040	37.743684	-122.426806	6576	Customer
4223	37.771917	-122.453704	5860	Subscriber
4233	37.763847	-122.413004	6558	Subscriber
4658	37.770030	-122.411726	5769	Customer
4828	37.333988	-121.894902	2412	Customer
4855	37.868813	-122.258764	4543	Customer
...	...	...	...	...
175322	37.789677	-122.390428	5149	Customer
175831	37.752428	-122.420628	5405	Subscriber
175857	37.775946	-122.437777	5472	Subscriber
175860	37.775946	-122.437777	5004	Subscriber
176130	37.778768	-122.415929	335	Subscriber
176188	37.772406	-122.435650	2191	Customer
176490	37.771917	-122.453704	1161	Customer
176639	37.804770	-122.403234	4602	Customer
176707	37.773507	-122.416040	5271	Subscriber
177144	37.866418	-122.253799	847	Subscriber
177219	37.771917	-122.453704	4370	Subscriber
177251	37.764765	-122.420091	5351	Subscriber
177279	37.775910	-122.402575	1401	Customer
177337	37.779047	-122.447291	4377	Subscriber
177484	37.764765	-122.420091	4729	Subscriber
177662	37.743684	-122.426806	5382	Subscriber
177908	37.795392	-122.394203	5284	Customer
177912	37.795392	-122.394203	5132	Customer
178099	37.767037	-122.415443	4597	Customer

```
178416            37.797280          -122.398436    5110  Subscriber
178649            37.880222          -122.269592    4642    Customer
178656            37.783899          -122.408445    5105  Subscriber
179375            37.804770          -122.403234    5020    Customer
179529            37.794130          -122.394430    5059    Customer
179535            37.794130          -122.394430    4814    Customer
179732            37.791300          -122.399051    5561  Subscriber
180303            37.796248          -122.279352    4528  Subscriber
182133            37.771917          -122.453704    4956  Subscriber
182411            37.773492          -122.403673    4944  Subscriber
183326            37.783813          -122.434559    5366  Subscriber

        member_birth_year member_gender bike_share_for_all_trip  duration_min  \
0                    1984          Male                      No        869.75
2                    1972          Male                      No       1030.90
3                    1989         Other                      No        608.17
91                   1997        Female                      No         93.68
199                  1980          Male                      No        252.05
297                  1987          Male                      No        217.68
511                  1975          Male                     Yes        123.68
524                  1994          Male                      No        107.45
779                  1991        Female                      No        603.17
790                  1994        Female                      No         98.78
813                  1968        Female                      No        166.57
926                  1989          Male                      No         84.75
939                  1974          Male                      No        280.07
945                  1974          Male                      No        279.47
1023                 1983          Male                     Yes         83.82
1826                 1997          Male                      No        152.90
2188                 1990          Male                      No        157.07
2298                 1992        Female                      No        335.93
2544                 1984        Female                      No        527.22
2588                 1988          Male                      No        511.10
2693                 1980        Female                     Yes        364.72
3401                 1989        Female                      No       1040.87
3530                 1995        Female                      No        230.73
3985                 1966          Male                      No        145.50
4040                 1985          Male                      No        137.02
4223                 1994          Male                      No        157.87
4233                 1986          Male                     Yes        449.80
4658                 1978          Male                      No        153.08
4828                 1982        Female                      No        110.28
4855                 1988          Male                      No        103.70
...                   ...           ...                     ...           ...
175322               1994        Female                      No        229.12
175831               1990          Male                     Yes        179.08
175857               1986        Female                      No        132.63
175860               1991          Male                      No        134.32
```

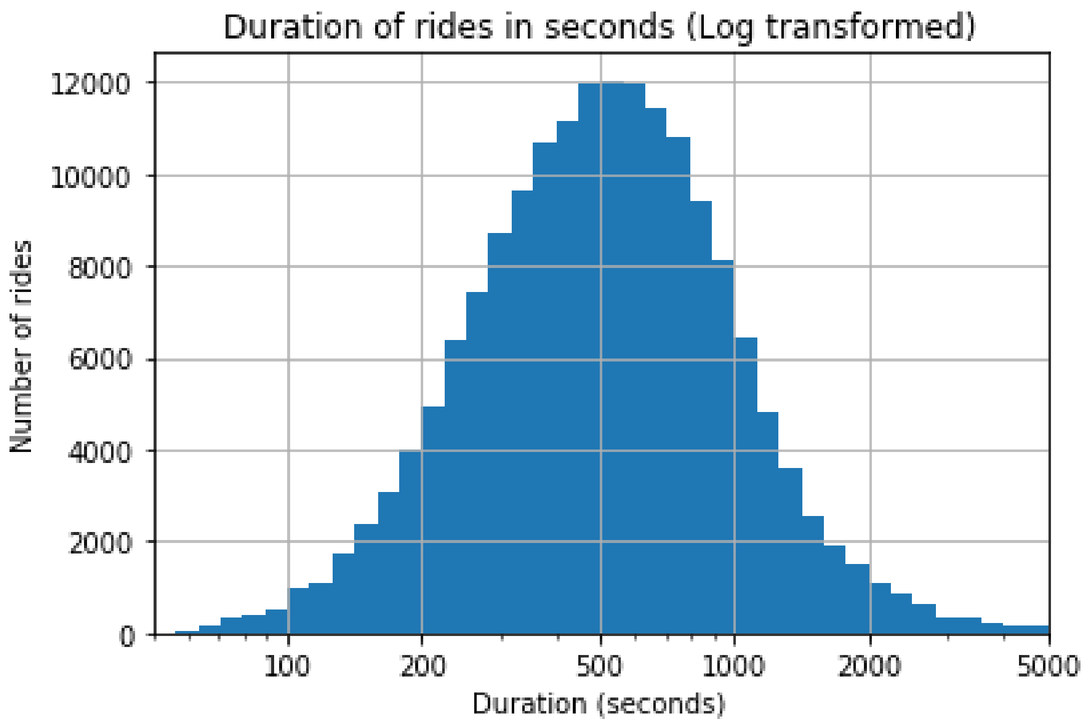
176130	1990	Female	No	996.88
176188	1991	Female	No	85.22
176490	1993	Female	No	86.15
176639	2001	Male	No	184.38
176707	1971	Female	No	99.45
177144	1997	Male	Yes	858.13
177219	1971	Other	No	104.50
177251	1995	Female	No	186.13
177279	1988	Male	No	512.78
177337	1971	Other	No	141.13
177484	1995	Female	No	228.22
177662	1978	Male	No	202.57
177908	1987	Female	No	259.42
177912	1984	Male	No	259.02
178099	1965	Male	No	159.80
178416	1989	Male	No	520.05
178649	1977	Other	No	238.68
178656	1996	Other	Yes	214.90
179375	2001	Male	No	108.47
179529	1954	Male	No	116.23
179535	1954	Male	No	116.05
179732	1989	Male	No	176.13
180303	1964	Other	Yes	110.48
182133	1971	Other	No	101.43
182411	1982	Male	Yes	226.82
183326	1972	Male	No	95.22

	member_age	dist_bet_stations
0	35	32.26
2	47	176.67
3	30	17.30
91	22	50.31
199	39	200.38
297	32	0.00
511	44	412.94
524	25	91.26
779	28	185.59
790	25	181.41
813	51	69.22
926	30	59.56
939	45	237.45
945	45	237.45
1023	36	135.06
1826	22	50.33
2188	29	0.00
2298	27	23.88
2544	35	37.05
2588	31	178.29

```
2693            39             27.31
3401            30            123.02
3530            24             88.77
3985            53             51.60
4040            34            180.57
4223            25            321.59
4233            33             82.04
4658            41             30.36
4828            37             91.00
4855            31            257.07
...            ...               ...
175322          25             88.84
175831          29             38.28
175857          33              0.00
175860          28              0.00
176130          29             49.26
176188          28             55.22
176490          26             49.14
176639          18            128.09
176707          48             52.03
177144          22            195.62
177219          48              0.00
177251          24              0.92
177279          31             27.27
177337          48             27.24
177484          24              0.00
177662          41            155.60
177908          32              0.00
177912          35              0.00
178099          54              0.00
178416          30            171.95
178649          42            170.47
178656          23              0.00
179375          18            128.09
179529          65             65.99
179535          65             65.99
179732          30            135.81
180303          55             71.05
182133          48             61.02
182411          37            183.62
183326          47              0.00

[884 rows x 19 columns]
```

In [60]:
```
# Log transformation of the duration in seconds variable
log_binsize = 0.05
bins= 10 ** np.arange(1.7, np.log(df_clean['duration_sec'].max())+log_binsize, log_bins
```

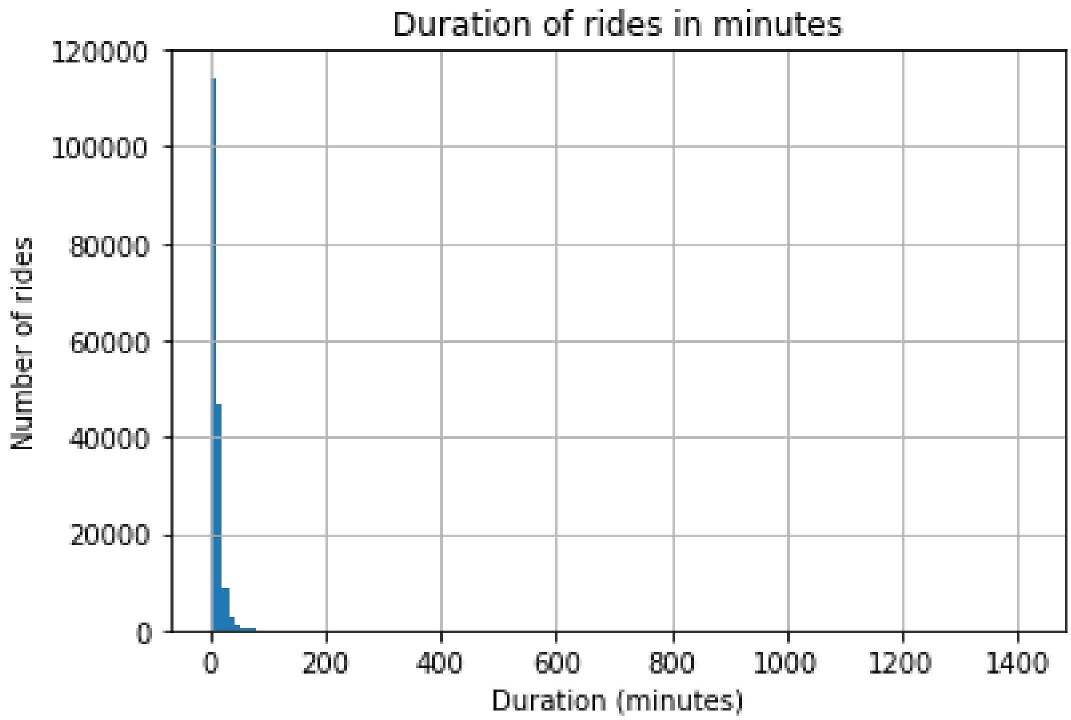
```
df_clean.duration_sec.hist(bins=bins)
plt.xscale('log')
x_ticks = [100,200,500,1000,2000,5000]
plt.xticks(x_ticks, x_ticks)
plt.xlim(50,5000)
plt.xlabel('Duration (seconds)')
plt.ylabel('Number of rides')
plt.title('Duration of rides in seconds (Log transformed)');
```



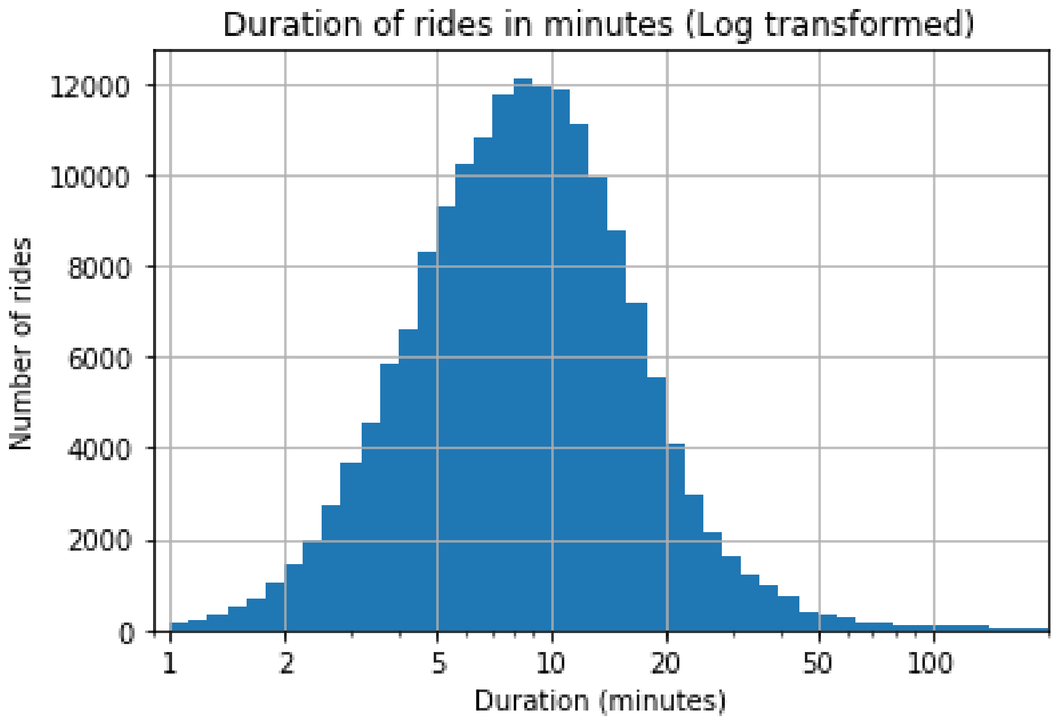
The initial plot of the duration variable was right skewed with large gaps between bins and some value(s) over 80,000 secs and thus revealed the presence of outlier(s). This meant it had to be log transformed. From the log transformed distribution, a unimodal distribution is observed and it can be seen that most rides were between 240 to 1200 seconds.

Next up, the duration in minutes will be examined.

```
In [61]: # Examining the distribution of the duration in minutes
         binsize = 10
         bins = np.arange(1, df_clean['duration_min'].max()+binsize ,binsize)
         df_clean.duration_min.hist(bins=bins)
         plt.xlabel('Duration (minutes)')
         plt.ylabel('Number of rides')
         plt.title('Duration of rides in minutes');
```



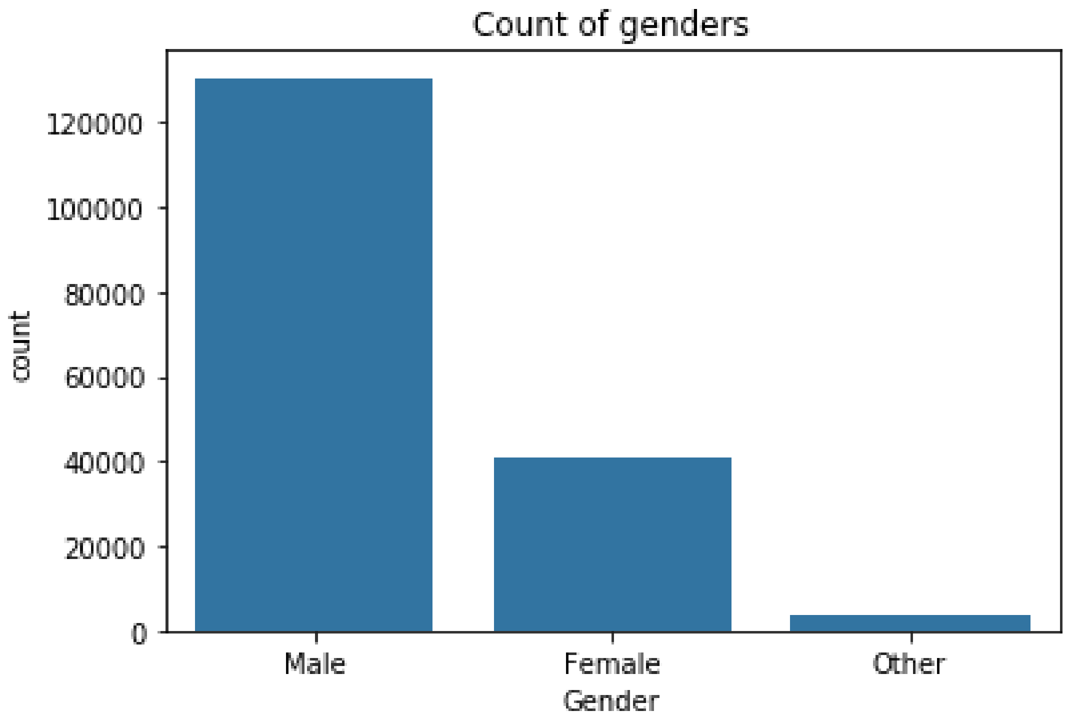

```
In [62]: # Log transformation of the duration in minutes variable
         log_binsize = 0.05
         bins = 10 ** np.arange(0, np.log(df_clean['duration_min'].max())+log_binsize, log_binsi
         df_clean.duration_min.hist(bins=bins)
         plt.xscale('log')
         x_ticks = [1,2,5,10,20,50,100]
         plt.xticks(x_ticks, x_ticks)
         plt.xlabel('Duration (minutes)')
         plt.ylabel('Number of rides')
         plt.title('Duration of rides in minutes (Log transformed)')
         plt.xlim(0.9,200);
```



Similar to what was observed in the plots of the duration in seconds. The graph had to be transformed into a logarithmic scale since the standard plot was hightly right skewed and outlier(s) were present. A unimodal distribution is observed with its peaks between 4 to 20 minutes. This outcome was expected because the minutes variable was created from the duration in seconds variable.

Next up, their gender will be explored.

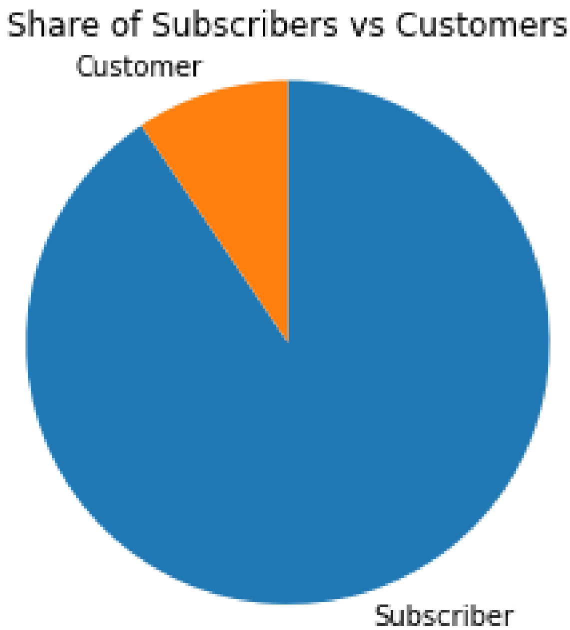
In [63]:
```
# Exploring the gender variable
color = sb.color_palette()[0]
gender_order = ['Male', 'Female', 'Other']
sb.countplot(data=df_clean, x='member_gender', order=gender_order, color=color)
plt.xlabel('Gender')
plt.title('Count of genders');
```



The number of men are at least 3 times more than the number of women that use the bikes with a count of over 120,000 and 40,000 respectively.

Further, I will be digging into the user types of users.

```
In [64]: # viewing the proportion of the type of users
         users = df_clean.user_type.value_counts()
         plt.pie(x=users, labels=users.index, startangle=90, counterclock = False)
         plt.axis('square')
         plt.title('Share of Subscribers vs Customers');
```

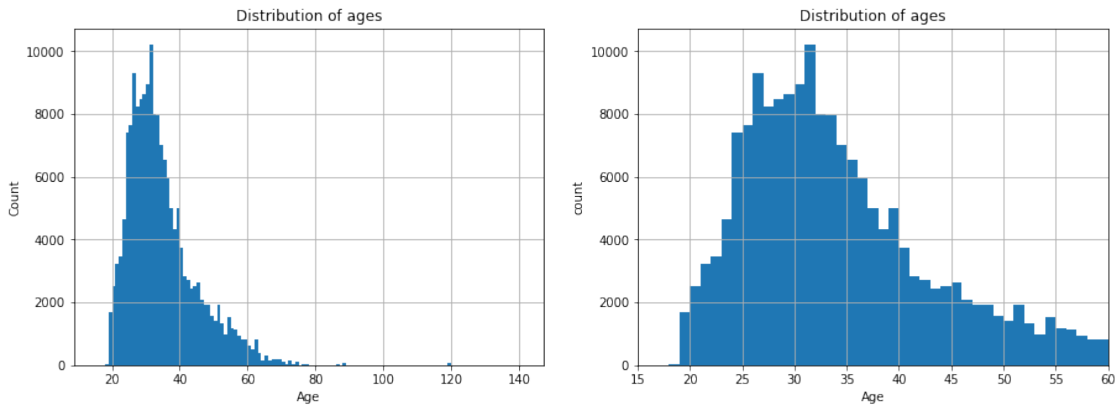


The pie chart reveals that about a whopping 90% of the users are subscribers while only about 10% are single use customers. I believe this is because subscribers will get a cheaper rate. Hence, it is more economical to be a subscriber if you intend to use the service frequently.

The distribution of ages will be plotted next.

In [65]:
```
# Plotting the distribution of ages
plt.figure(figsize=[15,5])
plt.subplot(1,2,1)
binsize = 1 # Every bin represents an age
bins = np.arange(15, df_clean['member_age'].max()+binsize ,binsize)
df_clean['member_age'].hist(bins=bins)
plt.xlabel('Age')
plt.ylabel('Count')
plt.title('Distribution of ages')

plt.subplot(1,2,2) # Zooming in to the plot for more clarity
df_clean['member_age'].hist(bins=bins)
plt.xlim(15,60)
plt.xlabel('Age')
plt.ylabel('count')
plt.title('Distribution of ages')
```

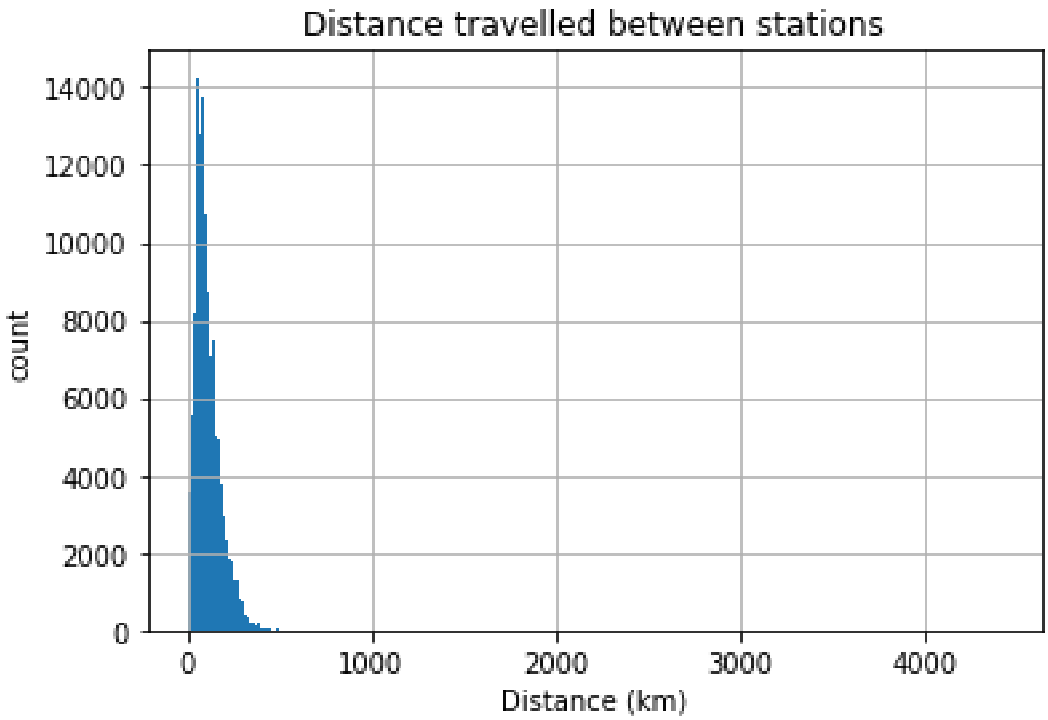
Out[65]:
```
Text(0.5,1,'Distribution of ages')
```



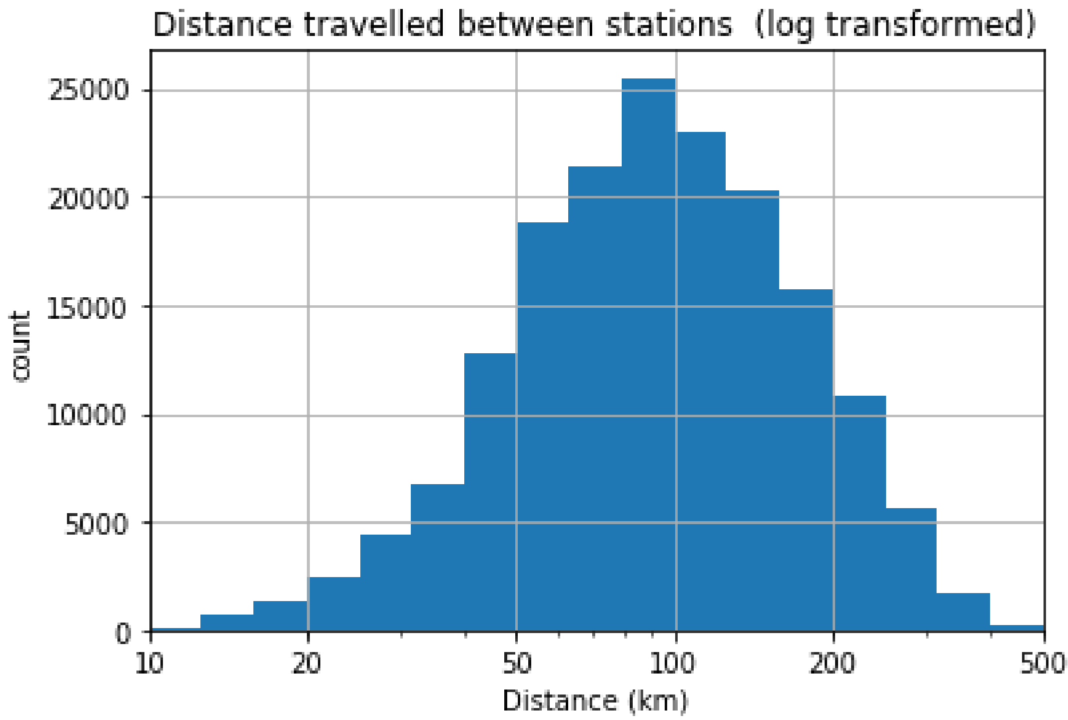
The first subplot (left) above looks right skewed. This was expected because the minimum age to access the bike sharing service was 18 but there was no maximum age. There were also outliers with the highest at over 140. This meant we had to zoom into the plot to understand and interpret it better. Thus, it can be seen from the right plot that most users were aged between 23 and 40.

An examination of the distribution of the distance travelled between stations follows.

```
In [66]: # Examining the distance travelled between stations
         binsize = 10
         bins = np.arange(0, df_clean['dist_bet_stations'].max()+binsize, binsize)
         df_clean.dist_bet_stations.hist(bins=bins)
         plt.xlabel('Distance (km)')
         plt.ylabel('count')
         plt.title('Distance travelled between stations')

Out[66]: Text(0.5,1,'Distance travelled between stations')
```




```
In [67]: # Log transformation of the distance travelled
         log_binsize = 0.1
         bins = 10 ** np.arange(0, np.log(df_clean['dist_bet_stations'].max())+log_binsize, log_
         df_clean.dist_bet_stations.hist(bins=bins)
         plt.xscale('log')
         x_ticks = [10,20,50,100,200,500,1000,2000]
         plt.xticks(x_ticks, x_ticks)
         plt.xlabel('Distance (km)')
         plt.ylabel('count')
         plt.title('Distance travelled between stations  (log transformed)')
         plt.xlim(10,500)

Out[67]: (10, 500)
```



The log transformation shows a unimodal distribution with the average distance travelled from one station to the other is between 40km and 200km.

```
In [68]: df_clean.head()
```

```
Out[68]:    duration_sec               start_time                  end_time  \
         0         52185 2019-02-28 17:32:10.145 2019-03-01 08:01:55.975
         2         61854 2019-02-28 12:13:13.218 2019-03-01 05:24:08.146
         3         36490 2019-02-28 17:54:26.010 2019-03-01 04:02:36.842
         4          1585 2019-02-28 23:54:18.549 2019-03-01 00:20:44.074
         5          1793 2019-02-28 23:49:58.632 2019-03-01 00:19:51.760

           start_station_id                                start_station_name  \
         0               21  Montgomery St BART Station (Market St at 2nd St)
         2               86                             Market St at Dolores St
         3              375                             Grove St at Masonic Ave
         4                7                                 Frank H Ogawa Plaza
         5               93                        4th St at Mission Bay Blvd S

            start_station_latitude  start_station_longitude end_station_id  \
         0               37.789625              -122.400811             13
         2               37.769305              -122.426826              3
         3               37.774836              -122.446546             70
         4               37.804562              -122.271738            222
         5               37.770407              -122.391198            323
```

```
                        end_station_name  end_station_latitude  \
0            Commercial St at Montgomery St             37.794231
2  Powell St BART Station (Market St at 4th St)         37.786375
3                    Central Ave at Fell St             37.773311
4                     10th Ave at E 15th St             37.792714
5                        Broadway at Kearny             37.798014

   end_station_longitude bike_id   user_type  member_birth_year member_gender  \
0            -122.402923    4902    Customer               1984          Male
2            -122.404904    5905    Customer               1972          Male
3            -122.444293    6638  Subscriber               1989         Other
4            -122.248780    4898  Subscriber               1974          Male
5            -122.405950    5200  Subscriber               1959          Male

  bike_share_for_all_trip  duration_min  member_age  dist_bet_stations
0                      No        869.75          35              32.26
2                      No       1030.90          47             176.67
3                      No        608.17          30              17.30
4                     Yes         26.42          45             163.95
5                      No         29.88          60             199.25
```

1.4.1 Discuss the distribution(s) of your variable(s) of interest. Were there any unusual points? Did you need to perform any transformations?

The primary variable of interest is the duration of rides (minutes). This was engineered from the duration of rides (seconds) which was provided in the original dataset. As expected, both variables had a similar distributions. They were both right skewed and had outliers. Hence, they had to be plotted on logarithmic scales. From the log transformed distributions, unimodal distributions were observed and it was clear that most rides were between 240 to 1200 seconds and between 5 to 16 minutes. However, only the minutes variable will be used in our analysis moving forward because it is a better descriptor of time. For instance, it is better to say "10 minutes" than "600 seconds".

1.4.2 Of the features you investigated, were there any unusual distributions? Did you perform any operations on the data to tidy, adjust, or change the form of the data? If so, why did you do this?

A number of variables were right skewed with strong outliers. However, an examination of the outliers revealed that they were legitimate observations. Thus, they remained as part of the dataset.

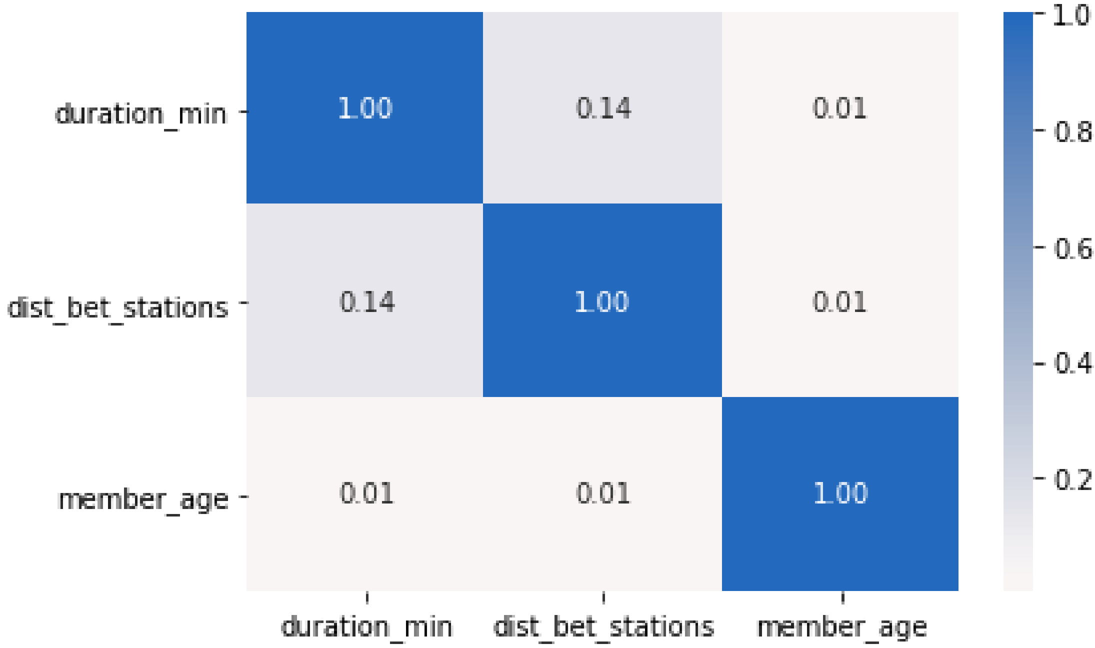
1.5 Bivariate Exploration

This section seeks to investigate relationships between pairs of variables that were introduced in the previous section (univariate exploration).

I will start by viewing the correlation of the numeric variables in a heatmap.

```
In [69]: # correlation plot of numeric variables
         numeric_vars = ['duration_min', 'dist_bet_stations', 'member_age']
         sb.heatmap(df_clean[numeric_vars].corr(), annot = True, fmt = '.2f', cmap = 'vlag_r', c
```

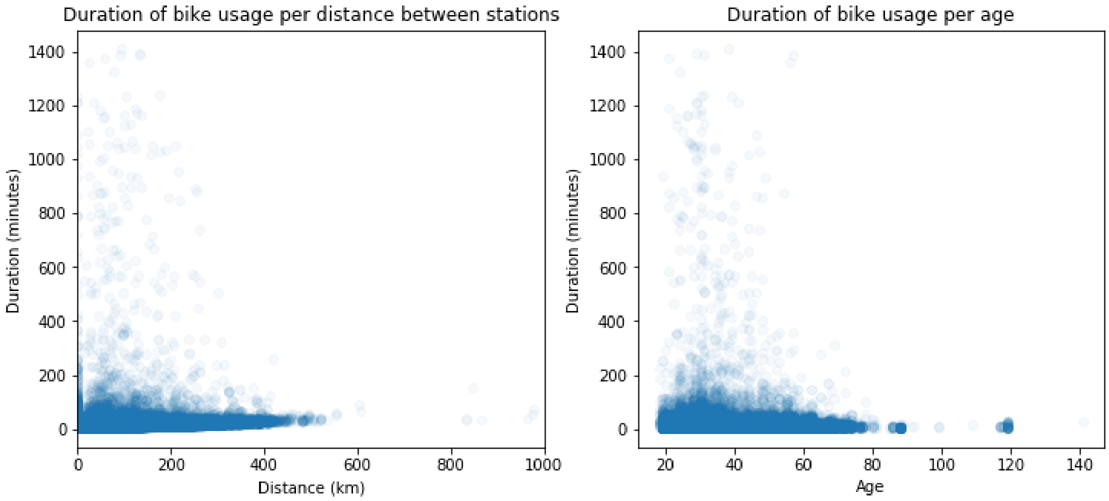
```
Out[69]: <matplotlib.axes._subplots.AxesSubplot at 0x7f3657547710>
```



There seems to be little to no correlation between the pairs of numeric variables plotted in the heatmap.

Let's take a look at the scatterplots between the pairs.

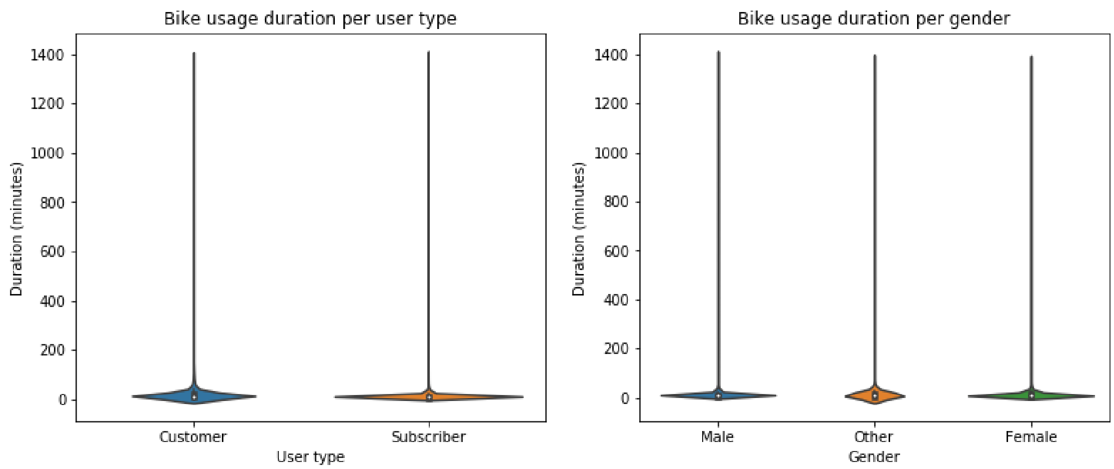
```
In [70]: # Scatter plots of age and distance between stations in relation to the main feature of
         plt.figure(figsize=[12,5])

         plt.subplot(1,2,1) # Left plot
         plt.scatter(data=df_clean, x='dist_bet_stations', y='duration_min', alpha=0.04)
         plt.xlabel('Distance (km)')
         plt.ylabel('Duration (minutes)')
         plt.title('Duration of bike usage per distance between stations')
         plt.xlim(0,1000);

         plt.subplot(1,2,2) # Right plot
         plt.scatter(data=df_clean, x='member_age', y='duration_min', alpha=0.04)
         plt.xlabel('Age')
         plt.ylabel('Duration (minutes)')
         plt.title('Duration of bike usage per age');
```



The absence of a relationship between the pairs is confirmed here. Nonetheless, it is noteworthy from the left plot that most people used the bikes for less than 100 minutes and travelled between 0 to 400km. Many rented a bike and returned it to the same station, this is the reason for the concentration of zero distances at the bottom left of the plot. The right plot also reveals that most users were aged below 80 and also used the bikes for about 100 minutes or less.

An investigation into the relationship between the categorical variables and the primary variable of interest follows.

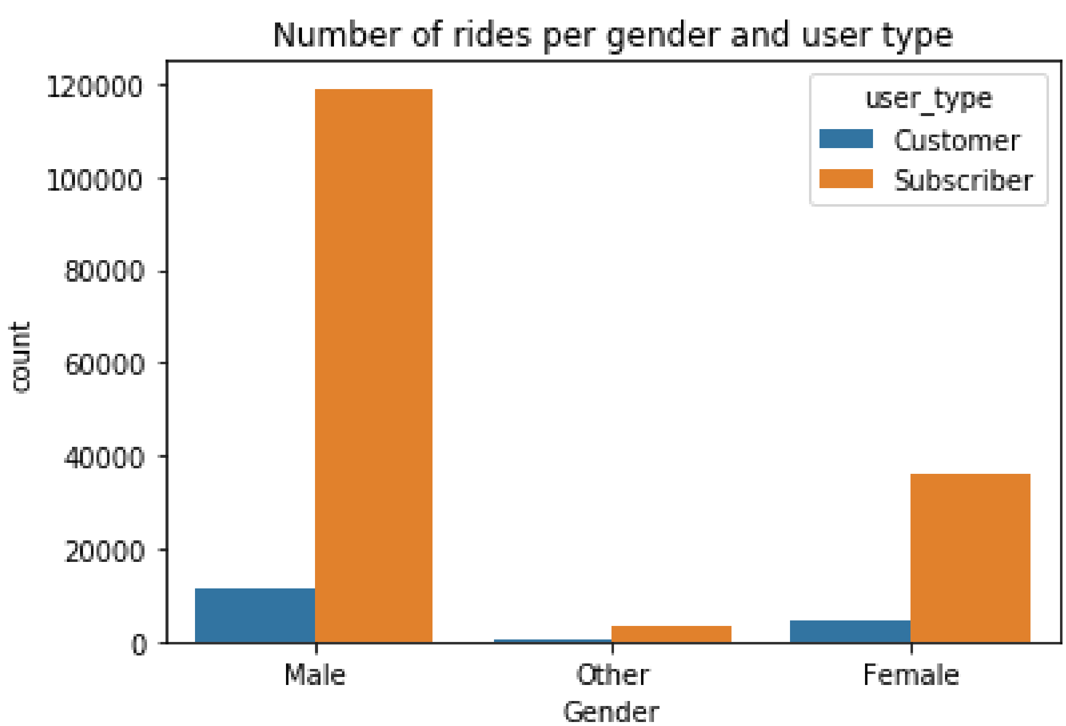
In [71]:
```
# Violin plots of the user types and gender in relation to the duration in minutes
plt.figure(figsize=[13,5])

plt.subplot(1,2,1) # Left plot
sb.violinplot(data=df_clean, x='user_type', y='duration_min')
plt.xlabel('User type')
plt.ylabel('Duration (minutes)')
plt.title('Bike usage duration per user type');

plt.subplot(1,2,2) # Right plot
sb.violinplot(data=df_clean, x='member_gender', y='duration_min')
plt.xlabel('Gender')
plt.ylabel('Duration (minutes)')
plt.title('Bike usage duration per gender');
```



An upside-down "T" is observed for all variables in both plots. This indicates a deep concentration at the base below 100 minutes regardless of their subscription status or gender.

Finally, let's look at the relationship be the two qualitative variables: user type and gender.

In [72]:
```
# Clustered bar chart of the user types and gender.
sb.countplot(data=df_clean, x='member_gender', hue='user_type')
plt.xlabel('Gender')
plt.title('Number of rides per gender and user type');
```



The plot above shows that more subscribers are males. This is not a surprise because men are generally more physically active.

1.5.1 Talk about some of the relationships you observed in this part of the investigation. How did the feature(s) of interest vary with other features in the dataset?

No real relationship was observerd between the key feature of interest and other features in the dataset. However, one thing stood out amongst the plots relating to the main feature of interest regardless of age, gender, subscription status or distance and that is the average rider uses the bike for less than 100 minutes before returning it to a station.

1.5.2 Did you observe any interesting relationships between the other features (not the main feature(s) of interest)?

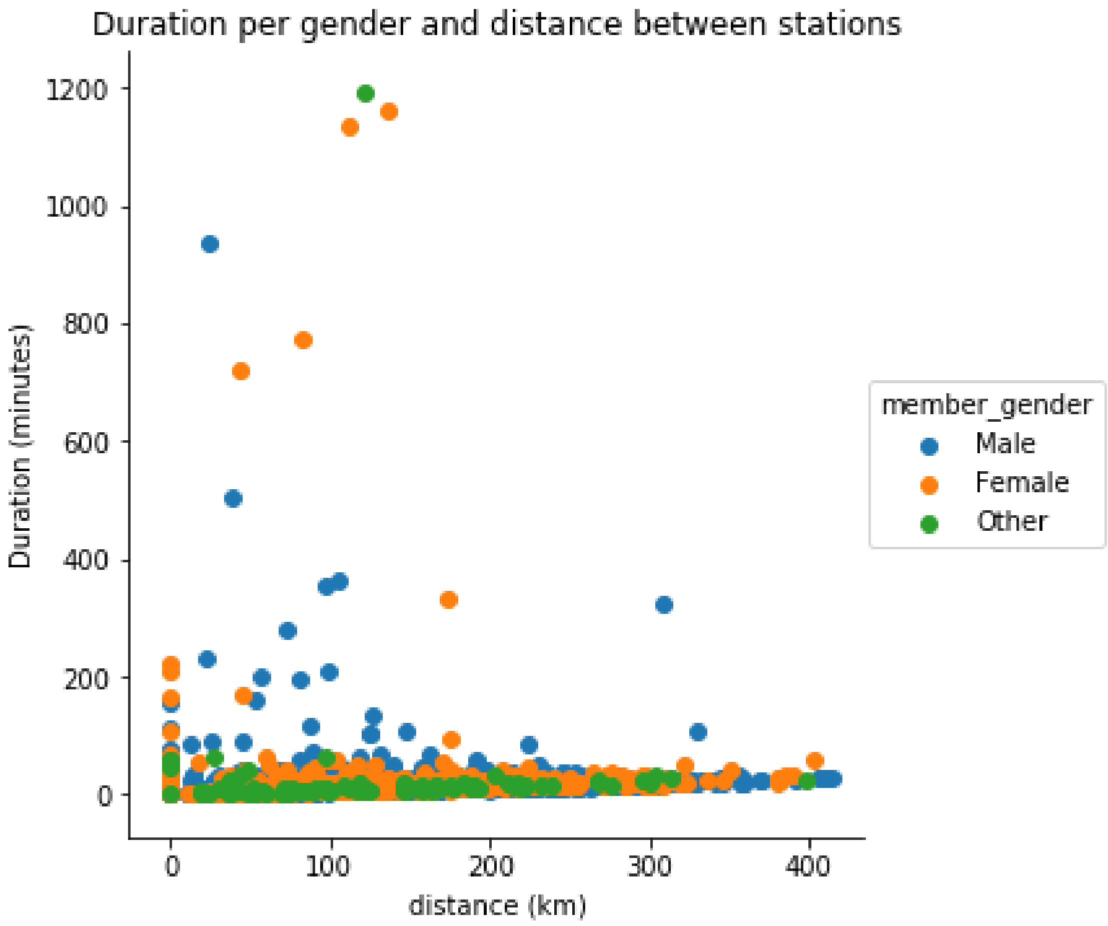
Most of the users were subscribers and most of the subscribers are males. However, this was not much of a surprise because males are generally sportier than females.

1.6 Multivariate Exploration

This section seeks to investigate relationships between three variables or more that have been introduced in the previous sections.

```
In [73]: samples = np.random.choice(df_clean.shape[0], 5000, replace = False)
         df_sample = df_clean.loc[samples]

         facet = sb.FacetGrid(data = df_sample, hue = 'member_gender', size = 5)
         facet.map(plt.scatter, 'dist_bet_stations', 'duration_min')
         facet.add_legend()
         plt.xlabel('distance (km)')
         plt.ylabel('Duration (minutes)')
         plt.title('Duration per gender and distance between stations');
```

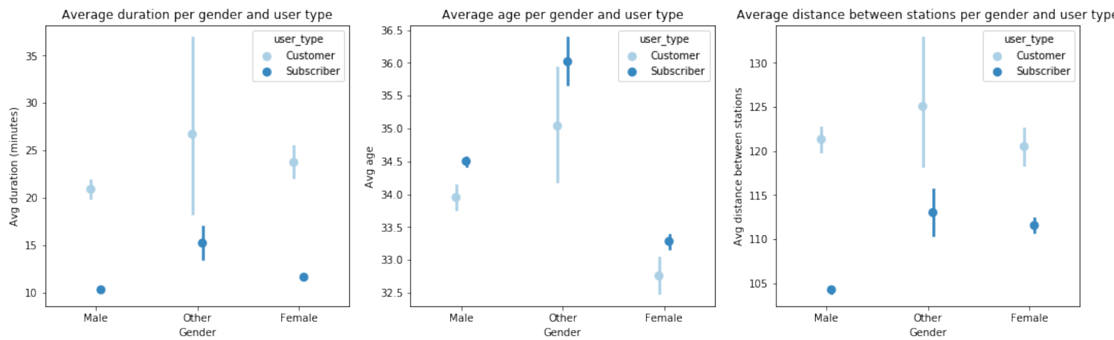
```
/opt/conda/lib/python3.6/site-packages/ipykernel_launcher.py:2: FutureWarning:
Passing list-likes to .loc or [] with any missing label will raise
KeyError in the future, you can use .reindex() as an alternative.

See the documentation here:
https://pandas.pydata.org/pandas-docs/stable/indexing.html#deprecate-loc-reindex-listlike
```



We see a strong concentration at the base of the plot just below 100 minutes and before 400km. It also looks like males have a much higher chance of using the bikes for more than 100 munites.

In [74]:
```
# point plots of the user types and gender in relation to the duration, age and distanc
plt.figure(figsize=[18,5])

plt.subplot(1,3,1) # Left plot
sb.pointplot(data=df_clean, x='member_gender', y='duration_min', hue='user_type', palet
plt.xlabel('Gender')
plt.ylabel('Avg duration (minutes)')
plt.title('Average duration per gender and user type');

plt.subplot(1,3,2) # Centre plot
sb.pointplot(data=df_clean, x='member_gender', y='member_age', hue='user_type', palette
plt.xlabel('Gender')
plt.ylabel('Avg age')
plt.title('Average age per gender and user type');

plt.subplot(1,3,3) # Right plot
```

```
sb.pointplot(data=df_clean, x='member_gender', y='dist_bet_stations', hue='user_type',
plt.xlabel('Gender')
plt.ylabel('Avg distance between stations')
plt.title('Average distance between stations per gender and user type');
```



From the left plot, we see that subscribers generally use the bike sharing service for an average of about 15 minutes which is 10 minutes less than the average usage of non-subscribers at 25 minutes. We also see from the centre plot that subscribers are generally older than non-subscribers. Finally, the last plot reveals that subscribers generally also rode the bikes over shorter distances between stations than non-subscribers.

1.6.1 Talk about some of the relationships you observed in this part of the investigation. Were there features that strengthened each other in terms of looking at your feature(s) of interest?

Although the average time per ride was less than 100 minutes, males were more likely to use the bikes for longer than females.

1.6.2 Were there any interesting or surprising interactions between features?

It's interesting to not that subscribers were generally older, travelled for fewer distances between stations and also used the service for about 10 minutes less per ride than non-subscribers.

1.7 Conclusions

The number of men that used the bike sharing service are at least 3 times more than the number of women with about 130,000 and 40,000 respectively. The average user of the service was between 23 and 40 years old. Although most rides were were less than 100 minutes, its peak was between 4 to 20 minutes and the average distance travelled from one station to another was between 40km and 200km. About 90% of the bike sharing service users are subscribers while only about 10% are single use customers. Subscribers are largely males and are generally older than non-subscribers, they also rode the bikes over shorter distances between stations than non-subscribers and generally used the bikes for an average of about 15 minutes which is 10 minutes less than the average time of non-subscribers at 25 minutes.

```
In [ ]:
```