Monthly Membership Giving Trends

This project was established to examine monthly membership giving for the past 10 years. The purpose of examining the trend is to better predict cashflow, identify any potential growth, and help set goals as compared to a standard benchmark.

Given that membership patterns year-over-year can be very cyclical in nature, a pure month-over-month comparisons won't necessarily tell the complete story of potential growth. With this in mind, we are seeing growth in the November/December and July/August months, but we are seeing a flatter pattern on number of member transactions for winter and spring months.

Results show that the highest number of transactions per month for the past 2 years occurs during the span of the Museum's summer installation, making the case that offering something special each summer is imperative to our growth.

In 2014, we saw a spike in memberships for the month of April. It appears that the membership team at that time conducted a membership acquisition earlier than future years. The campaign centered on exhibition Designing for Disaster, a project focused on architecture & design and natural disasters.

In [1]:
```
import pandas as pd

data = pd.read_csv("H:/membs_months.csv", encoding = "Latin1")
```

In [2]:
```
#map numbers for months in order to properly sort months by fiscal year (Octob
er - September)
data["num_month"] = data["month"].map({"October": 1, "November": 2, "December"
: 3,
                                       "January": 4, "February": 5, "March": 6
,
                                      "April": 7, "May": 8, "June": 9, "July":
 10,
                                      "August": 11, "September": 12
                                      })
```

In [3]:
```
data.head(24)
```

Out[3]:

	fiscal_year	month	num_transactions	num_month
0	2007	October	220	1
1	2007	November	241	2
2	2007	December	334	3
3	2007	January	425	4
4	2007	February	265	5
5	2007	March	296	6
6	2007	April	245	7
7	2007	May	277	8
8	2007	June	225	9
9	2007	July	182	10
10	2007	August	178	11
11	2007	September	243	12
12	2008	October	184	1
13	2008	November	174	2
14	2008	December	497	3
15	2008	January	330	4
16	2008	February	207	5
17	2008	March	218	6
18	2008	April	230	7
19	2008	May	207	8
20	2008	June	185	9
21	2008	July	151	10
22	2008	August	165	11
23	2008	September	215	12

In [4]:
```
#plot pivot table to show number of transactions for each month over 10 years.

import numpy as np

data_pivot = data.pivot(index="num_month", columns="fiscal_year", values="num_
transactions")
data_pivot
```

Out[4]:

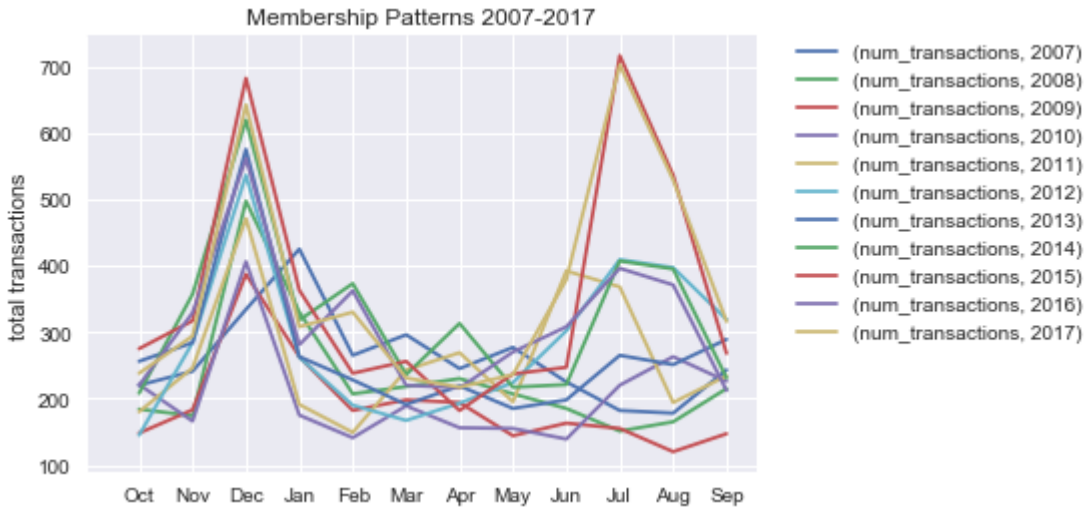
fiscal_year	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017
num_month											
1	220	184	148	221	180	145	256	208	275	220	238
2	241	174	183	166	246	284	284	357	317	329	294
3	334	497	387	406	471	536	575	619	682	561	642
4	425	330	263	175	191	262	263	318	363	281	308
5	265	207	182	141	149	190	228	373	238	362	330
6	296	218	198	189	243	167	191	236	256	220	231
7	245	230	194	156	269	193	219	313	182	216	217
8	277	207	144	155	195	224	185	217	237	270	236
9	225	185	163	139	392	302	198	221	247	308	379
10	182	151	155	220	368	409	265	407	716	396	702
11	178	165	120	263	194	397	251	395	536	371	529
12	243	215	147	226	234	319	289	231	268	212	316

In [5]:
```
import seaborn as sns
import matplotlib.pyplot as plt
```

In [6]:
```
%matplotlib inline
sns.set()

data.pivot(index="num_month", columns="fiscal_year").plot()
plt.ylabel("total transactions")
plt.xlabel("")
plt.xticks(np.arange(13), ("", "Oct", "Nov", "Dec", "Jan", "Feb", "Mar", "Apr"
, "May", "Jun", "Jul", "Aug", "Sep"))
plt.title("Membership Patterns 2007-2017")
plt.legend(bbox_to_anchor=(1.05, 1), loc=2, borderaxespad=0.)
```

Out[6]: <matplotlib.legend.Legend at 0xa437358>

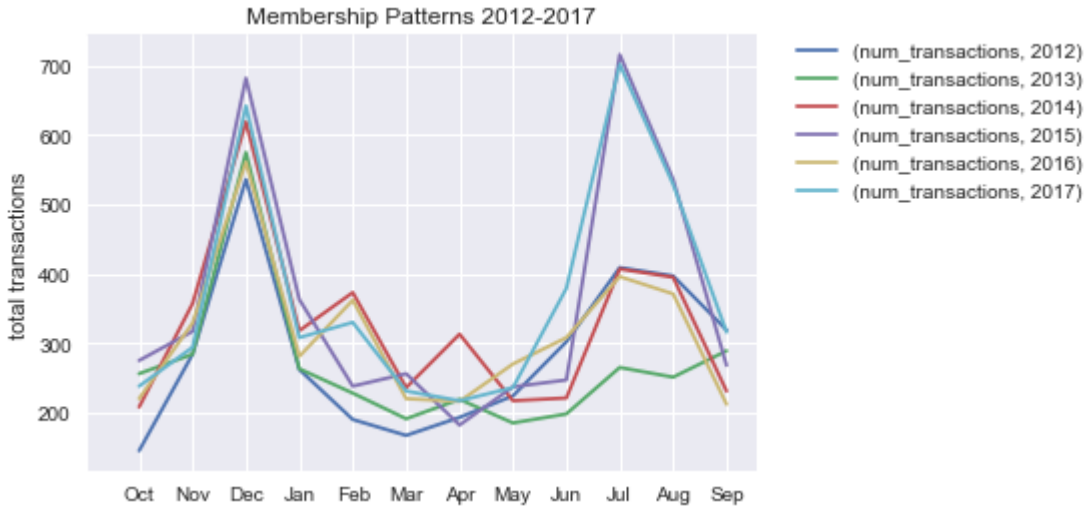


In [7]:
```
#graph past 5 years of data
data_5 = data[data["fiscal_year"] >= 2012]

data_pivot2 = data_5.pivot(index="num_month", columns="fiscal_year", values="n
um_transactions")
data_pivot2

data_5.pivot(index="num_month", columns="fiscal_year").plot()
plt.ylabel("total transactions")
plt.xlabel("")
plt.xticks(np.arange(13), ("", "Oct", "Nov", "Dec", "Jan", "Feb", "Mar", "Apr"
, "May", "Jun", "Jul", "Aug", "Sep"))
plt.title("Membership Patterns 2012-2017")
plt.legend(bbox_to_anchor=(1.05, 1), loc=2, borderaxespad=0.)
```

Out[7]: <matplotlib.legend.Legend at 0xb530898>

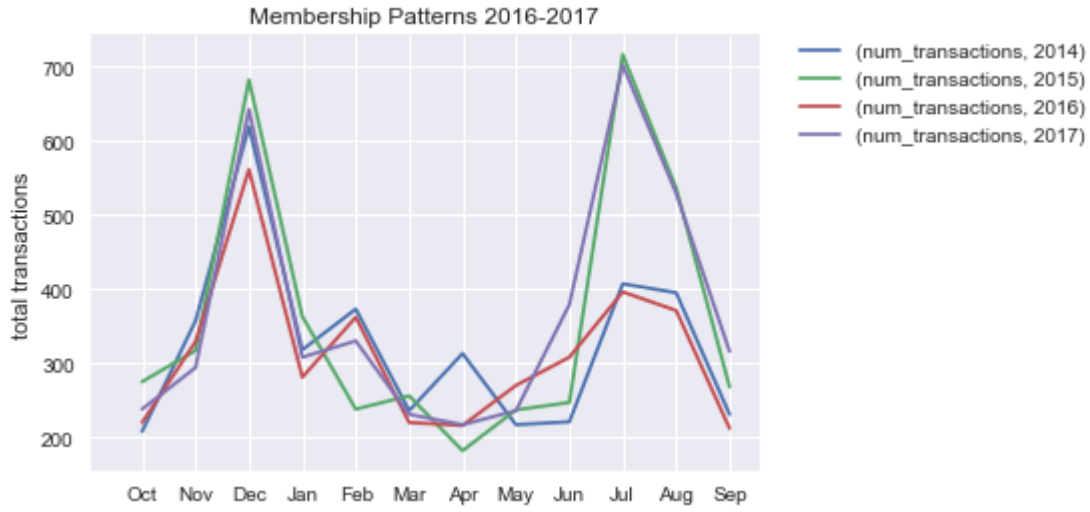


In [11]:
```
#Graph 2014-2017 data to further help compare 2017 results
data_1417 = data[data["fiscal_year"] >= 2014]

data_pivot5 = data_1417.pivot(index="num_month", columns="fiscal_year", values
="num_transactions")

data_1417.pivot(index="num_month", columns="fiscal_year").plot()
plt.ylabel("total transactions")
plt.xlabel("")
plt.xticks(np.arange(13), ("", "Oct", "Nov", "Dec", "Jan", "Feb", "Mar", "Apr"
, "May", "Jun", "Jul", "Aug", "Sep"))
plt.title("Membership Patterns 2016-2017")
plt.legend(bbox_to_anchor=(1.05, 1), loc=2, borderaxespad=0.)
```

Out[11]: <matplotlib.legend.Legend at 0xbd0c2e8>

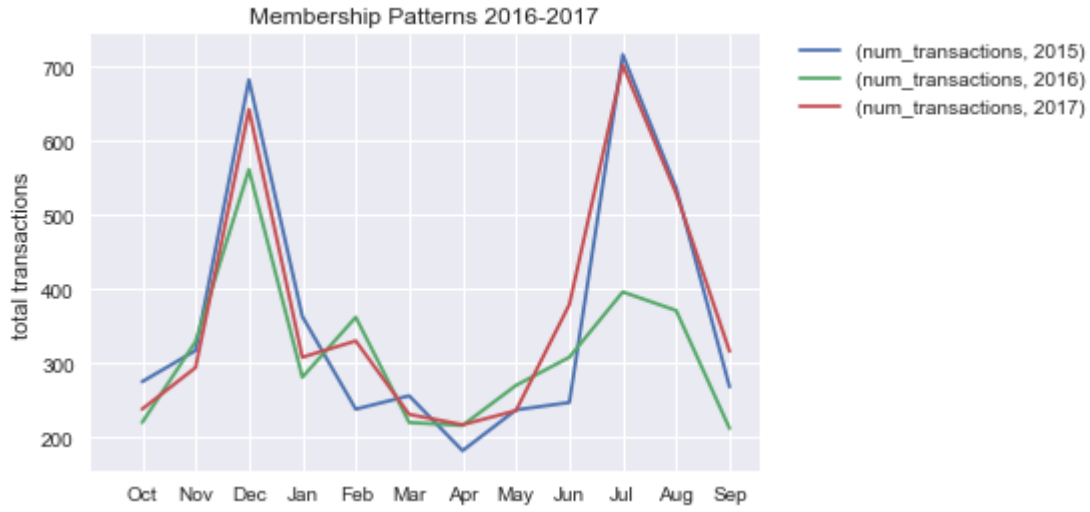


In [10]:
```
#Graph 2015-2017 data to further help compare 2017 results
data_1517 = data[data["fiscal_year"] >= 2015]

data_pivot4 = data_1517.pivot(index="num_month", columns="fiscal_year", values
="num_transactions")

data_1517.pivot(index="num_month", columns="fiscal_year").plot()
plt.ylabel("total transactions")
plt.xlabel("")
plt.xticks(np.arange(13), ("", "Oct", "Nov", "Dec", "Jan", "Feb", "Mar", "Apr"
, "May", "Jun", "Jul", "Aug", "Sep"))
plt.title("Membership Patterns 2016-2017")
plt.legend(bbox_to_anchor=(1.05, 1), loc=2, borderaxespad=0.)
```

Out[10]: <matplotlib.legend.Legend at 0xba9f198>

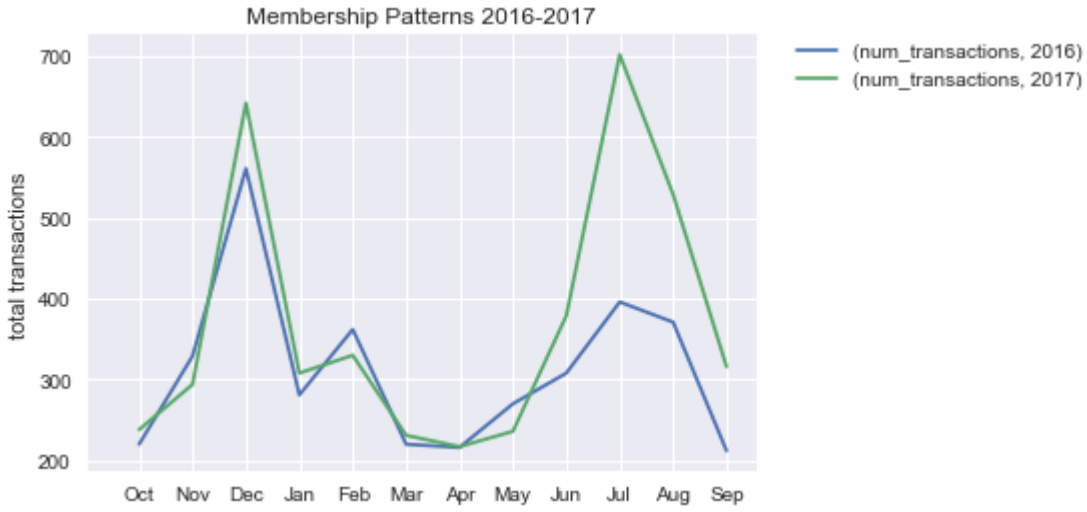


In [9]:
```
#graph 2016 and 2017 data to help compare 2017 results
data_1617 = data[data["fiscal_year"] >= 2016]

data_pivot3 = data_1617.pivot(index="num_month", columns="fiscal_year", values
="num_transactions")

data_1617.pivot(index="num_month", columns="fiscal_year").plot()
plt.ylabel("total transactions")
plt.xlabel("")
plt.xticks(np.arange(13), ("", "Oct", "Nov", "Dec", "Jan", "Feb", "Mar", "Apr"
, "May", "Jun", "Jul", "Aug", "Sep"))
plt.title("Membership Patterns 2016-2017")
plt.legend(bbox_to_anchor=(1.05, 1), loc=2, borderaxespad=0.)
```

Out[9]: <matplotlib.legend.Legend at 0xba20ac8>



In [8]:
```
#graph 2017 data
data_2017 = data[data["fiscal_year"] == 2017]
data_pivot3 = data_2017.pivot(index="num_month", columns="fiscal_year", values
="num_transactions")
data_pivot3

data_2017.pivot(index="num_month", columns="fiscal_year").plot()
plt.ylabel("total transactions")
plt.xlabel("")
plt.xticks(np.arange(13), ("", "Oct", "Nov", "Dec", "Jan", "Feb", "Mar", "Apr"
, "May", "Jun", "Jul", "Aug", "Sep"))
plt.title("Membership Pattern 2017")
plt.legend(bbox_to_anchor=(1.05, 1), loc=2, borderaxespad=0.)
```

Out[8]: <matplotlib.legend.Legend at 0xb5d0470>