

magicpin is India's #1 savings app. No exaggerations.

We are on a mission to make hyperlocal magical. And with more than 1.5 lakh merchant partners and 5 million+ users across a dozen cities, we know a thing or two about scale.

Here's how it works: Pin a location on the map to find the coolest people in the neighbourhood and where they hang out - cafes, restaurants, trendy fashion boutiques, luxurious spas, and health clubs. Find the buzz in your locality through our user generated pictures, videos, and reviews to discover interesting people and places and get cashback/rewards for visiting those places. Yes, including your local grocery stores and pharmacies.

magicpin drives discovery that leads to business for local retailers across categories like restaurants, fashion, beauty, grocery, spa, gyms, and more. For their spending, users get rewarded in magicPoints that can be used to buy more services and experiences at magicpin partner merchants (or you can magicOrder from the comfort of your home!) magicPoints can be earned at ALL stores in a city, and it doesn't matter whether you pay through cash, credit/debit card or e-wallet.

Our users can tap into our community, known as "magicians", who have their finger on the pulse of their city's neighbourhoods. Their picture stories inspire other users to have those experiences. For retailers and brands, magicpin is the go-to platform to reach out to relevant local customers, highlight their uniqueness, reward loyal customers through magicPoints and vouchers, run real-time promotions, augmented reality games, micro-influencer campaigns and more.

magicpin builds engagement and visibility for its partner retailers and brands to help drive up their new and repeat business.

magicpin is by far the leader in offline marketing for retailers in India. We are live in Gurgaon, Noida, Bangalore, Mumbai, Pune, Hyderabad, Chandigarh, Jaipur, Goa, Chennai, Ahmedabad and is quickly spreading to more cities and countries.

## Questions to solve!?

1) What should be the ideal time of day, where we should push our promotions? Which are the best localities to have the promotions between 12 PM to 4 PM time window?

2) In which localities should magicpin push to acquire more users i.e. where is magicpin getting more business but not getting new users. Name 10 such localities and share the step by step approach for the same?

3) Repeat rate: Users who have done their first transaction today and does another transaction in the next 7 days(D7 repeat), Next 15 days(D15 repeat). What is the repeat rate of new users in the system, Share the stepwise approach you took?

4) Based on the above observations, what will be your top 3 recommendations on increasing transactions on the platform and why?

## Feature descriptions!!

First transaction - 0, 1 - 0 means not the first transaction (repeat user), 1 means first transaction (new user)

Funding transaction - Transaction took place at a merchant who is our partner (has paid us money to drive business)

Bill Size - Size of the bill Cashback - Cashback amount that the customer gets

Followers - Followers of the customer

In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [2]:

```
pwd
```

Out[2]:

```
'C:\\Users\\shaqu\\Desktop\\company questionnaires\\MagicPin'
```

In [3]:

```
df= pd.read_excel('magicpin assignment (1).xlsx', parse_dates=True)
df.head()
```

Out[3]:

	ID	customer_id	merchant_id	merchant_name	trxn_locality	home_locality	work_locality	date	time	first transaction?	funding txn?
0	4482549	948930	2722	McDonald's	Sector 18	Chandkheda	Chandkheda	2017-11-15	04:27:39	0	0
1	4482551	1110785	260571	JMart	Sector 45	Sector 48	Sector 135	2017-11-15	04:39:47	0	0
2	4482552	1130583	54220	Gallery cafe - Hyatt Place	Sector 18	Kalbadevi	Andheri East	2017-11-15	04:43:54	1	0
3	4482559	1087586	177490	Railway Yard, The Mid Night Cafe	Ashok Vihar	Ashok Vihar Phase 3	Rana Pratap Bagh	2017-11-15	05:05:04	0	0
4	4482560	1024798	264552	Twenty Four Seven Convenience Store	Saket	Saket	Saket	2017-11-15	05:10:07	0	0

In [4]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 247591 entries, 0 to 247590
Data columns (total 14 columns):
ID                247591 non-null int64
customer_id       247591 non-null int64
merchant_id       247591 non-null int64
merchant_name     247591 non-null object
trxn_locality     247591 non-null object
home_locality     247591 non-null object
work_locality     247591 non-null object
date              247591 non-null datetime64[ns]
time              247591 non-null object
first transaction? 247591 non-null int64
funding txn?      247591 non-null int64
bill size         247591 non-null float64
cashback          247590 non-null float64
followers         247591 non-null int64
dtypes: datetime64[ns](1), float64(2), int64(6), object(5)
memory usage: 26.4+ MB
```

In [5]:

```
df.describe()
```

Out[5]:

	ID	customer_id	merchant_id	first transaction?	funding txn?	bill size	cashback	followers
count	2.475910e+05	2.475910e+05	247591.000000	247591.000000	247591.000000	247591.000000	247590.000000	247591.000000
mean	4.670815e+06	8.221916e+05	136827.268988	0.100662	0.073298	817.401965	30.904070	5000.981865
std	1.082548e+05	3.345909e+05	109197.512474	0.300881	0.260626	2186.034815	35.046481	2886.365406
min	4.482549e+06	2.400000e+01	456.000000	0.000000	0.000000	6.000000	0.000000	1.000000
25%	4.576862e+06	5.807600e+05	16269.000000	0.000000	0.000000	186.000000	9.000000	2500.000000
50%	4.670645e+06	9.464360e+05	102299.000000	0.000000	0.000000	378.000000	22.000000	5011.000000
75%	4.763772e+06	1.109921e+06	242662.000000	0.000000	0.000000	793.000000	44.000000	7498.000000
max	4.861247e+06	1.192475e+06	407529.000000	1.000000	1.000000	453285.000000	425.000000	10000.000000

In [6]:

```
df.isna().sum()
```

Out[6]:

```
ID                0
customer_id       0
merchant_id       0
merchant_name     0
txn_locality      0
home_locality     0
work_locality     0
date              0
time              0
first transaction? 0
funding txn?      0
bill size         0
cashback          1
followers         0
dtype: int64
```

**1) What should be the ideal time of day, where we should push our promotions? Which are the best localities to have the promotions between 12 PM to 4 PM time window?**

In [7]:

```
import datetime
```

In [8]:

```
df['time']
```

Out[8]:

```
0      04:27:39
1      04:39:47
2      04:43:54
3      05:05:04
4      05:10:07
...
247586 23:59:07
247587 23:59:39
247588 23:59:56
247589 23:59:57
247590 23:59:58
Name: time, Length: 247591, dtype: object
```

In [9]:

```
#datetime.datetime.strptime(df['time'], "%H:%M:%S")
```

In [10]:

```
## how to convert time format into hours!!

hours= df['time'].apply(lambda x: x.strftime('%H'))
hours.values
```

Out[10]:

```
array(['04', '04', '04', ..., '23', '23', '23'], dtype=object)
```

In [11]:

```
## grouping the first traction wiht hourly distribution!!

trac1_hour= df['first transaction?'].groupby(hours).sum()
trac1_hour.tail()
```

Out[11]:

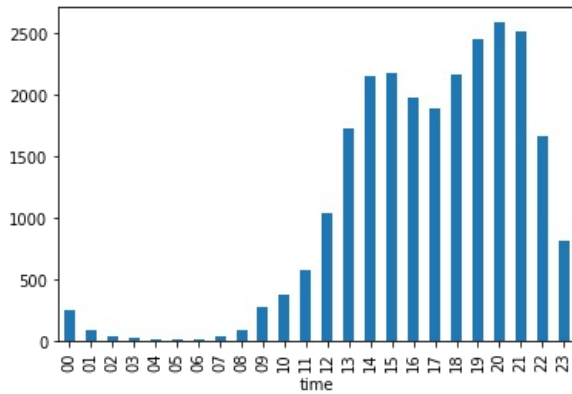
```
time
19    2447
20    2587
21    2517
22    1659
23     809
Name: first transaction?, dtype: int64
```

In [12]:

```
trac1_hour.plot(kind='bar')
```

Out[12]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x18ce21bd648>



Majority of the first tractions done during the evening time when the people goes outside for some snacks and dinner!

from 2pm to 10pm our customers are generally high as this is the they use digital payment method for their meals and snacks in different restraurants and cafeterias!

and the highest first tractions is also go for around 2500 which means some group of friends also uses our service not just individually!

In [13]:

```
bill_hour= df['bill size'].groupby(hours).sum()  
bill_hour.head()
```

Out[13]:

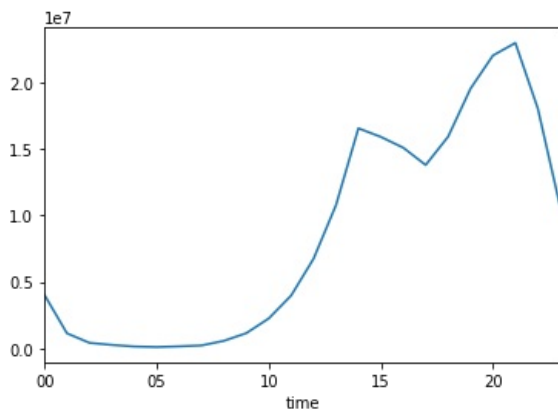
```
time  
00    4024337.86  
01    1148503.45  
02    432480.83  
03    276246.51  
04    156148.63  
Name: bill size, dtype: float64
```

In [14]:

```
bill_hour.plot()
```

Out[14]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x18ce2508a88>

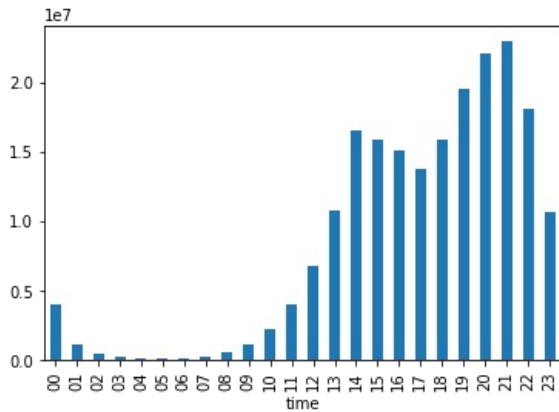


In [15]:

```
bill_hour.plot(kind='bar')
```

Out[15]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x18ce25f10c8>



In the above graph we have showed that from 2pm there is a exponential increase in the bill\_size of our magic pin budget.

and it goes to peak during evening till night around 6pm to 9:30 pm because most of the customers goes to restaurants for having meals and snacks during evening.

So we can also conclude that our main customers come during evening time. And we can target them during evening more easily!!

**Grouping the localities with hours so that we can identify which locality is the best place to promote our promotions between 12 pm to 16pm!!**

In [16]:

```
locality_hour= df['home_locality'].groupby(hours).sum()  
locality_hour.head()
```

Out[16]:

```
time  
00    HinjawadiKarve NagarNITMalad WestMahipalpurMSX...  
01    Vile Parle WestSushant Lok 2KharUttam NagarMum...  
02    Sector 39DT Mega MallMagadi RoadMahimMahimMahi...  
03    Sector 18JangpuraAsaf Ali RoadElectronic CityE...  
04    ChandkhedaSector 48KalbadeviElectronic CityEle...  
Name: home_locality, dtype: object
```

In [17]:

```
# how to print the long text completely in series ?
```

```
pd.options.display.max_colwidth = 100
```

In [18]:

```
# pd.DataFrame(locality_hour)
```

In [19]:

```
locality_hour12_16= df['home_locality'].groupby(hours).sum()  
hr12_16pm= pd.DataFrame(locality_hour12_16[12:17])  
hr12_16pm.head()
```

Out[19]:

		home_locality
time		
12	Sector 42Sector 12 AA	lipurBorivali WestNajafgarhMahipalpurSector 3BTMKudlu GateMG RoadVikaspuriD...
13	ChamarajpetBanjara Hills	ManikondaMajnu ka TilaRajarajeshwari NagarKudlu GateWagle Estate, Thane ...
14	MahadevapuraKurlaHitech City	Sector 24JeedimetlaKammanahalliPrashant ViharDundaheraSector 27LoniS...
15	HBR LayoutSector 23Chamarajpet	Sector 34BTMNehru GroundSwargateAndheri WestSector 70Lado SaraiWil...
16	BTMHAL AllRaj NagarDLF Phase 3	Kempegowda International AirportBavdhanSector 17Sector 25Kalyan Ea...

In [20]:

```
hr12_16pm.T.head()
```

Out[20]:

time		12	13	14	15	
home_locality	WestNajafgarh	Sector 42Sector 12	ChamarajpetBanjara HillsManikondaMajnu ka	MahadevapuraKurlaHitech CitySector	HBR LayoutSector 23ChamarajpetSector 34BTMNehru	BTMHAL All NagarDLF Ph 3Kempegov Internatic
	3BTMKudlu GateMG RoadVikaspuriD...	AAlipurBorivali	TilaRajarajeshwari NagarKudlu GateWagle Estate, Thane ...	24JeedimetlaKammanahalliPrashant ViharDundaheraSector 27LoniS...	GroundSwargateAndheri WestSector 70Lado SaraiWil...	AirportBavdhanSe 17Sector 25Kal E

In [21]:

```
# hr12pm.values.nunique()

# unique, counts= numpy.unique(hr12pm, return_counts= True)
# dict(zip(unique, counts))
```

In [22]:

```
# R0ugh!!

# import numpy

# a = numpy.array(['aa', 3, 0, 1, 0, 1, 2, 'aa', 0, 0, 0, 0, 1, 3, 4])
# unique, counts = numpy.unique(a, return_counts=True)
# dict(zip(unique, counts))
```

In [23]:

```
# Top 10 home locality!

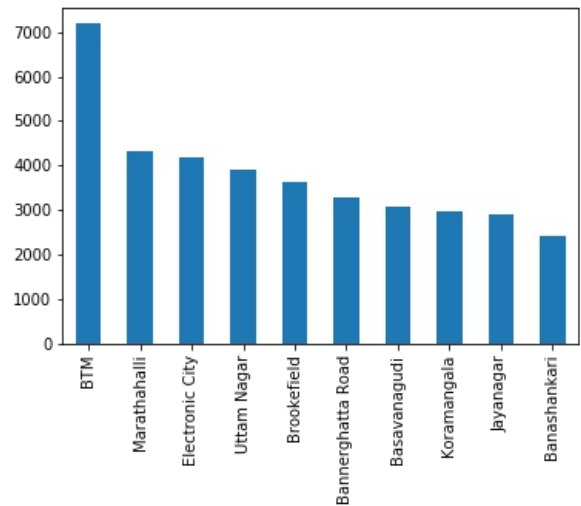
hm_loc= df['home_locality'].value_counts()
hm_loc = hm_loc[0:10]
hm_loc
```

Out[23]:

```
BTM          7182
Marathahalli 4309
Electronic City 4167
Uttam Nagar  3894
Brookefield  3631
Bannerghatta Road 3286
Basavanagudi 3057
Koramangala  2980
Jayanagar    2895
Banashankari 2407
Name: home_locality, dtype: int64
```

In [24]:

```
hm_loc.plot(kind= 'bar')
plt.xticks(rotation='vertical', size=10)
plt.show()
```



In [25]:

```
wrk_loc= df['work_locality'].value_counts()  
wrk_loc.head(10)
```

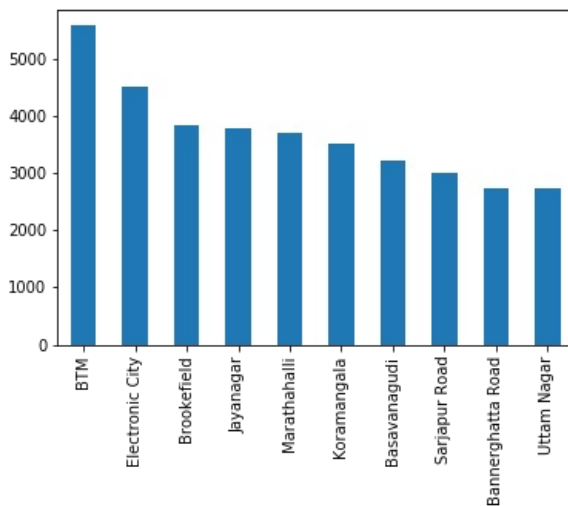
Out[25]:

```
BTM                5568  
Electronic City    4510  
Brookefield        3844  
Jayanagar          3787  
Marathahalli       3700  
Koramangala        3510  
Basavanagudi       3206  
Sarjapur Road      2994  
Bannerghatta Road  2731  
Uttam Nagar        2721  
Name: work_locality, dtype: int64
```

In [26]:

```
# plotting top 10 work localities!!
```

```
wrk_loc = wrk_loc[0:10]  
wrk_loc.plot(kind= 'bar')  
plt.xticks(rotation='vertical', size=10)  
plt.show()
```



In [27]:

```
trx_loc= df['txn_locality'].value_counts()  
trx_loc.head(10)
```

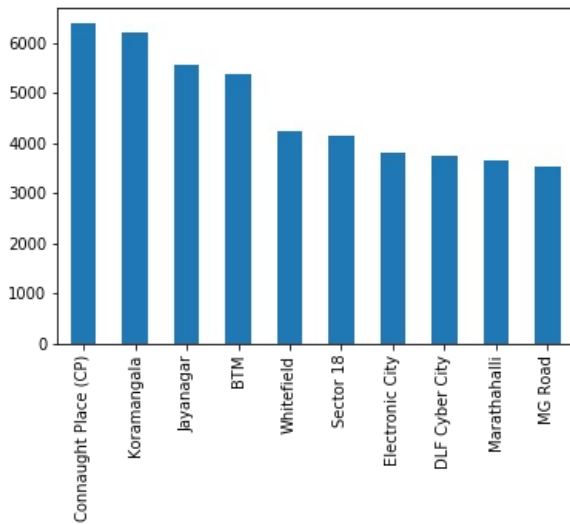
Out[27]:

```
Connaught Place (CP)  6381  
Koramangala           6196  
Jayanagar             5577  
BTM                   5369  
Whitefield            4233  
Sector 18             4137  
Electronic City       3799  
DLF Cyber City        3735  
Marathahalli          3663  
MG Road               3535  
Name: txn_locality, dtype: int64
```

In [28]:

```
# plotting top 10 transaction localities!!
```

```
trx_loc = trx_loc[0:10]
trx_loc.plot(kind= 'bar')
plt.xticks(rotation='vertical', size=10)
plt.show()
```



In [29]:

```
hours.value_counts().head()
```

Out[29]:

```
20    27663
21    27434
19    24376
14    21387
18    19891
Name: time, dtype: int64
```

In [ ]:

In [30]:

```
# hours[12:16].value_counts().groupby(trx_loc.values[:4]).sum()

# # hours.value_counts()
```

In [31]:

```
t= hours.value_counts()
x= t.loc[['12','13', '14', '15', '16']]
x
```

Out[31]:

```
12    9982
13   16613
14   21387
15   18920
16   17675
Name: time, dtype: int64
```

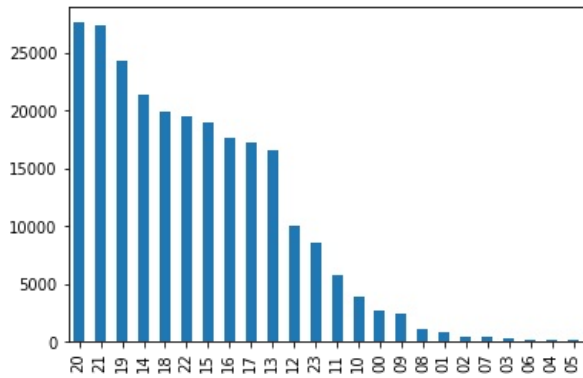


In [32]:

```
hours.value_counts().plot(kind='bar')
```

Out[32]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x18ce29cf1c8>

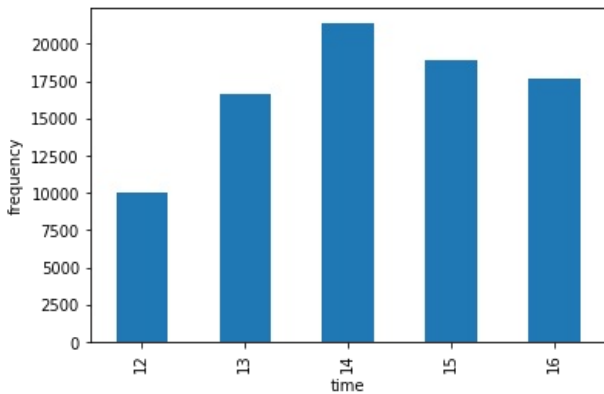


In [33]:

```
x.plot(kind='bar')  
plt.xlabel('time')  
plt.ylabel('frequency')
```

Out[33]:

Text(0, 0.5, 'frequency')



In [34]:

```
x.values.mean()
```

Out[34]:

16915.4

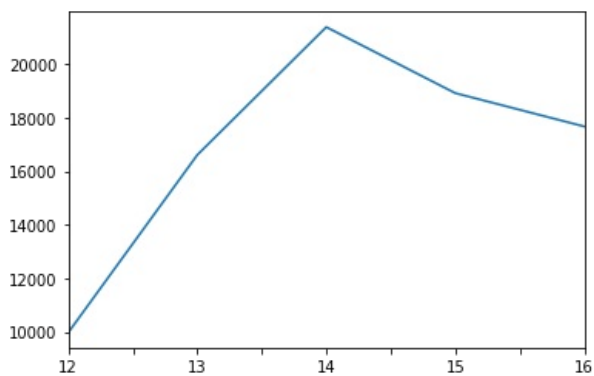
So during the 12 pm to 16 pm the average number of times the customers reach to our service is around 16,915 times!

In [35]:

```
x.plot()
```

Out[35]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x18ce2889248>



***So we can say from our data insights that most of the transactions done from the top 10 localities:***

1) Cannaught Place (CP) 2) Koramangala 3) Jayanagar 4) BTM 5) Whitefield 6) Sector 18 7) Electronics City 8) DLF Cyber city 9) marathahalli 10) MG Road

***And From 12 pm to 16 pm the best localities to promote our promotions will be in localities of Cannaught Palace, BTM, , Jayanagar and Koramangla***

**2) In which localities should magcopin push to acquire more users i.e. where is magicpin getting more business but not getting new users. Name 10 such localities and share the step by step approach for the same?**

In [36]:

```
trx_loc= df['txn_locality'].value_counts()
print('top 10: ', trx_loc.head(10))
print('\n')

print('least 10: ', trx_loc.tail(10))
```

```
top 10:  Connaught Place (CP)      6381
        Koramangala              6196
        Jayanagar                5577
        BTM                     5369
        Whitefield               4233
        Sector 18                4137
        Electronic City          3799
        DLF Cyber City           3735
        Marathahalli             3663
        MG Road                  3535
Name: txn_locality, dtype: int64
```

```
least 10:  Landour              1
          pitampura             1
          Vettuvankeni          1
          Lavasa                1
          Aurangzeb Road        1
          Mayakund              1
          Phase 11              1
          Trilokpuri            1
          GTB nagar             1
          Dhalpur               1
Name: txn_locality, dtype: int64
```

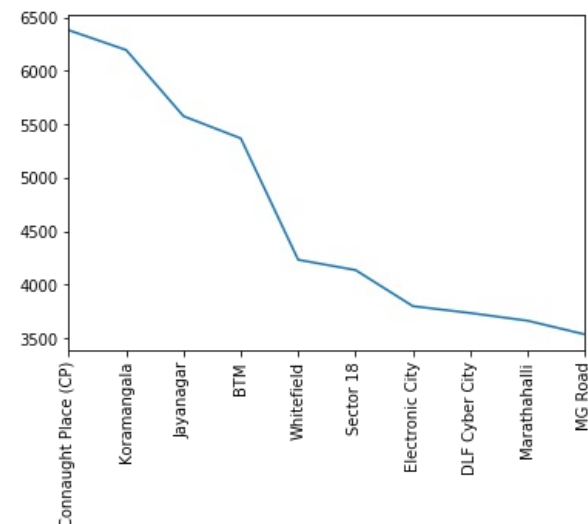
In [37]:

```
trx_loc_plt= trx_loc[:10]

trx_loc_plt.plot()
plt.xticks(rotation='vertical', size=10)
```

Out[37]:

```
(array([0., 1., 2., 3., 4., 5., 6., 7., 8., 9.]),
 <a list of 10 Text xticklabel objects>)
```



Number of times transaction occurred in each locality! max is in Connaught Place, and lowest is in Library Road

In [38]:

```
df['first transaction?'].value_counts()

# 1 mean first transaction, 0 means not first transaction!
```

Out[38]:

```
0    222668
1     24923
Name: first transaction?, dtype: int64
```

In [39]:

```
bill_txn= df['bill size'].groupby(df['txn_locality']).sum()
bill_txn.head()
```

Out[39]:

```
txn_locality
A S Rao Nagar      5633.00
Aaya Nagar         66393.65
Abids              170359.40
Adarsh Nagar       82770.55
Adchini            31452.00
Name: bill size, dtype: float64
```

In [40]:

```
sorted(bill_txn)[-10:]
```

Out[40]:

```
[2979402.129999999,
 2980735.4400000013,
 3216833.1399999997,
 3392528.110000001,
 4138985.7700000005,
 4616727.3399999997,
 4882715.62,
 4905217.7199999995,
 5530471.8300000004,
 6311005.869999999]
```

In [41]:

```
# top 10 localities with more businesses!!
bill_txn.sort_values(ascending= False)[:10]
```

Out[41]:

```
txn_locality
Connaught Place (CP)  6311005.87
Koramangala           5530471.83
Sector 18             4905217.72
Jayanagar             4882715.62
MG Road              4616727.34
Marathahalli          4138985.77
BTM                  3392528.11
Whitefield            3216833.14
DLF Cyber City        2980735.44
JP Nagar              2979402.13
Name: bill size, dtype: float64
```

In [42]:

```
plt_bill_txn= bill_txn.sort_values(ascending= False)[:10]  
plt_bill_txn
```

Out[42]:

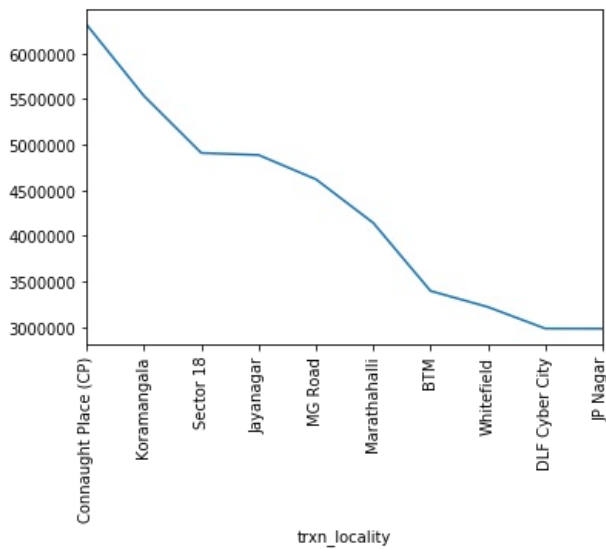
```
txn_locality  
Connaught Place (CP)    6311005.87  
Koramangala             5530471.83  
Sector 18               4905217.72  
Jayanagar               4882715.62  
MG Road                 4616727.34  
Marathahalli            4138985.77  
BTM                     3392528.11  
Whitefield              3216833.14  
DLF Cyber City          2980735.44  
JP Nagar                2979402.13  
Name: bill size, dtype: float64
```

In [43]:

```
plt_bill_txn.plot(kind= 'line')  
plt.xticks(rotation='vertical', size=10)
```

Out[43]:

```
(array([0., 1., 2., 3., 4., 5., 6., 7., 8., 9.]),  
<a list of 10 Text xticklabel objects>)
```



MAximum transaction amount occur in cannaught paalace with bill size around 6311005 rupees! and the top 5 localities with bill size are :

Connaught Place (CP) 6311005.87

Koramangala 5530471.83

Sector 18 4905217.72

Jayanagar 4882715.62

MG Road 4616727.34

Marathahalli 4138985.77

BTM 3392528.11

Whitefield 3216833.14

DLF Cyber City 2980735.44

JP Nagar 2979402.13

In [44]:

```
# df['txn_locality'].groupby(df['first transaction?'])
```

```
df['first transaction?'].value_counts()
```

```
0      222668
1      24923
Name: first transaction?, dtype: int64
```

```
df[['first transaction?', 'txn_locality']]
```

first transaction?	txn_locality
0	Sector 18
1	Sector 45
2	Sector 18
3	Ashok Vihar
4	Saket
...	...
247586	Sinhgad Road
247587	DLF Phase 3
247588	Bannerghatta Road
247589	Magarpatta
247590	FC Road

In [47]:

```
df.groupby(['first transaction?', 'trxn_locality']).ngroups
```

1706

```
df.groupby(['txn_locality']).count().head()
```

[illegible]

In [49]:

```
first_transaction_locality= df['first transaction?'].groupby(df['txn_locality']).sum().sort_values(ascending= False)
first_transaction_locality
```

Out[49]:

```
txn_locality
Connaught Place (CP)    1562
Jayanagar                609
BTM                     571
FC Road                 546
Koramangala             489
...
Narayan Singh Circle    0
Nayagaon                0
Neelangerai             0
Netaji Nagar            0
sector 14               0
Name: first transaction?, Length: 950, dtype: int64
```

first transaction is highest in Delhi Connaught palace CP, Jayanaagr, BTM, FC Road GTB Naagr areas Andheri West and Ghotkopar East

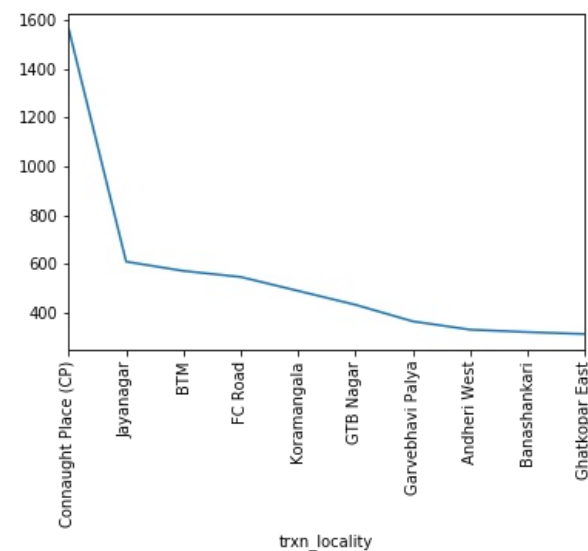
Connaught Place (CP) 1562 Jayanagar 609 BTM 571 FC Road 546 Koramangala 489 GTB Nagar 432 Garvebhavi Palya 364 Andheri West 330 Banashankari 320 Ghatkopar East 312

In [50]:

```
first_transaction_locality[:10].plot()
plt.xticks(rotation='vertical', size=10)
```

Out[50]:

```
(array([0., 1., 2., 3., 4., 5., 6., 7., 8., 9.]),
 <a list of 10 Text xticklabel objects>)
```



first transaction is highest in Delhi Connaught palace CP, Jayanaagr, BTM, FC Road GTB Naagr areas Andheri West

and Sector 18 and Mg road have high business but less new users which is only 218 in sector 18 and 151 in MG road!!

In [51]:

```
df['bill size'].mean()
# The average bill size of the customer uses the magicpipn service is 817 rupees!
```

Out[51]:

```
817.4019645706026
```

In [52]:

```
df['bill size'].median()
```

Out[52]:

```
378.0
```

**3)Repeat rate: Users who have done their first transaction today and does another transaction in the next 7 days(D7 repeat), Next 15 days(D15 repeat). What is the repeat rate of new users in the system, Share the stepwise approach you took?**

In [53]:

```
df['first transaction?'].groupby(df['customer_id']).count().sort_values(ascending= False)
```

Out[53]:

```
customer_id
1130580    127
1136339     89
994954     86
114833     76
1063810     75
...
1022665     1
1022626     1
1022621     1
1022567     1
958352      1
Name: first transaction?, Length: 83105, dtype: int64
```

In [54]:

```
# top 10 customers with first trabsactions are:
top_10cus= df['first transaction?'].groupby(df['customer_id']).count().sort_values(ascending= False)[:10]
top_10cus
```

Out[54]:

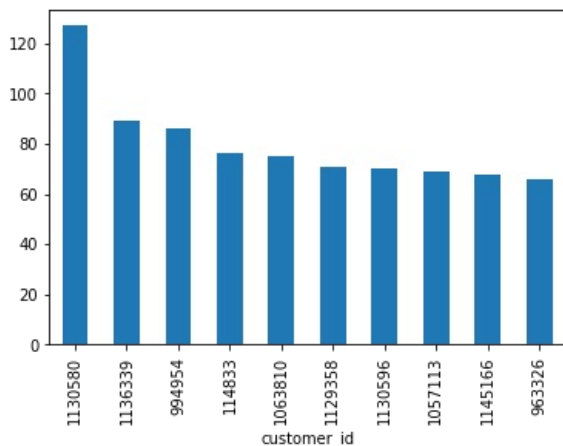
```
customer_id
1130580    127
1136339     89
994954     86
114833     76
1063810     75
1129358     71
1130596     70
1057113     69
1145166     68
963326     66
Name: first transaction?, dtype: int64
```

In [55]:

```
top_10cus.plot(kind= 'bar')
```

Out[55]:

<matplotlib.axes.\_subplots.AxesSubplot at 0x18cded92648>



customer with customer id 1130580 is around 127 times, and id with 1136339 is 89

In [56]:

```
df['first transaction?'].groupby(df['customer_id']).count().sort_values(ascending= False)[:10]
```

Out[56]:

```
customer_id
1130580      127
1136339      89
994954       86
114833       76
1063810      75
1129358      71
1130596      70
1057113      69
1145166      68
963326       66
Name: first transaction?, dtype: int64
```

In [57]:

```
df['customer_id'].groupby(df['first transaction?']).value_counts(sort= True, ascending = False)
```

Out[57]:

```
first transaction?  customer_id
0                  1130580      127
                  1136339      88
                  994954      86
                  114833      76
                  1063810      75
                  ...
1                  1192420       1
                  1192432       1
                  1192441       1
                  1192453       1
                  1192475       1
Name: customer_id, Length: 93631, dtype: int64
```

from the above data of the first transaction and customer we can clearly see that maximum number of times a customer use the magicpin is 127 times so if we divide the maximum number by 7 days then it is repeating in a round 18 days consecutively

and we divide the repeat rate by 15 days then in every 8 days the customer with maximum transaction repeats!!

and for average customers the transaction repeats to every 10 days with 7 days repetition and with repetition of 15 days it will repeat in around every 5 days!!

#### 4)Based on the above observations, what will be your top 3 recommendations on increasing transactions on the platform and why?

Some of my recommendations will be :

- 1) Focus on the localities which have around the average transaction pay times will be provided promo codes and luring them to attract more to our services
- 2) As I optimize the localities who have high transactions areas but have low new transactions should be more focussed with proper advertising of our products!! We should focus on advertising our products to different restaurants to these areas and also tie up with different merchants to increase the reach of our service in these areas!!
- 3) Providing cashback is not a good idea for long run so we should also focus on building the trust of the customers by providing secure money transfer with more transparency but we can provide some useful coupons according to the users last purchase patterns so that the user also enjoys the same coupons for future uses!!
- 4) For the above recommendation we would also need to tie up with some big data e-commerce company so that we can track our users shopping preferences and according to that we provide coupons for their relevant uses!!

In [58]:

```
# pip install nbconvert
```

In [ ]:



In [ ]:

In [ ]: