Ecommerce Purchases Project

In [58]:

```
import pandas as pd
```

In [59]:

```
data = pd.read_csv('Ecommerce Purchases')
data
```

Out[59]:

	Address	Lot	AM or PM	Browser Info	Company	Credit Card	CC Exp Date	Sec
0	16629 Pace Camp Apt. 448\nAlexisborough, NE 77...	46 in	PM	Opera/9.56.(X11; Linux x86_64; sl-SI) Presto/2...	Martinez-Herman	6011929061123406	02/20	
1	9374 Jasmine Spurs Suite 508\nSouth John, TN 8...	28 rn	PM	Opera/8.93.(Windows 98; Win 9x 4.90; en-US) Pr...	Fletcher, Richards and Whitaker	3337758169645356	11/18	
2	Unit 0065 Box 5052\nDPO AP 27450	94 vE	PM	Mozilla/5.0 (compatible; MSIE 9.0; Windows NT ...	Simpson, Williams and Pham	675957666125	08/19	
3	7780 Julia Fords\nNew Stacy, WA 45798	36 vm	PM	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_8_0 ...	Williams, Marshall and Buchanan	6011578504430710	02/24	
4	23012 Munoz Drive Suite 337\nNew Cynthia, TX 5...	20 IE	AM	Opera/9.58.(X11; Linux x86_64; it-IT) Presto/2...	Brown, Watson and Andrews	6011456623207998	10/25	
...	...	...	...	...	...	...	...	
9995	966 Castaneda Locks\nWest Juliafurt, CO 96415	92 XI	PM	Mozilla/5.0 (Windows NT 5.1) AppleWebKit/5352 ...	Randall-Sloan	342945015358701	03/22	
9996	832 Curtis Dam Suite 785\nNorth Edwardburgh, T...	41 JY	AM	Mozilla/5.0 (compatible; MSIE 9.0; Windows NT ...	Hale, Collins and Wilson	210033169205009	07/25	
9997	Unit 4434 Box 6343\nDPO AE 28026-0283	74 Zh	AM	Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10_7...	Anderson Ltd	6011539787356311	05/21	
9998	0096 English Rest\nRoystad, IA 12457	74 cL	PM	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_8_8;...	Cook Inc	180003348082930	11/17	
9999	40674 Barrett Stravenue\nGrimesville, WI 79682	64 Hr	AM	Mozilla/5.0 (X11; Linux i686; rv:1.9.5.20) Gec...	Greene Inc	4139972901927273	02/19	

10000 rows × 14 columns

1. Display Top 10 Rows of The Dataset

In [60]:

```
data.head(10)
```

Out[60]:

	Address	Lot	AM or PM	Browser Info	Company	Credit Card	CC Exp Date
0	16629 Pace Camp Apt. 448\nAlexisborough, NE 77...	46 in	PM	Opera/9.56.(X11; Linux x86_64; sl-SI) Presto/2...	Martinez-Herman	6011929061123406	02/20
1	9374 Jasmine Spurs Suite 508\nSouth John, TN 8...	28 rn	PM	Opera/8.93.(Windows 98; Win 9x 4.90; en-US) Pr...	Fletcher, Richards and Whitaker	3337758169645356	11/18
2	Unit 0065 Box 5052\nDPO AP 27450	94 vE	PM	Mozilla/5.0 (compatible; MSIE 9.0; Windows NT ...	Simpson, Williams and Pham	675957666125	08/19
3	7780 Julia Fords\nNew Stacy, WA 45798	36 vm	PM	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_8_0 ...	Williams, Marshall and Buchanan	6011578504430710	02/24
4	23012 Munoz Drive Suite 337\nNew Cynthia, TX 5...	20 IE	AM	Opera/9.58.(X11; Linux x86_64; it-IT) Presto/2...	Brown, Watson and Andrews	6011456623207998	10/25
5	7502 Powell Mission Apt. 768\nTravisland, VA 3...	21 XT	PM	Mozilla/5.0 (Macintosh; U; PPC Mac OS X 10_8_5...	Silva-Anderson	30246185196287	07/25
6	93971 Conway Causeway\nAndersonburgh, AZ 75107	96 Xt	AM	Mozilla/5.0 (compatible; MSIE 7.0; Windows NT ...	Gibson and Sons	6011398782655569	07/24
7	260 Rachel Plains Suite 366\nCastroberg, WV 24...	96 pG	PM	Mozilla/5.0 (X11; Linux i686) AppleWebKit/5350...	Marshall-Collins	561252141909	06/25
8	2129 Dylan Burg\nNew Michelle, ME 28650	45 JN	PM	Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10_7...	Galloway and Sons	180041795790001	04/24
9	3795 Dawson Extensions\nLake Tinafort, ID 88739	15 Ug	AM	Mozilla/5.0 (X11; Linux i686; rv:1.9.7.20) Gec...	Rivera, Buchanan and Ramirez	4396283918371	01/17

2. Check Last 10 Rows of The Dataset

In [61]:

```
data.tail(10)
```

Out[61]:

	Address	Lot	AM or PM	Browser Info	Company	Credit Card	CC Exp Date	Sec
9990	75731 Molly Springs\nWest Danielle, VT 96934-5102	93 ty	PM	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_7_4;...	Pace, Vazquez and Richards	869968197049750	04/24	
9991	PSC 8165, Box 8498\nAPO AP 60327-0346	50 dA	AM	Mozilla/5.0 (compatible; MSIE 8.0; Windows NT ...	Snyder Inc	4221582137197481	02/24	
9992	885 Allen Mountains Apt. 230\nWallhaven, LA 16995	40 vH	PM	Mozilla/5.0 (Macintosh; PPC Mac OS X 10_6_5) A...	Wells Ltd	4664825258997302	10/20	
9993	7555 Larson Locks Suite 229\nEllisburgh, MA 34...	72 jg	PM	Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10_8...	Colon and Sons	30025560104631	10/25	
9994	6276 Rojas Hollow\nLake Louis, WY 56410-7837	93 Ex	PM	Opera/9.68.(X11; Linux x86_64; sl-SI) Presto/2...	Ritter-Smith	3112186784121077	01/25	
9995	966 Castaneda Locks\nWest Juliafurt, CO 96415	92 XI	PM	Mozilla/5.0 (Windows NT 5.1) AppleWebKit/5352 ...	Randall-Sloan	342945015358701	03/22	
9996	832 Curtis Dam Suite 785\nNorth Edwardburgh, T...	41 JY	AM	Mozilla/5.0 (compatible; MSIE 9.0; Windows NT ...	Hale, Collins and Wilson	210033169205009	07/25	
9997	Unit 4434 Box 6343\nDPO AE 28026-0283	74 Zh	AM	Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10_7...	Anderson Ltd	6011539787356311	05/21	
9998	0096 English Rest\nRoystad, IA 12457	74 cL	PM	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_8_8;...	Cook Inc	180003348082930	11/17	
9999	40674 Barrett Stravenue\nGrimesville, WI 79682	64 Hr	AM	Mozilla/5.0 (X11; Linux i686; rv:1.9.5.20) Gec...	Greene Inc	4139972901927273	02/19	

3. Check Datatype of Each Column

In [62]:

```
data.dtypes
```

```
Out[62]: Address              object
         Lot                  object
         AM or PM             object
         Browser Info         object
         Company              object
         Credit Card           int64
         CC Exp Date          object
         CC Security Code      int64
         CC Provider          object
         Email                object
         Job                  object
         IP Address           object
         Language             object
         Purchase Price      float64
         dtype: object
```

4. Check Null Values In the Dataset

```
In [63]: data.isnull().sum()
```

```
Out[63]: Address             0
         Lot                 0
         AM or PM            0
         Browser Info        0
         Company             0
         Credit Card         0
         CC Exp Date         0
         CC Security Code    0
         CC Provider         0
         Email               0
         Job                 0
         IP Address          0
         Language            0
         Purchase Price      0
         dtype: int64
```

5. How many Rows And Columns Are In Our Dataset?

```
In [64]: len(data.columns)
```

```
Out[64]: 14
```

```
In [65]: len(data)
```

```
Out[65]: 10000
```

```
In [66]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 14 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   Address           10000 non-null  object 
 1   Lot               10000 non-null  object 
 2   AM or PM          10000 non-null  object 
 3   Browser Info      10000 non-null  object 
 4   Company           10000 non-null  object 
 5   Credit Card       10000 non-null  int64  
 6   CC Exp Date       10000 non-null  object 
 7   CC Security Code  10000 non-null  int64  
 8   CC Provider       10000 non-null  object 
 9   Email             10000 non-null  object 
 10  Job               10000 non-null  object 
 11  IP Address        10000 non-null  object 
 12  Language          10000 non-null  object 
 13  Purchase Price    10000 non-null  float64
dtypes: float64(1), int64(2), object(11)
memory usage: 1.1+ MB
```

6. Highest And Lowest Purchase Prices

In [67]:

```
data.columns
```

Out[67]:

```
Index(['Address', 'Lot', 'AM or PM', 'Browser Info', 'Company', 'Credit Card',
       'CC Exp Date', 'CC Security Code', 'CC Provider', 'Email', 'Job',
       'IP Address', 'Language', 'Purchase Price'],
      dtype='object')
```

In [68]:

```
data['Purchase Price'].max()
```

Out[68]:

```
99.99
```

In [69]:

```
data['Purchase Price'].min()
```

Out[69]:

```
0.0
```

7. Average Purchase Price

In [70]:

```
data['Purchase Price'].mean()
```

Out[70]:

```
50.347302
```

8. How Many People Have French 'fr' As Their Language ?

In [71]:

```
len(data[data['Language']=='fr'])
```

Out[71]:

```
1097
```

In [72]:

```
data[data['Language']=='fr'].count()
```

Out[72]:

```
Address             1097
Lot                 1097
AM or PM            1097
Browser Info        1097
Company             1097
Credit Card         1097
CC Exp Date         1097
CC Security Code    1097
CC Provider         1097
Email               1097
Job                 1097
IP Address          1097
Language            1097
Purchase Price      1097
dtype: int64
```

9. Job Title Contains Engineer

In [73]:

```
data.columns
```

Out[73]:

```
Index(['Address', 'Lot', 'AM or PM', 'Browser Info', 'Company', 'Credit Card',
       'CC Exp Date', 'CC Security Code', 'CC Provider', 'Email', 'Job',
       'IP Address', 'Language', 'Purchase Price'],
      dtype='object')
```

In [74]:

```
len(data[data['Job'].str.contains('engineer', case=False)])
```

Out[74]:

```
984
```

10. Find Email of The Person With The Following IP address: 132.207.160.22

In [75]:

```
data[data['IP Address']=='132.207.160.22']['Email']
```

Out[75]:

```
2    amymiller@morales-harrison.com
Name: Email, dtype: object
```

11. How Many People Have Mastercard As Their Credit Card Provider And Made A Purchase Above 50?

In [76]:

```
data.columns
```

Out[76]:

```
Index(['Address', 'Lot', 'AM or PM', 'Browser Info', 'Company', 'Credit Card',
       'CC Exp Date', 'CC Security Code', 'CC Provider', 'Email', 'Job',
       'IP Address', 'Language', 'Purchase Price'],
      dtype='object')
```

In [77]:

```
len(data[(data['CC Provider']=='Mastercard') & (data['Purchase Price']>50)])
```

Out[77]:

```
405
```

In [78]:

```
data[(data['CC Provider']=='Mastercard') \
     & (data['Purchase Price']>50)].count()
```

Out[78]:

```
Address              405
Lot                  405
AM or PM             405
Browser Info         405
Company              405
Credit Card          405
CC Exp Date          405
CC Security Code     405
CC Provider          405
Email                405
Job                  405
IP Address           405
Language             405
Purchase Price       405
dtype: int64
```

12. Find Email of The Person With The Following Credit Card Number: 4664825258997302

In [79]:

```
data[data['Credit Card']==4664825258997302]['Email']
```

Out[79]:

```
9992    bberry@wright.net
Name: Email, dtype: object
```

13. How Many People Purchase During The AM and How Many People Purchase During PM ?

In [80]:

```
data .columns
```

Out[80]:

```
Index(['Address', 'Lot', 'AM or PM', 'Browser Info', 'Company', 'Credit Card',
       'CC Exp Date', 'CC Security Code', 'CC Provider', 'Email', 'Job',
       'IP Address', 'Language', 'Purchase Price'],
      dtype='object')
```

In [81]:

```
data['AM or PM'].value_counts()
```

Out[81]:

```
PM    5068
AM    4932
Name: AM or PM, dtype: int64
```

14. How many People Have A Credit Card That Expires In 2020?

In [82]:

```
data['CC Exp Date']
```

Out[82]:

```
0       02/20
1       11/18
2       08/19
3       02/24
4       10/25
        ...  
9995    03/22
9996    07/25
9997    05/21
9998    11/17
9999    02/19
Name: CC Exp Date, Length: 10000, dtype: object
```

In [87]:

```
def fun():
    count=0
    for date in data['CC Exp Date']:
        if date.split('/')[1]=='20':
            count=count+1
    print(count)
```

In [88]:

```
fun()
```

```
988
```

Another Way

In [91]:

```
len(data[data['CC Exp Date'].apply(lambda x:x[3:]=='20')])
```

Out[91]:

```
988
```

15. Top 5 Most Popular Email providers (e.g. gmail.com, yahoo.com, etc..)

In [94]:

```
list1=[]
for email in data['Email']:
    list1.append(email.split('@')[1])
```

In [95]:

```
data['temp']=list1
```

In [96]:

```
data.head(1)
```

Out[96]:

	Address	Lot	AM or PM	Browser Info	Company	Credit Card	CC Exp Date	CC Security Code	Provi
0	16629 Pace Camp Apt. 448\nAlexisborough, NE 77...	46 in	PM	Opera/9.56. (X11; Linux x86_64; sl-SI) Presto/2...	Martinez-Herman	6011929061123406	02/20	900	JCB d

In [98]:

```
data['temp'].value_counts().head()
```

Out[98]:

```
hotmail.com     1638
yahoo.com       1616
gmail.com       1605
smith.com         42
williams.com      37
Name: temp, dtype: int64
```

Another Way

In [99]:

```
data['Email'].apply(lambda x:x.split('@')[1]).value_counts().head()
```

Out[99]:

```
hotmail.com     1638
yahoo.com       1616
gmail.com       1605
smith.com         42
williams.com      37
Name: Email, dtype: int64
```