

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

```
In [6]: a= {'b': "C:/Users/Seth-Ese Joey/Downloads/brewery_data.csv",
          'c': "C:/Users/Seth-Ese Joey/Downloads/sales_reps.csv"}
a
```

```
Out[6]: {'b': 'C:/Users/Seth-Ese Joey/Downloads/brewery_data.csv',
        'c': 'C:/Users/Seth-Ese Joey/Downloads/sales_reps.csv'}
```

```
In [4]: df= pd.read_csv(a['b'])
df.head()
```

```
Out[4]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PF
0	10101	Jardine	jard@gmail.com	trophy	150	200	725	145000	:
1	10102	Gill	gillhell@uk.com	budweiser	250	500	815	407500	20
2	10103	Sorvino	sorvi2000@gmail.com	castle lite	180	450	937	421650	20
3	10104	Jones	jone.ai@yahoo.com	eagle lager	170	250	765	191250	0
4	10105	Andrews	andy@gmail.com	hero	150	200	836	167200	.

```
In [7]: dff=pd.read_csv(a['c'])
dff.head()
```

```
Out[7]:
```

	id	name	region_id
0	321500	Samuel Racine	1
1	321510	Eugena Esser	1
2	321520	Michel Averette	1
3	321530	Renetta Carew	1
4	321540	Cara Clarke	1

```
In [3]: df.columns
```

```
Out[3]: Index(['SALES_ID', 'SALES_REP', 'EMAILS', 'BRANDS', 'PLANT_COST', 'UNIT_PRICE',
              'QUANTITY', 'COST', 'PROFIT', 'COUNTRIES', 'REGION ', 'MONTHS',
              'YEARS'],
              dtype='object')
```

```
In [21]: df.rename(columns={'REGION ':'REGION'}, inplace=True)
df.head()
```

```
Out[21]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PF
0	10101	Jardine	jard@gmail.com	trophy	150	200	725	145000	:
1	10102	Gill	gillhell@uk.com	budweiser	250	500	815	407500	20
2	10103	Sorvino	sorvi2000@gmail.com	castle lite	180	450	937	421650	20
3	10104	Jones	jone.ai@yahoo.com	eagle lager	170	250	765	191250	0
4	10105	Andrews	andy@gmail.com	hero	150	200	836	167200	.

```

In [22]: len(df)
Out[22]: 1047

In [23]: df.columns
Out[23]: Index(['SALES_ID', 'SALES_REP', 'EMAILS', 'BRANDS', 'PLANT_COST', 'UNIT_PRICE',
               'QUANTITY', 'COST', 'PROFIT', 'COUNTRIES', 'REGION', 'MONTHS', 'YEARS'],
              dtype='object')

In [24]: df.shape # to know how many rows and columns
Out[24]: (1047, 13)

In [25]: df.columns
Out[25]: Index(['SALES_ID', 'SALES_REP', 'EMAILS', 'BRANDS', 'PLANT_COST', 'UNIT_PRICE',
               'QUANTITY', 'COST', 'PROFIT', 'COUNTRIES', 'REGION', 'MONTHS', 'YEARS'],
              dtype='object')

In [26]: df.info() # check if your data is clean
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1047 entries, 0 to 1046
Data columns (total 13 columns):
#   Column          Non-Null Count  Dtype
---  -
0   SALES_ID         1047 non-null  int64
1   SALES_REP        1047 non-null  object
2   EMAILS           1047 non-null  object
3   BRANDS           1047 non-null  object
4   PLANT_COST       1047 non-null  int64
5   UNIT_PRICE       1047 non-null  int64
6   QUANTITY         1047 non-null  int64
7   COST             1047 non-null  int64
8   PROFIT           1047 non-null  int64
9   COUNTRIES        1047 non-null  object
10  REGION           1047 non-null  object
11  MONTHS           1047 non-null  object
12  YEARS            1047 non-null  int64
dtypes: int64(7), object(6)
memory usage: 106.5+ KB

In [27]: df.isnull().sum() # to check if ur data is clean
Out[27]: SALES_ID         0
SALES_REP         0
EMAILS            0
BRANDS            0
PLANT_COST        0
UNIT_PRICE        0
QUANTITY          0
COST              0
PROFIT            0
COUNTRIES         0
REGION            0
MONTHS            0
YEARS             0
dtype: int64

In [28]: df.isna().sum()

```

```
Out[28]: SALES_ID      0
          SALES_REP    0
          EMAILS      0
          BRANDS      0
          PLANT_COST   0
          UNIT_PRICE   0
          QUANTITY     0
          COST         0
          PROFIT       0
          COUNTRIES    0
          REGION      0
          MONTHS       0
          YEARS        0
          dtype: int64
```

```
In [29]: df['COUNTRIES'].unique()
```

```
Out[29]: array(['Ghana', 'Nigeria', 'Togo', 'Benin', 'Senegal'], dtype=object)
```

```
In [30]: df['YEARS'].unique()
```

```
Out[30]: array([2019, 2018, 2017], dtype=int64)
```

PROFIT ANALYSIS

```
In [31]: # 1.Within the space of the last three years, what was the profit worth of the breweries
         # inclusive of the anglophone and the francophone territories?
```

```
sd=df['PROFIT'].sum()
print(f'The Profit worth for the past three years is {sd}')
```

The Profit worth for the past three years is 105587420

```
In [16]: # 2.Compare the total profit between these two territories in order for the territory ma
         # Mr. Stone made a strategic decision that will aid profit maximization in 2020.
         # For Francophone countries
```

```
df1=df[(df.COUNTRIES=='Benin')|(df.COUNTRIES=='Togo')|(df.COUNTRIES=='Senegal')]
df1.head()
```

```
Out[16]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PI
2	10103	Sorvino	sorvi2000@gmail.com	castle lite	180	450	937	421650	2
3	10104	Jones	jone.ai@yahoo.com	eagle lager	170	250	765	191250	
4	10105	Andrews	andy@gmail.com	hero	150	200	836	167200	
7	10108	Jones	jone.ai@yahoo.com	trophy	150	200	812	162400	
8	10109	Morgan	morganny@gmail.com	budweiser	250	500	700	350000	1

```
In [17]: df1_franco=df1['PROFIT'].sum()
         print(f'The profit of the francophone countries is {df1_franco}')
```

The profit of the francophone countries is 63198160

```
In [32]: df2= df[(df.COUNTRIES=='Nigeria')|(df.COUNTRIES=='Ghana')]
         df2.head()
```

Out[32]:

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT
0	10101	Jardine	jard@gmail.com	trophy	150	200	725	145000	362
1	10102	Gill	gillhell@uk.com	budweiser	250	500	815	407500	2037
5	10106	Jardine	jard@gmail.com	beta malt	80	150	798	119700	558
6	10107	Thompson	thomp@uk.com	grand malt	90	150	954	143100	572
10	10111	Parent	parentty@uk.com	eagle lager	170	250	861	215250	688

```
In [19]: df2_angloPROfit= df2['PROFIT'].sum()
print(f'The profit of the anglophone countries is {df2_angloPROfit}')
```

The profit of the anglophone countries is 42389260

```
In [33]: #3.Country that generated the highest profit in 2019
df2019=df[(df.YEARS==2019)]
df2019.head()
```

Out[33]:	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT
	0	10101	Jardine	jard@gmail.com	trophy	150	200	725	145000
	5	10106	Jardine	jard@gmail.com	beta malt	80	150	798	119700
	8	10109	Morgan	morganny@gmail.com	budweiser	250	500	700	350000
	11	10112	Jones	jone.ai@yahoo.com	hero	150	200	902	180400
	15	10116	Jones	jone.ai@yahoo.com	budweiser	250	500	709	354500

```
In [21]: df2019.groupby('COUNTRIES')['PROFIT'].sum().sort_values(ascending=False)
```

```
Out[21]: COUNTRIES
Ghana      7144070
Senegal    6687560
Togo       6109960
Benin      5273340
Nigeria    4805320
Name: PROFIT, dtype: int64
```

```
In [22]: #4.Help him find the year with the highest profit.
df.groupby('YEARS')['PROFIT'].sum().sort_values(ascending=False)
```

```
Out[22]: YEARS
2017      38503320
2018      37063850
2019      30020250
Name: PROFIT, dtype: int64
```

```
In [23]: #5.Which month in the three years was the least profit generated?
df.groupby('MONTHS')['PROFIT'].sum().sort_values(ascending=True)
```

Out[23]:

MONTHS	
April	8573830
September	8702660
January	8722990
August	8730940
October	8767100
May	8772250
July	8787010
December	8787290
November	8820330
June	8828440
February	9028450
March	9066130

Name: PROFIT, dtype: int64

In [24]:

```
#6.What was the minimum profit in the month of December 2018?
df_minProfit= df[(df.YEARS==2018) & (df.MONTHS=='December')]
df_minProfit.head()
```

Out[24]:

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST
71	10172	Jones	jone.ai@yahoo.com	budweiser	250	500	869	434500
95	10196	Sorvino	sorvi2000@gmail.com	hero	150	200	959	191800
107	10208	Jardine	jard@gmail.com	castle lite	180	450	878	395100
179	10280	Morgan	morganny@gmail.com	hero	150	200	777	155400
191	10292	Kivell	kivel_go@yahoo.com	castle lite	180	450	840	378000

In [25]:

```
df_minProfit['MONTHS'].unique()
```

Out[25]:

```
array(['December'], dtype=object)
```

In [26]:

```
df_minProfit.groupby('MONTHS')['PROFIT'].min()
```

Out[26]:

MONTHS	
December	38150

Name: PROFIT, dtype: int64

In [32]:

```
df_minProfit['PROFIT'].min()
```

Out[32]:

```
38150
```

In [33]:

```
dec= df_minProfit[(df_minProfit.PROFIT==38150)]
dec
```

Out[33]:

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	P
1019	11120	Jones	jone.ai@yahoo.com	hero	150	200	763	152600	

In []:

In [31]:

```
r=df_minProfit['PROFIT']==38150
min_dec=df_minProfit[r]
min_dec
```

Out[31]:

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	P
1019	11120	Jones	jone.ai@yahoo.com	hero	150	200	763	152600	

```
df2019.head()
```

```
Out[37]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	I
0	10101	Jardine	jard@gmail.com	trophy	150	200	725	145000	
5	10106	Jardine	jard@gmail.com	beta malt	80	150	798	119700	
8	10109	Morgan	morganny@gmail.com	budweiser	250	500	700	350000	
11	10112	Jones	jone.ai@yahoo.com	hero	150	200	902	180400	
15	10116	Jones	jone.ai@yahoo.com	budweiser	250	500	709	354500	

```
In [38]: sum_profit = df2019['PROFIT'].sum()  
sum_profit
```

```
Out[38]: 30020250
```

```
In [39]: dfp= df2019.groupby('MONTHS')['PROFIT'].sum().sort_values(ascending=False)  
dfp
```

```
Out[39]: MONTHS  
January      3263160  
August       2982800  
July         2945340  
April        2851470  
November     2675610  
June         2669080  
May          2573040  
March        2530620  
October      2220870  
December     2048780  
September    1892600  
February     1366880  
Name: PROFIT, dtype: int64
```

```
In [40]: #Another method usind dictionary direct  
for k,v in dfp.items():  
    percent= round((v*100)/sum_profit,2)  
    print(f'the percent of {k} is {percent}%')
```

```
the percent of January is 10.87%  
the percent of August is 9.94%  
the percent of July is 9.81%  
the percent of April is 9.5%  
the percent of November is 8.91%  
the percent of June is 8.89%  
the percent of May is 8.57%  
the percent of March is 8.43%  
the percent of October is 7.4%  
the percent of December is 6.82%  
the percent of September is 6.3%  
the percent of February is 4.55%
```

```
In [41]: vc= dfp.reset_index()  
vc
```

Out[41]:

	MONTHS	PROFIT
0	January	3263160
1	August	2982800
2	July	2945340
3	April	2851470
4	November	2675610
5	June	2669080
6	May	2573040
7	March	2530620
8	October	2220870
9	December	2048780
10	September	1892600
11	February	1366880

In [43]:

```
dvc=vc['MONTHS'].unique()
dvc
```

Out[43]:

```
array(['January', 'August', 'July', 'April', 'November', 'June', 'May',
      'March', 'October', 'December', 'September', 'February'],
      dtype=object)
```

In [44]:

```
ap=[]
for i in dfp:
    percent2= round((i*100)/sum_profit,2)
    ap.append(percent2)
print(ap)
```

```
[10.87, 9.94, 9.81, 9.5, 8.91, 8.89, 8.57, 8.43, 7.4, 6.82, 6.3, 4.55]
```

In [45]:

```
zipp= dict(zip(dvc,ap))
print(zipp)
for i,n in zipp.items():
    print(f'the percent of {i} is {n}%')
```

```
{'January': 10.87, 'August': 9.94, 'July': 9.81, 'April': 9.5, 'November': 8.91, 'June':
8.89, 'May': 8.57, 'March': 8.43, 'October': 7.4, 'December': 6.82, 'September': 6.3, 'F
ebruary': 4.55}
the percent of January is 10.87%
the percent of August is 9.94%
the percent of July is 9.81%
the percent of April is 9.5%
the percent of November is 8.91%
the percent of June is 8.89%
the percent of May is 8.57%
the percent of March is 8.43%
the percent of October is 7.4%
the percent of December is 6.82%
the percent of September is 6.3%
the percent of February is 4.55%
```

In []:

In [34]:

```
#8. Which particular brand generated the highest profit in Senegal?
senGal= df[(df.COUNTRIES=='Senegal')]
senGal.head()
```

```
Out[34]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	C
	4	10105	Andrews	andy@gmail.com	hero	150	200	836 167
	9	10110	Howard	howard_freeman@yahoo.com	castle lite	180	450	745 335
	14	10115	Morgan	morganny@gmail.com	trophy	150	200	939 187
	19	10120	Parent	parentty@uk.com	beta malt	80	150	731 109
	24	10125	Jardine	jard@gmail.com	eagle lager	170	250	715 178

```
In [35]: senGal['COUNTRIES'].unique()
```

```
Out[35]: array(['Senegal'], dtype=object)
```

```
In [36]: brands=senGal.groupby('BRANDS')['PROFIT'].sum().sort_values(ascending=False)
brands
```

```
Out[36]: BRANDS
castle lite    7012980
budweiser      6480750
eagle lager    2061680
beta malt      1793750
grand malt     1525680
hero           1331600
trophy         1278750
Name: PROFIT, dtype: int64
```

```
In [37]: print('CASTLE LITE generated the Highest profit in Senegal')
```

CASTLE LITE generated the Highest profit in Senegal

BRAND ANALYSIS

```
In [38]: # 1. Within the last two years, the brand manager wants to know the top three brands
# consumed in the francophone countries

df1.head()
```

```
Out[38]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PI
	2	10103	Sorvino	sorvi2000@gmail.com	castle lite	180	450	937 421650	2
	3	10104	Jones	jone.ai@yahoo.com	eagle lager	170	250	765 191250	
	4	10105	Andrews	andy@gmail.com	hero	150	200	836 167200	
	7	10108	Jones	jone.ai@yahoo.com	trophy	150	200	812 162400	
	8	10109	Morgan	morganny@gmail.com	budweiser	250	500	700 350000	1

```
In [39]: df_1819= df1[(df1.YEARS==2018) | (df1.YEARS==2019)]
df_1819.head()
```


Out[39]:	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST
	2	10103	Sorvino	sorvi2000@gmail.com	castle lite	180	450	937 421650
	3	10104	Jones	jone.ai@yahoo.com	eagle lager	170	250	765 191250
	7	10108	Jones	jone.ai@yahoo.com	trophy	150	200	812 162400
	8	10109	Morgan	morganny@gmail.com	budweiser	250	500	700 350000
	12	10113	Smith	smithMan@yahoo.com	beta malt	80	150	731 109650

```
In [40]: print(df_1819.groupby('BRANDS')['QUANTITY'].sum().sort_values(ascending=False).head(3))
print('The Top three products consumed by these countries are TROPHY,HERO,EAGLE LAGER')

BRANDS
trophy      52899
hero        50846
eagle lager  50630
Name: QUANTITY, dtype: int64
The Top three products consumed by these countries are TROPHY,HERO,EAGLE LAGER
```

```
In [41]: # 2.Find out the top two choice of consumer brands in Ghana
df_ghana= df2[(df2.COUNTRIES=='Ghana')]
df_ghana.head()
```

Out[41]:	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PR
	0	10101	Jardine	jard@gmail.com	trophy	150	200	725 145000	3
	5	10106	Jardine	jard@gmail.com	beta malt	80	150	798 119700	5
	10	10111	Parent	parentty@uk.com	eagle lager	170	250	861 215250	6
	15	10116	Jones	jone.ai@yahoo.com	budweiser	250	500	709 354500	17
	20	10121	Gill	gillhell@uk.com	grand malt	90	150	898 134700	5

```
In [42]: df_ghana['COUNTRIES'].unique()
```

```
Out[42]: array(['Ghana'], dtype=object)
```

```
In [43]: print(df_ghana.groupby('BRANDS')['QUANTITY'].sum().sort_values(ascending=False).head(2))
print('Two top choice brand in Ghana are EAGLE LAGER and CASTLE LITE')

BRANDS
eagle lager    25829
castle lite    25806
Name: QUANTITY, dtype: int64
Two top choice brand in Ghana are EAGLE LAGER and CASTLE LITE
```

```
In [44]: # 3. Find out the details of beers consumed in the past three years in the most oil reac
# country in West Africa.

df_Nigeria= df2[(df2.COUNTRIES=='Nigeria')]
df_Nigeria.head()
```

Out[44]:		SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST
	1	10102	Gill	gillhell@uk.com	budweiser	250	500	815	407500
	6	10107	Thompson	thomp@uk.com	grand malt	90	150	954	143100
	11	10112	Jones	jone.ai@yahoo.com	hero	150	200	902	180400
	16	10117	Parent	parentty@uk.com	castle lite	180	450	837	376650
	21	10122	Smith	smithMan@yahoo.com	trophy	150	200	860	172000

In [45]: `df_Nigeria['BRANDS'].unique()`

Out[45]: `array(['budweiser', 'grand malt', 'hero', 'castle lite', 'trophy', 'beta malt', 'eagle lager'], dtype=object)`

In [46]: `bear_Details=df_Nigeria[(df_Nigeria.BRANDS=='budweiser')|(df_Nigeria.BRANDS=='hero')|(df_Nigeria.BRANDS=='castle lite')|(df_Nigeria.BRANDS=='trophy')|(df_Nigeria.BRANDS=='eagle lager')]`
`bear_Details.head()`

Out[46]:		SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST
	1	10102	Gill	gillhell@uk.com	budweiser	250	500	815	407500
	11	10112	Jones	jone.ai@yahoo.com	hero	150	200	902	180400
	16	10117	Parent	parentty@uk.com	castle lite	180	450	837	376650
	21	10122	Smith	smithMan@yahoo.com	trophy	150	200	860	172000
	31	10132	Morgan	morganny@gmail.com	eagle lager	170	250	826	206500

In [47]: `bear_Details['BRANDS'].unique()`

Out[47]: `array(['budweiser', 'hero', 'castle lite', 'trophy', 'eagle lager'], dtype=object)`

In [48]: `bear_Details.groupby('BRANDS')['QUANTITY'].sum().sort_values(ascending=False)`

Out[48]: `BRANDS`
`budweiser 26153`
`eagle lager 25872`
`hero 25811`
`trophy 25743`
`castle lite 25681`
`Name: QUANTITY, dtype: int64`

In [49]: `# 4. Favorites malt brand in Anglophone region between 2018 and 2019`
`malt_brand= df2[(df2.BRANDS=='beta malt')|(df2.BRANDS=='grand malt')]`
`malt_brand.head()`

Out[49]:		SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST
	5	10106	Jardine	jard@gmail.com	beta malt	80	150	798	119700
	6	10107	Thompson	thomp@uk.com	grand malt	90	150	954	143100
	20	10121	Gill	gillhell@uk.com	grand malt	90	150	898	134700
	26	10127	Howard	howard_freeman@yahoo.com	beta malt	80	150	982	147300
	40	10141	Andrews	andy@gmail.com	beta malt	80	150	894	134100

In [50]: `malt_brand['BRANDS'].unique()`

```
Out[50]: array(['beta malt', 'grand malt'], dtype=object)
```

```
In [51]: malt_1819= malt_brand[(malt_brand.YEARS==2018)|(malt_brand.YEARS==2019)]
malt_1819.head()
```

```
Out[51]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT
5	10106	Jardine	jard@gmail.com	beta malt	80	150	798	119700	59850
40	10141	Andrews	andy@gmail.com	beta malt	80	150	894	134100	67050
61	10162	Gill	gillhell@uk.com	beta malt	80	150	931	139650	69825
90	10191	Jardine	jard@gmail.com	grand malt	90	150	902	135300	67650
96	10197	Jones	jone.ai@yahoo.com	beta malt	80	150	896	134400	67200

```
In [52]: malt_1819['YEARS'].unique()
```

```
Out[52]: array([2019, 2018], dtype=int64)
```

```
In [53]: print(malt_1819.groupby('BRANDS')['QUANTITY'].sum().sort_values(ascending=False))

""Grand Malt is the Favourite Malt brand in Anglophone countries between 2018 and 2019""

BRANDS
grand malt    33221
beta malt     32118
Name: QUANTITY, dtype: int64
```

```
Out[53]: 'Grand Malt is the Favourite Malt brand in Anglophone countries between 2018 and 2019'
```

```
In [54]: #5. Which brands sold the highest in 2019 in Nigeria?
df_Nigeria.head()
```

```
Out[54]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT
1	10102	Gill	gillhell@uk.com	budweiser	250	500	815	407500	203750
6	10107	Thompson	thomp@uk.com	grand malt	90	150	954	143100	71550
11	10112	Jones	jone.ai@yahoo.com	hero	150	200	902	180400	90200
16	10117	Parent	parentty@uk.com	castle lite	180	450	837	376650	188325
21	10122	Smith	smithMan@yahoo.com	trophy	150	200	860	172000	86000

```
In [55]: df_Nigeria['YEARS'].unique()
```

```
Out[55]: array([2018, 2017, 2019], dtype=int64)
```

```
In [56]: Nig_2019= df_Nigeria[(df_Nigeria.YEARS==2019)]
Nig_2019.head(10)
```

Out[56]:	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	
	11	10112	Jones	jone.ai@yahoo.com	hero	150	200	902 15
	51	10152	Parent	parentty@uk.com	castle lite	180	450	878 35
	66	10167	Andrews	andy@gmail.com	eagle lager	170	250	769 15
	81	10182	Andrews	andy@gmail.com	hero	150	200	773 15
	91	10192	Jardine	jard@gmail.com	trophy	150	200	981 15
	96	10197	Jones	jone.ai@yahoo.com	beta malt	80	150	896 15
	101	10202	Morgan	morganny@gmail.com	eagle lager	170	250	828 20
	106	10207	Jones	jone.ai@yahoo.com	budweiser	250	500	990 45
	116	10217	Howard	howard_freeman@yahoo.com	hero	150	200	957 15
	131	10232	Jardine	jard@gmail.com	beta malt	80	150	934 15

In [57]: Nig_2019['YEARS'].unique()

Out[57]: array([2019], dtype=int64)

In [58]: print(Nig_2019.groupby('BRANDS')['QUANTITY'].sum().sort_values(ascending=False))

""""HERO brand sold more in Nigeria at 2019""""

BRANDS

hero 9622

eagle lager 8401

beta malt 8389

grand malt 8125

trophy 5803

budweiser 5490

castle lite 3388

Name: QUANTITY, dtype: int64

Out[58]: 'HERO brand sold more in Nigeria at 2019'

In [59]: #6. Favorites brand in South_South region in Nigeria

df_Nigeria.head()

Out[59]:	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST
	1	10102	Gill	gillhell@uk.com	budweiser	250	500	815 407500
	6	10107	Thompson	thomp@uk.com	grand malt	90	150	954 143100
	11	10112	Jones	jone.ai@yahoo.com	hero	150	200	902 180400
	16	10117	Parent	parentty@uk.com	castle lite	180	450	837 376650
	21	10122	Smith	smithMan@yahoo.com	trophy	150	200	860 172000

In [60]: df_Nigeria.columns

Out[60]: Index(['SALES_ID', 'SALES_REP', 'EMAILS', 'BRANDS', 'PLANT_COST', 'UNIT_PRICE', 'QUANTITY', 'COST', 'PROFIT', 'COUNTRIES', 'REGION', 'MONTHS', 'YEARS'], dtype='object')

In [61]: df_Nigeria['REGION'].unique()

```
Out[61]: array(['west', 'Southeast', 'northcentral ', 'northeast ', 'northwest',  
        'southsouth'], dtype=object)
```

```
In [62]: Nig_Sregion= df_Nigeria[(df_Nigeria.REGION=='southsouth')]  
Nig_Sregion.head()
```

```
Out[62]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	C
	26	10127	Howard	howard_freeman@yahoo.com	beta malt	80	150	982 14
	56	10157	Jones	jone.ai@yahoo.com	trophy	150	200	920 18
	86	10187	Gill	gillhell@uk.com	castle lite	180	450	879 35
	116	10217	Howard	howard_freeman@yahoo.com	hero	150	200	957 19
	146	10247	Jardine	jard@gmail.com	grand malt	90	150	754 11

```
In [63]: print(Nig_Sregion.groupby('BRANDS')['QUANTITY'].sum().sort_values(ascending=False))  
  
        """The favourite brand sold in Nigeria is Eagle Lager"""
```

```
BRANDS  
eagle lager    4551  
trophy         4468  
hero           4456  
budweiser      4328  
castle lite    4287  
beta malt      4257  
grand malt     3914
```

```
Name: QUANTITY, dtype: int64
```

```
Out[63]: 'The favourite brand sold in Nigeria is Eagle Lager'
```

```
In [64]: # 7. Bear consumption in Nigeria  
  
bear_Details['COUNTRIES'].unique()
```

```
Out[64]: array(['Nigeria'], dtype=object)
```

```
In [65]: bear_Details.groupby('BRANDS')['QUANTITY'].sum().sort_values(ascending=False)
```

```
Out[65]: BRANDS  
budweiser    26153  
eagle lager  25872  
hero         25811  
trophy       25743  
castle lite  25681  
Name: QUANTITY, dtype: int64
```

```
In [66]: # 8. Level of consumption of Budweiser in the regions in Nigeria  
  
df_Nigeria['BRANDS'].unique()
```

```
Out[66]: array(['budweiser', 'grand malt', 'hero', 'castle lite', 'trophy',  
        'beta malt', 'eagle lager'], dtype=object)
```

```
In [67]: bud_consump= df_Nigeria[(df_Nigeria.BRANDS=='budweiser')]  
bud_consump.head()
```

```
Out[67]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PI
1	10102	Gill	gillhell@uk.com	budweiser	250	500	815	407500	2
36	10137	Thompson	thomp@uk.com	budweiser	250	500	821	410500	2
71	10172	Jones	jone.ai@yahoo.com	budweiser	250	500	869	434500	2
106	10207	Jones	jone.ai@yahoo.com	budweiser	250	500	990	495000	2
141	10242	Gill	gillhell@uk.com	budweiser	250	500	975	487500	2

```
In [68]: bud_consump['BRANDS'].unique()
```

```
Out[68]: array(['budweiser'], dtype=object)
```

```
In [69]: bud_consump.groupby('BRANDS')['QUANTITY'].sum()
```

```
Out[69]: BRANDS
budweiser    26153
Name: QUANTITY, dtype: int64
```

```
In [70]: # 9. Level of consumption of Budweiser in the regions in Nigeria in 2019 (Decision on Pr
Nig_2019.head()
```

```
Out[70]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PRC
11	10112	Jones	jone.ai@yahoo.com	hero	150	200	902	180400	45
51	10152	Parent	parentty@uk.com	castle lite	180	450	878	395100	237
66	10167	Andrews	andy@gmail.com	eagle lager	170	250	769	192250	63
81	10182	Andrews	andy@gmail.com	hero	150	200	773	154600	38
91	10192	Jardine	jard@gmail.com	trophy	150	200	981	196200	49

```
In [71]: Nig_2019['YEARS'].unique()
```

```
Out[71]: array([2019], dtype=int64)
```

```
In [72]: Nig_Budweiser= Nig_2019[(Nig_2019.BRANDS=='budweiser')]
Nig_Budweiser.head()
```

```
Out[72]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PRC
106	10207	Jones	jone.ai@yahoo.com	budweiser	250	500	990	495000	2
456	10557	Gill	gillhell@uk.com	budweiser	250	500	902	451000	2
561	10662	Howard	howard_freeman@yahoo.com	budweiser	250	500	948	474000	2
631	10732	Jones	jone.ai@yahoo.com	budweiser	250	500	884	442000	2
876	10977	Gill	gillhell@uk.com	budweiser	250	500	919	459500	2

```
In [73]: Nig_Budweiser['BRANDS'].unique()
```

```
Out[73]: array(['budweiser'], dtype=object)
```

```
In [74]: Nig_Budweiser.groupby('BRANDS')['QUANTITY'].sum().sort_values(ascending=False)
```

```
Out[74]: BRANDS
budweiser    5490
Name: QUANTITY, dtype: int64
```

```
In [ ]:
```

COUNTRY ANALYSIS

```
In [75]: # 1. Country with the highest consumption of beer.
df.head()
```

Out[75]:

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PF
0	10101	Jardine	jard@gmail.com	trophy	150	200	725	145000	:
1	10102	Gill	gillhell@uk.com	budweiser	250	500	815	407500	20
2	10103	Sorvino	sorvi2000@gmail.com	castle lite	180	450	937	421650	20
3	10104	Jones	jone.ai@yahoo.com	eagle lager	170	250	765	191250	0
4	10105	Andrews	andy@gmail.com	hero	150	200	836	167200	:

```
In [76]: beer_brands= df[(df.BRANDS=='budweiser')|(df.BRANDS=='hero')|(df.BRANDS=='castle lite')|
beer_brands.head()
```

Out[76]:

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PF
0	10101	Jardine	jard@gmail.com	trophy	150	200	725	145000	:
1	10102	Gill	gillhell@uk.com	budweiser	250	500	815	407500	20
2	10103	Sorvino	sorvi2000@gmail.com	castle lite	180	450	937	421650	20
3	10104	Jones	jone.ai@yahoo.com	eagle lager	170	250	765	191250	0
4	10105	Andrews	andy@gmail.com	hero	150	200	836	167200	:

```
In [77]: beer_brands['COUNTRIES'].unique()
```

```
Out[77]: array(['Ghana', 'Nigeria', 'Togo', 'Benin', 'Senegal'], dtype=object)
```

```
In [78]: bear_countries =beer_brands[(beer_brands.COUNTRIES=='Ghana')|(beer_brands.COUNTRIES=='Ni
bear_countries.head()
```

Out[78]:

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PF
0	10101	Jardine	jard@gmail.com	trophy	150	200	725	145000	:
1	10102	Gill	gillhell@uk.com	budweiser	250	500	815	407500	20
2	10103	Sorvino	sorvi2000@gmail.com	castle lite	180	450	937	421650	20
3	10104	Jones	jone.ai@yahoo.com	eagle lager	170	250	765	191250	0
4	10105	Andrews	andy@gmail.com	hero	150	200	836	167200	:

```
In [79]: bear_countries.groupby('COUNTRIES')['QUANTITY'].sum().sort_values(ascending=False)
```

```
Out[79]: COUNTRIES
        Senegal    129875
        Nigeria    129260
        Benin      127455
        Ghana      127232
        Togo       125548
        Name: QUANTITY, dtype: int64
```

```
In [80]: # 2.Highest sales personnel of Budweiser in Senegal
```

```
senegal= df[(df.BRANDS=='budweiser')& (df.COUNTRIES=='Senegal')]
senegal.head()
```

```
Out[80]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST
29	10130	Kivell	kivel_go@yahoo.com	budweiser	250	500	945	472500
64	10165	Sorvino	sorvi2000@gmail.com	budweiser	250	500	780	390000
99	10200	Thompson	thomp@uk.com	budweiser	250	500	729	364500
134	10235	Gill	gillhell@uk.com	budweiser	250	500	966	483000
169	10270	Smith	smithMan@yahoo.com	budweiser	250	500	863	431500

```
In [81]: senegal['COUNTRIES'].unique()
```

```
Out[81]: array(['Senegal'], dtype=object)
```

```
In [82]: senegal['SALES_REP'].unique()
```

```
Out[82]: array(['Kivell', 'Sorvino', 'Thompson', 'Gill', 'Smith', 'Andrews',
                'Parent', 'Jardine', 'Jones', 'Howard', 'Morgan'], dtype=object)
```

```
In [83]: senegal.groupby('SALES_REP')['QUANTITY'].sum().sort_values(ascending=False)
```

```
Out[83]: SALES_REP
        Jones      5917
        Sorvino    4246
        Andrews    3336
        Kivell     2813
        Jardine    2507
        Parent     1842
        Smith      1673
        Gill       966
        Howard     952
        Morgan     942
        Thompson    729
        Name: QUANTITY, dtype: int64
```

```
In [84]: # 3. Country with the highest profit of the fourth quarter in 2019
```

CLASS WORK

```
In [85]: # 1.country with the highest profit in 2018 using gmail
        # 2.the profit of yahoo mail in the last 2 years in francophone countries
        # 3.highest sales personal of budweiser in senegal using uk.com mail
        # 4.which brand that the uses yahoomail sold the highest in 2019 in southsouth nigeria
        # 5.profit of yahoo mail in december 2019
        # 6.the profit of gmail in the Anglophone countries in the last two years
        # 7.display the dataframe that shows all the columns for the minimum profit
```



```
Out[85]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PF
0	10101	Jardine	jard@gmail.com	trophy	150	200	725	145000	:
1	10102	Gill	gillhell@uk.com	budweiser	250	500	815	407500	20
2	10103	Sorvino	sorvi2000@gmail.com	castle lite	180	450	937	421650	25
3	10104	Jones	jone.ai@yahoo.com	eagle lager	170	250	765	191250	6
4	10105	Andrews	andy@gmail.com	hero	150	200	836	167200	4

```
In [86]: dff= df.copy()
dff.head()
```

```
Out[86]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PF
0	10101	Jardine	jard@gmail.com	trophy	150	200	725	145000	:
1	10102	Gill	gillhell@uk.com	budweiser	250	500	815	407500	20
2	10103	Sorvino	sorvi2000@gmail.com	castle lite	180	450	937	421650	25
3	10104	Jones	jone.ai@yahoo.com	eagle lager	170	250	765	191250	6
4	10105	Andrews	andy@gmail.com	hero	150	200	836	167200	4

```
In [87]: dff[['EMAILS','MAILINK']] = dff['EMAILS'].str.split('@',expand=True)
dff.head()
```

```
Out[87]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT	CO
0	10101	Jardine	jard	trophy	150	200	725	145000	36250	
1	10102	Gill	gillhell	budweiser	250	500	815	407500	203750	
2	10103	Sorvino	sorvi2000	castle lite	180	450	937	421650	252990	
3	10104	Jones	jone.ai	eagle lager	170	250	765	191250	61200	
4	10105	Andrews	andy	hero	150	200	836	167200	41800	

```
In [88]: dff['MAILINK'].unique()
```

```
Out[88]: array(['gmail.com', 'uk.com', 'yahoo.com'], dtype=object)
```

```
In [89]: # 1.country with the highest profit in 2018 using gmail
dff_2018=dff[(df.YEARS==2018)]
dff_2018.head()
```

```
Out[89]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT	CC
1	10102	Gill	gillhell	budweiser	250	500	815	407500	203750	
2	10103	Sorvino	sorvi2000	castle lite	180	450	937	421650	252990	
3	10104	Jones	jone.ai	eagle lager	170	250	765	191250	61200	
7	10108	Jones	jone.ai	trophy	150	200	812	162400	40600	
12	10113	Smith	smithMan	beta malt	80	150	731	109650	51170	

```
In [90]: dff_2018['MAILINK'].unique()
```

Out[90]: array(['uk.com', 'gmail.com', 'yahoo.com'], dtype=object)

```
In [91]: gmail18= dff_2018[(dff_2018.MAILINK=='gmail.com')]  
gmail18.head()
```

Out[91]:

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT	CO
2	10103	Sorvino	sorvi2000	castle lite	180	450	937	421650	252990	
14	10115	Morgan	morganny	trophy	150	200	939	187800	46950	
24	10125	Jardine	jard	eagle lager	170	250	715	178750	57200	
25	10126	Andrews	andy	hero	150	200	999	199800	49950	
33	10134	Sorvino	sorvi2000	beta malt	80	150	804	120600	56280	

```
In [92]: gmail18.groupby('COUNTRIES')['PROFIT'].sum().sort_values(ascending=False).head(1)
```

Out[92]: COUNTRIES
Togo 4182880
Name: PROFIT, dtype: int64

```
In [93]: dff['COUNTRIES'].unique()
```

Out[93]: array(['Ghana', 'Nigeria', 'Togo', 'Benin', 'Senegal'], dtype=object)

```
In [94]: # 2.the profit of yahoo mail in the last 2 years in francophone countries  
dff1=dff[(dff.COUNTRIES=='Togo')|(dff.COUNTRIES=='Benin')|(dff.COUNTRIES=='Senegal')]  
dff1.head()
```

Out[94]:

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT	COI
2	10103	Sorvino	sorvi2000	castle lite	180	450	937	421650	252990	
3	10104	Jones	jone.ai	eagle lager	170	250	765	191250	61200	
4	10105	Andrews	andy	hero	150	200	836	167200	41800	
7	10108	Jones	jone.ai	trophy	150	200	812	162400	40600	
8	10109	Morgan	morganny	budweiser	250	500	700	350000	175000	

```
In [95]: dff_1819= dff1[(dff1.YEARS==2018)|(dff1.YEARS==2019)]  
dff_1819.head()
```

Out[95]:

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT	CC
2	10103	Sorvino	sorvi2000	castle lite	180	450	937	421650	252990	
3	10104	Jones	jone.ai	eagle lager	170	250	765	191250	61200	
7	10108	Jones	jone.ai	trophy	150	200	812	162400	40600	
8	10109	Morgan	morganny	budweiser	250	500	700	350000	175000	
12	10113	Smith	smithMan	beta malt	80	150	731	109650	51170	

```
In [96]: dff_1819['COUNTRIES'].unique()
```

Out[96]: array(['Togo', 'Benin', 'Senegal'], dtype=object)

```
In [97]: yahoo1819= dff_1819[(dff_1819.MAILINK=='yahoo.com')]
```

```
yahoo1819.head()
```

```
Out[97]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT	CO
	3	10104	Jones	jone.ai	eagle lager	170	250	765	191250	61200
	7	10108	Jones	jone.ai	trophy	150	200	812	162400	40600
	12	10113	Smith	smithMan	beta malt	80	150	731	109650	51170
	17	10118	Kivell	kivel_go	eagle lager	170	250	910	227500	72800
	18	10119	Smith	smithMan	hero	150	200	996	199200	49800

```
In [98]: yahoo1819["MAILINK"].unique()
```

```
Out[98]: array(['yahoo.com'], dtype=object)
```

```
In [99]: yahoo1819.groupby('COUNTRIES')['PROFIT'].sum().sort_values(ascending=False)
```

```
Out[99]: COUNTRIES
Benin      5418730
Senegal    5254690
Togo       3886870
Name: PROFIT, dtype: int64
```

```
In [100]: dff["MAILINK"].unique()
```

```
Out[100]: array(['gmail.com', 'uk.com', 'yahoo.com'], dtype=object)
```

```
In [101]: # 3.highest sales personal of budweiser in senegal using uk.com mail
ukmail= dff[(dff.MAILINK=='uk.com')]
ukmail.head(7)
```

```
Out[101]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT	CC
	1	10102	Gill	gillhell	budweiser	250	500	815	407500	203750
	6	10107	Thompson	thomp	grand malt	90	150	954	143100	57240
	10	10111	Parent	parentty	eagle lager	170	250	861	215250	68880
	16	10117	Parent	parentty	castle lite	180	450	837	376650	225990
	19	10120	Parent	parentty	beta malt	80	150	731	109650	51170
	20	10121	Gill	gillhell	grand malt	90	150	898	134700	53880
	27	10128	Gill	gillhell	grand malt	90	150	920	138000	55200

```
In [102]: ukmail["MAILINK"].unique()
```

```
Out[102]: array(['uk.com'], dtype=object)
```

```
In [103]: senegal_Uk= ukmail[(ukmail.COUNTRIES=='Senegal')]
senegal_Uk.head()
```

Out[103]:	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT	CC
19	10120	Parent	parentty	beta malt	80	150	731	109650	51170	
34	10135	Gill	gillhell	grand malt	90	150	890	133500	53400	
69	10170	Gill	gillhell	grand malt	90	150	940	141000	56400	
94	10195	Gill	gillhell	eagle lager	170	250	910	227500	72800	
99	10200	Thompson	thomp	budweiser	250	500	729	364500	182250	

```
In [104... SenBudweiser= senegal_Uk[(senegal_Uk.BRANDS=='budweiser')]
SenBudweiser.head()
```

Out[104]:	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT	C
99	10200	Thompson	thomp	budweiser	250	500	729	364500	182250	
134	10235	Gill	gillhell	budweiser	250	500	966	483000	241500	
274	10375	Parent	parentty	budweiser	250	500	928	464000	232000	
554	10655	Parent	parentty	budweiser	250	500	914	457000	228500	

```
In [105... SenBudweiser['SALES_REP'].unique()
```

```
Out[105]: array(['Thompson', 'Gill', 'Parent'], dtype=object)
```

```
In [106... SenBudweiser.groupby('SALES_REP')['QUANTITY'].sum().sort_values(ascending=False).head(1)
```

```
Out[106]: SALES_REP
Parent    1842
Name: QUANTITY, dtype: int64
```

```
In [107... # 4.which brand that the uses yahoomail sold the highest in 2019 in southsouth nigeria
yahoo=dff[(dff.MAILINK=='yahoo.com')]
yahoo.head()
```

Out[107]:	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROI
3	10104	Jones	jone.ai	eagle lager	170	250	765	191250	612
7	10108	Jones	jone.ai	trophy	150	200	812	162400	406
9	10110	Howard	howard_freeman	castle lite	180	450	745	335250	2011
11	10112	Jones	jone.ai	hero	150	200	902	180400	451
12	10113	Smith	smithMan	beta malt	80	150	731	109650	511

```
In [108... yahoo_2019= yahoo[(yahoo.YEARS==2019)&(yahoo.REGION=='southsouth')&(yahoo.COUNTRIES=='Ni
yahoo_2019.head()
```

```
Out[108]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PRC
116	10217	Howard	howard_freeman	hero	150	200	957	191400	47
236	10337	Jones	jone.ai	beta malt	80	150	827	124050	57
476	10577	Jones	jone.ai	trophy	150	200	724	144800	36
746	10847	Jones	jone.ai	hero	150	200	932	186400	46
926	11027	Kivell	kivel_go	castle lite	180	450	847	381150	228

```
In [109... yahoo_2019.groupby('BRANDS')['QUANTITY'].sum().sort_values(ascending=False).head(4)
```

```
Out[109]:
```

BRANDS	
hero	2760
budweiser	847
castle lite	847
beta malt	827

Name: QUANTITY, dtype: int64

```
In [110... # 5.profit of yahoo mail in december 2019
yah19= yahoo[(yahoo.MONTHS=='December')&(yahoo.YEARS==2019)]
yah19.head()
```

```
Out[110]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT
11	10112	Jones	jone.ai	hero	150	200	902	180400	45100
119	10220	Smith	smithMan	trophy	150	200	917	183400	45850
155	10256	Jones	jone.ai	budweiser	250	500	909	454500	227250
227	10328	Jones	jone.ai	eagle lager	170	250	706	176500	56480
671	10772	Kivell	kivel_go	grand malt	90	150	804	120600	48240

```
In [111... yah19['PROFIT'].sum()
```

```
Out[111]: 662410
```

```
In [114... dff.head()
```

```
Out[114]:
```

	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT	CC
0	10101	Jardine	jard	trophy	150	200	725	145000	36250	
1	10102	Gill	gillhell	budweiser	250	500	815	407500	203750	
2	10103	Sorvino	sorvi2000	castle lite	180	450	937	421650	252990	
3	10104	Jones	jone.ai	eagle lager	170	250	765	191250	61200	
4	10105	Andrews	andy	hero	150	200	836	167200	41800	

```
In [119... # 6.the profit of gmail in the Anglophone countries in the last two years

dff2= dff[(dff.COUNTRIES=='Ghana')|(dff.COUNTRIES=='Nigeria')]
dff2.head()
```

Out[119]:	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT	CC
0	10101	Jardine	jard	trophy	150	200	725	145000	36250	
1	10102	Gill	gillhell	budweiser	250	500	815	407500	203750	
5	10106	Jardine	jard	beta malt	80	150	798	119700	55860	
6	10107	Thompson	thomp	grand malt	90	150	954	143100	57240	
10	10111	Parent	parentty	eagle lager	170	250	861	215250	68880	

```
In [131... dff2_1819=dff2[(dff2.YEARS==2018)|(dff2.YEARS==2019)]
dff2_1819.head()
```

Out[131]:	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT	CC
0	10101	Jardine	jard	trophy	150	200	725	145000	36250	
1	10102	Gill	gillhell	budweiser	250	500	815	407500	203750	
5	10106	Jardine	jard	beta malt	80	150	798	119700	55860	
11	10112	Jones	jone.ai	hero	150	200	902	180400	45100	
15	10116	Jones	jone.ai	budweiser	250	500	709	354500	177250	

```
In [121... dff2_1819['YEARS'].unique()
```

```
Out[121]: array([2019, 2018], dtype=int64)
```

```
In [133... dff2_gmail=dff2_1819[(dff2_1819.MAILINK=='gmail.com')]
dff2_gmail.head()
```

Out[133]:	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT	CO
0	10101	Jardine	jard	trophy	150	200	725	145000	36250	
5	10106	Jardine	jard	beta malt	80	150	798	119700	55860	
25	10126	Andrews	andy	hero	150	200	999	199800	49950	
40	10141	Andrews	andy	beta malt	80	150	894	134100	62580	
66	10167	Andrews	andy	eagle lager	170	250	769	192250	61520	

```
In [134... dff2_gmail['MAILINK'].unique()
```

```
Out[134]: array(['gmail.com'], dtype=object)
```

```
In [132... dff2_gmail['PROFIT'].sum()
```

```
Out[132]: 9091730
```

```
In [136... # 7.display the dataframe that shows all the columns for the minimum profit
dff.head()
```

Out[136]:	SALES_ID	SALES_REP	EMAILS	BRANDS	PLANT_COST	UNIT_PRICE	QUANTITY	COST	PROFIT	CC
0	10101	Jardine	jard	trophy	150	200	725	145000	36250	
1	10102	Gill	gillhell	budweiser	250	500	815	407500	203750	
2	10103	Sorvino	sorvi2000	castle lite	180	450	937	421650	252990	
3	10104	Jones	jone.ai	eagle lager	170	250	765	191250	61200	
4	10105	Andrews	andy	hero	150	200	836	167200	41800	

In [139... `dff.groupby('COUNTRIES')['PROFIT'].min()`

Out[139]:

COUNTRIES	
Benin	35100
Ghana	35200
Nigeria	35000
Senegal	35200
Togo	35250

Name: PROFIT, dtype: int64

In [1]: `pip install nbconvert[webpdf]`

Requirement already satisfied: nbconvert[webpdf] in c:\programdata\anaconda3\lib\site-packages (6.4.4)

Requirement already satisfied: nbformat>=4.4 in c:\programdata\anaconda3\lib\site-packages (from nbconvert[webpdf]) (5.3.0)

Requirement already satisfied: nbclient<0.6.0,>=0.5.0 in c:\programdata\anaconda3\lib\site-packages (from nbconvert[webpdf]) (0.5.13)

Requirement already satisfied: jupyter-core in c:\programdata\anaconda3\lib\site-packages (from nbconvert[webpdf]) (4.9.2)

Requirement already satisfied: pandocfilters>=1.4.1 in c:\programdata\anaconda3\lib\site-packages (from nbconvert[webpdf]) (1.5.0)

Requirement already satisfied: bleach in c:\programdata\anaconda3\lib\site-packages (from nbconvert[webpdf]) (4.1.0)

Requirement already satisfied: entrypoints>=0.2.2 in c:\programdata\anaconda3\lib\site-packages (from nbconvert[webpdf]) (0.4)

Requirement already satisfied: mistune<2,>=0.8.1 in c:\programdata\anaconda3\lib\site-packages (from nbconvert[webpdf]) (0.8.4)

Requirement already satisfied: defusedxml in c:\programdata\anaconda3\lib\site-packages (from nbconvert[webpdf]) (0.7.1)

Requirement already satisfied: testpath in c:\programdata\anaconda3\lib\site-packages (from nbconvert[webpdf]) (0.5.0)

Requirement already satisfied: jupyterlab-pygments in c:\programdata\anaconda3\lib\site-packages (from nbconvert[webpdf]) (0.1.2)

Requirement already satisfied: beautifulsoup4 in c:\programdata\anaconda3\lib\site-packages (from nbconvert[webpdf]) (4.11.1)

Requirement already satisfied: pygments>=2.4.1 in c:\programdata\anaconda3\lib\site-packages (from nbconvert[webpdf]) (2.11.2)

Requirement already satisfied: jinja2>=2.4 in c:\programdata\anaconda3\lib\site-packages (from nbconvert[webpdf]) (2.11.3)

Requirement already satisfied: traitlets>=5.0 in c:\programdata\anaconda3\lib\site-packages (from nbconvert[webpdf]) (5.1.1)

Collecting pyppeteer<1.1,>=1

 Downloading pyppeteer-1.0.2-py3-none-any.whl (83 kB)

Requirement already satisfied: MarkupSafe>=0.23 in c:\programdata\anaconda3\lib\site-packages (from jinja2>=2.4->nbconvert[webpdf]) (2.0.1)

Requirement already satisfied: jupyter-client>=6.1.5 in c:\programdata\anaconda3\lib\site-packages (from nbclient<0.6.0,>=0.5.0->nbconvert[webpdf]) (6.1.12)

Requirement already satisfied: nest-asyncio in c:\programdata\anaconda3\lib\site-packages (from nbclient<0.6.0,>=0.5.0->nbconvert[webpdf]) (1.5.5)

Requirement already satisfied: tornado>=4.1 in c:\programdata\anaconda3\lib\site-packages (from jupyter-client>=6.1.5->nbclient<0.6.0,>=0.5.0->nbconvert[webpdf]) (6.1)

Requirement already satisfied: python-dateutil>=2.1 in c:\programdata\anaconda3\lib\site-packages (from jupyter-client>=6.1.5->nbclient<0.6.0,>=0.5.0->nbconvert[webpdf]) (2.8.2)

Requirement already satisfied: pyzmq>=13 in c:\programdata\anaconda3\lib\site-packages (from jupyter-client>=6.1.5->nbclient<0.6.0,>=0.5.0->nbconvert[webpdf]) (22.3.0)

Requirement already satisfied: pywin32>=1.0 in c:\programdata\anaconda3\lib\site-packages (from jupyter-core->nbconvert[webpdf]) (302)

Requirement already satisfied: fastjsonschema in c:\programdata\anaconda3\lib\site-packages (from nbformat>=4.4->nbconvert[webpdf]) (2.15.1)

Requirement already satisfied: jsonschema>=2.6 in c:\programdata\anaconda3\lib\site-packages (from nbformat>=4.4->nbconvert[webpdf]) (4.4.0)

Requirement already satisfied: attrs>=17.4.0 in c:\programdata\anaconda3\lib\site-packages (from jsonschema>=2.6->nbformat>=4.4->nbconvert[webpdf]) (21.4.0)

Requirement already satisfied: pyparsing!=0.17.0,!=0.17.1,!=0.17.2,>=0.14.0 in c:\programdata\anaconda3\lib\site-packages (from jsonschema>=2.6->nbformat>=4.4->nbconvert[webpdf]) (0.18.0)

Collecting pyee<9.0.0,>=8.1.0

 Downloading pyee-8.2.2-py2.py3-none-any.whl (12 kB)

Requirement already satisfied: urllib3<2.0.0,>=1.25.8 in c:\programdata\anaconda3\lib\site-packages (from pyppeteer<1.1,>=1->nbconvert[webpdf]) (1.26.9)Note: you may need to restart the kernel to use updated packages.

Requirement already satisfied: appdirs<2.0.0,>=1.4.3 in c:\programdata\anaconda3\lib\site-packages (from pyppeteer<1.1,>=1->nbconvert[webpdf]) (1.4.4)

Requirement already satisfied: tqdm<5.0.0,>=4.42.1 in c:\programdata\anaconda3\lib\site-


```
packages (from pyppeteer<1.1,>=1->nbconvert[webpdf]) (4.64.0)
Collecting websockets<11.0,>=10.0
  Downloading websockets-10.3-cp39-cp39-win_amd64.whl (98 kB)
Requirement already satisfied: certifi>=2021 in c:\programdata\anaconda3\lib\site-packages (from pyppeteer<1.1,>=1->nbconvert[webpdf]) (2021.10.8)
Requirement already satisfied: importlib-metadata>=1.4 in c:\programdata\anaconda3\lib\site-packages (from pyppeteer<1.1,>=1->nbconvert[webpdf]) (4.11.3)
Requirement already satisfied: zipp>=0.5 in c:\programdata\anaconda3\lib\site-packages (from importlib-metadata>=1.4->pyppeteer<1.1,>=1->nbconvert[webpdf]) (3.7.0)
Requirement already satisfied: six>=1.5 in c:\programdata\anaconda3\lib\site-packages (from python-dateutil>=2.1->jupyter-client>=6.1.5->nbclient<0.6.0,>=0.5.0->nbconvert[webpdf]) (1.16.0)
Requirement already satisfied: colorama in c:\programdata\anaconda3\lib\site-packages (from tqdm<5.0.0,>=4.42.1->pyppeteer<1.1,>=1->nbconvert[webpdf]) (0.4.4)
Requirement already satisfied: soupsieve>1.2 in c:\programdata\anaconda3\lib\site-packages (from beautifulsoup4->nbconvert[webpdf]) (2.3.1)
Requirement already satisfied: packaging in c:\programdata\anaconda3\lib\site-packages (from bleach->nbconvert[webpdf]) (21.3)
Requirement already satisfied: webencodings in c:\programdata\anaconda3\lib\site-packages (from bleach->nbconvert[webpdf]) (0.5.1)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in c:\programdata\anaconda3\lib\site-packages (from packaging->bleach->nbconvert[webpdf]) (3.0.4)
Installing collected packages: websockets, pyee, pyppeteer
Successfully installed pyee-8.2.2 pyppeteer-1.0.2 websockets-10.3
```

```
In [2]: jupyter nbconvert --to webpdf --allow-chromium-download DAY_TWO.ipynb
```

Input In [2]

```
jupyter nbconvert --to webpdf --allow-chromium-download DAY_TWO.ipynb
```

^

SyntaxError: invalid syntax

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```