Project Title - Customer Personality Analysis

The Customer Personality data Analysis is one of the best analysis to collect a information from the customer and in which data maximum information is given. We analysis some informations to get important data like customer in which product to money investing. This data set collecting from kaggle.com, This dataset contains 29 variables and 2240 observations about different customers.

Here's a brief version of the data description file.

People

ID: Customer's unique identifier

Year_Birth: Customer's birth year

Education: Customer's education level

Marital_Status: Customer's marital status

Income: Customer's yearly household income

Kidhome: Number of children in customer's household

Teenhome: Number of teenagers in customer's household

Dt_Customer: Date of customer's enrollment with the company

Recency: Number of days since customer's last purchase

Complain: 1 if customer complained in the last 2 years, 0 otherwise

Products

MntWines: Amount spent on wine in last 2 years

MntFruits: Amount spent on fruits in last 2 years

MntMeatProducts: Amount spent on meat in last 2 years

MntFishProducts: Amount spent on fish in last 2 years

MntSweetProducts: Amount spent on sweets in last 2 years

MntGoldProds: Amount spent on gold in last 2 years

Promotion

NumDealsPurchases: Number of purchases made with a discount

AcceptedCmp1: 1 if customer accepted the offer in the 1st campaign, 0 otherwise

AcceptedCmp2: 1 if customer accepted the offer in the 2nd campaign, 0 otherwise

AcceptedCmp3: 1 if customer accepted the offer in the 3rd campaign, 0 otherwise

AcceptedCmp4: 1 if customer accepted the offer in the 4th campaign, 0 otherwise

AcceptedCmp5: 1 if customer accepted the offer in the 5th campaign, 0 otherwise

Response: 1 if customer accepted the offer in the last campaign, 0 otherwise

Place

NumWebPurchases: Number of purchases made through the company's web site

NumCatalogPurchases: Number of purchases made using a catalogue

NumStorePurchases: Number of purchases made directly in stores

NumWebVisitsMonth: Number of visits to company's web site in the last month

In [1]:

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

In [2]:

```
cust_per_df = pd.read_csv("marketing_campaign_new.csv")
cust_per_df.head()
```

Out[2]:

	ID	Year_Birth	Education	Marital_Status	Income	Kidhome	Teenhome	Dt_Customer	Rec
0	5524	1957	Graduation	Single	58138.0	0	0	04-09-2012	
1	2174	1954	Graduation	Single	46344.0	1	1	08-03-2014	
2	4141	1965	Graduation	Together	71613.0	0	0	21-08-2013	
3	6182	1984	Graduation	Together	26646.0	1	0	10-02-2014	
4	5324	1981	PhD	Married	58293.0	1	0	19-01-2014	

5 rows × 29 columns

In [3]:

```
cust_per_df.shape
```

Out[3]:

```
(2240, 29)
```

In [4]:
```
cust_per_df1 = cust_per_df.copy()
cust_per_df1
```

Out[4]:

	ID	Year_Birth	Education	Marital_Status	Income	Kidhome	Teenhome	Dt_Customer
0	5524	1957	Graduation	Single	58138.0	0	0	04-09-2012
1	2174	1954	Graduation	Single	46344.0	1	1	08-03-2014
2	4141	1965	Graduation	Together	71613.0	0	0	21-08-2013
3	6182	1984	Graduation	Together	26646.0	1	0	10-02-2014
4	5324	1981	PhD	Married	58293.0	1	0	19-01-2014
...	...	...	...	...	...	...	...	...
2235	10870	1967	Graduation	Married	61223.0	0	1	13-06-2013
2236	4001	1946	PhD	Together	64014.0	2	1	10-06-2014
2237	7270	1981	Graduation	Divorced	56981.0	0	0	25-01-2014
2238	8235	1956	Master	Together	69245.0	0	1	24-01-2014
2239	9405	1954	PhD	Married	52869.0	1	1	15-10-2012

2240 rows × 29 columns

In [5]:
```
# check the null values
cust_per_df1.isna().sum()
```

Out[5]:
```
ID                      0
Year_Birth              0
Education               0
Marital_Status          0
Income                 24
Kidhome                 0
Teenhome                0
Dt_Customer             0
Recency                 0
MntWines                0
MntFruits               0
MntMeatProducts         0
MntFishProducts         0
MntSweetProducts        0
MntGoldProds            0
NumDealsPurchases       0
NumWebPurchases         0
NumCatalogPurchases     0
NumStorePurchases       0
NumWebVisitsMonth       0
AcceptedCmp3            0
AcceptedCmp4            0
AcceptedCmp5            0
AcceptedCmp1            0
AcceptedCmp2            0
Complain                0
Z_CostContact           0
Z_Revenue               0
Response                0
dtype: int64
```

In [6]:
```
cust_per_df1["child"] = cust_per_df1["Kidhome"] + cust_per_df1["Teenhome"]
```

In [7]:
```
cust_per_df1
```

Out[7]:

	ID	Year_Birth	Education	Marital_Status	Income	Kidhome	Teenhome	Dt_Customer
0	5524	1957	Graduation	Single	58138.0	0	0	04-09-2012
1	2174	1954	Graduation	Single	46344.0	1	1	08-03-2014
2	4141	1965	Graduation	Together	71613.0	0	0	21-08-2013
3	6182	1984	Graduation	Together	26646.0	1	0	10-02-2014
4	5324	1981	PhD	Married	58293.0	1	0	19-01-2014
...	...	...	...	...	...	...	...	...
2235	10870	1967	Graduation	Married	61223.0	0	1	13-06-2013
2236	4001	1946	PhD	Together	64014.0	2	1	10-06-2014
2237	7270	1981	Graduation	Divorced	56981.0	0	0	25-01-2014
2238	8235	1956	Master	Together	69245.0	0	1	24-01-2014
2239	9405	1954	PhD	Married	52869.0	1	1	15-10-2012

2240 rows × 30 columns

In [8]:
```
cust_per_df1.columns
```

Out[8]:
```
Index(['ID', 'Year_Birth', 'Education', 'Marital_Status', 'Income', 'Kidhom
e',
       'Teenhome', 'Dt_Customer', 'Recency', 'MntWines', 'MntFruits',
       'MntMeatProducts', 'MntFishProducts', 'MntSweetProducts',
       'MntGoldProds', 'NumDealsPurchases', 'NumWebPurchases',
       'NumCatalogPurchases', 'NumStorePurchases', 'NumWebVisitsMonth',
       'AcceptedCmp3', 'AcceptedCmp4', 'AcceptedCmp5', 'AcceptedCmp1',
       'AcceptedCmp2', 'Complain', 'Z_CostContact', 'Z_Revenue', 'Response',
       'child'],
      dtype='object')
```

In [9]:
```
len(cust_per_df1.columns)
```

Out[9]:
```
30
```

In [10]:
```
cust_per_df1["total_amount_spent"] = cust_per_df1.MntWines + cust_per_df1.MntF
                                    cust_per_df1.MntFishProducts + cust_per_df
```

In [11]:
```
cust_per_df1.head()
```

Out[11]:

	ID	Year_Birth	Education	Marital_Status	Income	Kidhome	Teenhome	Dt_Customer	Rec
0	5524	1957	Graduation	Single	58138.0	0	0	04-09-2012	
1	2174	1954	Graduation	Single	46344.0	1	1	08-03-2014	
2	4141	1965	Graduation	Together	71613.0	0	0	21-08-2013	
3	6182	1984	Graduation	Together	26646.0	1	0	10-02-2014	
4	5324	1981	PhD	Married	58293.0	1	0	19-01-2014	

5 rows × 31 columns

In [12]:
```
cust_per_df1["Dt_Customer"] = pd.to_datetime(cust_per_df1.Dt_Customer)
```

```
C:\Users\DHRUV\AppData\Local\Temp\ipykernel_1188\3321778344.py:1: UserWarn
ing: Parsing '21-08-2013' in DD/MM/YYYY format. Provide format or specify
infer_datetime_format=True for consistent parsing.
  cust_per_df1["Dt_Customer"] = pd.to_datetime(cust_per_df1.Dt_Customer)
C:\Users\DHRUV\AppData\Local\Temp\ipykernel_1188\3321778344.py:1: UserWarn
ing: Parsing '19-01-2014' in DD/MM/YYYY format. Provide format or specify
infer_datetime_format=True for consistent parsing.
  cust_per_df1["Dt_Customer"] = pd.to_datetime(cust_per_df1.Dt_Customer)
C:\Users\DHRUV\AppData\Local\Temp\ipykernel_1188\3321778344.py:1: UserWarn
ing: Parsing '13-11-2012' in DD/MM/YYYY format. Provide format or specify
infer_datetime_format=True for consistent parsing.
  cust_per_df1["Dt_Customer"] = pd.to_datetime(cust_per_df1.Dt_Customer)
C:\Users\DHRUV\AppData\Local\Temp\ipykernel_1188\3321778344.py:1: UserWarn
ing: Parsing '13-03-2014' in DD/MM/YYYY format. Provide format or specify
infer_datetime_format=True for consistent parsing.
  cust_per_df1["Dt_Customer"] = pd.to_datetime(cust_per_df1.Dt_Customer)
C:\Users\DHRUV\AppData\Local\Temp\ipykernel_1188\3321778344.py:1: UserWarn
ing: Parsing '15-11-2013' in DD/MM/YYYY format. Provide format or specify
infer_datetime_format=True for consistent parsing.
  cust_per_df1["Dt_Customer"] = pd.to_datetime(cust_per_df1.Dt_Customer)
```

In [13]:
```
cust_per_df1
```

Out[13]:

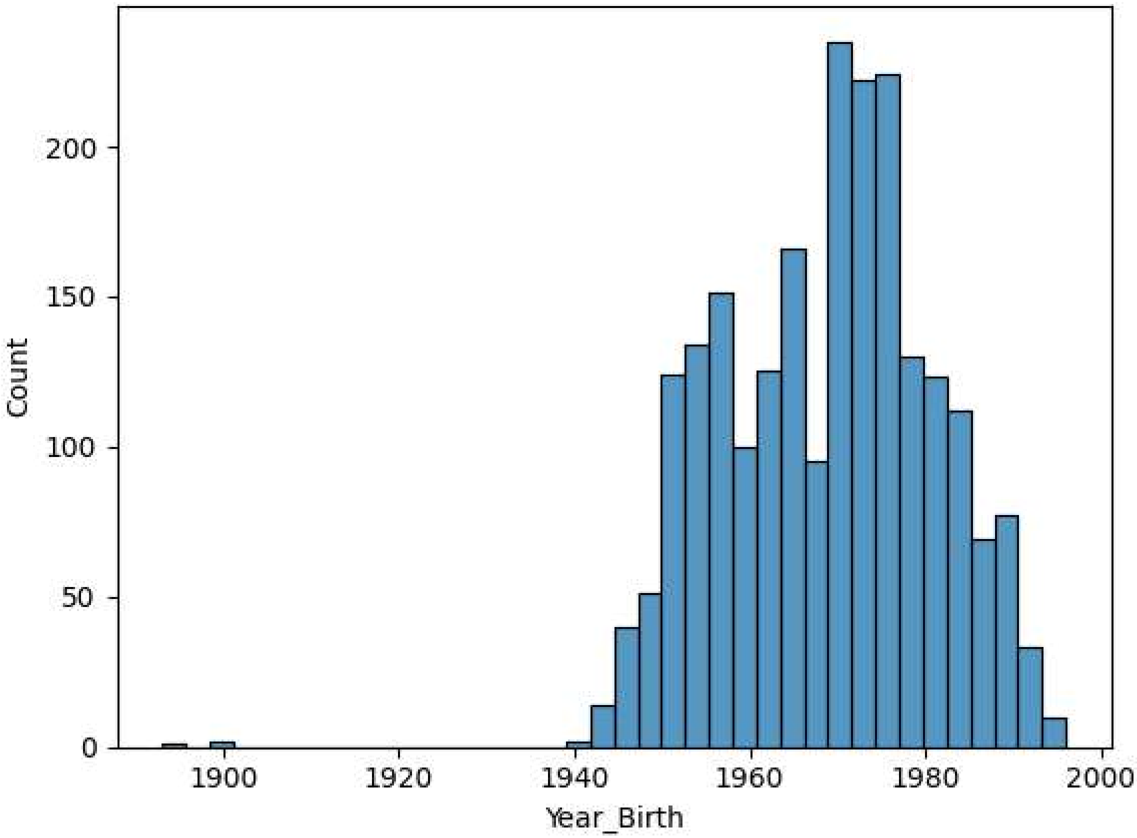
	ID	Year_Birth	Education	Marital_Status	Income	Kidhome	Teenhome	Dt_Customer
0	5524	1957	Graduation	Single	58138.0	0	0	2012-04-09
1	2174	1954	Graduation	Single	46344.0	1	1	2014-08-03
2	4141	1965	Graduation	Together	71613.0	0	0	2013-08-21
3	6182	1984	Graduation	Together	26646.0	1	0	2014-10-02
4	5324	1981	PhD	Married	58293.0	1	0	2014-01-19
...	...	...	...	...	...	...	...	...
2235	10870	1967	Graduation	Married	61223.0	0	1	2013-06-13
2236	4001	1946	PhD	Together	64014.0	2	1	2014-10-06
2237	7270	1981	Graduation	Divorced	56981.0	0	0	2014-01-25
2238	8235	1956	Master	Together	69245.0	0	1	2014-01-24
2239	9405	1954	PhD	Married	52869.0	1	1	2012-10-15

2240 rows × 31 columns

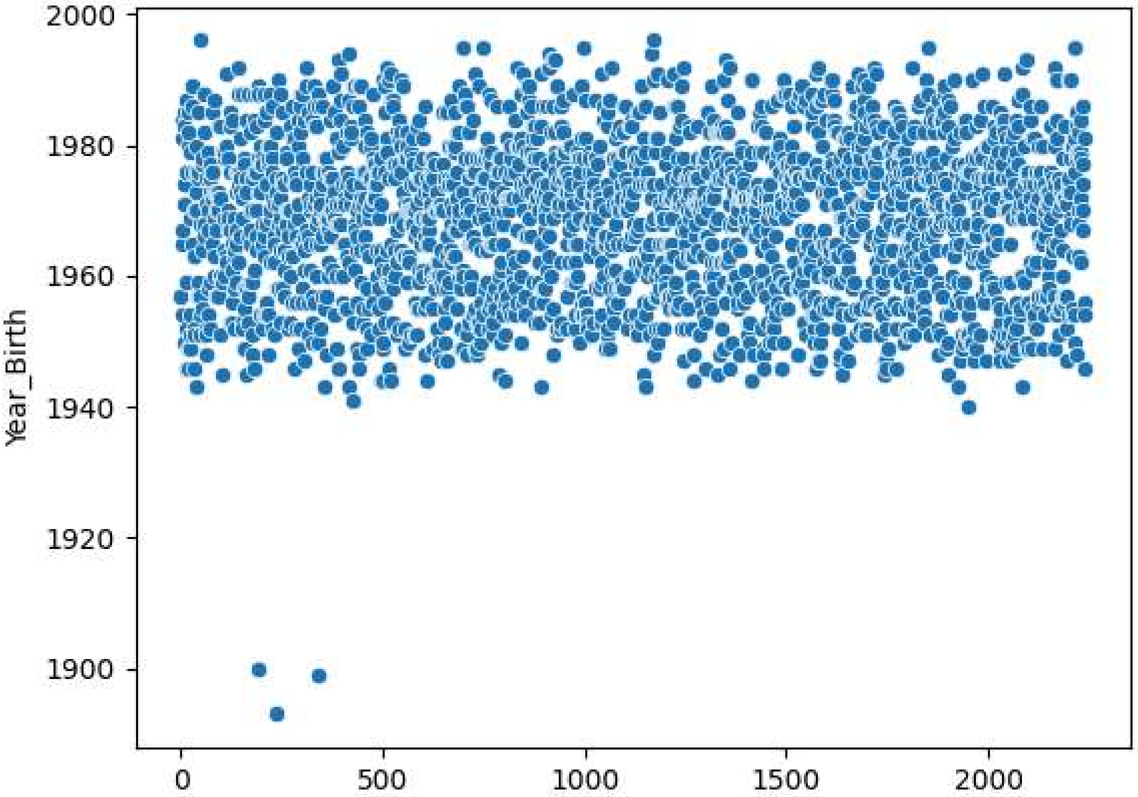
In [14]:
```
cust_per_df1.Year_Birth.unique()
```

Out[14]:
```
array([1957, 1954, 1965, 1984, 1981, 1967, 1971, 1985, 1974, 1950, 1983,
       1976, 1959, 1952, 1987, 1946, 1980, 1949, 1982, 1979, 1951, 1969,
       1986, 1989, 1963, 1970, 1973, 1943, 1975, 1996, 1968, 1964, 1977,
       1978, 1955, 1966, 1988, 1948, 1958, 1972, 1960, 1945, 1991, 1962,
       1953, 1961, 1956, 1992, 1900, 1893, 1990, 1947, 1899, 1993, 1994,
       1941, 1944, 1995, 1940], dtype=int64)
```

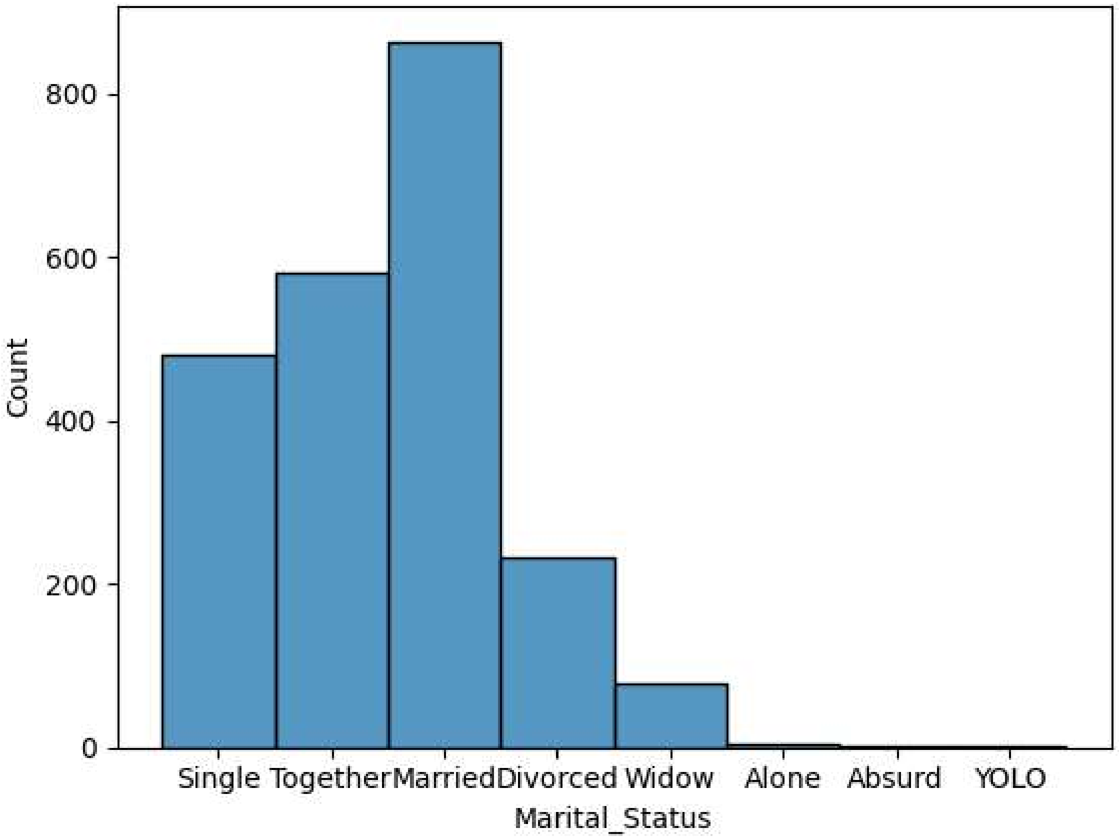
In [15]:
```
sns.histplot(cust_per_df1.Year_Birth)
plt.show()
```



In [16]:
```
sns.scatterplot(cust_per_df1.Year_Birth)
plt.show()
```

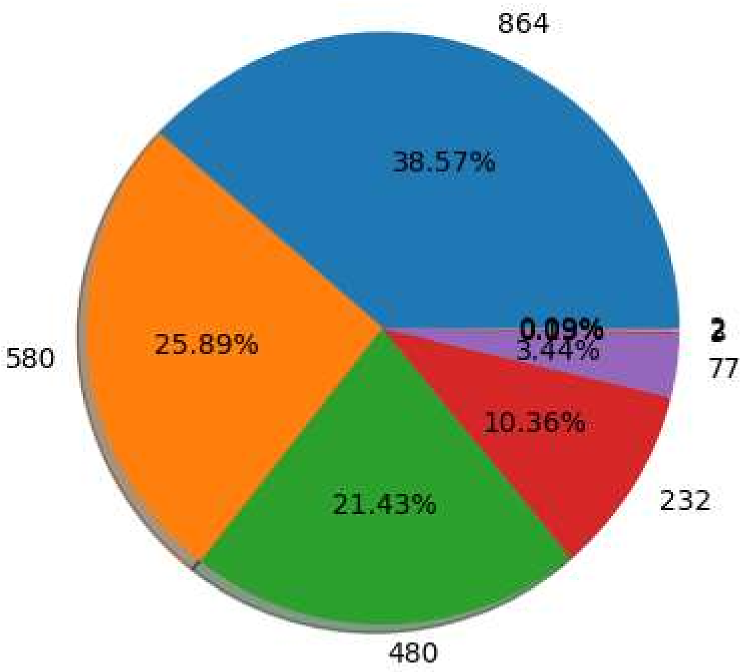


In [17]:
```
sns.histplot(cust_per_df1.Marital_Status)
plt.show()
```



In [18]:

```
plt.pie(cust_per_df1.Marital_Status.value_counts(),labels=cust_per_df1.Marital
plt.show()
```



In [20]:

```
cust_per_df1
```

Out[20]:

Year_Birth	Education	Marital_Status	Income	Kidhome	Teenhome	Dt_Customer	Recency	Mr
1957	Graduation	Single	58138.0	0	0	2012-04-09	58	
1954	Graduation	Single	46344.0	1	1	2014-08-03	38	
1965	Graduation	Together	71613.0	0	0	2013-08-21	26	
1984	Graduation	Together	26646.0	1	0	2014-10-02	26	
1981	PhD	Married	58293.0	1	0	2014-01-19	94	
...	...	...	...	...	...	...	...	
1967	Graduation	Married	61223.0	0	1	2013-06-13	46	
1946	PhD	Together	64014.0	2	1	2014-10-06	56	
1981	Graduation	Divorced	56981.0	0	0	2014-01-25	91	
1956	Master	Together	69245.0	0	1	2014-01-24	8	
1954	PhD	Married	52869.0	1	1	2012-10-15	40	

31 columns

Q1 How many customer income have more than 50000?

In [21]:
```
high_income = cust_per_df1.Income > 50000
high_income
```

Out[21]:
```
0        True
1       False
2        True
3       False
4        True
        ...
2235     True
2236     True
2237     True
2238     True
2239     True
Name: Income, Length: 2240, dtype: bool
```

In [24]:
```
high_income_df = cust_per_df1[high_income]
high_income_df.Income.count()
print(f"the total number of income is having more than 50k custers are {high_i
```

```
the total number of income is having more than 50k custers are 1156
```

In [25]:
```
high_income_df
```

Out[25]:

	ID	Year_Birth	Education	Marital_Status	Income	Kidhome	Teenhome	Dt_Customer
0	5524	1957	Graduation	Single	58138.0	0	0	2012-04-09
2	4141	1965	Graduation	Together	71613.0	0	0	2013-08-21
4	5324	1981	PhD	Married	58293.0	1	0	2014-01-19
5	7446	1967	Master	Together	62513.0	0	1	2013-09-09
6	965	1971	Graduation	Divorced	55635.0	0	1	2012-11-13
...	...	...	...	...	...	...	...	...
2235	10870	1967	Graduation	Married	61223.0	0	1	2013-06-13
2236	4001	1946	PhD	Together	64014.0	2	1	2014-10-06
2237	7270	1981	Graduation	Divorced	56981.0	0	0	2014-01-25
2238	8235	1956	Master	Together	69245.0	0	1	2014-01-24
2239	9405	1954	PhD	Married	52869.0	1	1	2012-10-15

1156 rows × 31 columns

Q2. How many money spent overall in wines and give its percentage overall spent money?

In [29]:
```
# only 10 columns display here
high_income_df.columns[0:10]
```

Out[29]:
```
Index(['ID', 'Year_Birth', 'Education', 'Marital_Status', 'Income', 'Kidhom
e',
       'Teenhome', 'Dt_Customer', 'Recency', 'MntWines'],
      dtype='object')
```

In [31]:
```
# Wine total
overall_wines = cust_per_df1.MntWines.sum()
overall_wines
```

Out[31]:
```
680816
```

In [32]:
```
# Total Spent

Overall_amount_spent = cust_per_df1.total_amount_spent.sum()
Overall_amount_spent
```

Out[32]:
```
1356988
```

In [35]:
```
Percent_wines =  round(((overall_wines * 100)/Overall_amount_spent),2)
#ercent_wines
print(f"the money spent overall data n wines are {overall_wines} and its perce
```

```
the money spent overall data n wines are 680816 and its percentage 50.17%
```

Q3 which year customer spent maximum amount?

In [37]:
```
cust_per_df1['Year'] = pd.DatetimeIndex(cust_per_df1.Dt_Customer).year
cust_per_df1['Month'] = pd.DatetimeIndex(cust_per_df1.Dt_Customer).month
cust_per_df1['Day'] = pd.DatetimeIndex(cust_per_df1.Dt_Customer).day
```

In [38]:
```
cust_per_df1
```

Out[38]:

Teenhome	Dt_Customer	Recency	MntWines	...	AcceptedCmp2	Complain	Z_CostContact	Z_Rev
0	2012-04-09	58	635	...	0	0	3	
1	2014-08-03	38	11	...	0	0	3	
0	2013-08-21	26	426	...	0	0	3	
0	2014-10-02	26	11	...	0	0	3	
0	2014-01-19	94	173	...	0	0	3	
...	...	...	...	...	...	...	...	
1	2013-06-13	46	709	...	0	0	3	
1	2014-10-06	56	406	...	0	0	3	
0	2014-01-25	91	908	...	0	0	3	
1	2014-01-24	8	428	...	0	0	3	
1	2012-10-15	40	84	...	0	0	3	

In [45]:
```
cust_per_df1.groupby('Year')[["total_amount_spent"]].sum()
```

Out[45]:

	total_amount_spent
Year	
2012	368269
2013	715425
2014	273294

In [57]:
```
cust_per_df1.groupby('Year')[["total_amount_spent"]].sum().max()
```

Out[57]:
```
total_amount_spent    715425
dtype: int64
```

Q4 Maximum income of the month?

In [59]:

```
cust_per_df1.nlargest(10,'Month')
```

Out[59]:

	ID	Year_Birth	Education	Marital_Status	Income	Kidhome	Teenhome	Dt_Customer	Re
16	9736	1980	Graduation	Married	41850.0	1	1	2012-12-24	
22	1993	1949	PhD	Married	58607.0	0	1	2012-12-23	
36	4339	1970	PhD	Married	67353.0	0	1	2013-12-31	
41	503	1985	Master	Married	20559.0	1	0	2013-12-03	
42	8430	1957	Graduation	Together	21994.0	0	1	2012-12-24	
50	8614	1957	Graduation	Widow	65486.0	0	1	2014-12-05	
54	9381	1978	Graduation	Married	66373.0	1	1	2013-12-06	
75	5846	1977	Graduation	Divorced	40246.0	1	0	2012-12-19	
77	7503	1976	Graduation	Single	75825.0	0	0	2012-12-10	
88	8504	1973	Graduation	Married	79593.0	0	0	2014-12-05	

10 rows × 34 columns

In [ ]: