Excel Tasks:

1. Data Exploration:

Task 1: Create a statistical summary for numerical features

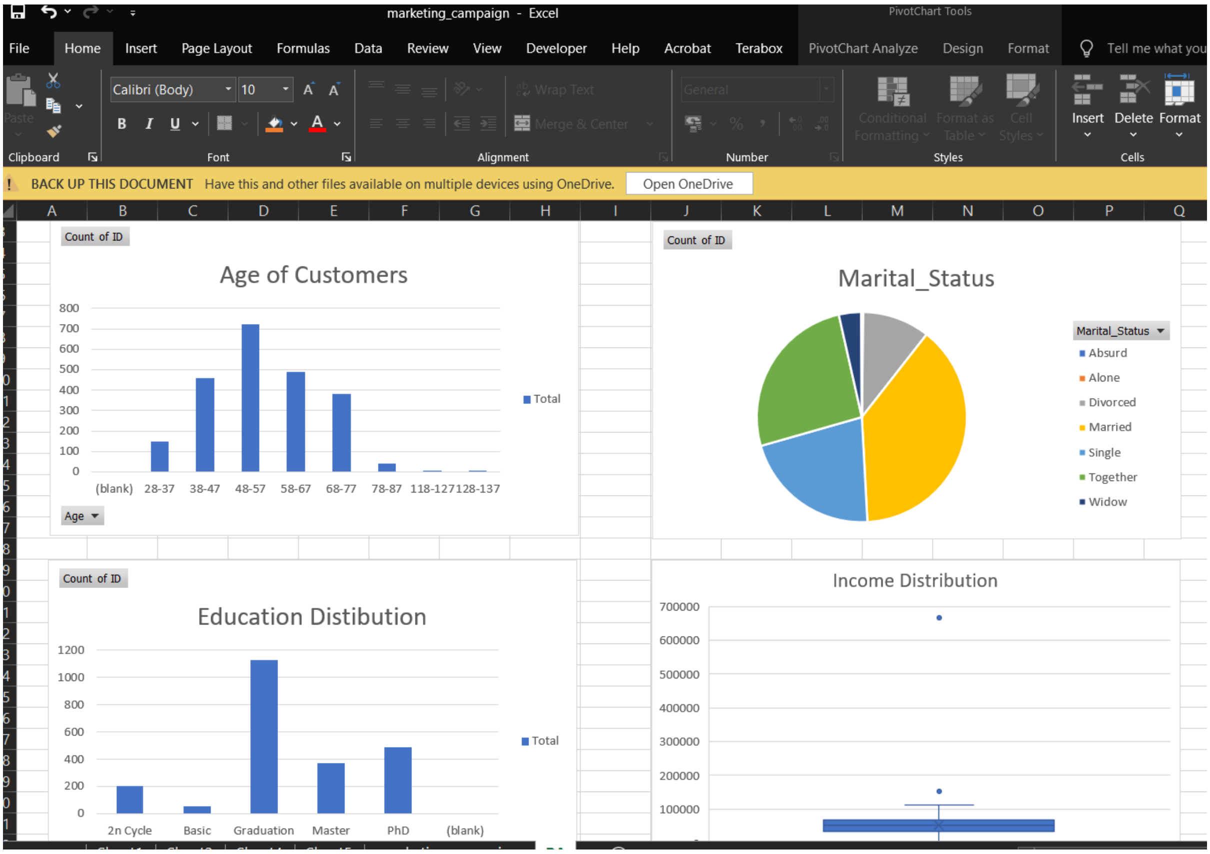
Task 2: Create a line chart for the number of enrolments by year

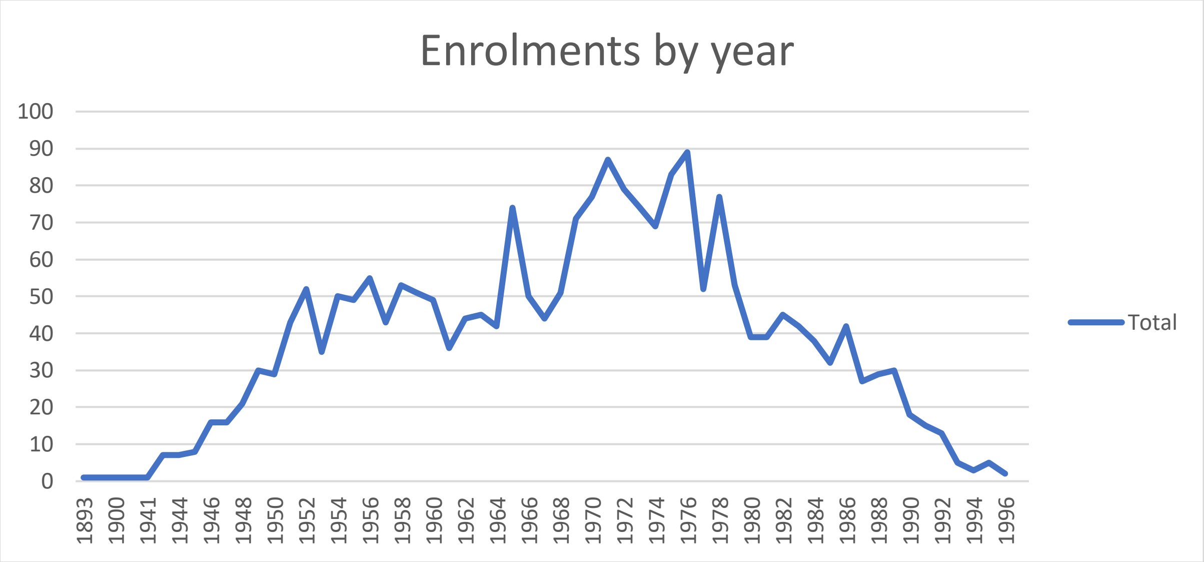
Task 3: Give a cross-tabulated count for response values against education

Task 4: Make a boxplot on income and write your observations

Task 5: Calculate the age of customers and make a histogram of that

Task 6: Visualize the response against Marital_Status



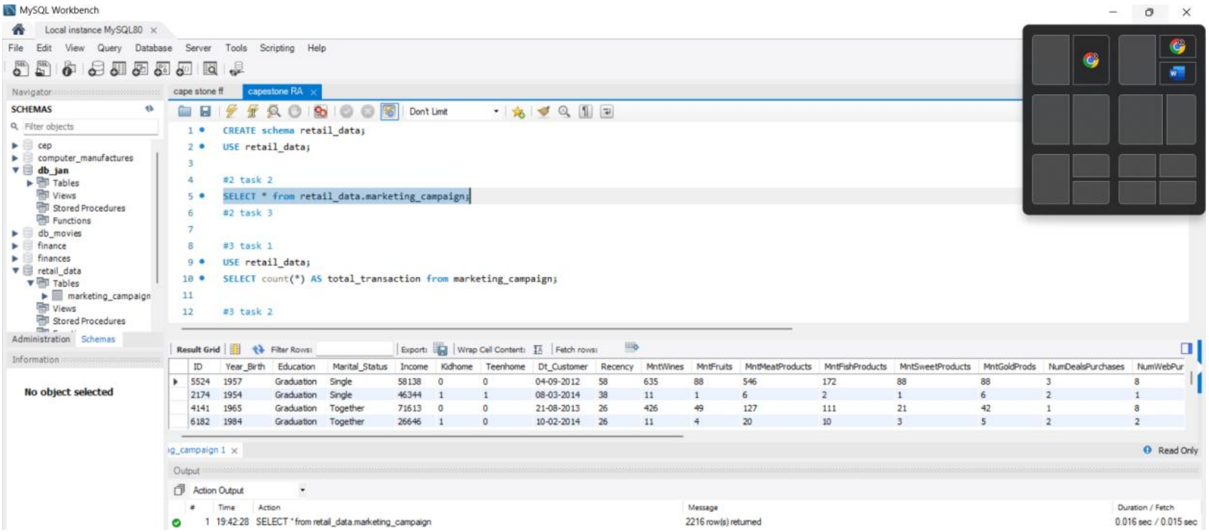


Data Loading: Create a schema named "retail_data"

Set "retail_data" as the default schema

Create tables to store the retail transaction data

Set ct_customer as the datetime field while loading the data and apply the appropriate date format



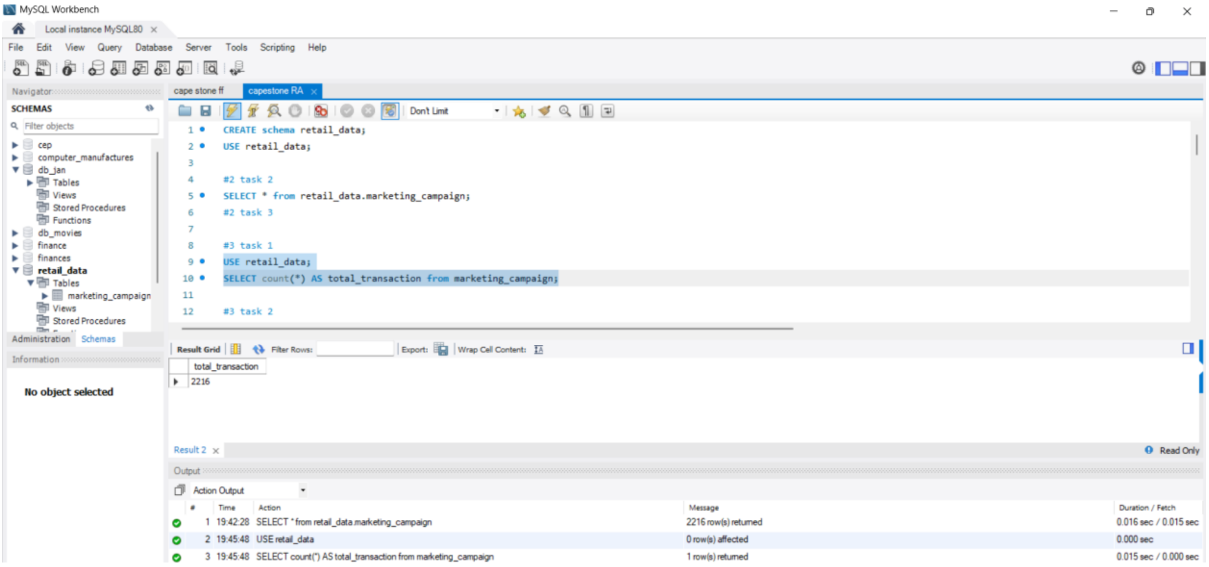
```
CREATE schema retail_data;
USE retail_data;

#2 task 2
SELECT * from retail_data.marketing_campaign;
#2 task 3

#3 task 1
USE retail_data;
SELECT count(*) AS total_transaction from marketing_campaign;

#3 task 2
```

ID	Year_Birth	Education	Marital_Status	Income	Kidhome	Teenhome	Dt_Customer	Recency	MntWines	MntFruits	MntMeatProducts	MntFishProducts	MntSweetProducts	MntGoldProds	NumDealsPurchases	NumWebPur
5524	1957	Graduation	Single	58138	0	0	04-09-2012	58	635	88	546	172	88	88	3	8
2174	1954	Graduation	Single	46344	1	1	08-03-2014	38	11	1	6	2	1	6	2	1
4141	1965	Graduation	Together	71613	0	0	21-08-2013	26	426	49	127	111	21	42	1	8
6182	1984	Graduation	Together	26646	1	0	10-02-2014	26	11	4	20	10	3	5	2	2

```
#   Time      Action                                        Message                Duration / Fetch
1   19:42:28  SELECT * from retail_data.marketing_campaign  2216 row(s) returned   0.016 sec / 0.015 sec
```

1 Data Preprocessing:

Calculate the total number of customer encounters in the marketing campaign dataset

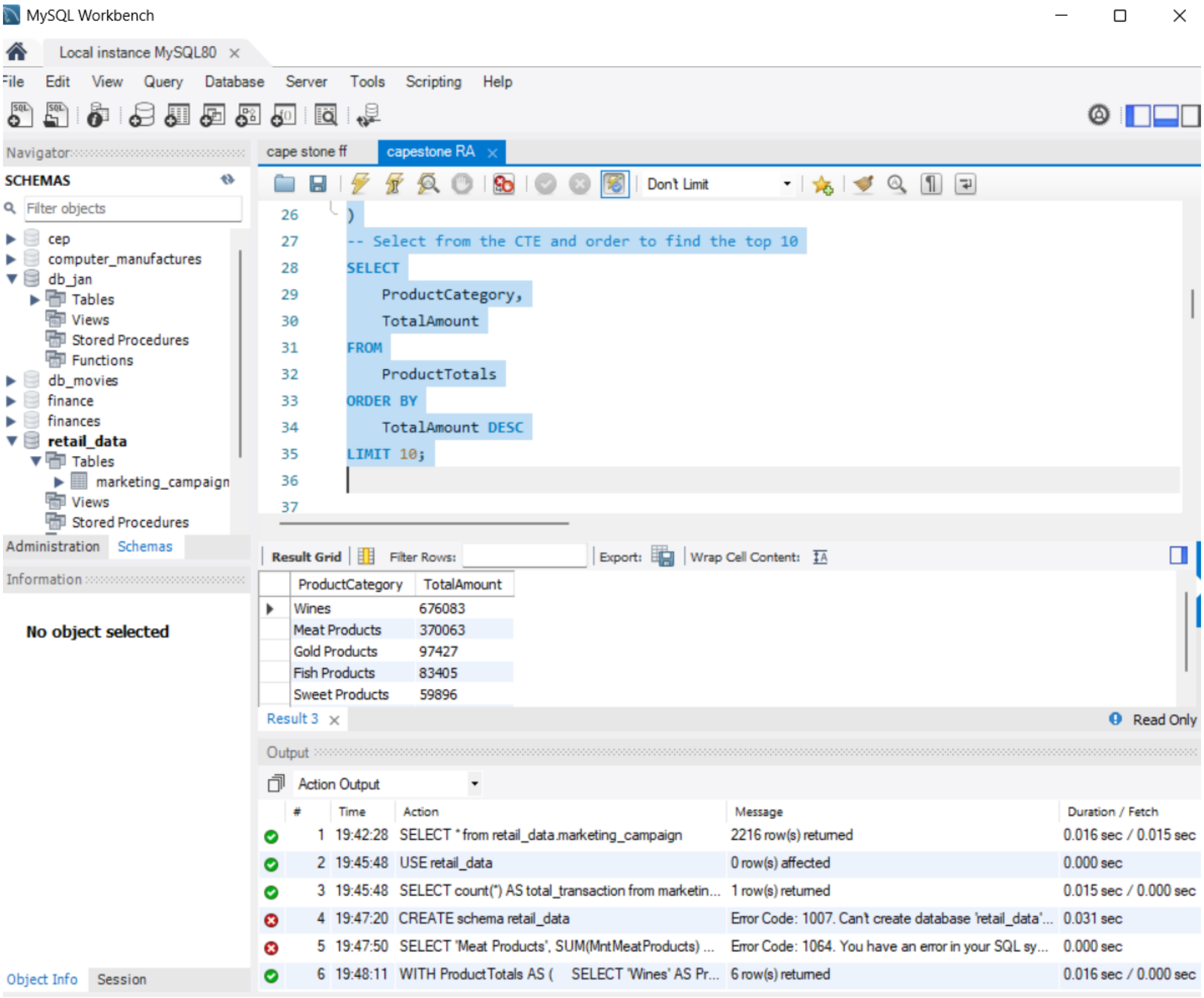


```sql
CREATE schema retail_data;
USE retail_data;

#2 task 2
SELECT * from retail_data.marketing_campaign;
#2 task 3

#3 task 1
USE retail_data;
SELECT count(*) AS total_transaction from marketing_campaign;

#3 task 2
```

	total_transaction
▶	2216

	#	Time	Action	Message	Duration / Fetch
✓	1	19:42:28	SELECT * from retail_data.marketing_campaign	2216 row(s) returned	0.016 sec / 0.015 sec
✓	2	19:45:48	USE retail_data	0 row(s) affected	0.000 sec
✓	3	19:45:48	SELECT count(*) AS total_transaction from marketing_campaign	1 row(s) returned	0.015 sec / 0.000 sec

Identify the top 10 most purchased products in the dataset, such as Wines, Meat Products, etc.

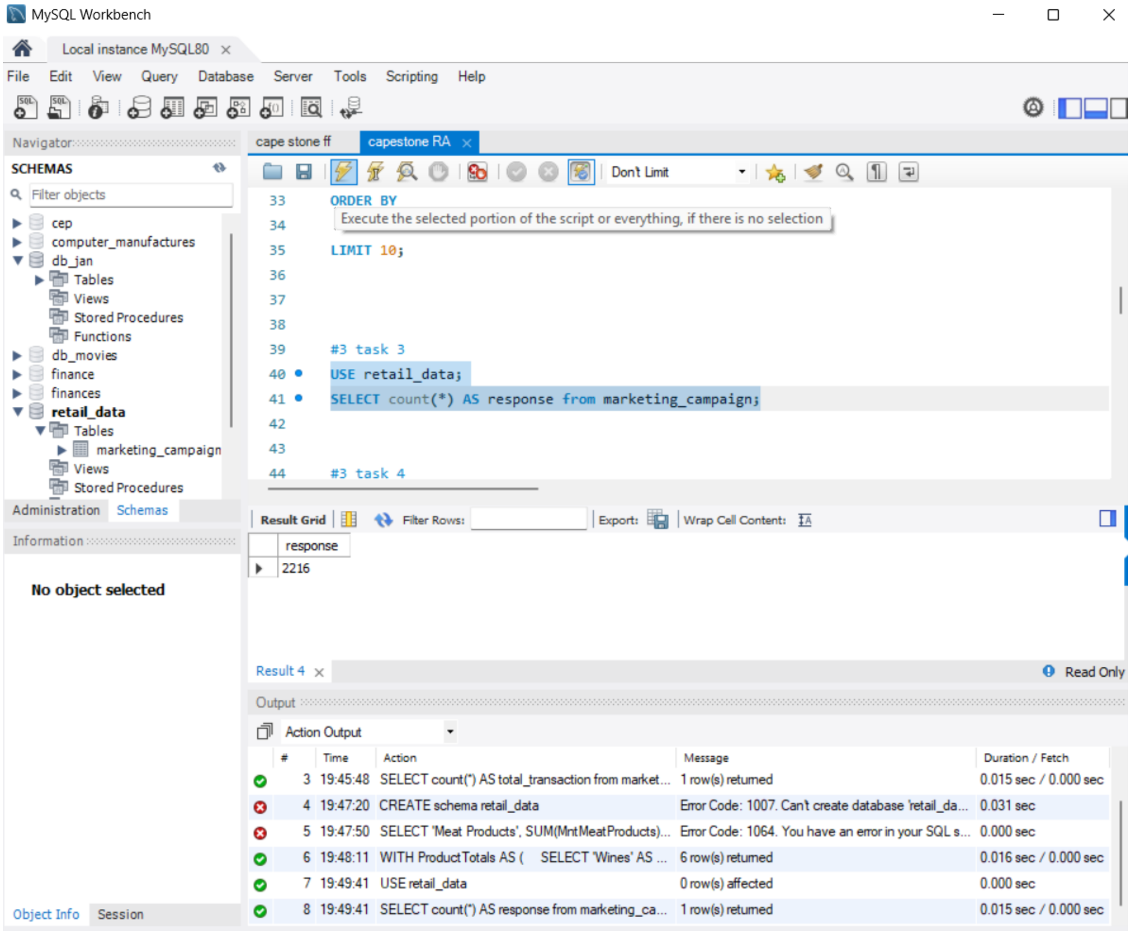


```sql
)
-- Select from the CTE and order to find the top 10
SELECT
    ProductCategory,
    TotalAmount
FROM
    ProductTotals
ORDER BY
    TotalAmount DESC
LIMIT 10;
```

	ProductCategory	TotalAmount
▶	Wines	676083
	Meat Products	370063
	Gold Products	97427
	Fish Products	83405
	Sweet Products	59896

	#	Time	Action	Message	Duration / Fetch
✓	1	19:42:28	SELECT * from retail_data.marketing_campaign	2216 row(s) returned	0.016 sec / 0.015 sec
✓	2	19:45:48	USE retail_data	0 row(s) affected	0.000 sec
✓	3	19:45:48	SELECT count(*) AS total_transaction from marketin...	1 row(s) returned	0.015 sec / 0.000 sec
✗	4	19:47:20	CREATE schema retail_data	Error Code: 1007. Can't create database 'retail_data'...	0.031 sec
✗	5	19:47:50	SELECT 'Meat Products', SUM(MntMeatProducts) ...	Error Code: 1064. You have an error in your SQL sy...	0.000 sec
✓	6	19:48:11	WITH ProductTotals AS (SELECT 'Wines' AS Pr...	6 row(s) returned	0.016 sec / 0.000 sec

Find the count of response values

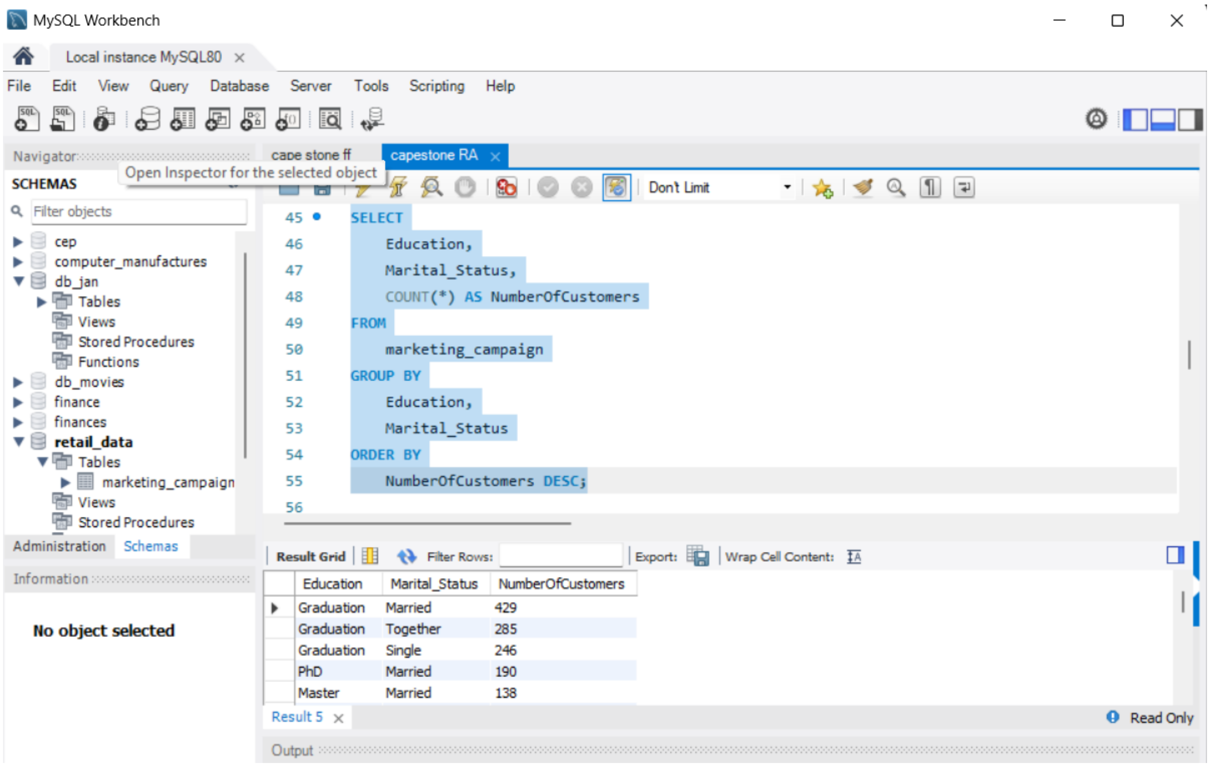


```
ORDER BY

LIMIT 10;



#3 task 3
USE retail_data;
SELECT count(*) AS response from marketing_campaign;


#3 task 4
```

```
response
2216
```

	#	Time	Action	Message	Duration / Fetch
✓	3	19:45:48	SELECT count(*) AS total_transaction from market...	1 row(s) returned	0.015 sec / 0.000 sec
☒	4	19:47:20	CREATE schema retail_data	Error Code: 1007. Can't create database 'retail_da...	0.031 sec
☒	5	19:47:50	SELECT 'Meat Products', SUM(MntMeatProducts)...	Error Code: 1064. You have an error in your SQL s...	0.000 sec
✓	6	19:48:11	WITH ProductTotals AS (SELECT 'Wines' AS ...	6 row(s) returned	0.016 sec / 0.000 sec
✓	7	19:49:41	USE retail_data	0 row(s) affected	0.000 sec
✓	8	19:49:41	SELECT count(*) AS response from marketing_ca...	1 row(s) returned	0.015 sec / 0.000 sec

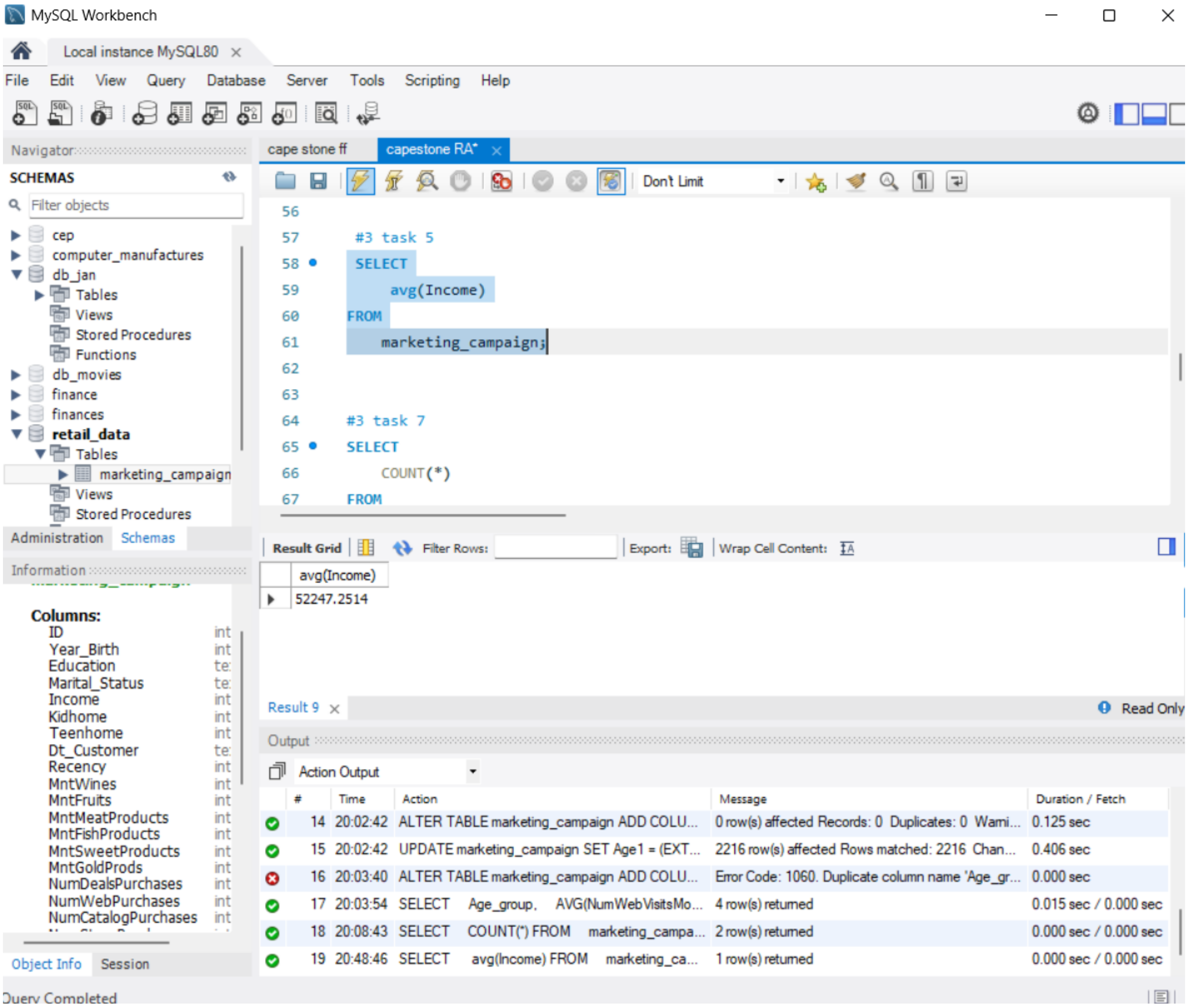
Determine the distribution of customers based on their education level and marital status



```
SELECT
    Education,
    Marital_Status,
    COUNT(*) AS NumberOfCustomers
FROM
    marketing_campaign
GROUP BY
    Education,
    Marital_Status
ORDER BY
    NumberOfCustomers DESC;
```

Education	Marital_Status	NumberOfCustomers
Graduation	Married	429
Graduation	Together	285
Graduation	Single	246
PhD	Married	190
Master	Married	138

Identify the average income of customers who participated in the marketing campaign



```
#3 task 5
SELECT
    avg(Income)
FROM
    marketing_campaign;


#3 task 7
SELECT
    COUNT(*)
FROM
```

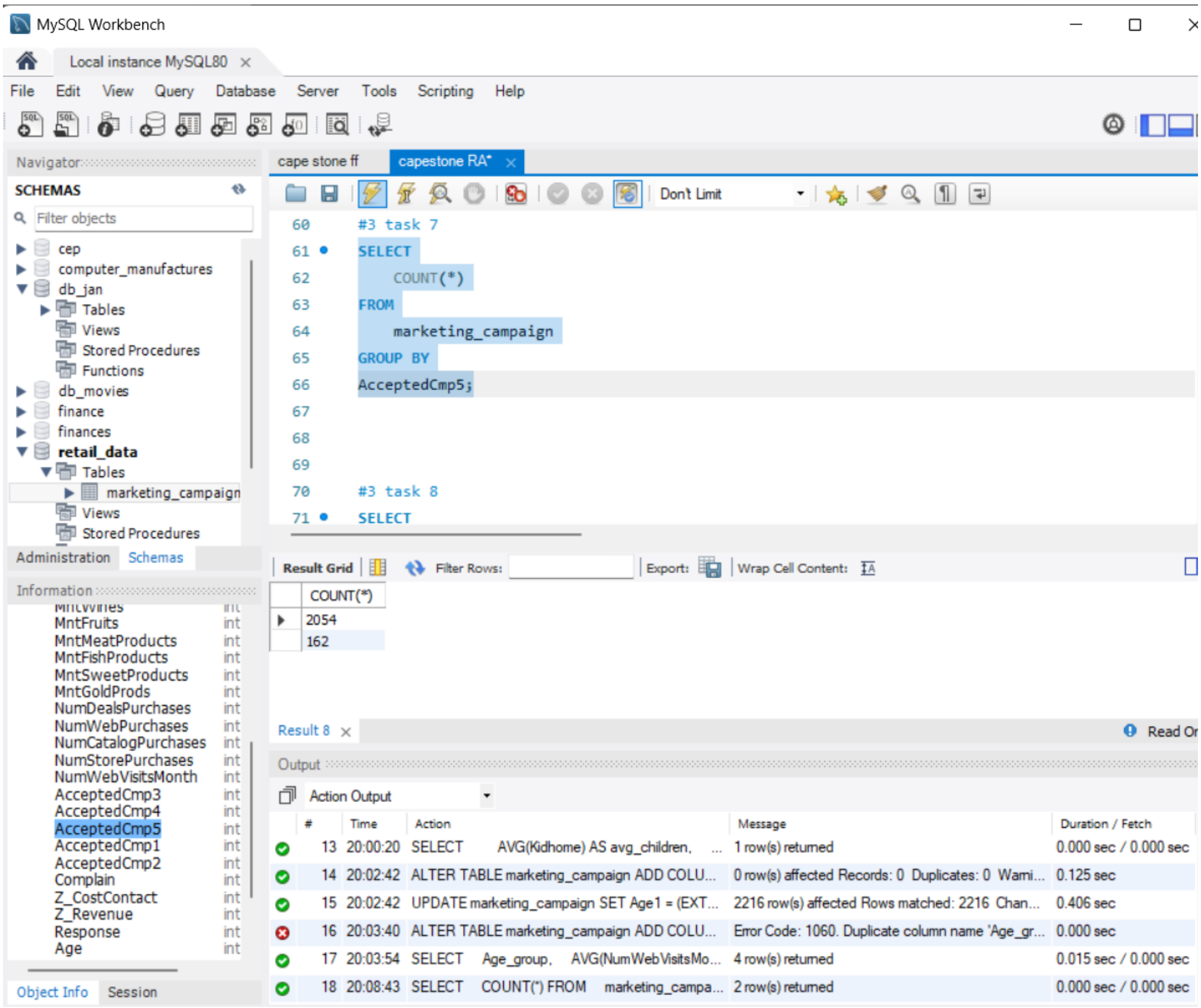
avg(Income)
52247.2514

Columns:

ID	int
Year_Birth	int
Education	te:
Marital_Status	te:
Income	int
Kidhome	int
Teenhome	int
Dt_Customer	te:
Recency	int
MntWines	int
MntFruits	int
MntMeatProducts	int
MntFishProducts	int
MntSweetProducts	int
MntGoldProds	int
NumDealsPurchases	int
NumWebPurchases	int
NumCatalogPurchases	int

	#	Time	Action	Message	Duration / Fetch
✓	14	20:02:42	ALTER TABLE marketing_campaign ADD COLU...	0 row(s) affected Records: 0 Duplicates: 0 Wami...	0.125 sec
✓	15	20:02:42	UPDATE marketing_campaign SET Age1 = (EXT...	2216 row(s) affected Rows matched: 2216 Chan...	0.406 sec
✗	16	20:03:40	ALTER TABLE marketing_campaign ADD COLU...	Error Code: 1060. Duplicate column name 'Age_gr...	0.000 sec
✓	17	20:03:54	SELECT Age_group, AVG(NumWebVisitsMo...	4 row(s) returned	0.015 sec / 0.000 sec
✓	18	20:08:43	SELECT COUNT(*) FROM marketing_campa...	2 row(s) returned	0.000 sec / 0.000 sec
✓	19	20:48:46	SELECT avg(Income) FROM marketing_ca...	1 row(s) returned	0.000 sec / 0.000 sec

Identify the distribution of customers' responses to the last campaign



```
#3 task 7
SELECT
    COUNT(*)
FROM
    marketing_campaign
GROUP BY
AcceptedCmp5;


#3 task 8
SELECT
```

COUNT(*)
2054
162

MntFruits	int
MntMeatProducts	int
MntFishProducts	int
MntSweetProducts	int
MntGoldProds	int
NumDealsPurchases	int
NumWebPurchases	int
NumCatalogPurchases	int
NumStorePurchases	int
NumWebVisitsMonth	int
AcceptedCmp3	int
AcceptedCmp4	int
AcceptedCmp5	int
AcceptedCmp1	int
AcceptedCmp2	int
Complain	int
Z_CostContact	int
Z_Revenue	int
Response	int
Age	int

```
 #  Time      Action                                      Message                                              Duration / Fetch
 13 20:00:20  SELECT     AVG(Kidhome) AS avg_children,  ... 1 row(s) returned                                  0.000 sec / 0.000 sec
 14 20:02:42  ALTER TABLE marketing_campaign ADD COLU...   0 row(s) affected Records: 0  Duplicates: 0  Warni... 0.125 sec
 15 20:02:42  UPDATE marketing_campaign SET Age1 = (EXT... 2216 row(s) affected Rows matched: 2216  Chan...      0.406 sec
 16 20:03:40  ALTER TABLE marketing_campaign ADD COLU...   Error Code: 1060. Duplicate column name 'Age_gr...   0.000 sec
 17 20:03:54  SELECT    Age_group,    AVG(NumWebVisitsMo... 4 row(s) returned                                  0.015 sec / 0.000 sec
 18 20:08:43  SELECT    COUNT(*) FROM    marketing_campa... 2 row(s) returned                                  0.000 sec / 0.000 sec
```

Calculate the average number of children and teenagers in customers' households

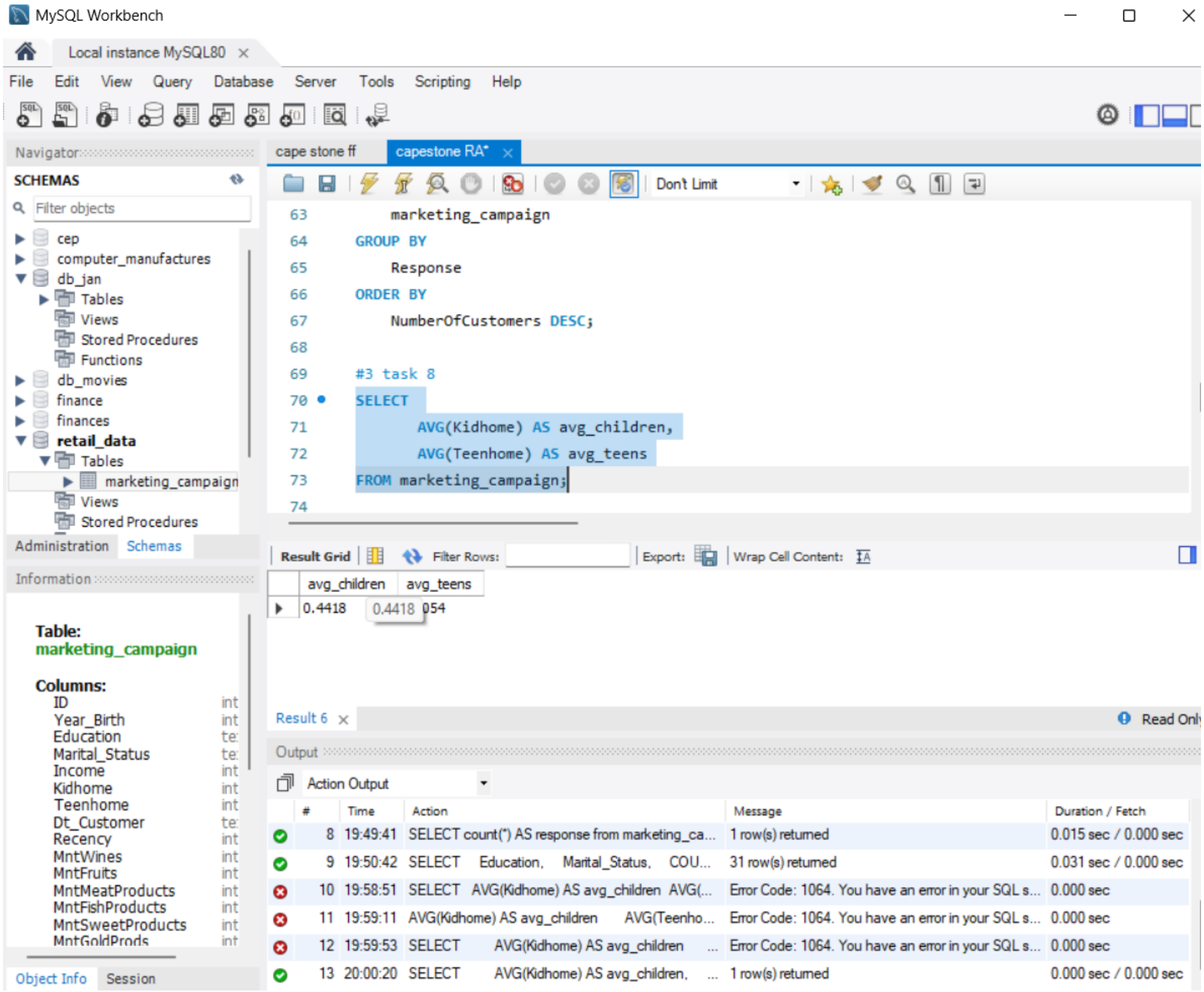


Table: marketing_campaign

Columns:

ID	int
Year_Birth	int
Education	te
Marital_Status	te
Income	int
Kidhome	int
Teenhome	int
Dt_Customer	te
Recency	int
MntWines	int
MntFruits	int
MntMeatProducts	int
MntFishProducts	int
MntSweetProducts	int
MntGoldProds	int

```sql
    marketing_campaign
GROUP BY
    Response
ORDER BY
    NumberOfCustomers DESC;

#3 task 8
SELECT
        AVG(Kidhome) AS avg_children,
        AVG(Teenhome) AS avg_teens
FROM marketing_campaign;

```

avg_children	avg_teens
0.4418	0.4418 054

```
#   Time      Action                                            Message                                            Duration / Fetch
8   19:49:41  SELECT count(*) AS response from marketing_ca...  1 row(s) returned                                  0.015 sec / 0.000 sec
9   19:50:42  SELECT    Education,    Marital_Status,    COU...  31 row(s) returned                                 0.031 sec / 0.000 sec
10  19:58:51  SELECT   AVG(Kidhome) AS avg_children  AVG(...     Error Code: 1064. You have an error in your SQL s...  0.000 sec
11  19:59:11  AVG(Kidhome) AS avg_children      AVG(Teenho...    Error Code: 1064. You have an error in your SQL s...  0.000 sec
12  19:59:53  SELECT      AVG(Kidhome) AS avg_children      ...  Error Code: 1064. You have an error in your SQL s...  0.000 sec
13  20:00:20  SELECT      AVG(Kidhome) AS avg_children,     ...  1 row(s) returned                                  0.000 sec / 0.000 sec
```

Create an Age column by subtracting year_birth from the current year

- Create Age_group columns based on the below condition:

WHEN Age BETWEEN 18 AND 25 THEN '18-25'

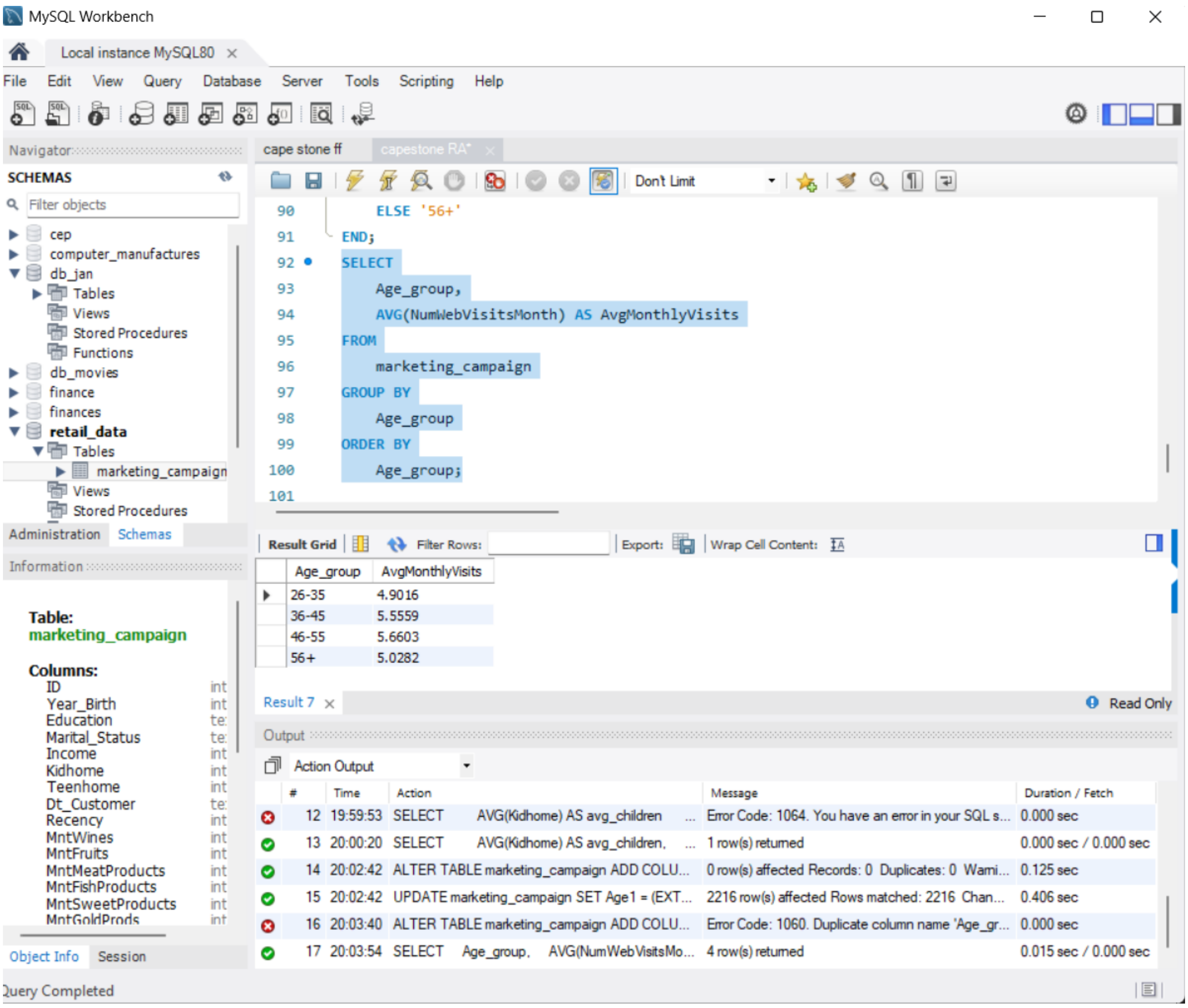
WHEN Age BETWEEN 26 AND 35 THEN '26-35'

WHEN Age BETWEEN 36 AND 45 THEN '36-45'

WHEN Age BETWEEN 46 AND 55 THEN '46-55'

ELSE '56+'

- Determine the average number of visits per month for customers in each age group



```sql
        ELSE '56+'
    END;
SELECT
    Age_group,
    AVG(NumWebVisitsMonth) AS AvgMonthlyVisits
FROM
    marketing_campaign
GROUP BY
    Age_group
ORDER BY
    Age_group;
```

Age_group	AvgMonthlyVisits
26-35	4.9016
36-45	5.5559
46-55	5.6603
56+	5.0282

```
#   Time      Action                                            Message                                               Duration / Fetch
12  19:59:53  SELECT     AVG(Kidhome) AS avg_children      ...  Error Code: 1064. You have an error in your SQL s...  0.000 sec
13  20:00:20  SELECT     AVG(Kidhome) AS avg_children,     ...  1 row(s) returned                                     0.000 sec / 0.000 sec
14  20:02:42  ALTER TABLE marketing_campaign ADD COLU...        0 row(s) affected Records: 0  Duplicates: 0  Wami...  0.125 sec
15  20:02:42  UPDATE marketing_campaign SET Age1 = (EXT...      2216 row(s) affected Rows matched: 2216  Chan...      0.406 sec
16  20:03:40  ALTER TABLE marketing_campaign ADD COLU...        Error Code: 1060. Duplicate column name 'Age_gr...    0.000 sec
17  20:03:54  SELECT   Age_group,   AVG(NumWebVisitsMo...       4 row(s) returned                                     0.015 sec / 0.000 sec
```

Table: marketing_campaign

Columns:

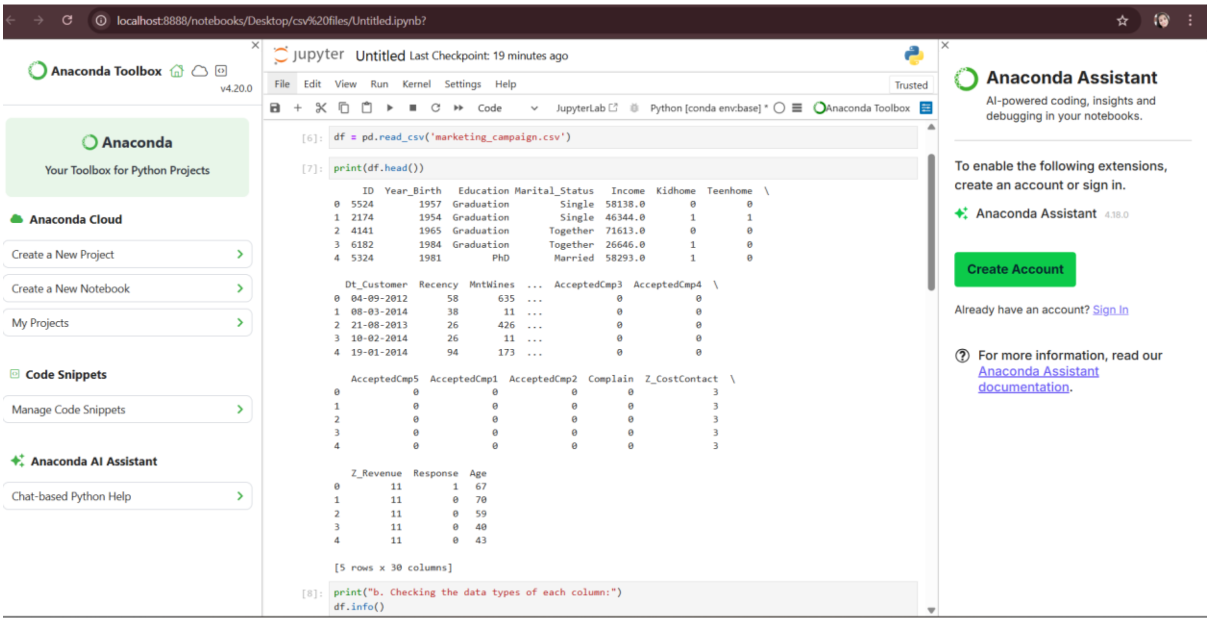
ID	int
Year_Birth	int
Education	te:
Marital_Status	te:
Income	int
Kidhome	int
Teenhome	int
Dt_Customer	te:
Recency	int
MntWines	int
MntFruits	int
MntMeatProducts	int
MntFishProducts	int
MntSweetProducts	int
MntGoldProds	int

Task 1: General Data Overview:

Exploratory Data Analysis Check the first few rows of the dataset to understand its structure

Check the data types of each column

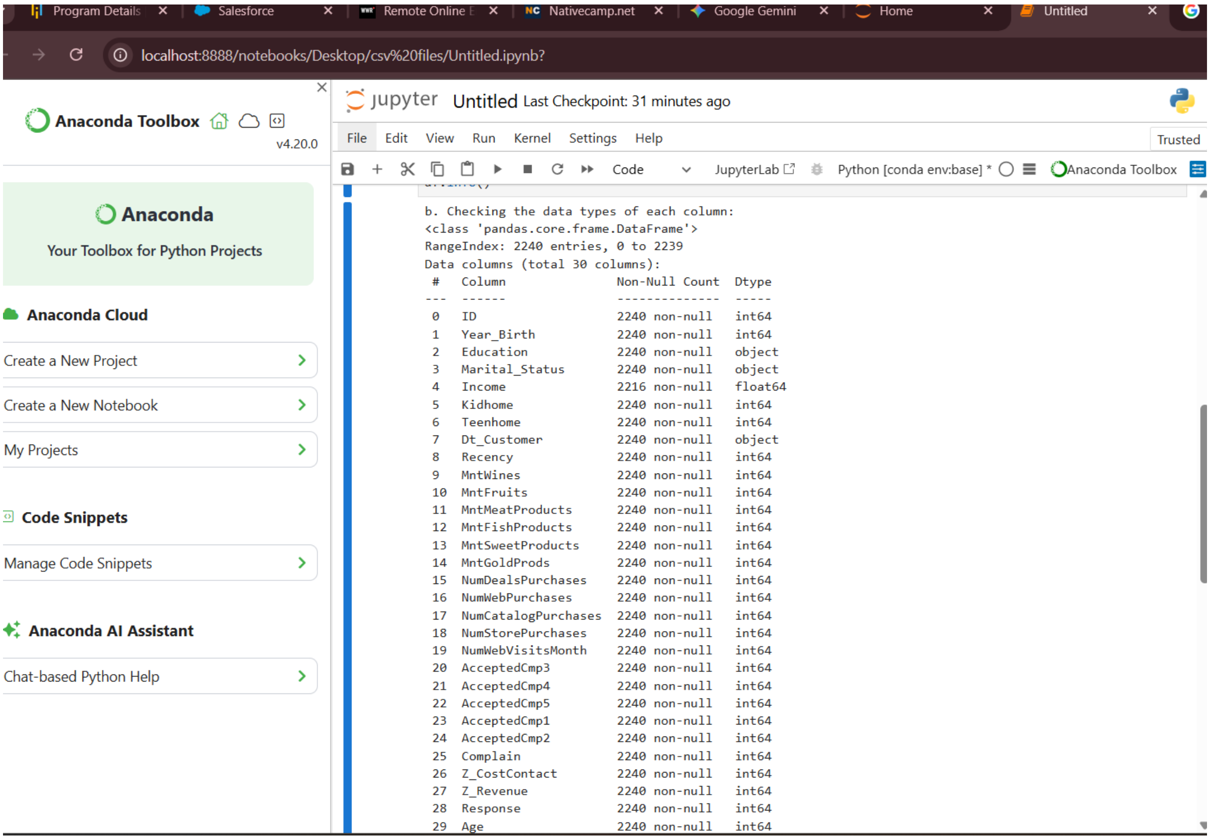
Check for any missing values in the dataset


localhost:8888/notebooks/Desktop/csv%20files/Untitled.ipynb?
Anaconda Toolbox
v4.20.0
Anaconda
Your Toolbox for Python Projects
Anaconda Cloud
Create a New Project
Create a New Notebook
My Projects
Code Snippets
Manage Code Snippets
Anaconda AI Assistant
Chat-based Python Help
jupyter Untitled Last Checkpoint: 19 minutes ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python [conda env:base] *
Anaconda Toolbox
Anaconda Assistant
AI-powered coding, insights and debugging in your notebooks.
Anaconda Assistant 4.18.0
Create Account
Already have an account? Sign In

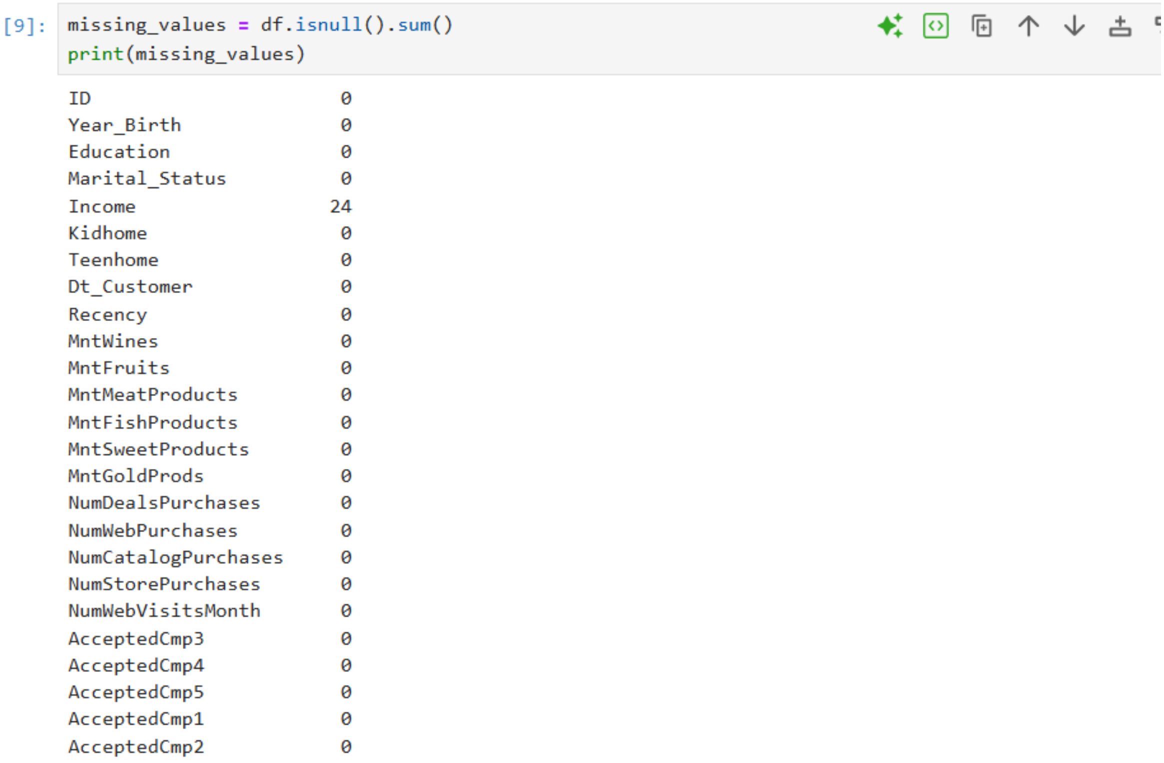
To enable the following extensions, create an account or sign in.

For more information, read our Anaconda Assistant documentation.

```
[6]: df = pd.read_csv('marketing_campaign.csv')

[7]: print(df.head())

     ID  Year_Birth   Education Marital_Status   Income  Kidhome  Teenhome  \
0  5524        1957  Graduation         Single  58138.0        0         0
1  2174        1954  Graduation         Single  46344.0        1         1
2  4141        1965  Graduation       Together  71613.0        0         0
3  6182        1984  Graduation       Together  26646.0        1         0
4  5324        1981         PhD        Married  58293.0        1         0

  Dt_Customer  Recency  MntWines  ...  AcceptedCmp3  AcceptedCmp4  \
0  04-09-2012       58       635  ...             0             0
1  08-03-2014       38        11  ...             0             0
2  21-08-2013       26       426  ...             0             0
3  10-02-2014       26        11  ...             0             0
4  19-01-2014       94       173  ...             0             0

   AcceptedCmp5  AcceptedCmp1  AcceptedCmp2  Complain  Z_CostContact  \
0             0             0             0         0              3
1             0             0             0         0              3
2             0             0             0         0              3
3             0             0             0         0              3
4             0             0             0         0              3

   Z_Revenue  Response  Age
0         11         1   67
1         11         0   70
2         11         0   59
3         11         0   40
4         11         0   43

[5 rows x 30 columns]

[8]: print("b. Checking the data types of each column:")
     df.info()
```



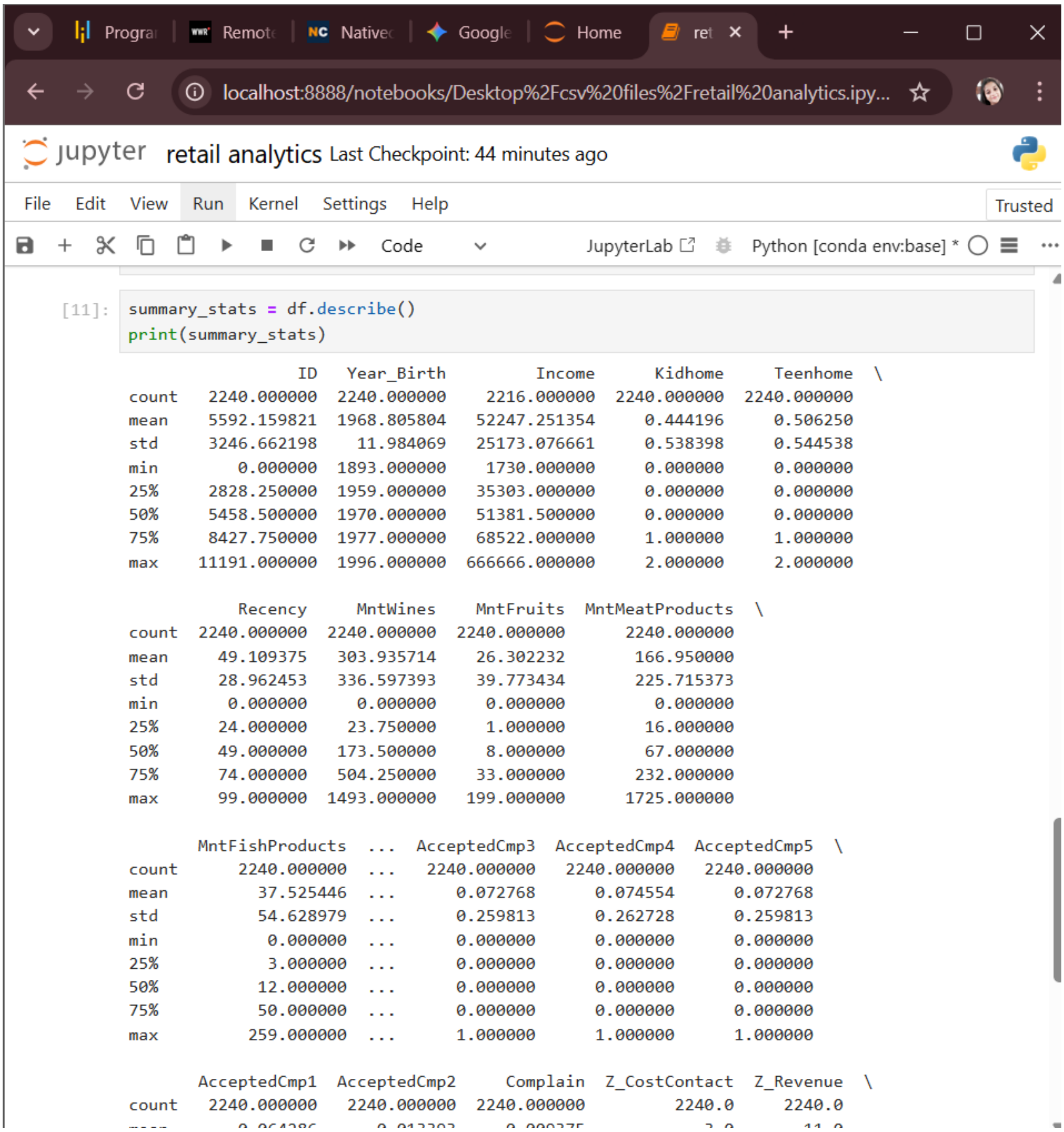
```
b. Checking the data types of each column:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2240 entries, 0 to 2239
Data columns (total 30 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   ID                   2240 non-null   int64
 1   Year_Birth           2240 non-null   int64
 2   Education            2240 non-null   object
 3   Marital_Status       2240 non-null   object
 4   Income               2216 non-null   float64
 5   Kidhome              2240 non-null   int64
 6   Teenhome             2240 non-null   int64
 7   Dt_Customer          2240 non-null   object
 8   Recency              2240 non-null   int64
 9   MntWines             2240 non-null   int64
 10  MntFruits            2240 non-null   int64
 11  MntMeatProducts      2240 non-null   int64
 12  MntFishProducts      2240 non-null   int64
 13  MntSweetProducts     2240 non-null   int64
 14  MntGoldProds         2240 non-null   int64
 15  NumDealsPurchases    2240 non-null   int64
 16  NumWebPurchases      2240 non-null   int64
 17  NumCatalogPurchases  2240 non-null   int64
 18  NumStorePurchases    2240 non-null   int64
 19  NumWebVisitsMonth    2240 non-null   int64
 20  AcceptedCmp3         2240 non-null   int64
 21  AcceptedCmp4         2240 non-null   int64
 22  AcceptedCmp5         2240 non-null   int64
 23  AcceptedCmp1         2240 non-null   int64
 24  AcceptedCmp2         2240 non-null   int64
 25  Complain             2240 non-null   int64
 26  Z_CostContact        2240 non-null   int64
 27  Z_Revenue            2240 non-null   int64
 28  Response             2240 non-null   int64
 29  Age                  2240 non-null   int64
```



```
[9]: missing_values = df.isnull().sum()
     print(missing_values)
```

```
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
```

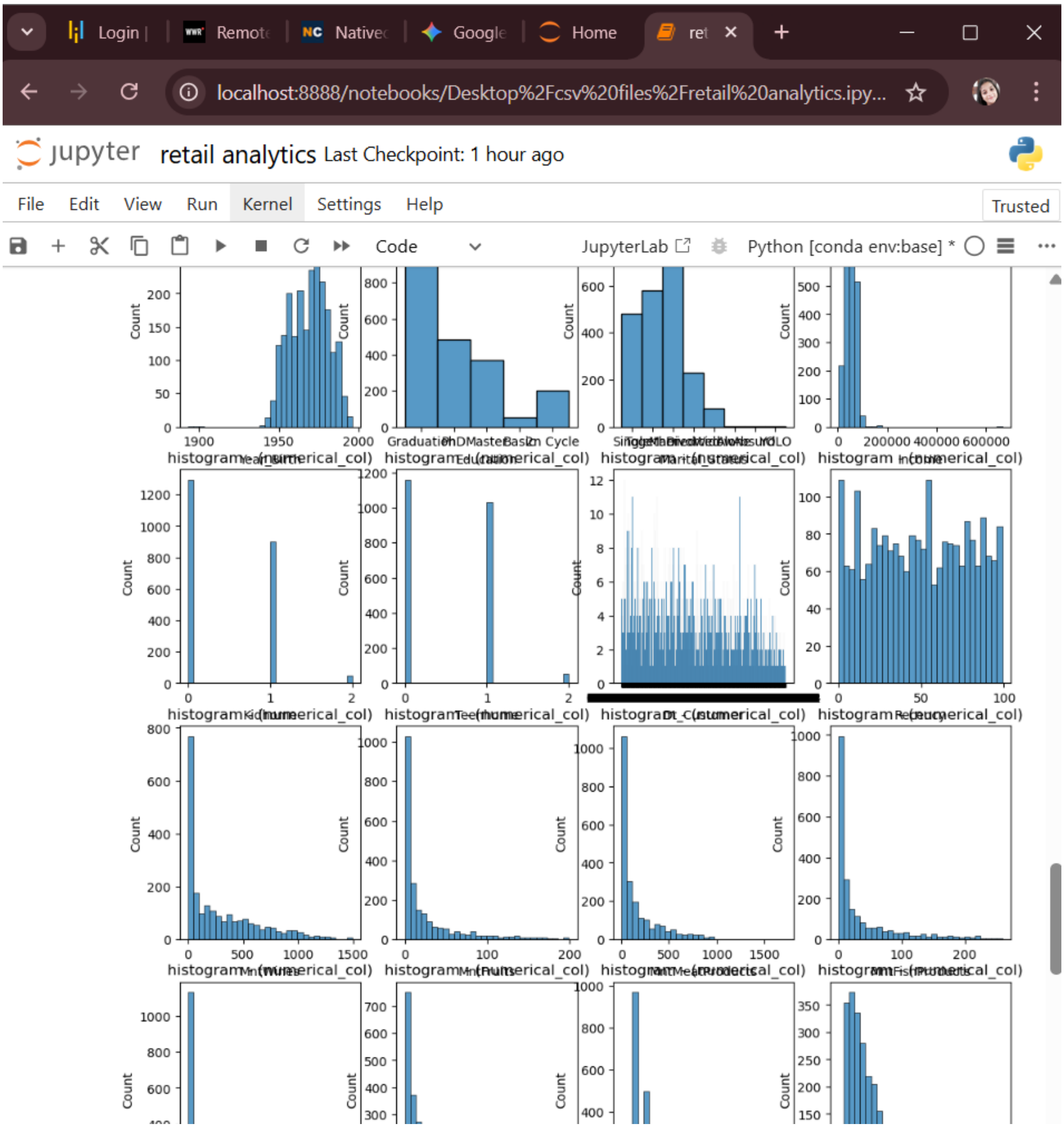
Compute summary statistics for numerical columns (mean, median, min, max, and standard deviation)


Progra
Remote
Nativec
Google
Home
ret
localhost:8888/notebooks/Desktop%2Fcsv%20files%2Fretail%20analytics.ipy...
jupyter
retail analytics Last Checkpoint: 44 minutes ago
File
Edit
View
Run
Kernel
Settings
Help
Trusted
Code
JupyterLab
Python [conda env:base] *

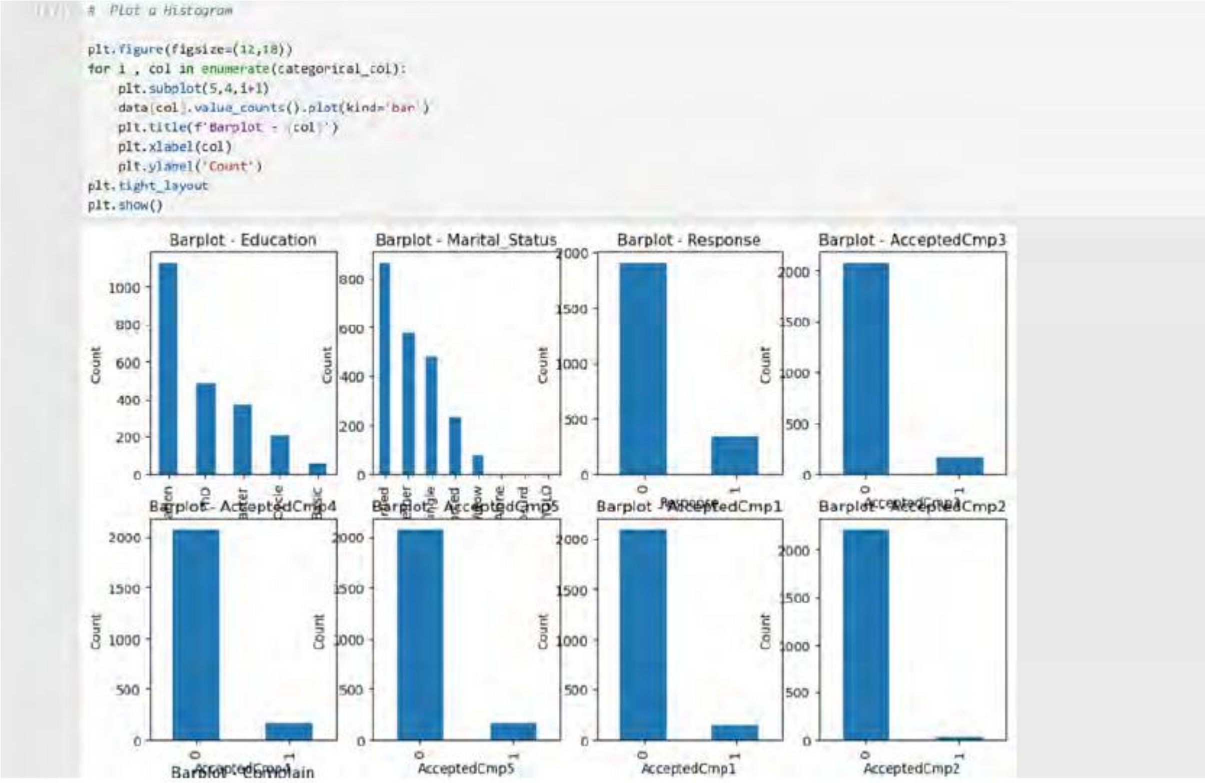
```
[11]: summary_stats = df.describe()
      print(summary_stats)
```

```
                 ID   Year_Birth         Income      Kidhome     Teenhome  \
count   2240.000000  2240.000000    2216.000000  2240.000000  2240.000000
mean    5592.159821  1968.805804   52247.251354     0.444196     0.506250
std     3246.662198    11.984069   25173.076661     0.538398     0.544538
min        0.000000  1893.000000    1730.000000     0.000000     0.000000
25%     2828.250000  1959.000000   35303.000000     0.000000     0.000000
50%     5458.500000  1970.000000   51381.500000     0.000000     0.000000
75%     8427.750000  1977.000000   68522.000000     1.000000     1.000000
max    11191.000000  1996.000000  666666.000000     2.000000     2.000000

           Recency     MntWines    MntFruits  MntMeatProducts  \
count  2240.000000  2240.000000  2240.000000      2240.000000
mean     49.109375   303.935714    26.302232       166.950000
std      28.962453   336.597393    39.773434       225.715373
min       0.000000     0.000000     0.000000         0.000000
25%      24.000000    23.750000     1.000000        16.000000
50%      49.000000   173.500000     8.000000        67.000000
75%      74.000000   504.250000    33.000000       232.000000
max      99.000000  1493.000000   199.000000      1725.000000

       MntFishProducts  ...  AcceptedCmp3  AcceptedCmp4  AcceptedCmp5  \
count      2240.000000  ...   2240.000000   2240.000000   2240.000000
mean         37.525446  ...      0.072768      0.074554      0.072768
std          54.628979  ...      0.259813      0.262728      0.259813
min           0.000000  ...      0.000000      0.000000      0.000000
25%           3.000000  ...      0.000000      0.000000      0.000000
50%          12.000000  ...      0.000000      0.000000      0.000000
75%          50.000000  ...      0.000000      0.000000      0.000000
max         259.000000  ...      1.000000      1.000000      1.000000

       AcceptedCmp1  AcceptedCmp2     Complain  Z_CostContact  Z_Revenue  \
count   2240.000000   2240.000000  2240.000000         2240.0     2240.0
mean       0.064286      0.013393     0.009375            3.0       11.0
```

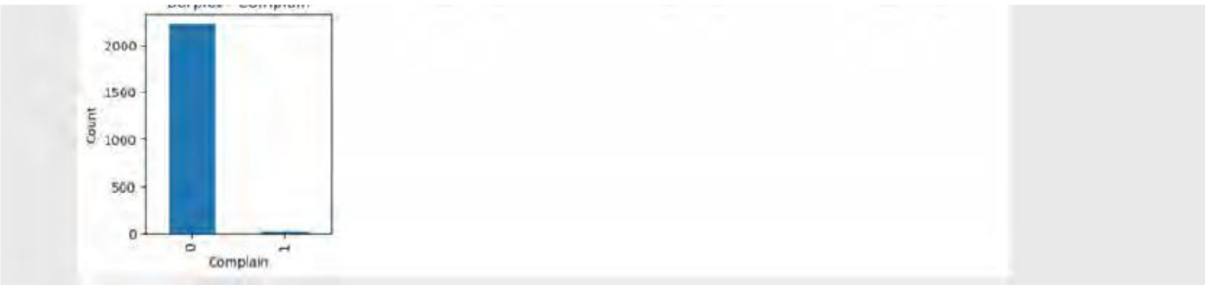
Explore the distribution of numerical variables using histograms or box plots



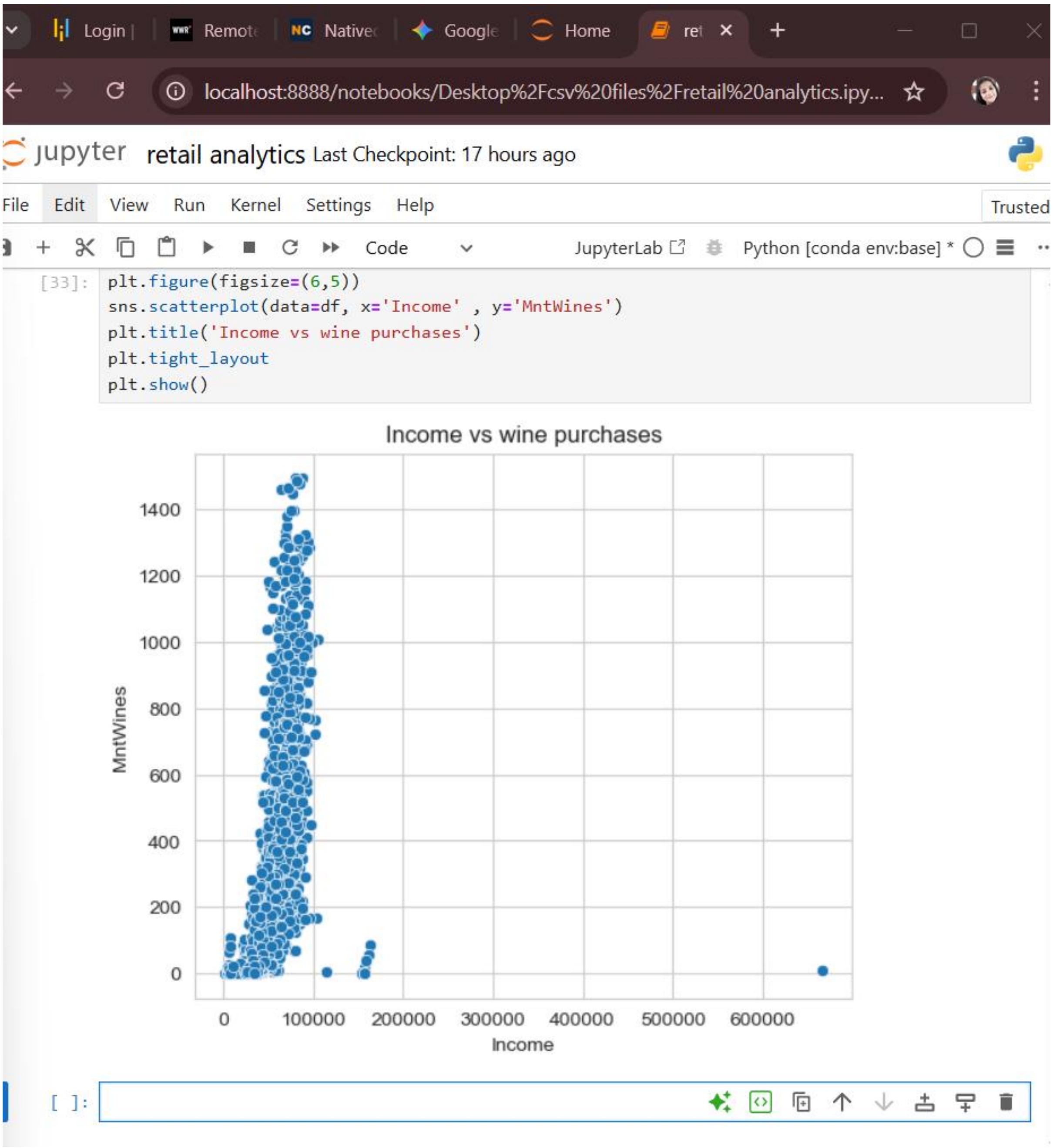
Explore the distribution of each categorical variable using bar plots or pie charts



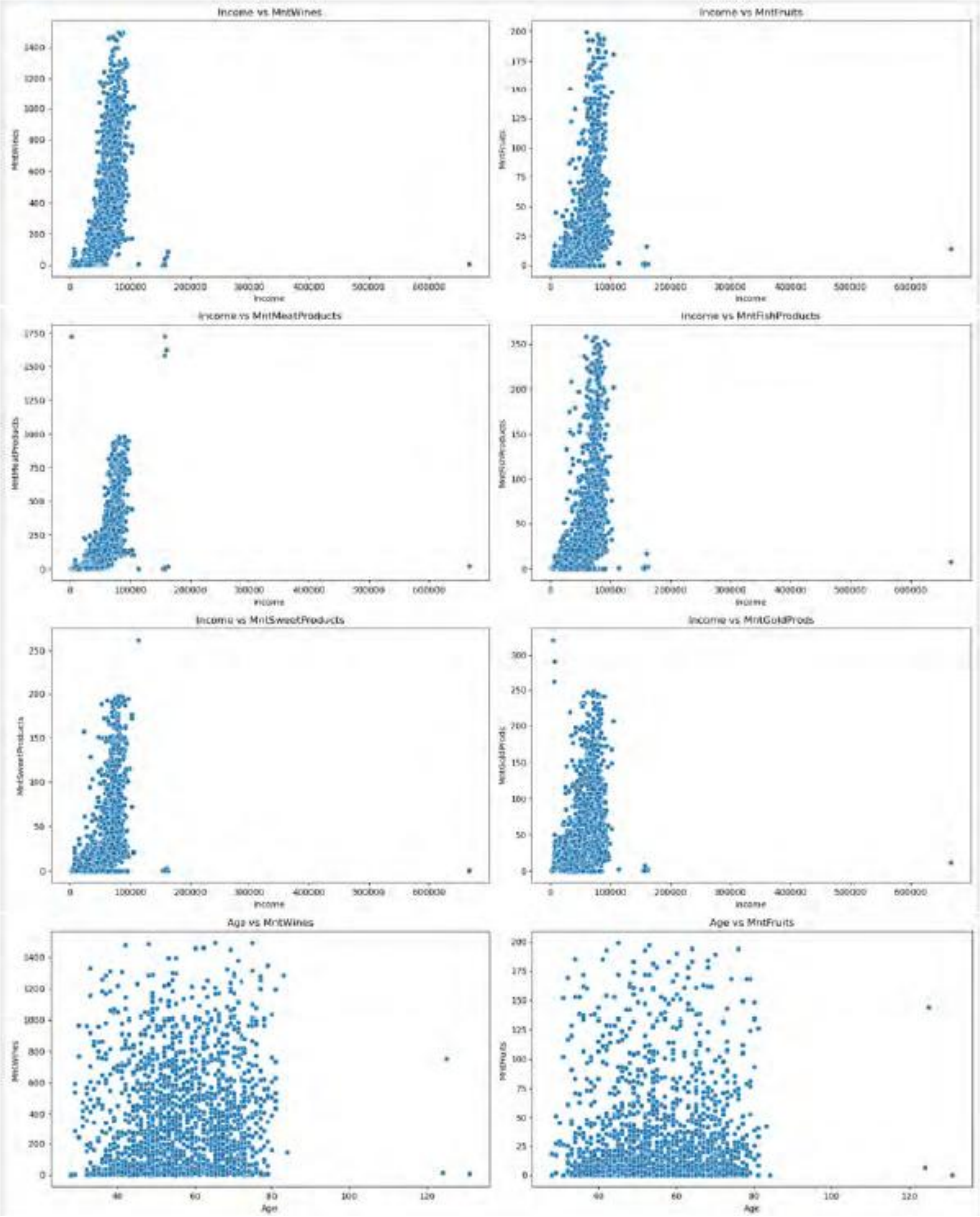
```
# Plot a Histogram

plt.figure(figsize=(12,18))
for i , col in enumerate(categorical_col):
    plt.subplot(5,4,i+1)
    data[col].value_counts().plot(kind='bar')
    plt.title(f'Barplot - {col}')
    plt.xlabel(col)
    plt.ylabel('Count')
plt.tight_layout
plt.show()
```



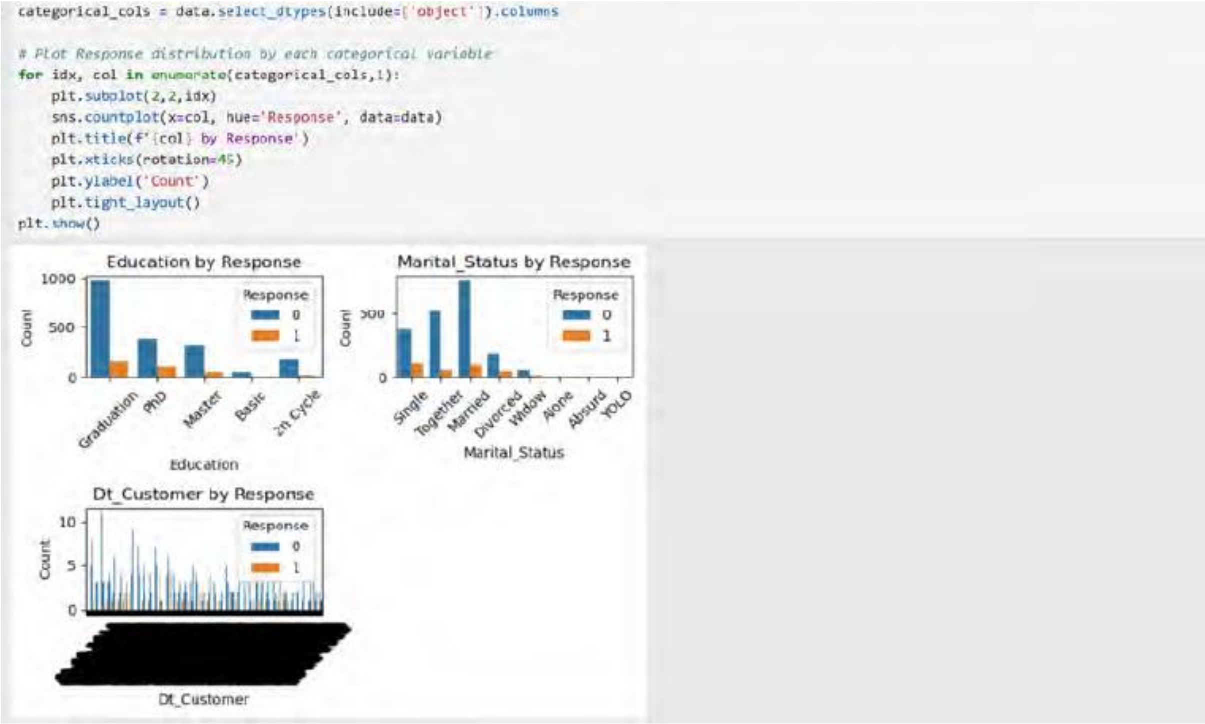
Identify outliers in numerical variables using box plots or scatter plots



```
plt.figure(figsize=(6,5))
sns.scatterplot(data=df, x='Income' , y='MntWines')
plt.title('Income vs wine purchases')
plt.tight_layout
plt.show()
```

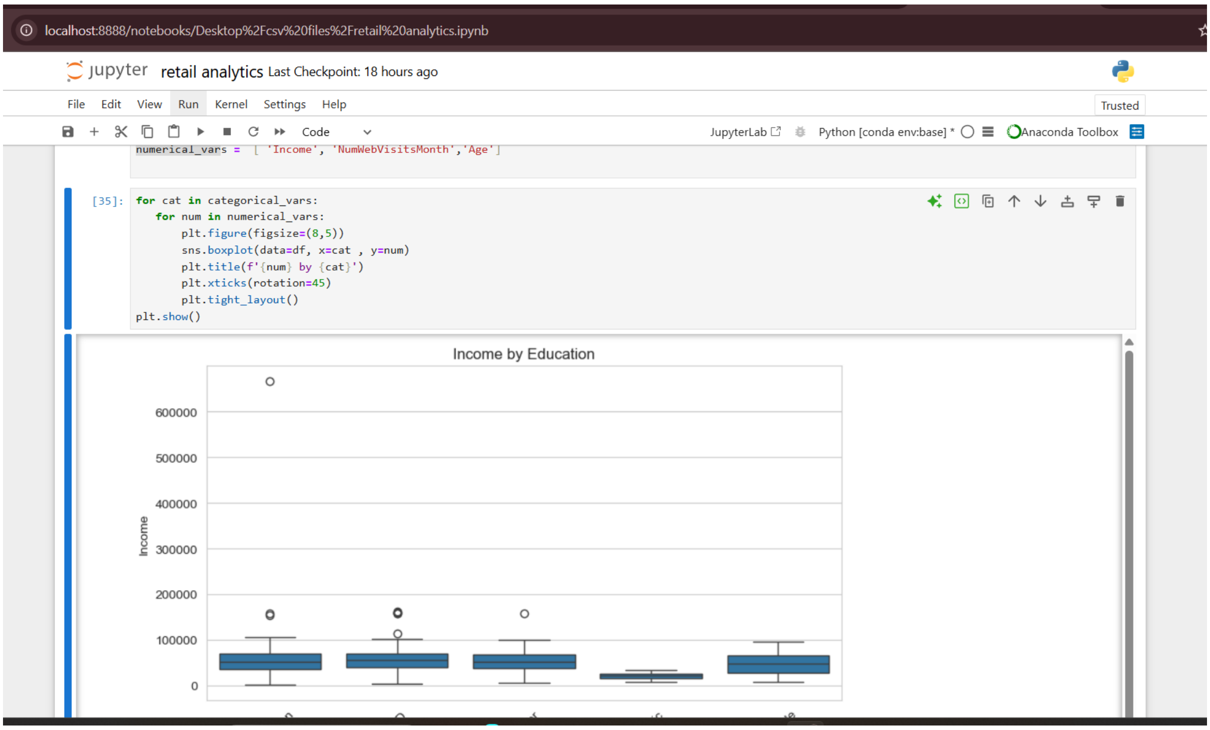


Explore the relationship between categorical variables and the target variable using bar plots or chi-square tests



```
categorical_cols = data.select_dtypes(include=['object']).columns

# Plot Response distribution by each categorical variable
for idx, col in enumerate(categorical_cols,1):
    plt.subplot(2,2,idx)
    sns.countplot(x=col, hue='Response', data=data)
    plt.title(f'{col} by Response')
    plt.xticks(rotation=45)
    plt.ylabel('Count')
    plt.tight_layout()
plt.show()
```

Explore the relationship between numerical and categorical variables using box plots or violin plots

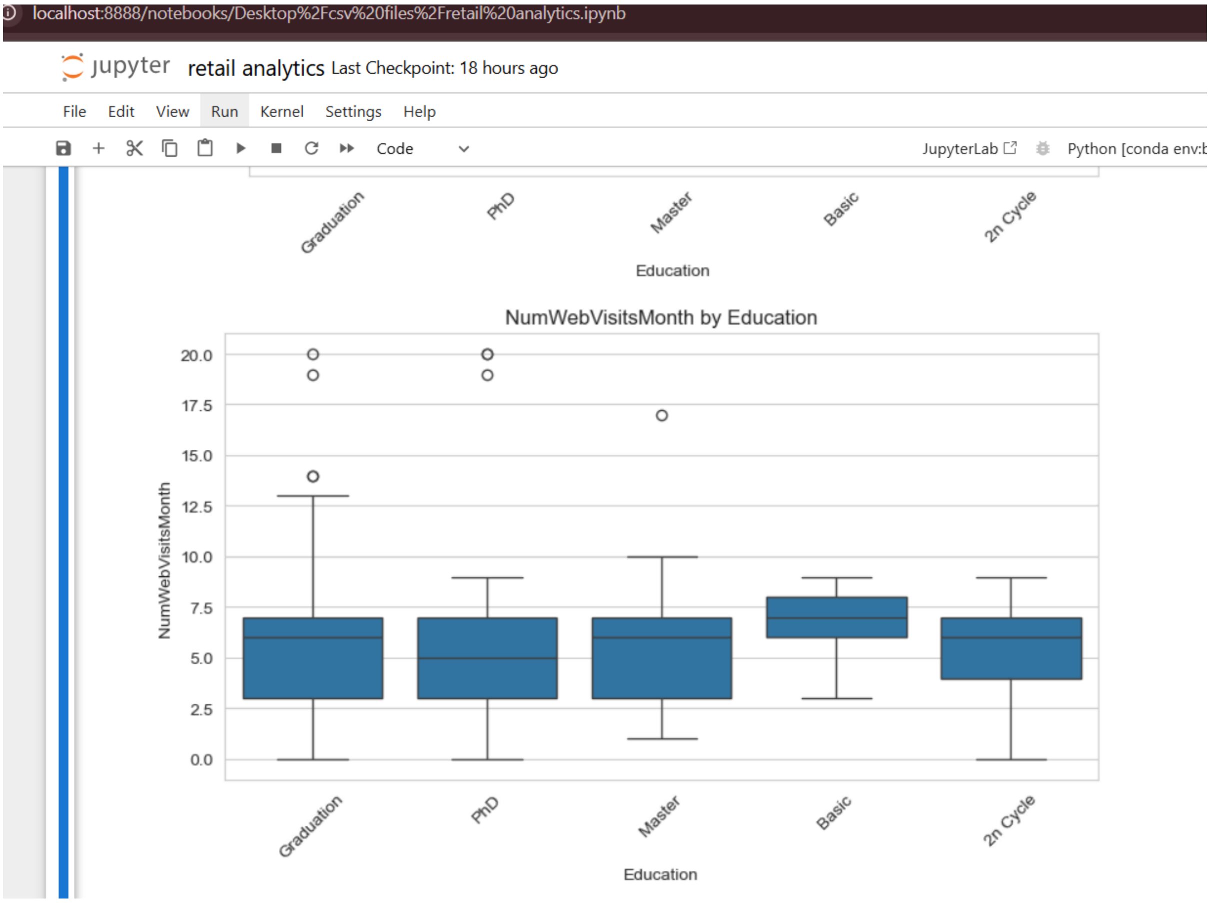


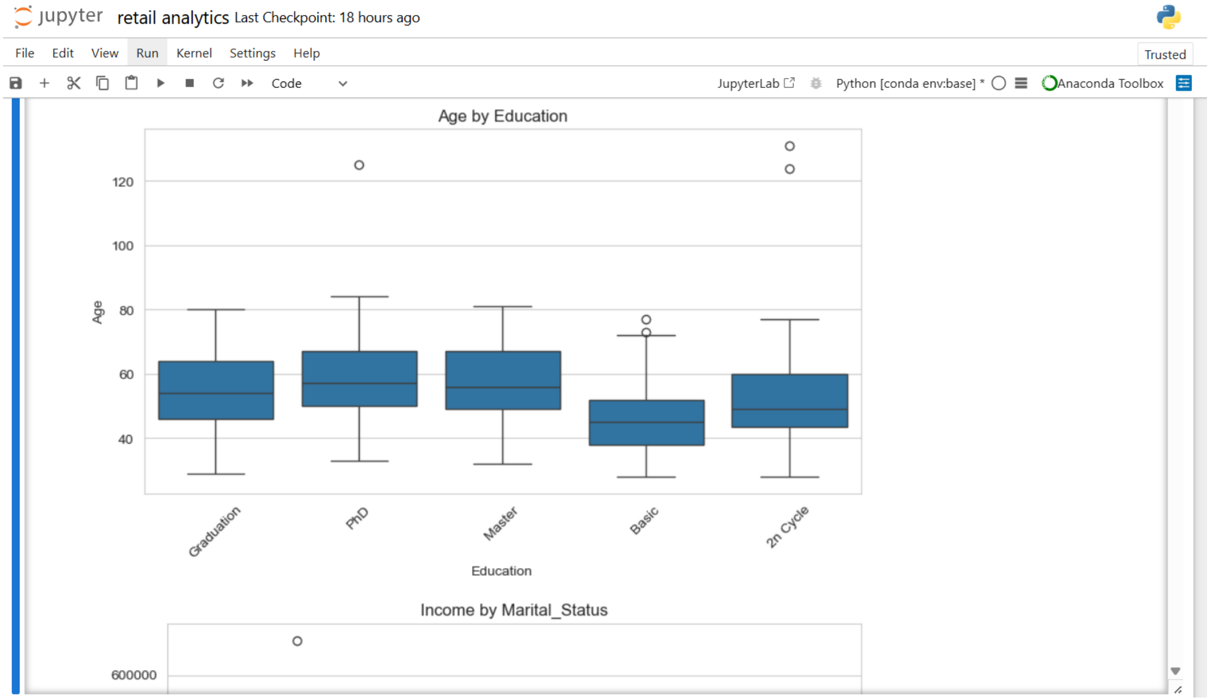
localhost:8888/notebooks/Desktop%2Fcsv%20files%2Fretail%20analytics.ipynb
jupyter
retail analytics Last Checkpoint: 18 hours ago
File
Edit
View
Run
Kernel
Settings
Help
Trusted
Code
JupyterLab
Python [conda env:base] *
Anaconda Toolbox

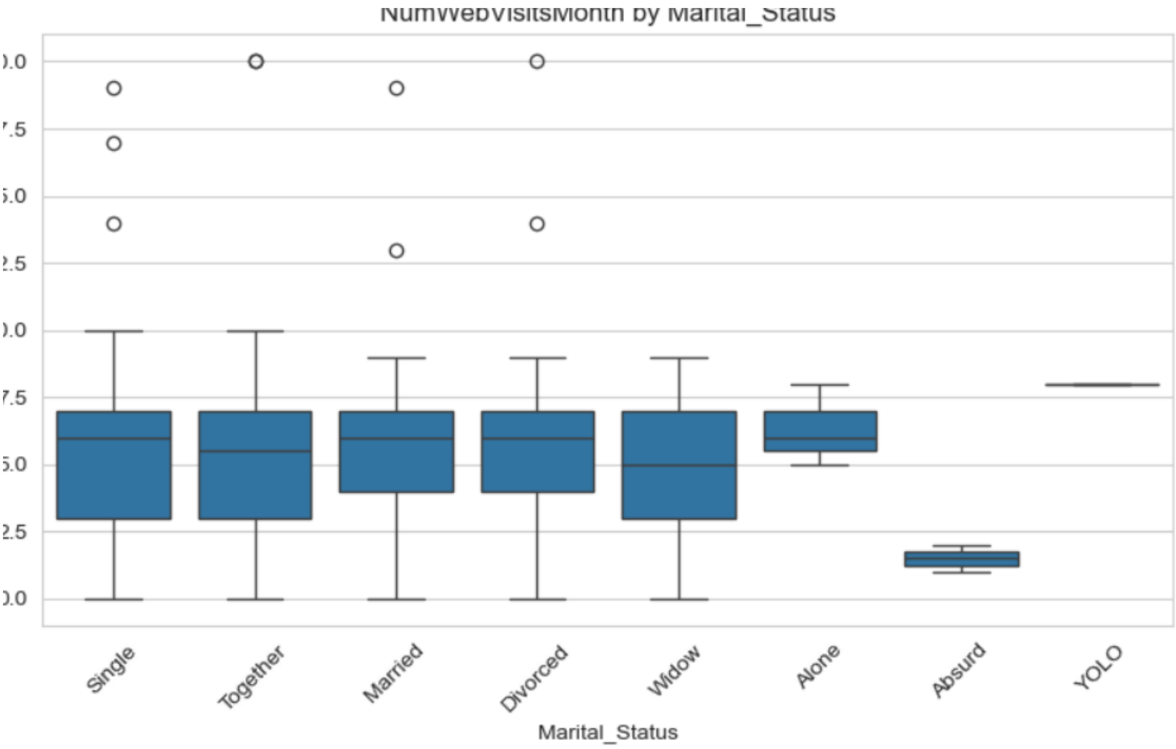
```
numerical_vars = ['Income', 'NumWebVisitsMonth','Age']
```

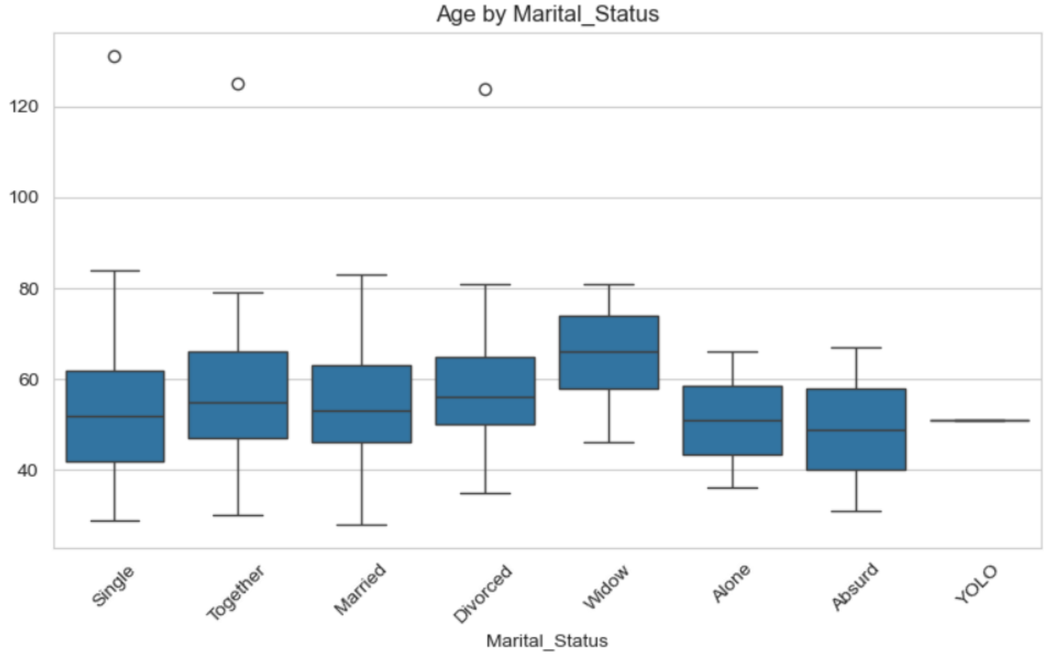
```
[35]: for cat in categorical_vars:
          for num in numerical_vars:
              plt.figure(figsize=(8,5))
              sns.boxplot(data=df, x=cat , y=num)
              plt.title(f'{num} by {cat}')
              plt.xticks(rotation=45)
              plt.tight_layout()
      plt.show()
```

Income by Education
600000
500000
400000
300000
200000
100000
0
Income

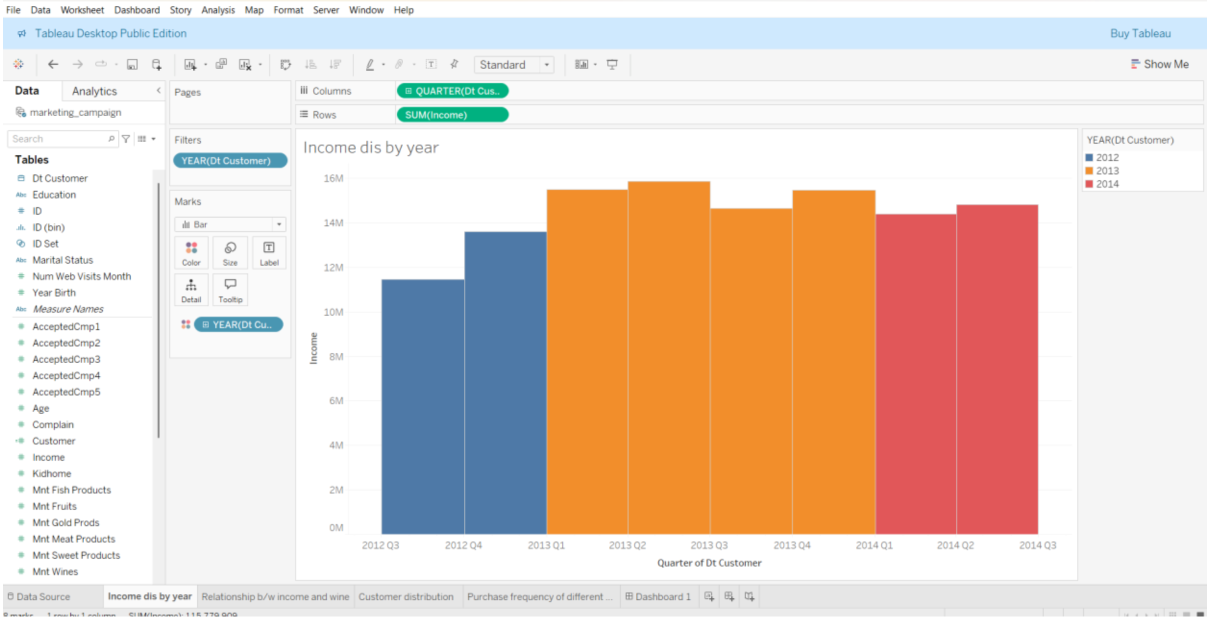

localhost:8888/notebooks/Desktop%2Fcsv%20files%2Fretail%20analytics.ipynb
jupyter
retail analytics Last Checkpoint: 18 hours ago
File
Edit
View
Run
Kernel
Settings
Help
Code
JupyterLab
Python [conda env:t
Graduation
PhD
Master
Basic
2n Cycle
Education
NumWebVisitsMonth by Education
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
NumWebVisitsMonth
Graduation
PhD
Master
Basic
2n Cycle
Education


jupyter
retail analytics Last Checkpoint: 18 hours ago
File
Edit
View
Run
Kernel
Settings
Help
Trusted
Code
JupyterLab
Python [conda env:base] *
Anaconda Toolbox
Age by Education
120
100
80
60
40
Age
Graduation
PhD
Master
Basic
2n Cycle
Education
Income by Marital_Status
600000





Task 1: Visualize Customer Income Distribution grouped by Year of Registration



Columns: QUARTER(Dt Cus..

Rows: SUM(Income)

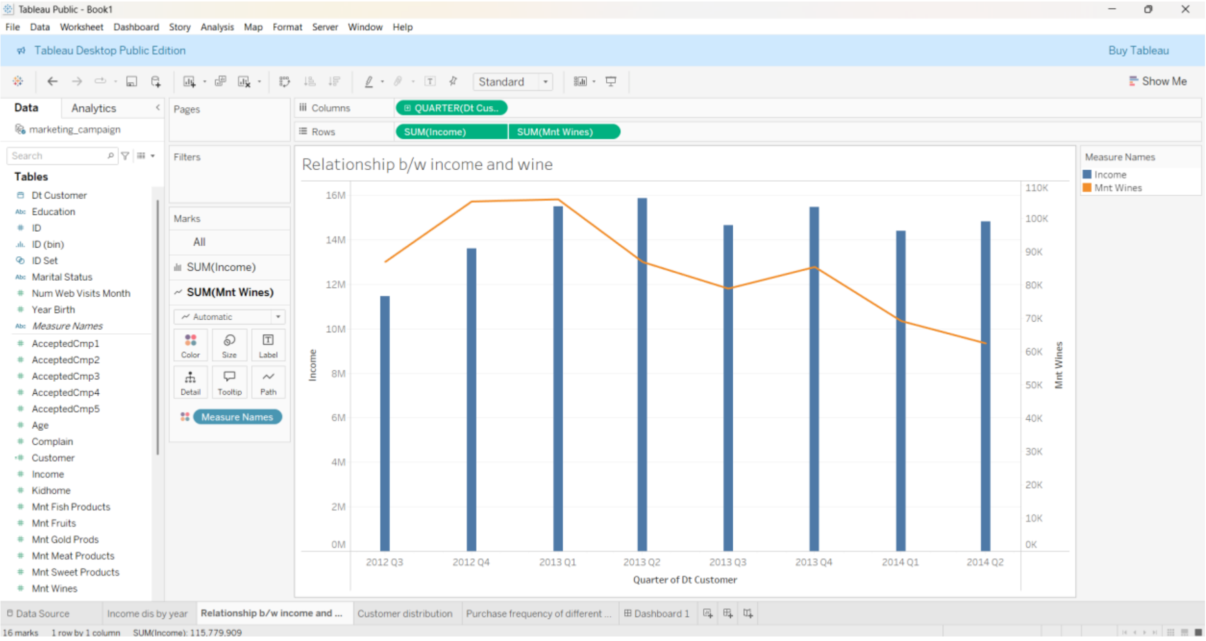
Filters: YEAR(Dt Customer)

Marks: Bar

Color: YEAR(Dt Cu..

8 marks 1 row by 1 column SUM(Income): 115,779,909

Task 2: Examine the breakdown of education levels and marital status using side-by-side circles to represent count



Columns: QUARTER(Dt Cus..

Rows: SUM(Income) SUM(Mnt Wines)

Marks: SUM(Income)

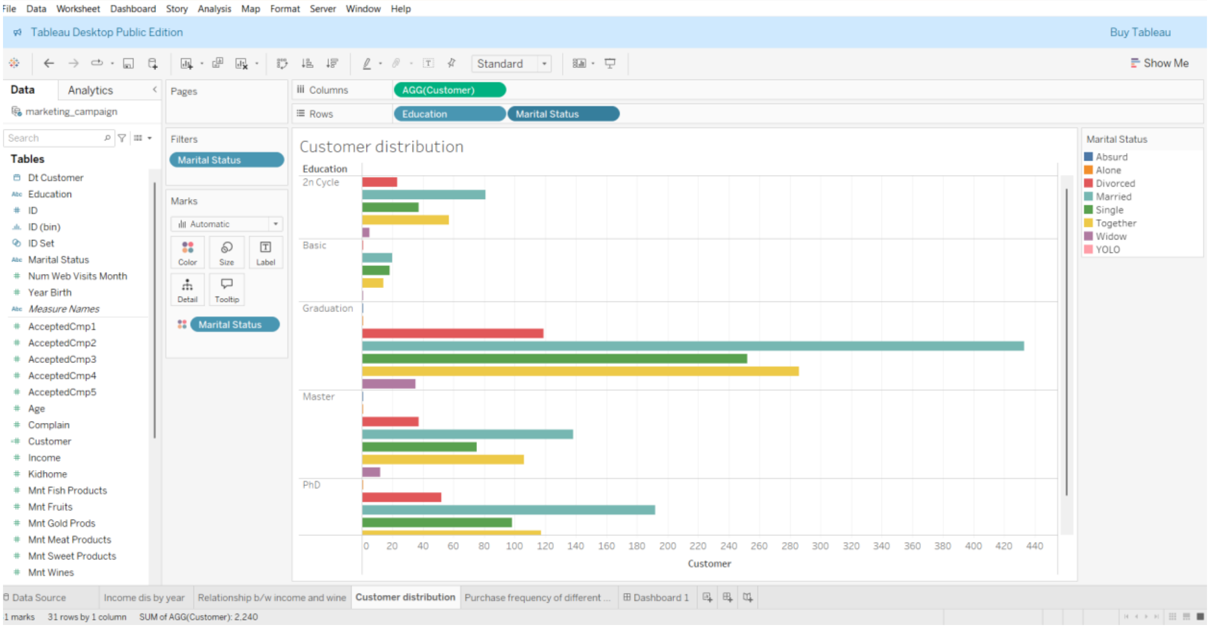
SUM(Mnt Wines)

Automatic

Color: Measure Names

16 marks 1 row by 1 column SUM(Income): 115,779,909

Task 3: Explore the relationship between Income and Wine Spending



Columns: AGG(Customer)

Rows: Education, Marital Status

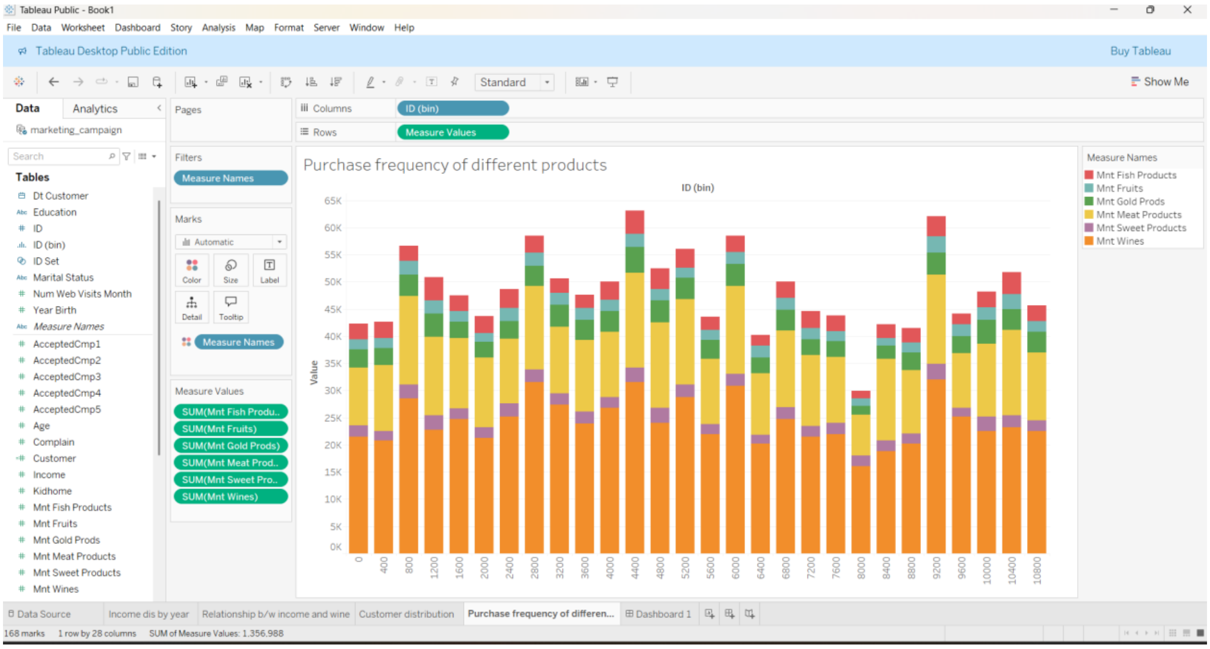
Filters: Marital Status

Marks: Automatic

Color: Marital Status

1 marks 31 rows by 1 column SUM of AGG(Customer): 2,240

Task 4: Analyze the frequency of purchases across various product categories



Columns: ID (bin)

Rows: Measure Values

Filters: Measure Names

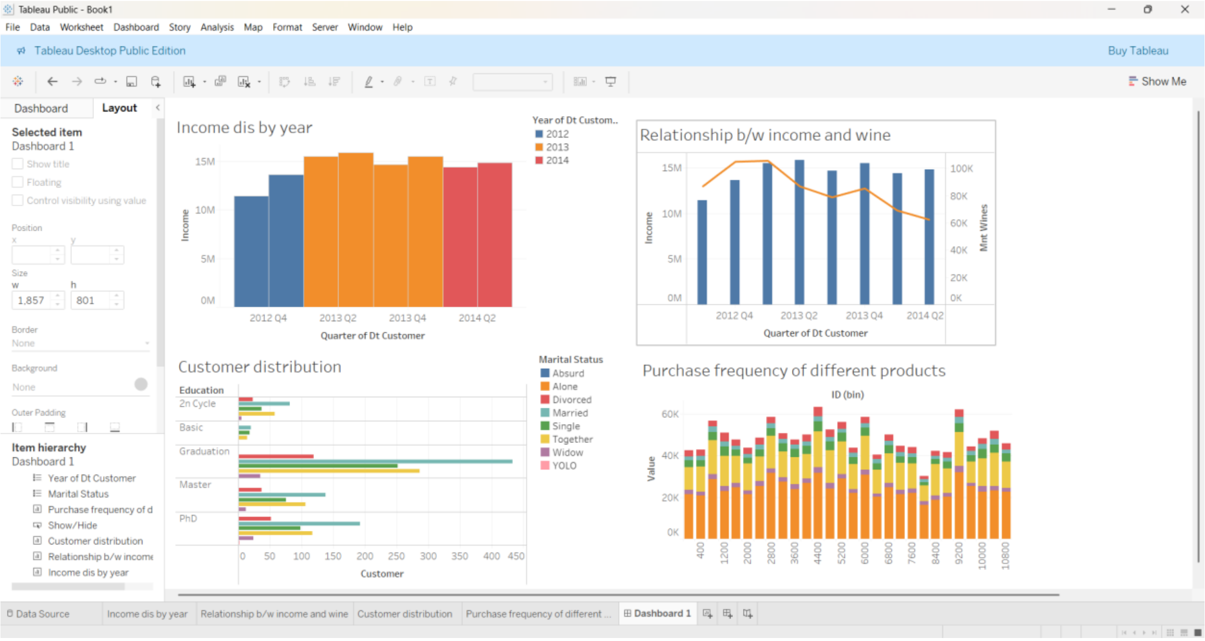
Marks: Automatic

Color: Measure Names

Measure Values:
- SUM(Mnt Fish Produ..
- SUM(Mnt Fruits)
- SUM(Mnt Gold Prods)
- SUM(Mnt Meat Prod..
- SUM(Mnt Sweet Pro..
- SUM(Mnt Wines)

168 marks 1 row by 28 columns SUM of Measure Values: 1,356,988

Task 5: Create a dashboard using all the visualizations



Selected item
Dashboard 1

□ Show title

□ Floating

□ Control visibility using value

Position
x
y

Size
w 1,857
h 801

Border
None

Background
None

Outer Padding

Item hierarchy
Dashboard 1
- Year of Dt Customer
- Marital Status
- Purchase frequency of d
- Show/Hide
- Customer distribution
- Relationship b/w income
- Income dis by year