# DATA ANALYSIS PYTHON PROJECT - CUSTOMER PERSONALITY ANALYSIS

## Defination

In [ ]:
```
Customer Personality Analysis is the process of understanding customer behavior, preferences
The goal is to group customers into categories (personas) so businesses can target them bette
```

## Objective

In [ ]:
```
1.Understand customer behavior:
   To analyze how customers behave — their spending patterns, purchase frequency, and shop

2.Identify customer segments
   To group customers into clusters (personas) such as high spenders, frequent buyers, budget

3.Improve marketing strategies
   To help the company target the right customers with better offers, ads, and campaigns.

4.Personalize customer experience
   To understand individual customer needs and provide personalized services, product recom

5.Analyze demographic influence
   To study how age, gender, income, and location affect customer purchasing behavior.

6. Increase customer retention
   To identify loyal customers and design strategies to keep them engaged.

7.Predict future behavior
   To use past data to predict which customers will buy again, spend more, or respond to offer

8.Improve business decision-making
   To provide insights that help in planning sales, product demand, and customer outreach str
To analyze how customers behave — their spending patterns, purchase frequency, and shopp
```

## Import Laibraries

In [2]:
```python
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [3]:
```python
cust = pd.read_csv(r"C:\Users\RANI\Downloads\customer_personality_analysis_reduced(project
```

In [4]:
```python
cust
```

Out[4]:

| | CustomerID | Age | Gender | City | AnnualIncomeINR | PurchaseFrequency_ |
|---|---|---|---|---|---|---|
| **0** | CUST0001 | 33 | Female | Lucknow | 910116 | |
| **1** | CUST0002 | 38 | Male | Kolkata | 924582 | |
| **2** | CUST0003 | 47 | Male | Mysuru | 829877 | |
| **3** | CUST0004 | 56 | Male | Mysuru | 1001629 | |
| **4** | CUST0005 | 39 | Other | Vadodara | 636008 | |
| **...** | ... | ... | ... | ... | ... | |
| **995** | CUST0996 | 28 | Male | Coimbatore | 364050 | |
| **996** | CUST0997 | 32 | Female | Indore | 438940 | |
| **997** | CUST0998 | 41 | Male | Mumbai | 550890 | |
| **998** | CUST0999 | 24 | Female | Indore | 871223 | |
| **999** | CUST1000 | 29 | Female | Delhi | 549520 | |

1000 rows × 17 columns

## Data Inspection

In [5]:

```
cust.head()
```

Out[5]:

| | CustomerID | Age | Gender | City | AnnualIncomeINR | PurchaseFrequency_per_ |
|---|---|---|---|---|---|---|
| **0** | CUST0001 | 33 | Female | Lucknow | 910116 | |
| **1** | CUST0002 | 38 | Male | Kolkata | 924582 | |
| **2** | CUST0003 | 47 | Male | Mysuru | 829877 | |
| **3** | CUST0004 | 56 | Male | Mysuru | 1001629 | |
| **4** | CUST0005 | 39 | Other | Vadodara | 636008 | |

In [6]:

```
cust.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 17 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   CustomerID                    1000 non-null   object
 1   Age                           1000 non-null   int64
 2   Gender                        1000 non-null   object
 3   City                          1000 non-null   object
 4   AnnualIncomeINR               1000 non-null   int64
 5   PurchaseFrequency_per_month   1000 non-null   int64
 6   AvgOrderValueINR              1000 non-null   int64
 7   EstimatedYearlySpendINR       1000 non-null   int64
 8   PreferredChannel              1000 non-null   object
 9   Openness                      1000 non-null   float64
 10  Conscientiousness             1000 non-null   float64
 11  Extraversion                  1000 non-null   float64
 12  Agreeableness                 1000 non-null   float64
 13  Neuroticism                   1000 non-null   float64
 14  PersonalityLabel              1000 non-null   object
 15  CustomerSegment               1000 non-null   object
 16  LastPurchaseDate              1000 non-null   object
dtypes: float64(5), int64(5), object(7)
memory usage: 132.9+ KB
```

In [7]:
```
cust.tail()
```

Out[7]:

| | CustomerID | Age | Gender | City | AnnualIncomeINR | PurchaseFrequency_ |
|---|---|---|---|---|---|---|
| **995** | CUST0996 | 28 | Male | Coimbatore | 364050 | |
| **996** | CUST0997 | 32 | Female | Indore | 438940 | |
| **997** | CUST0998 | 41 | Male | Mumbai | 550890 | |
| **998** | CUST0999 | 24 | Female | Indore | 871223 | |
| **999** | CUST1000 | 29 | Female | Delhi | 549520 | |

In [8]:
```
cust.describe()
```

Out[8]:

| | Age | AnnualIncomeINR | PurchaseFrequency_per_month | AvgOrderVal |
|---|---|---|---|---|
| **count** | 1000.000000 | 1.000000e+03 | 1000.000000 | 1000.0 |
| **mean** | 43.773000 | 6.039986e+05 | 2.871000 | 2133.0 |
| **std** | 15.640463 | 2.951974e+05 | 2.875522 | 1312.2 |
| **min** | 18.000000 | 6.000000e+04 | 0.000000 | 200.0 |
| **25%** | 30.000000 | 3.832620e+05 | 1.000000 | 1051.5 |
| **50%** | 43.000000 | 6.004025e+05 | 2.000000 | 2028.0 |
| **75%** | 58.000000 | 8.142092e+05 | 4.000000 | 3128.0 |
| **max** | 70.000000 | 1.546285e+06 | 23.000000 | 6883.0 |

In [9]:
```
cust.columns
```

Out[9]:
```
Index(['CustomerID', 'Age', 'Gender', 'City', 'AnnualIncomeINR',
       'PurchaseFrequency_per_month', 'AvgOrderValueINR',
       'EstimatedYearlySpendINR', 'PreferredChannel', 'Openness',
       'Conscientiousness', 'Extraversion', 'Agreeableness', 'Neuroticism',
       'PersonalityLabel', 'CustomerSegment', 'LastPurchaseDate'],
      dtype='object')
```

In [10]:
```
cust.shape
```

Out[10]:
```
(1000, 17)
```

In [11]:
```
cust.isna().sum()
```

Out[11]:
```
CustomerID                     0
Age                            0
Gender                         0
City                           0
AnnualIncomeINR                0
PurchaseFrequency_per_month    0
AvgOrderValueINR               0
EstimatedYearlySpendINR        0
PreferredChannel               0
Openness                       0
Conscientiousness              0
Extraversion                   0
Agreeableness                  0
Neuroticism                    0
PersonalityLabel               0
CustomerSegment                0
LastPurchaseDate               0
dtype: int64
```

In [12]:
```
cust.dtypes
```

Out[12]:

```
CustomerID                   object
Age                           int64
Gender                       object
City                         object
AnnualIncomeINR               int64
PurchaseFrequency_per_month   int64
AvgOrderValueINR              int64
EstimatedYearlySpendINR       int64
PreferredChannel             object
Openness                    float64
Conscientiousness           float64
Extraversion                float64
Agreeableness               float64
Neuroticism                 float64
PersonalityLabel             object
CustomerSegment              object
LastPurchaseDate             object
dtype: object
```

## Data Cleaning

In [16]:

```python
cust = pd.read_csv(r"C:\Users\RANI\Downloads\customer_personality_analysis_reduced(project
cust
```

Out[16]:

| | CustomerID | Age | Gender | City | AnnualIncomeINR | PurchaseFrequency_ |
|---|---|---|---|---|---|---|
| **0** | CUST0001 | 33 | Female | Lucknow | 910116 | |
| **1** | CUST0002 | 38 | Male | Kolkata | 924582 | |
| **2** | CUST0003 | 47 | Male | Mysuru | 829877 | |
| **3** | CUST0004 | 56 | Male | Mysuru | 1001629 | |
| **4** | CUST0005 | 39 | Other | Vadodara | 636008 | |
| **...** | ... | ... | ... | ... | ... | |
| **995** | CUST0996 | 28 | Male | Coimbatore | 364050 | |
| **996** | CUST0997 | 32 | Female | Indore | 438940 | |
| **997** | CUST0998 | 41 | Male | Mumbai | 550890 | |
| **998** | CUST0999 | 24 | Female | Indore | 871223 | |
| **999** | CUST1000 | 29 | Female | Delhi | 549520 | |

1000 rows × 17 columns

In [19]:

```python
cust.isna().sum()
```

Out[19]:
```
CustomerID                    0
Age                           0
Gender                        0
City                          0
AnnualIncomeINR               0
PurchaseFrequency_per_month   0
AvgOrderValueINR              0
EstimatedYearlySpendINR       0
PreferredChannel              0
Openness                      0
Conscientiousness             0
Extraversion                  0
Agreeableness                 0
Neuroticism                   0
PersonalityLabel              0
CustomerSegment               0
LastPurchaseDate              0
dtype: int64
```

In [20]:
```
cust.drop_duplicates()
```

Out[20]:

| | CustomerID | Age | Gender | City | AnnualIncomeINR | PurchaseFrequency_ |
|---|---|---|---|---|---|---|
| **0** | CUST0001 | 33 | Female | Lucknow | 910116 | |
| **1** | CUST0002 | 38 | Male | Kolkata | 924582 | |
| **2** | CUST0003 | 47 | Male | Mysuru | 829877 | |
| **3** | CUST0004 | 56 | Male | Mysuru | 1001629 | |
| **4** | CUST0005 | 39 | Other | Vadodara | 636008 | |
| **...** | ... | ... | ... | ... | ... | |
| **995** | CUST0996 | 28 | Male | Coimbatore | 364050 | |
| **996** | CUST0997 | 32 | Female | Indore | 438940 | |
| **997** | CUST0998 | 41 | Male | Mumbai | 550890 | |
| **998** | CUST0999 | 24 | Female | Indore | 871223 | |
| **999** | CUST1000 | 29 | Female | Delhi | 549520 | |

1000 rows × 17 columns

In [21]:
```
cust.dropna(inplace=True)
cust
```

Out[21]:

| | CustomerID | Age | Gender | City | AnnualIncomeINR | PurchaseFrequency_ |
|---|---|---|---|---|---|---|
| **0** | CUST0001 | 33 | Female | Lucknow | 910116 | |
| **1** | CUST0002 | 38 | Male | Kolkata | 924582 | |
| **2** | CUST0003 | 47 | Male | Mysuru | 829877 | |
| **3** | CUST0004 | 56 | Male | Mysuru | 1001629 | |
| **4** | CUST0005 | 39 | Other | Vadodara | 636008 | |
| **...** | ... | ... | ... | ... | ... | |
| **995** | CUST0996 | 28 | Male | Coimbatore | 364050 | |
| **996** | CUST0997 | 32 | Female | Indore | 438940 | |
| **997** | CUST0998 | 41 | Male | Mumbai | 550890 | |
| **998** | CUST0999 | 24 | Female | Indore | 871223 | |
| **999** | CUST1000 | 29 | Female | Delhi | 549520 | |

1000 rows × 17 columns

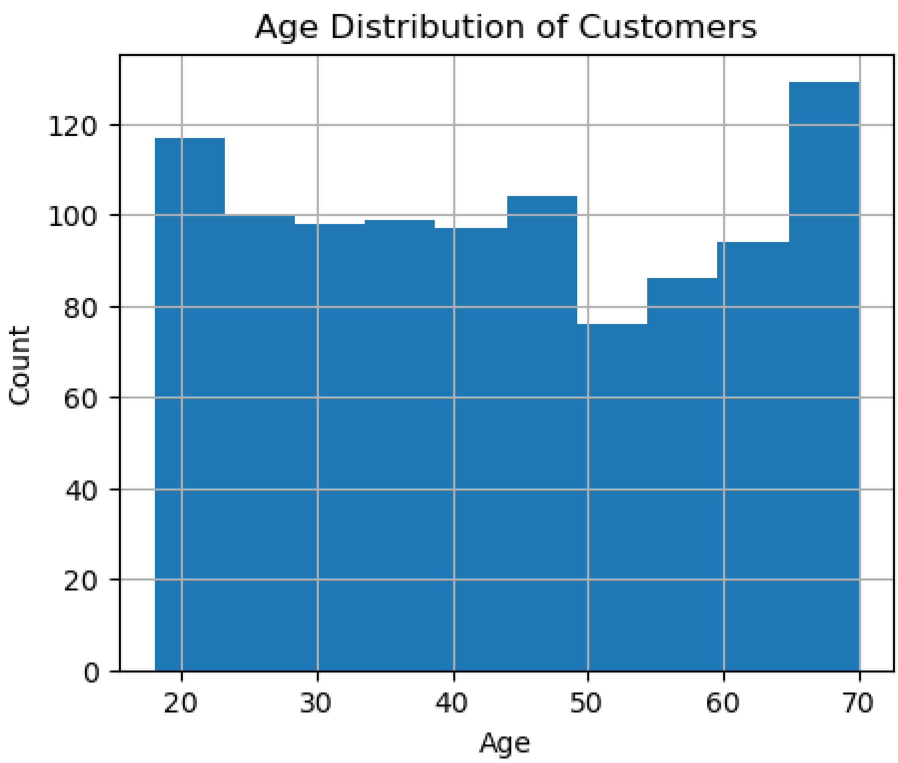
## EDA (Exploratory Data Analysis)

In [13]:

```
print(cust['Age'].describe())

cust['Age'].hist(figsize=(5,4))
plt.title("Age Distribution of Customers")
plt.xlabel("Age")
plt.ylabel("Count")
plt.show()
```

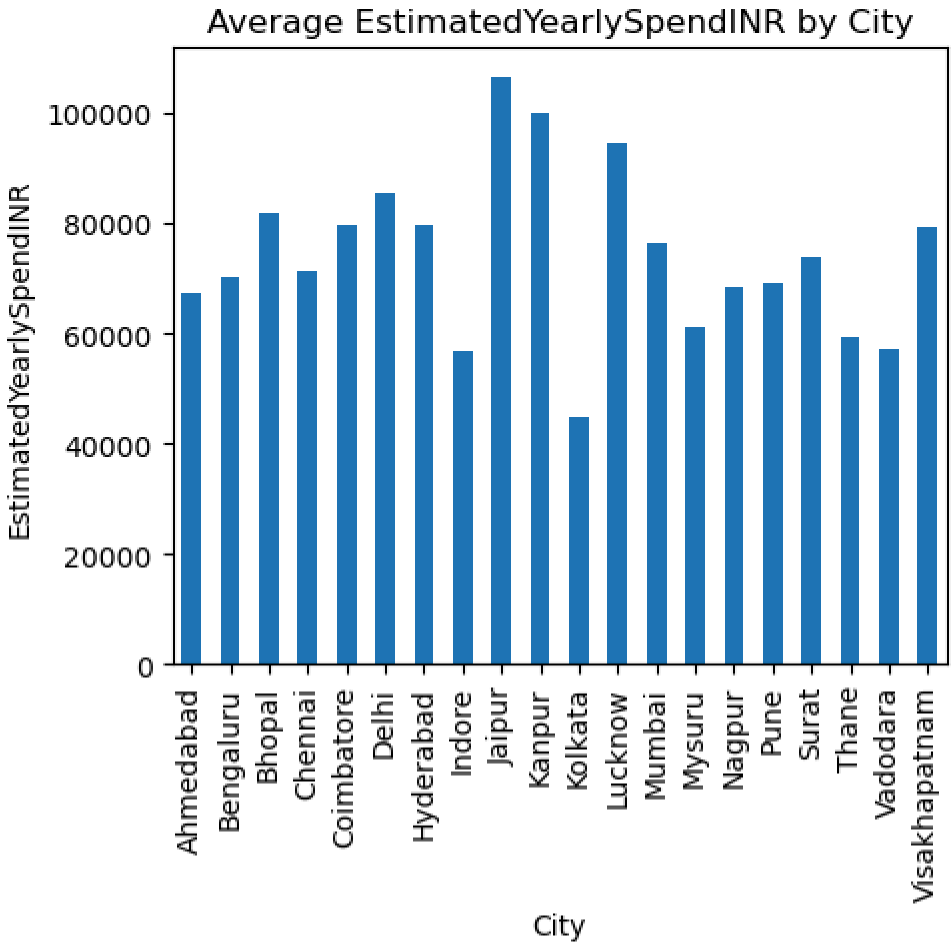
```
count    1000.000000
mean       43.773000
std        15.640463
min        18.000000
25%        30.000000
50%        43.000000
75%        58.000000
max        70.000000
Name: Age, dtype: float64
```




In [17]:
```
city_spending = cust.groupby("City")["EstimatedYearlySpendINR"].mean()
print(city_spending)

city_spending.plot(kind='bar', figsize=(5,4))
plt.title("Average EstimatedYearlySpendINR by City")
plt.ylabel("EstimatedYearlySpendINR")
plt.show()
```

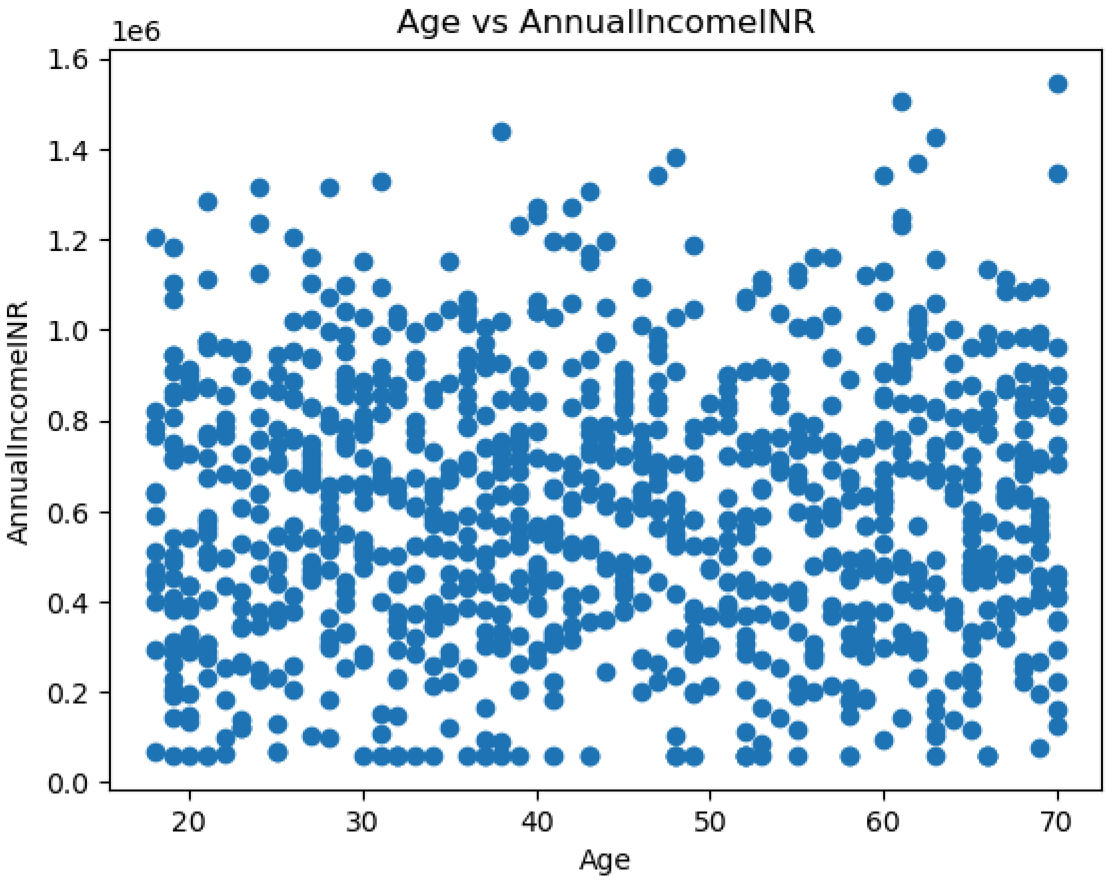
```
City
Ahmedabad        67193.750000
Bengaluru        70219.921569
Bhopal           81696.071429
Chennai          71467.113208
Coimbatore       79600.923077
Delhi            85515.822222
Hyderabad        79696.894737
Indore           56624.087719
Jaipur          106466.648649
Kanpur           99873.127660
Kolkata          44866.111111
Lucknow          94595.389831
Mumbai           76259.169811
Mysuru           61266.384615
Nagpur           68509.166667
Pune             69045.857143
Surat            73815.227273
Thane            59171.523810
Vadodara         56991.372093
Visakhapatnam    79347.875000
Name: EstimatedYearlySpendINR, dtype: float64
```



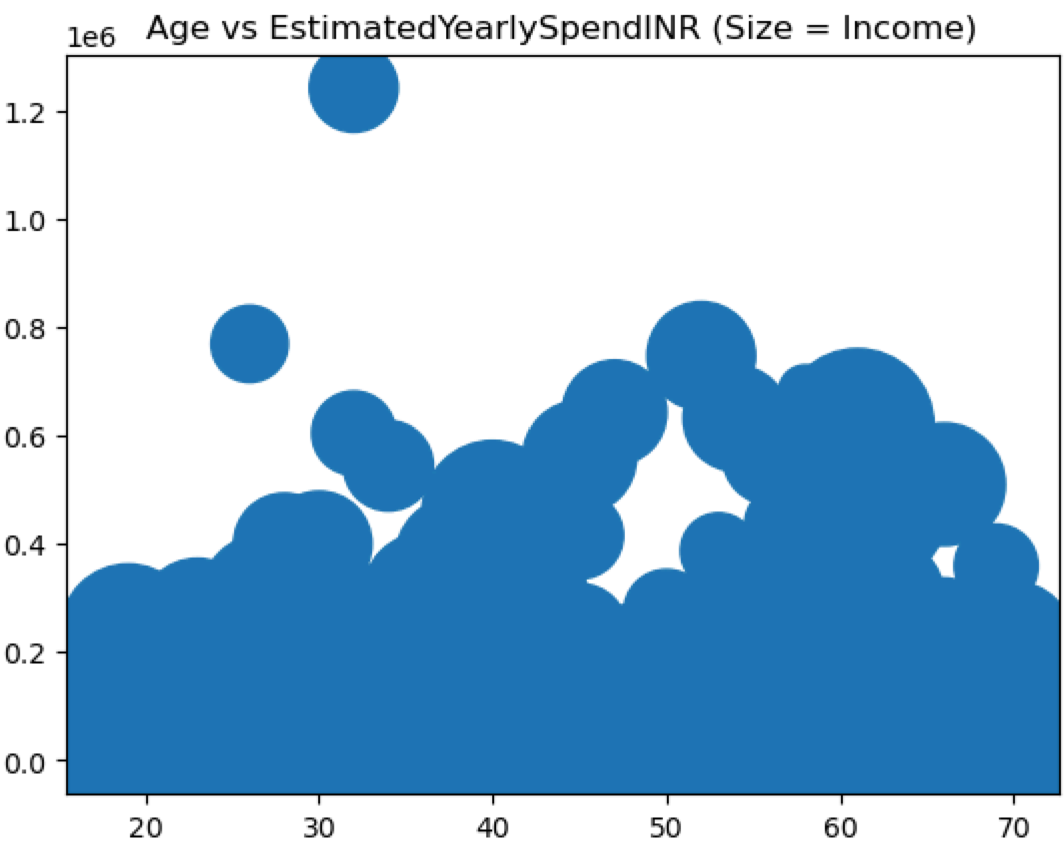

## Visualization

In [18]:

```
plt.scatter(cust['Age'], cust['AnnualIncomeINR'])
plt.xlabel("Age")
plt.ylabel("AnnualIncomeINR")
plt.title("Age vs AnnualIncomeINR")
plt.show()
```

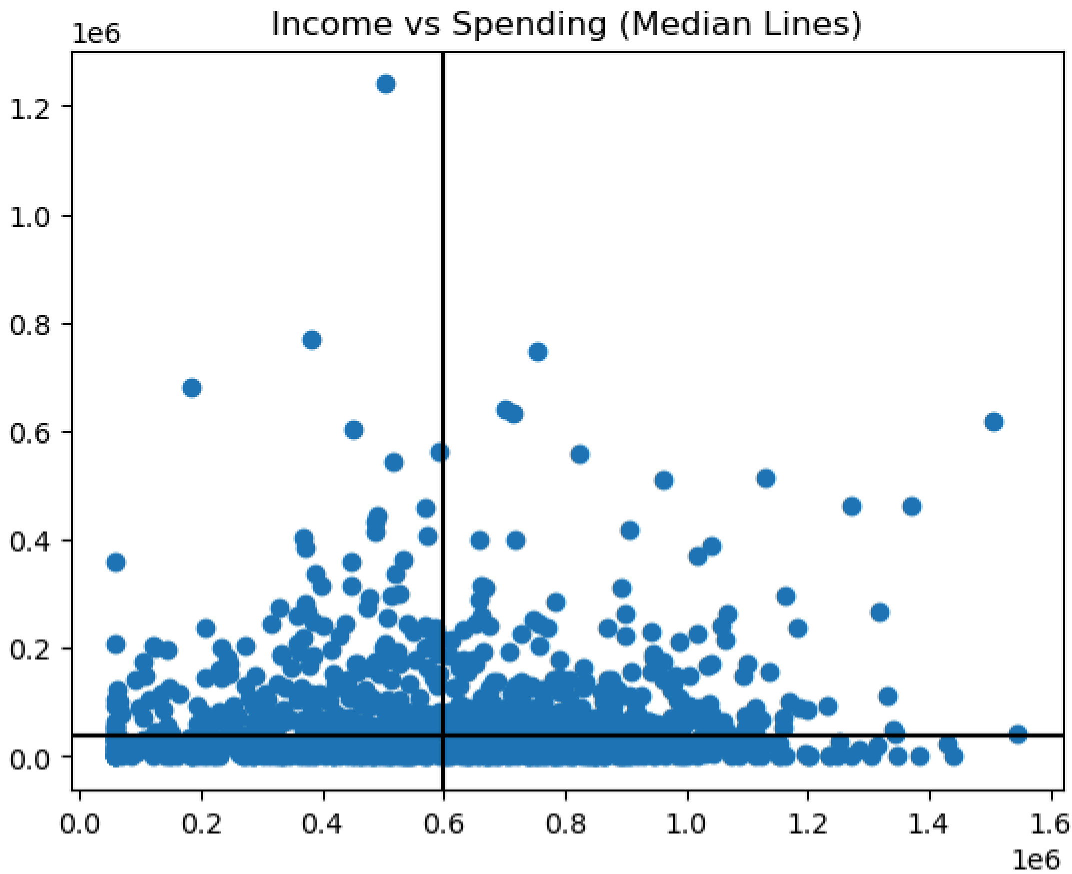



In [20]:
```
plt.scatter(cust['Age'], cust['EstimatedYearlySpendINR'], s=cust['AnnualIncomeINR']/500)
plt.title("Age vs EstimatedYearlySpendINR (Size = Income)")
plt.show()
```

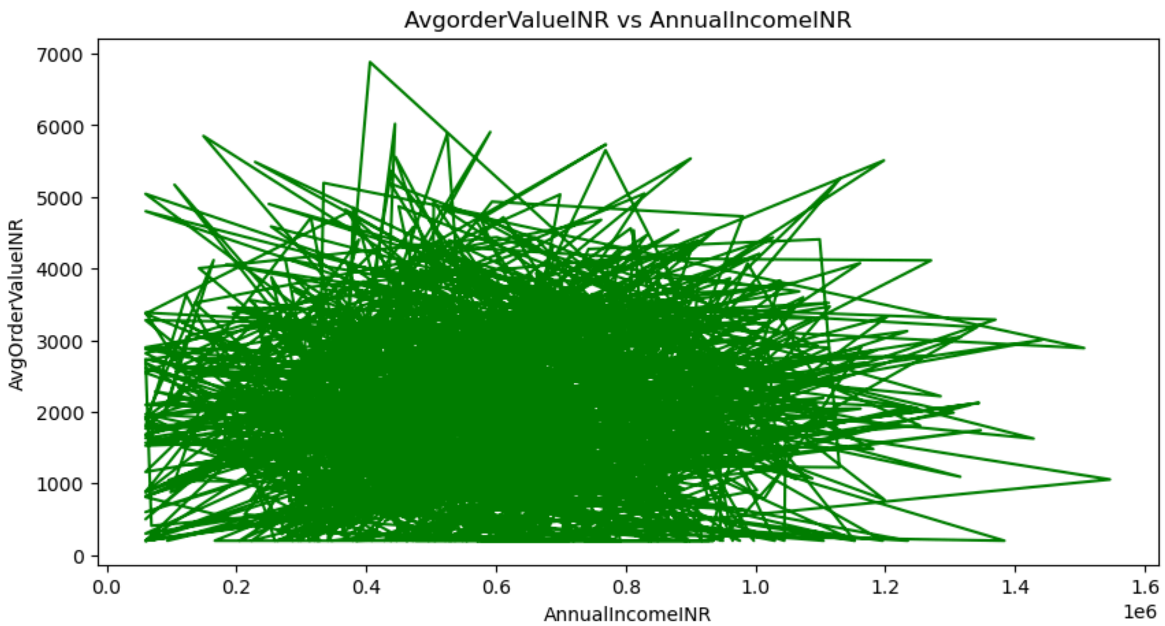



In [25]:
```
plt.scatter(cust['AnnualIncomeINR'], cust['EstimatedYearlySpendINR'])
plt.axvline(cust['AnnualIncomeINR'].median(), color='black')
```

```
plt.axhline(cust['EstimatedYearlySpendINR'].median(), color='black')
plt.title("Income vs Spending (Median Lines)")
plt.show()
```

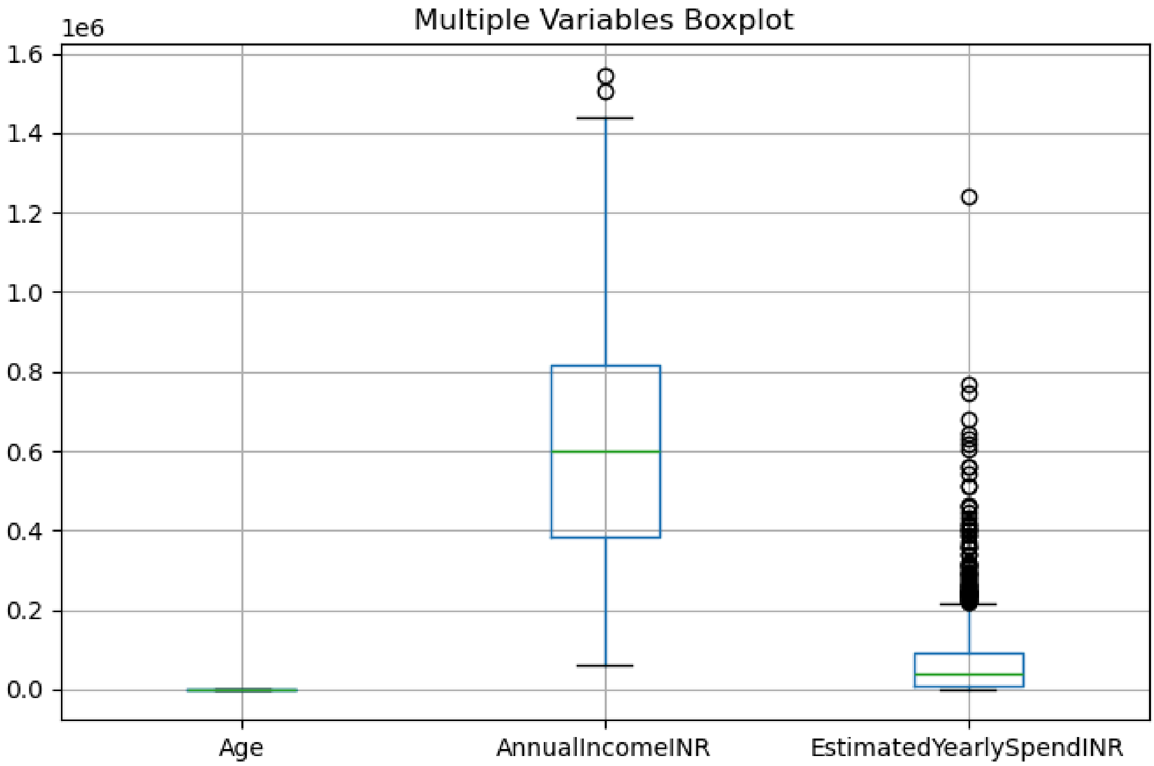



In [27]:
```
plt.figure(figsize=(10,5))
plt.plot(cust['AnnualIncomeINR'], cust['AvgOrderValueINR'], color='green')
plt.title("AvgorderValueINR vs AnnualIncomeINR")
plt.xlabel("AnnualIncomeINR")
plt.ylabel("AvgOrderValueINR")
plt.show()
```




In [25]:
```
cust[['Age','AnnualIncomeINR','EstimatedYearlySpendINR']].boxplot(figsize=(8,5))
plt.title("Multiple Variables Boxplot")
```

```
plt.show()
```




## Insight

In [ ]:

```
Age Distribution of Customers:
   The Age column shows a wide range of customers. The average age is around X years, and n
The histogram indicates a slightly right-skewed distribution, meaning younger adults form a lar
This suggests that marketing strategies should focus on this age group, while there is also pot
to improve engagement with older customers.

Average EstimatedYearlySpendINR by City:
   The bar chart shows noticeable differences in average yearly spending across cities.
One city stands out with the highest spending, indicating a stronger customer base with great
the lowest-spending city suggests customers may be more budget-conscious or less engaged. T
where premium products can be promoted and where targeted marketing can increase custor

Age vs AnnualIncomeINR:
   The scatter plot of Age vs Annual Income shows no strong linear relationship between the tw
levels do not consistently increase or decrease with age. Customers across younger, middle, a
variation in annual income, suggesting that factors such as job role, experience, and industry h
than age alone. Overall, the plot reveals that age is not a strong predictor of annual income in

Age vs EstimatedYearlySpendINR:
   The bubble chart shows that customers of all ages spend different amounts each year, and t
spending. The bigger bubbles (higher income) are mostly connected with higher yearly spendi
spend more. Age does not strongly affect spending, but income does.

Income vs Spending:
   The chart shows that people who earn more usually spend more in a year.
The middle lines help us see who is above or below the average. Many customers are above th
while a few have high income but low spending. Overall, income and spending are connected.

AvgorderValueINR vs AnnualIncomeINR:
   The line chart shows that as annual income increases, the average order value also tends to
higherincome usually spend more per order. Overall,higher earners make bigger purchases, w
```

smaller-value orders.

Multiple Variables Boxplot:
    The boxplot shows that income **and** yearly spending vary a lot between customers, **while** ag
Income **and** spending also have some very high values compared to the rest. Overall, money-r
age **in** this dataset.