```
import pandas as pd

# Pakai pemisah tab
df = pd.read_csv("marketing_campaign.csv", sep='\t')

# Tampilkan 5 baris awal
df.head()
```

	ID	Year_Birth	Education	Marital_Status	Income	Kidhome	Teenhome	Dt_Customer	Recency	MntWines	...	NumWebVisitsMonth	A
0	5524	1957	Graduation	Single	58138.0	0	0	04-09-2012	58	635	...	7	
1	2174	1954	Graduation	Single	46344.0	1	1	08-03-2014	38	11	...	5	
2	4141	1965	Graduation	Together	71613.0	0	0	21-08-2013	26	426	...	4	
3	6182	1984	Graduation	Together	26646.0	1	0	10-02-2014	26	11	...	6	
4	5324	1981	PhD	Married	58293.0	1	0	19-01-2014	94	173	...	5	

5 rows × 29 columns

```
# Cek jumlah baris dan kolom
print("Jumlah baris dan kolom:", df.shape)

# Cek tipe data dan jumlah null
df.info()

# Cek total nilai null per kolom
df.isnull().sum()
```

```
Jumlah baris dan kolom: (2240, 29)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2240 entries, 0 to 2239
Data columns (total 29 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
dtypes: float64(1), int64(25), object(3)
memory usage: 507.6+ KB
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
```

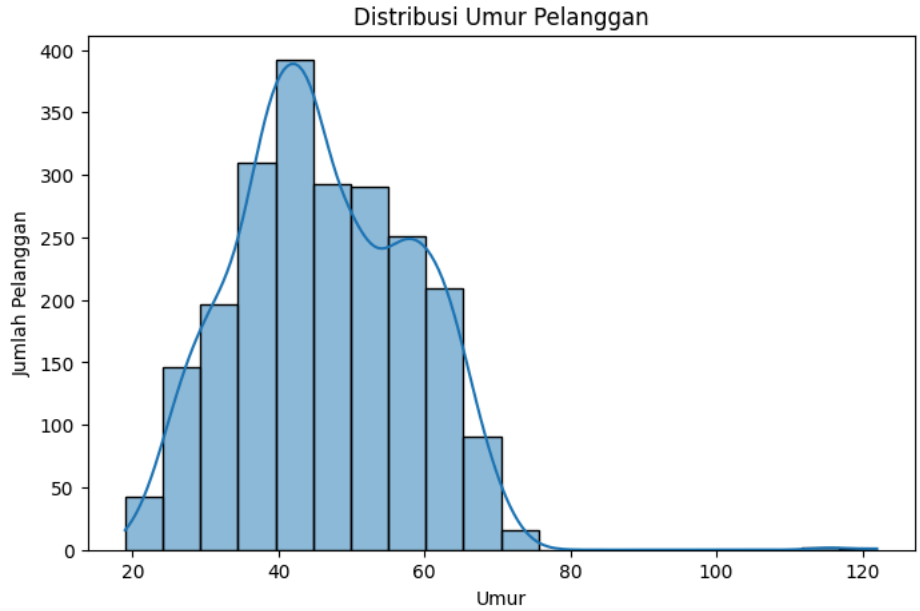
```
from datetime import datetime

# Ubah kolom tanggal ke datetime
df['Dt_Customer'] = pd.to_datetime(df['Dt_Customer'], format='%d-%m-%Y', errors='coerce')

# Hitung umur pelanggan (misalnya tahun data terakhir = 2015)
df['Age'] = 2015 - df['Year_Birth']
df[['Year_Birth', 'Age', 'Kids_Teen', 'Total_Spending']].head()
#df['Dt_Customer'].head()
```

	Year_Birth	Age	Kids_Teen	Total_Spending
0	1957	58	0	1617
1	1954	61	2	27
2	1965	50	0	776
3	1984	31	1	53
4	1981	34	1	422

```
# Jumlah anak di rumah
df['Kids_Teen'] = df['Kidhome'] + df['Teenhome']

# Total spending dari semua kategori produk
spending_cols = ['MntWines', 'MntFruits', 'MntMeatProducts',
                 'MntFishProducts', 'MntSweetProducts', 'MntGoldProds']
df['Total_Spending'] = df[spending_cols].sum(axis=1)
df[['Age', 'Income', 'Total_Spending']].describe()
```

	Age	Income	Total_Spending
count	2240.000000	2216.000000	2240.000000
mean	46.194196	52247.251354	605.798214
std	11.984069	25173.076661	602.249288
min	19.000000	1730.000000	5.000000
25%	38.000000	35303.000000	68.750000
50%	45.000000	51381.500000	396.000000
75%	56.000000	68522.000000	1045.500000
max	122.000000	666666.000000	2525.000000

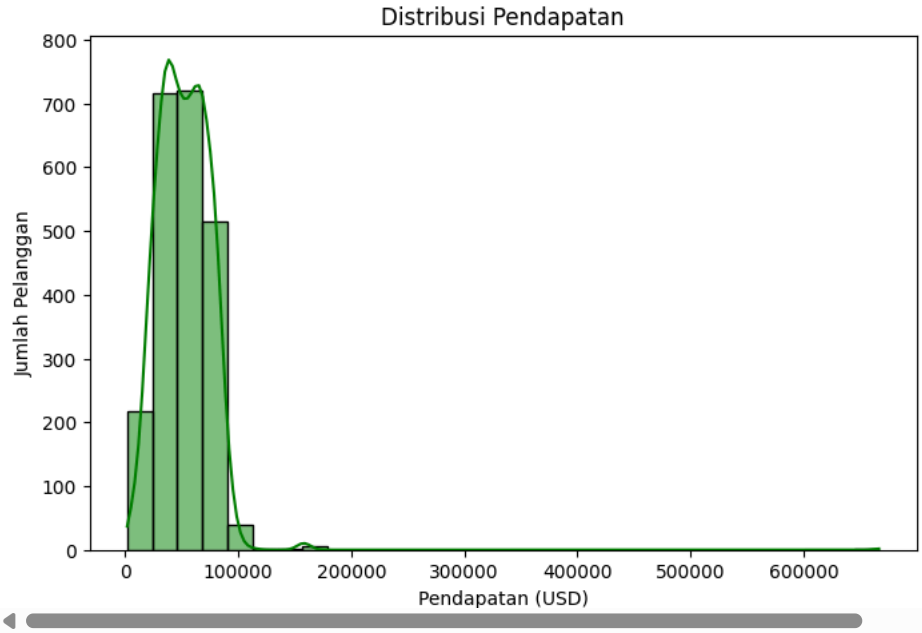
```
import matplotlib.pyplot as plt
import seaborn as sns

plt.figure(figsize=(8, 5))
sns.histplot(df['Age'], bins=20, kde=True)
plt.title('Distribusi Umur Pelanggan')
plt.xlabel('Umur')
plt.ylabel('Jumlah Pelanggan')
plt.show()
```




```
print("Statistik Umur Pelanggan:")
df = df[df['Age'] <= 100]
print(df['Age'].describe())
```

```
Statistik Umur Pelanggan:
count    2237.000000
mean       46.098346
std        11.701917
min        19.000000
25%        38.000000
50%        45.000000
75%        56.000000
max        75.000000
Name: Age, dtype: float64
```

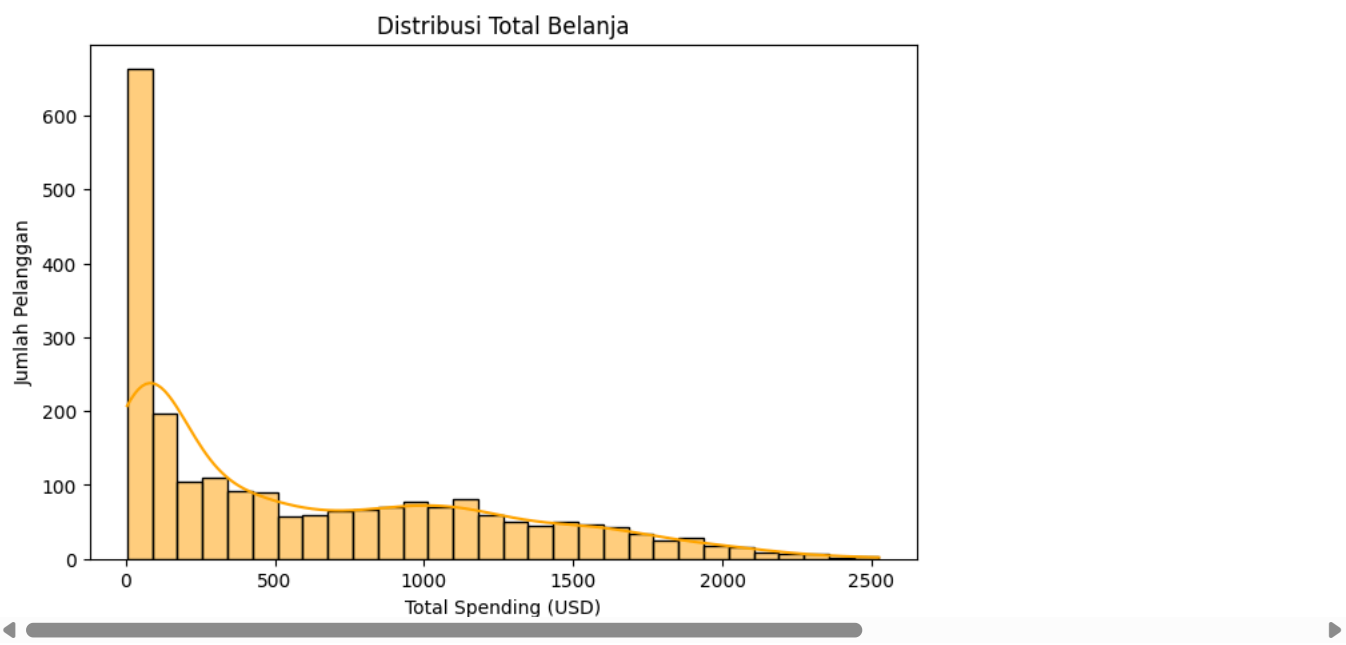
```
plt.figure(figsize=(8, 5))
sns.histplot(df['Income'], bins=30, kde=True, color='green')
plt.title('Distribusi Pendapatan')
plt.xlabel('Pendapatan (USD)')
plt.ylabel('Jumlah Pelanggan')
plt.show()
```




```
print("Statistik Pendapatan Pelanggan:")
print(df['Income'].describe())
```

```
Statistik Pendapatan Pelanggan:
count      2216.000000
mean      52247.251354
std       25173.076661
min        1730.000000
25%       35303.000000
50%       51381.500000
75%       68522.000000
max      666666.000000
Name: Income, dtype: float64
```

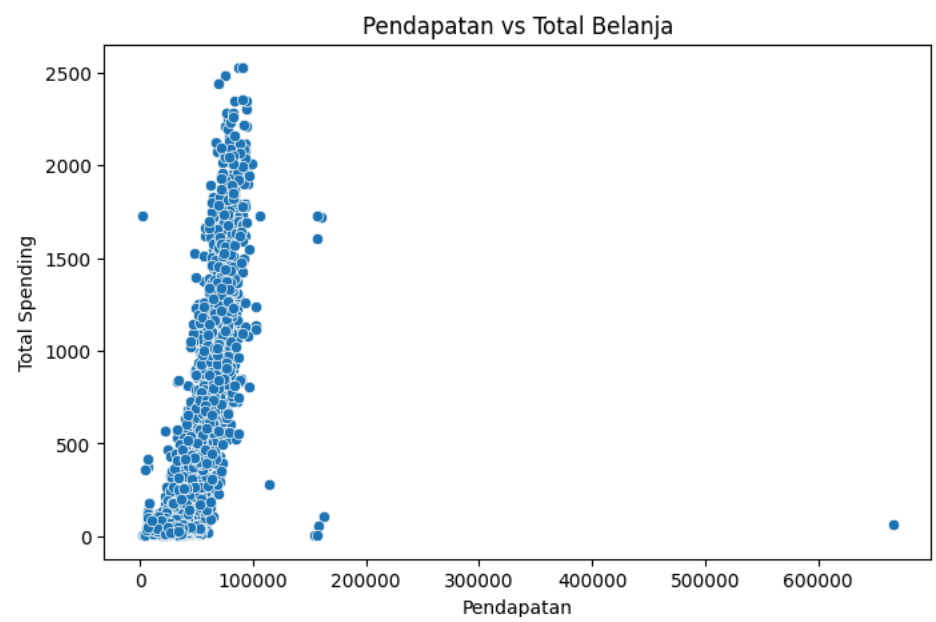
```
plt.figure(figsize=(8, 5))
sns.histplot(df['Total_Spending'], bins=30, kde=True, color='orange')
plt.title('Distribusi Total Belanja')
plt.xlabel('Total Spending (USD)')
plt.ylabel('Jumlah Pelanggan')
plt.show()
```




```
print("Statistik Total Belanja Pelanggan:")
print(df['Total_Spending'].describe())
```

```
Statistik Total Belanja Pelanggan:
count    2240.000000
mean      605.798214
std       602.249288
min         5.000000
25%        68.750000
50%       396.000000
75%      1045.500000
max      2525.000000
Name: Total_Spending, dtype: float64
```

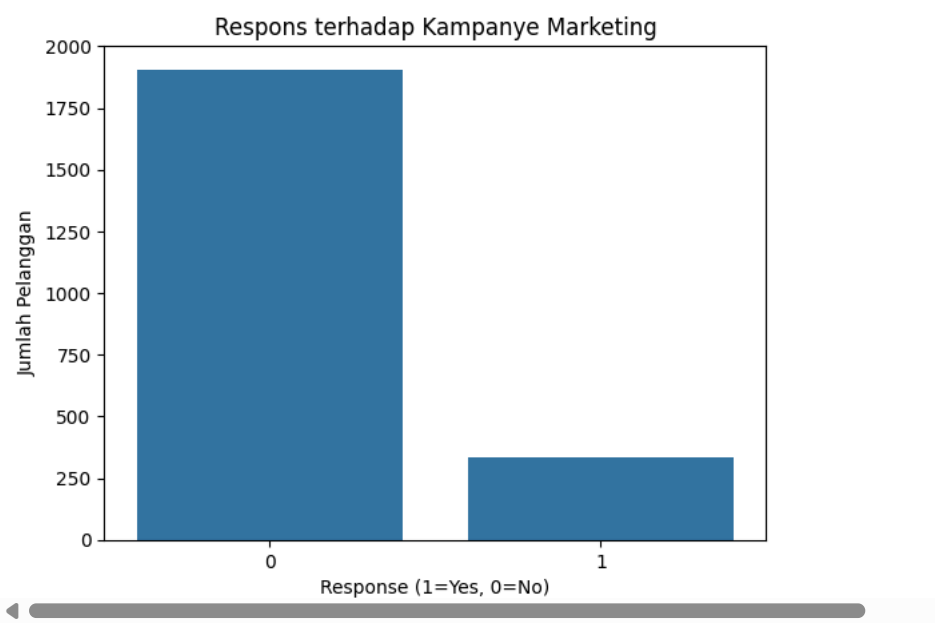
```
plt.figure(figsize=(8, 5))
sns.scatterplot(x='Income', y='Total_Spending', data=df)
plt.title('Pendapatan vs Total Belanja')
plt.xlabel('Pendapatan')
plt.ylabel('Total Spending')
plt.show()
```




```
correlation = df[['Income', 'Total_Spending']].corr()
print("Korelasi antara Income dan Total Spending:")
print(correlation)
```

```
Korelasi antara Income dan Total Spending:
                  Income  Total_Spending
Income          1.000000        0.667576
Total_Spending  0.667576        1.000000
```

```
sns.countplot(x='Response', data=df)
plt.title('Respons terhadap Kampanye Marketing')
plt.xlabel('Response (1=Yes, 0=No)')
plt.ylabel('Jumlah Pelanggan')
plt.show()
```




```
print("Distribusi Respons Kampanye:")
print(df['Response'].value_counts(normalize=True) * 100)
```

```
Distribusi Respons Kampanye:
Response
0    85.089286
1    14.910714
Name: proportion, dtype: float64
```

```
print("Median umur:", df['Age'].median())
print("Pendidikan terbanyak:", df['Education'].mode()[0])
```

```
Median umur: 45.0
Pendidikan terbanyak: Graduation
```

```
# Segmentasi pelanggan berdasarkan total spending
df['Spending_Segment'] = pd.qcut(df['Total_Spending'], q=3, labels=['Low Spender', 'Medium Spender', 'High Spender'])

# Cek jumlah per segmen
print(df['Spending_Segment'].value_counts())
```

```
Spending_Segment
Low Spender       746
High Spender      746
Medium Spender    745
Name: count, dtype: int64
```

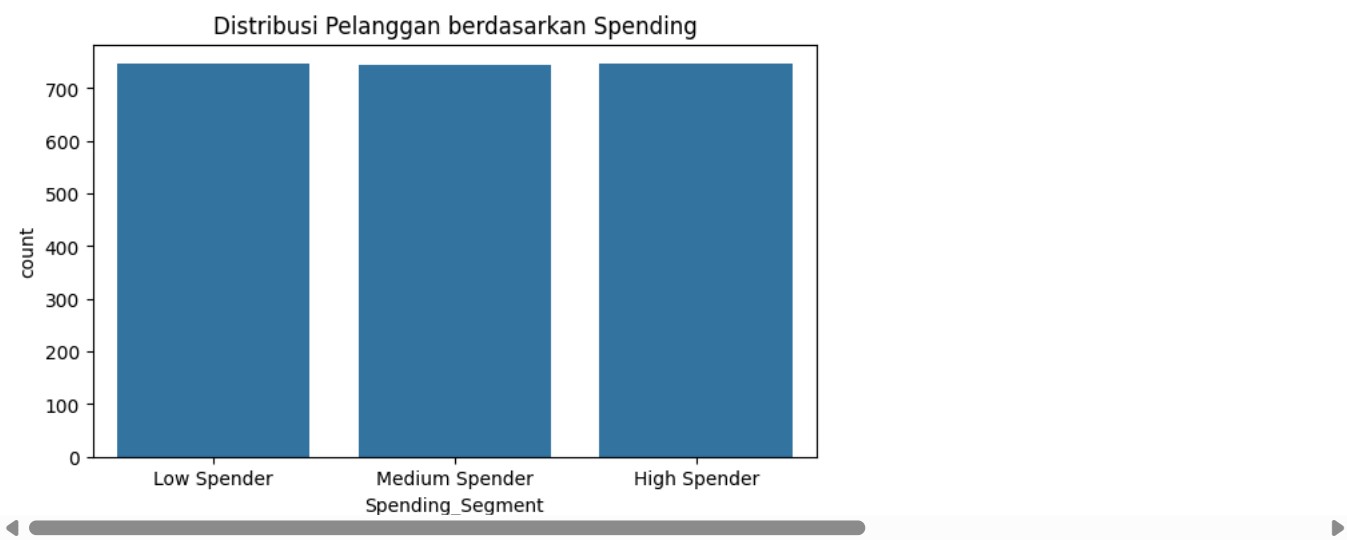
```
df['Income_Segment'] = pd.qcut(df['Income'], q=3, labels=['Low Income', 'Medium Income', 'High Income'])

print(df['Income_Segment'].value_counts())
```

```
Income_Segment
Low Income       738
High Income      738
Medium Income    737
Name: count, dtype: int64
```

```
def age_category(age):
    if age <= 30:
        return 'Young'
    elif age <= 50:
        return 'Adult'
    else:
        return 'Senior'

df['Age_Segment'] = df['Age'].apply(age_category)

print(df['Age_Segment'].value_counts())
```

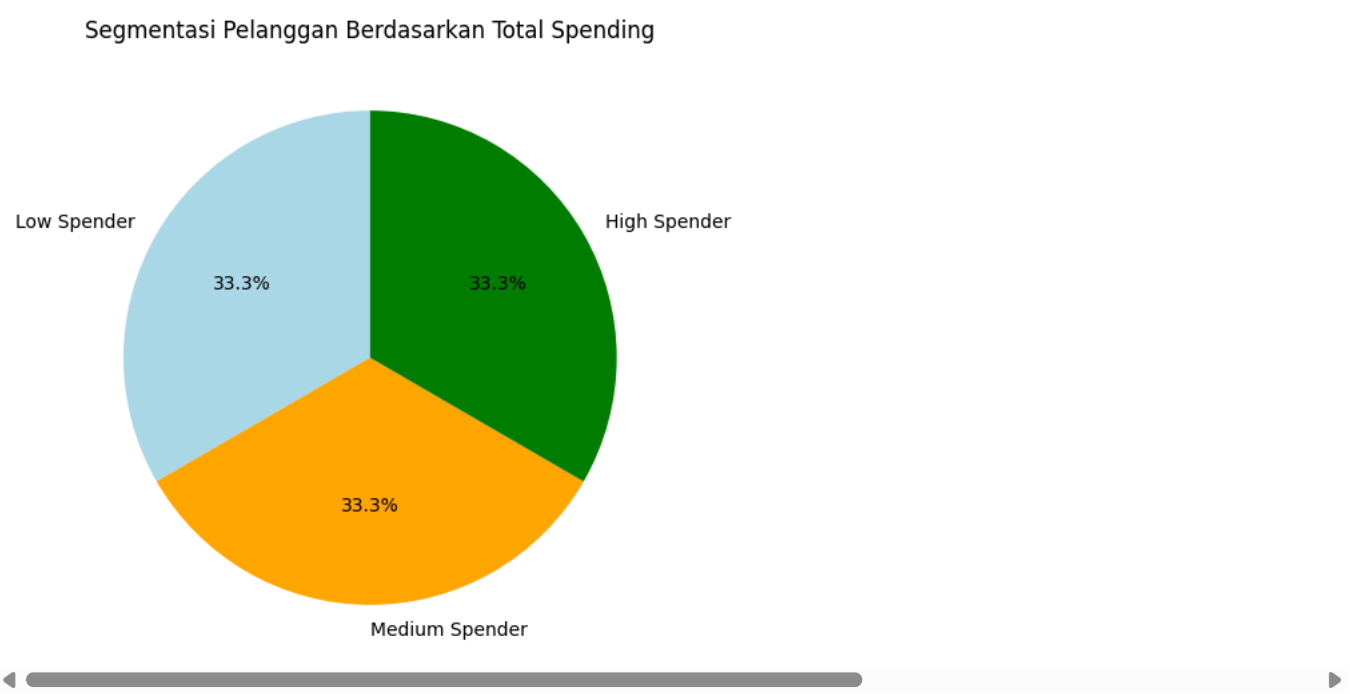
```
Age_Segment
Adult     1233
Senior     783
Young      221
Name: count, dtype: int64
```

```
import seaborn as sns
import matplotlib.pyplot as plt

plt.figure(figsize=(7, 4))
sns.countplot(x='Spending_Segment', data=df, order=['Low Spender', 'Medium Spender', 'High Spender'])
plt.title('Distribusi Pelanggan berdasarkan Spending')
plt.show()
```




```
# Hitung jumlah pelanggan per segmen
segment_counts = df['Spending_Segment'].value_counts().reindex(['Low Spender', 'Medium Spender', 'High Spender'])
```

```
# Buat pie chart
plt.figure(figsize=(6, 6))
segment_counts.plot.pie(autopct='%1.1f%%', startangle=90, colors=['lightblue', 'orange', 'green'])
plt.title('Segmentasi Pelanggan Berdasarkan Total Spending')
plt.ylabel('')  # Hilangkan label Y
plt.show()
```




```
# Bandingkan karakteristik berdasarkan Spending Segment
df.groupby('Spending_Segment')[['Income', 'Age', 'Total_Spending', 'Kids_Teen', 'Response']].mean().round(1)
```

```
C:\Users\lenovo ideapad 330\AppData\Local\Temp\ipykernel_924\934148667.py:2: FutureWarning: The default of observed=False is deprec
  df.groupby('Spending_Segment')[['Income', 'Age', 'Total_Spending', 'Kids_Teen', 'Response']].mean().round(1)
```

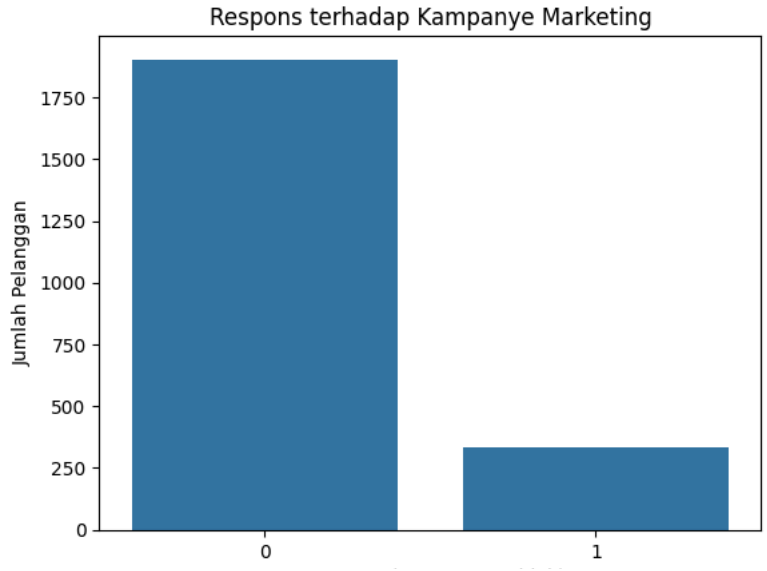
Spending_Segment	Income	Age	Total_Spending	Kids_Teen	Response
Low Spender	32737.7	42.9	51.2	1.3	0.1
Medium Spender	50832.6	47.5	413.0	1.1	0.1
High Spender	73022.8	47.9	1352.8	0.5	0.2

```
df.groupby('Income_Segment')[['Income', 'Total_Spending', 'Response']].mean().round(1)
df.groupby('Age_Segment')[['Income', 'Total_Spending', 'Response']].mean().round(1)
```

```
C:\Users\lenovo ideapad 330\AppData\Local\Temp\ipykernel_924\3817086190.py:1: FutureWarning: The default of observed=False is deprec
  df.groupby('Income_Segment')[['Income', 'Total_Spending', 'Response']].mean().round(1)
```

Age_Segment	Income	Total_Spending	Response
Adult	50094.3	538.7	0.2
Senior	57313.1	710.4	0.1
Young	46295.0	609.1	0.2

```
import seaborn as sns
import matplotlib.pyplot as plt

sns.countplot(x='Response', data=df)
plt.title('Respons terhadap Kampanye Marketing')
plt.xlabel('Response (1 = Ya, 0 = Tidak)')
plt.ylabel('Jumlah Pelanggan')
plt.show()
```




```
response_rate = df['Response'].value_counts(normalize=True) * 100
print("Persentase Respons:")
print(response_rate)
```

```
Persentase Respons:
Response
0    85.069289
1    14.930711
Name: proportion, dtype: float64
```

```
# Bandingkan rata-rata income
print("Income rata-rata:")
print(df.groupby('Response')['Income'].mean())

# Bandingkan rata-rata total belanja
print("Total Spending rata-rata:")
print(df.groupby('Response')['Total_Spending'].mean())

# Bandingkan umur rata-rata
print("Umur rata-rata:")
print(df.groupby('Response')['Age'].mean())
```

```
Income rata-rata:
Response
0    50824.326064
1    60209.675676
Name: Income, dtype: float64
Total Spending rata-rata:
Response
0    538.759327
1    987.392216
Name: Total_Spending, dtype: float64
```