In [9]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

In [10]:
```
df_csv = pd.read_csv(r"C:\Users\chara\Downloads\saas_churn_dataset.csv")
df_csv.head()
```

Out[10]:

	CustomerID	Age	Gender	Country	SubscriptionType	MonthlySpend	ContractLeng
0	1.0	39.0	Male	US	Basic	14.842209	12
1	2.0	33.0	Male	US	Standard	67.183445	12
2	NaN	41.0	Male	Canada	Basic	26.983480	1
3	4.0	50.0	NaN	Germany	Standard	NaN	12
4	5.0	32.0	Female	US	NaN	14.728931	Na

In [8]:
```
df_csv.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 17 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   CustomerID         9500 non-null   float64
 1   Age                10000 non-null  float64
 2   Gender             9500 non-null   object 
 3   Country            9500 non-null   object 
 4   SubscriptionType   9500 non-null   object 
 5   MonthlySpend       9500 non-null   float64
 6   ContractLength     9500 non-null   float64
 7   TenureMonths       9500 non-null   float64
 8   SupportTickets     9500 non-null   float64
 9   LoginsPerMonth     10000 non-null  float64
 10  FeatureUsageScore  9500 non-null   float64
 11  Referral           9500 non-null   float64
 12  Churn              9500 non-null   float64
 13  MarketingSpend     9500 non-null   float64
 14  DiscountApplied    9500 non-null   float64
 15  AdSpend            9500 non-null   float64
 16  Revenue            9500 non-null   float64
dtypes: float64(14), object(3)
memory usage: 1.3+ MB
```

In [11]:
```
df_csv.isnull().sum()
```

Out[11]:
```
CustomerID           500
Age                    0
Gender               500
Country              500
SubscriptionType     500
MonthlySpend         500
ContractLength       500
TenureMonths         500
SupportTickets       500
LoginsPerMonth         0
FeatureUsageScore    500
Referral             500
Churn                500
MarketingSpend       500
DiscountApplied      500
AdSpend              500
Revenue              500
dtype: int64
```

Finding missing percentage

In [13]:
```
missing_percentage = (df_csv.isnull().mean()) * 100

missing_percentage
```

Out[13]:
```
CustomerID           5.0
Age                  0.0
Gender               5.0
Country              5.0
SubscriptionType     5.0
MonthlySpend         5.0
ContractLength       5.0
TenureMonths         5.0
SupportTickets       5.0
LoginsPerMonth       0.0
FeatureUsageScore    5.0
Referral             5.0
Churn                5.0
MarketingSpend       5.0
DiscountApplied      5.0
AdSpend              5.0
Revenue              5.0
dtype: float64
```

column renaming function

In [18]:
```
import re

def to_cap_snake(name):

    s = re.sub(r'([a-z])([A-Z])', r'\1 \2', name)
    # Step 2: Split acronyms (like ID, URL, etc.)
    s = re.sub(r'([A-Z]+)([A-Z][a-z])', r'\1 \2', s)
    # Step 3: Replace - and _ with space
    s = s.replace("-", " ").replace("_", " ")
    # Step 4: Capitalize words and join with underscore
```

```
    return "_".join([w.capitalize() for w in s.split()])

# Apply conversion to all columns
df_csv.columns = [to_cap_snake(col) for col in df_csv.columns]

# Show new column names
print(df_csv.columns)
```

```
Index(['Customer_Id', 'Age', 'Gender', 'Country', 'Subscription_Type',
       'Monthly_Spend', 'Contract_Length', 'Tenure_Months', 'Support_Tickets',
       'Logins_Per_Month', 'Feature_Usage_Score', 'Referral', 'Churn',
       'Marketing_Spend', 'Discount_Applied', 'Ad_Spend', 'Revenue'],
      dtype='object')
```

Basic Exploration

In [20]:
```
churn_rate = df_csv["Churn"].mean() * 100
print("Churn Rate:", round(churn_rate, 2), "%")
```

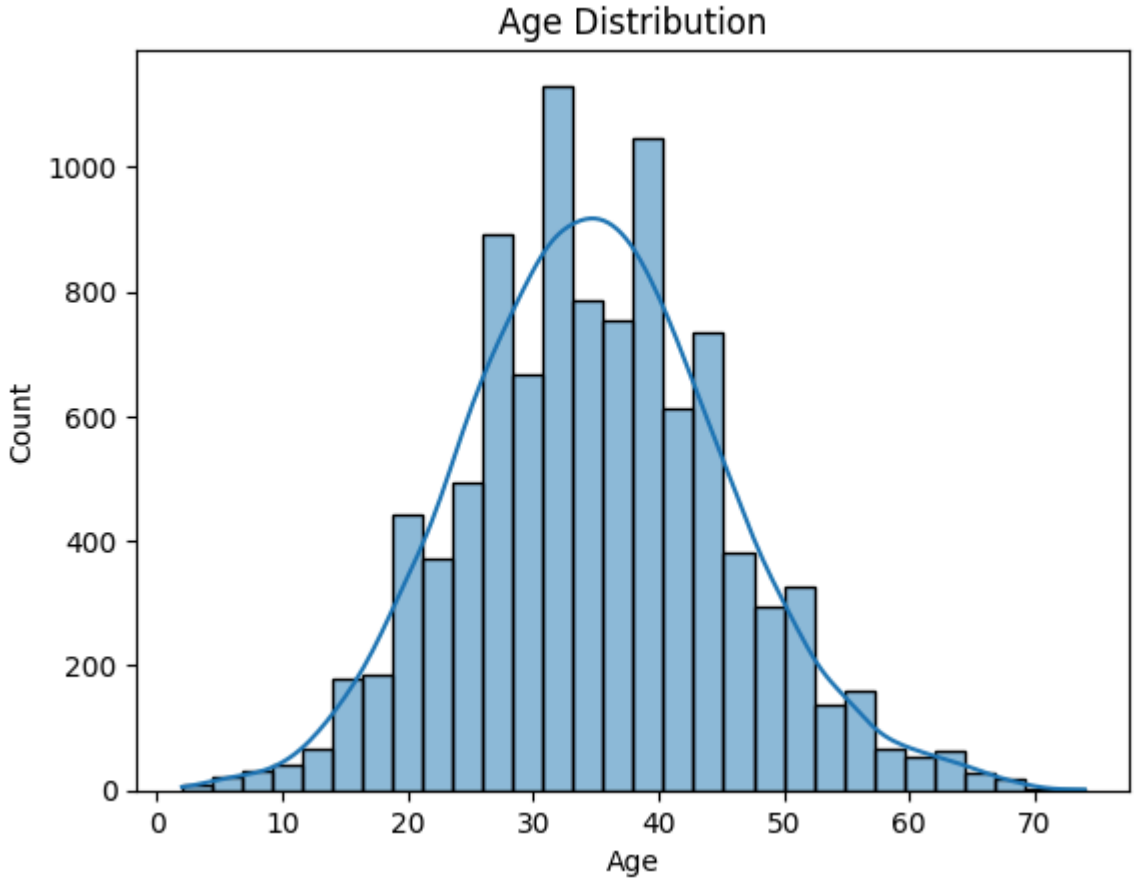
```
Churn Rate: 29.2 %
```

In [21]:
```
print("Shape:", df_csv.shape)
print(df_csv.info())
```

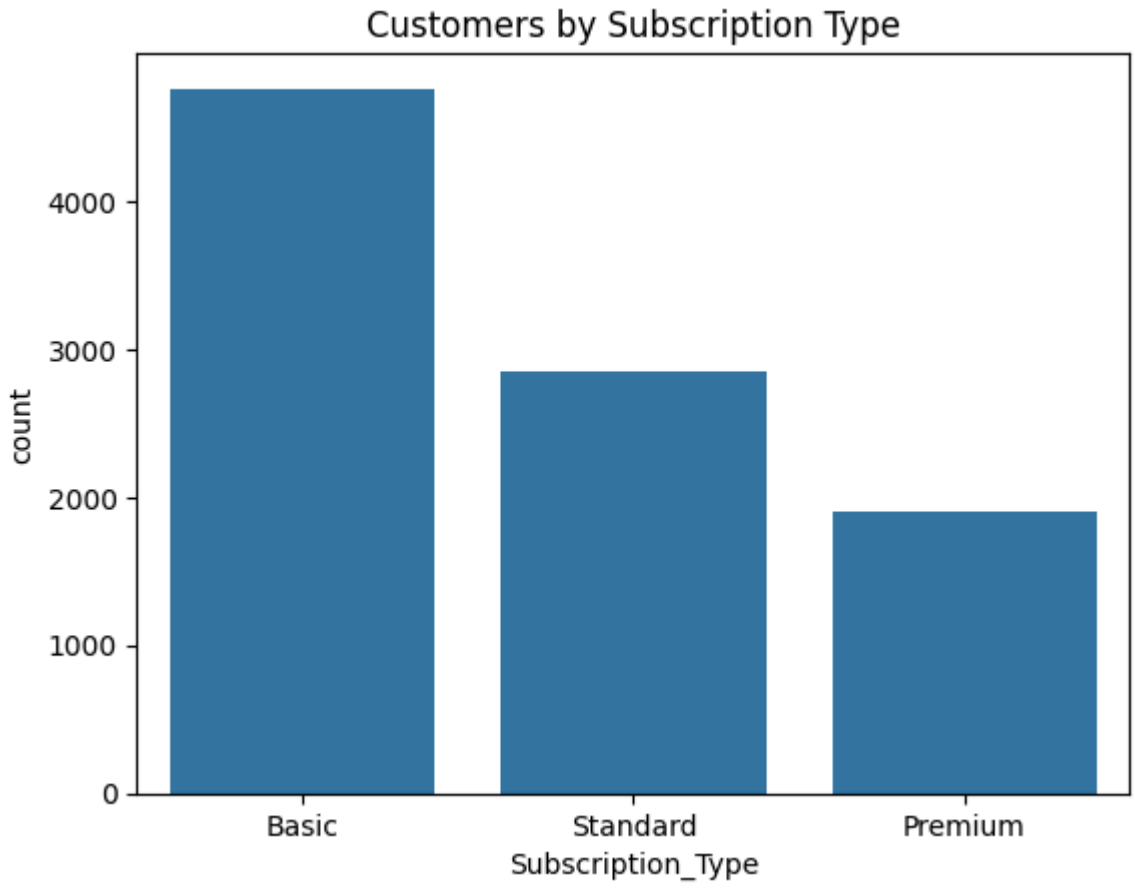
```
Shape: (10000, 17)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 17 columns):
 #   Column               Non-Null Count  Dtype  
---  ------               --------------  -----  
 0   Customer_Id          9500 non-null   float64
 1   Age                  10000 non-null  float64
 2   Gender               9500 non-null   object 
 3   Country              9500 non-null   object 
 4   Subscription_Type    9500 non-null   object 
 5   Monthly_Spend        9500 non-null   float64
 6   Contract_Length      9500 non-null   float64
 7   Tenure_Months        9500 non-null   float64
 8   Support_Tickets      9500 non-null   float64
 9   Logins_Per_Month     10000 non-null  float64
 10  Feature_Usage_Score  9500 non-null   float64
 11  Referral             9500 non-null   float64
 12  Churn                9500 non-null   float64
 13  Marketing_Spend      9500 non-null   float64
 14  Discount_Applied     9500 non-null   float64
 15  Ad_Spend             9500 non-null   float64
 16  Revenue              9500 non-null   float64
dtypes: float64(14), object(3)
memory usage: 1.3+ MB
None
```

Coustomer Profile Analysis

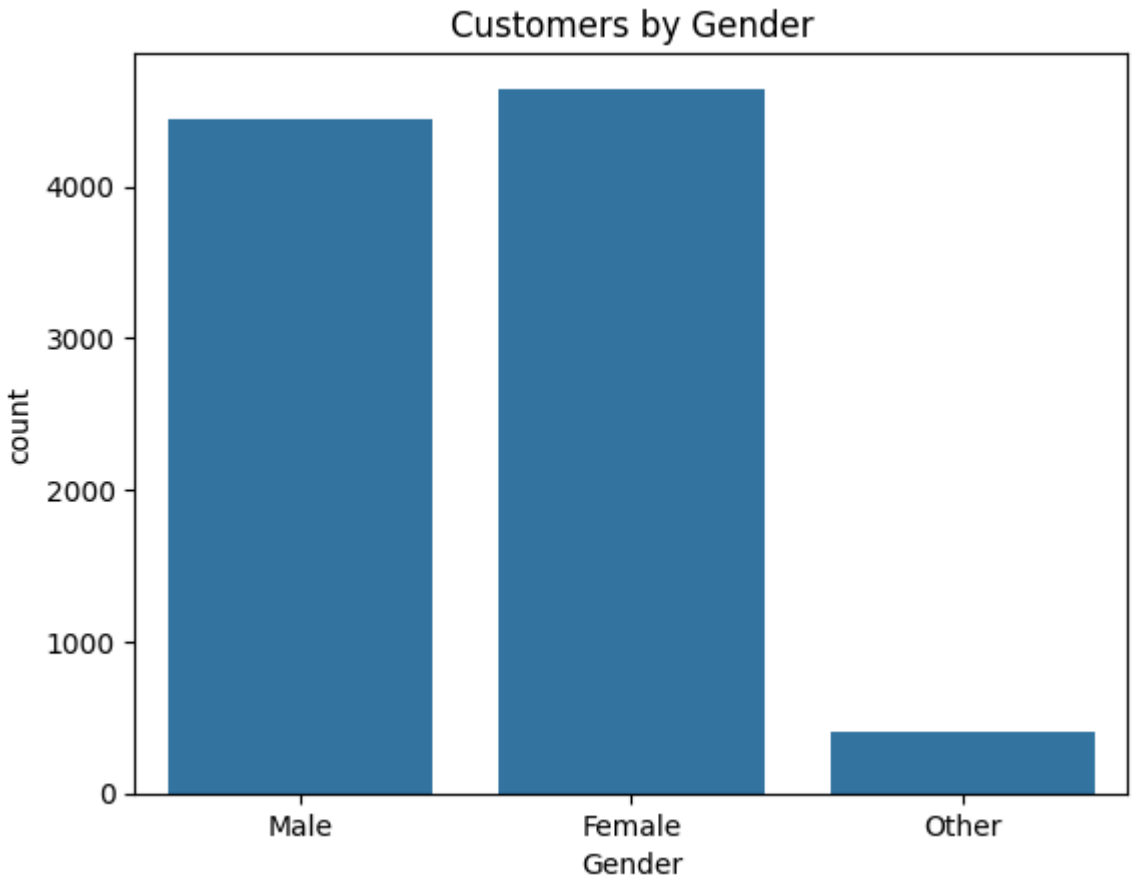
In [22]:
```
sns.histplot(df_csv["Age"], bins=30, kde=True)
plt.title("Age Distribution")
plt.show()
```



In [24]:
```
sns.countplot(x="Subscription_Type", data=df_csv)
plt.title("Customers by Subscription Type")
plt.show()
```



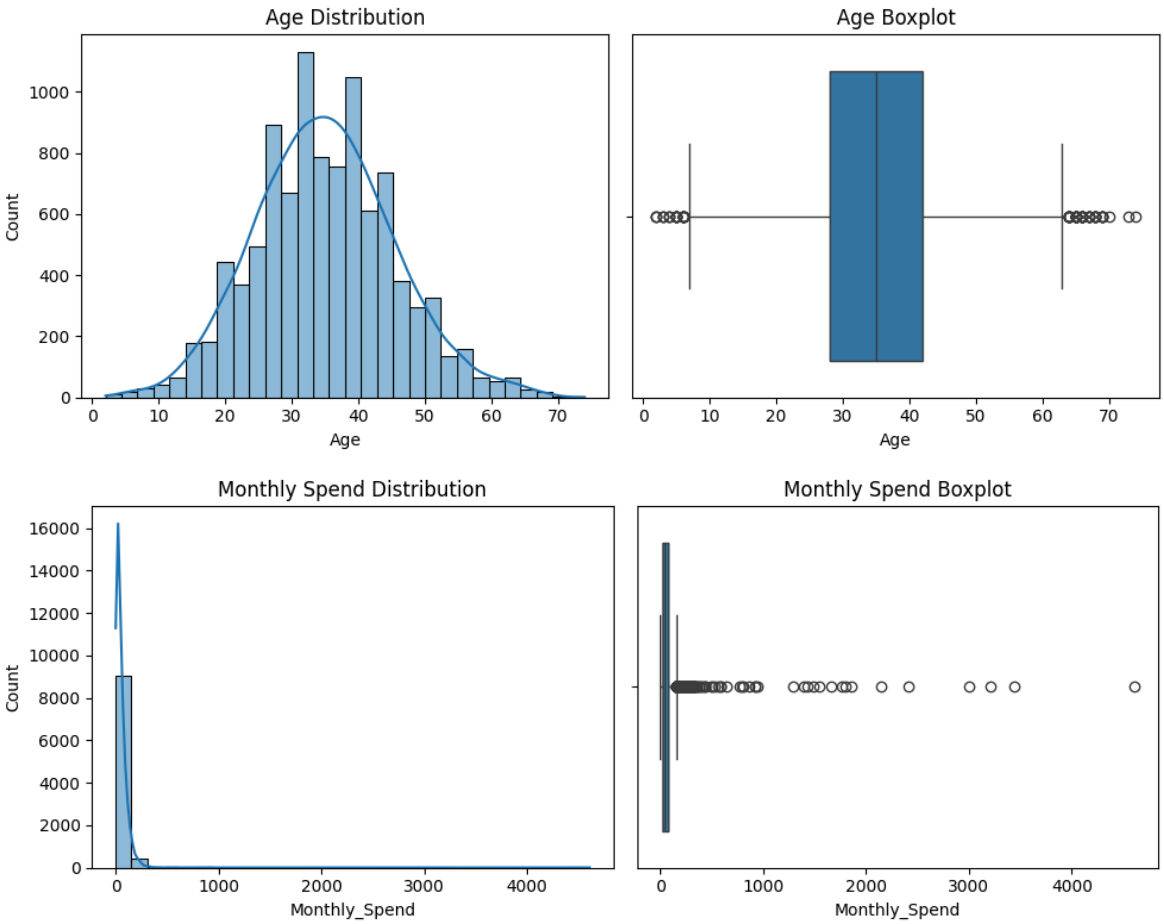
In [25]:
```
sns.countplot(x="Gender", data=df_csv)
plt.title("Customers by Gender")
plt.show()
```



Univariate Analysis

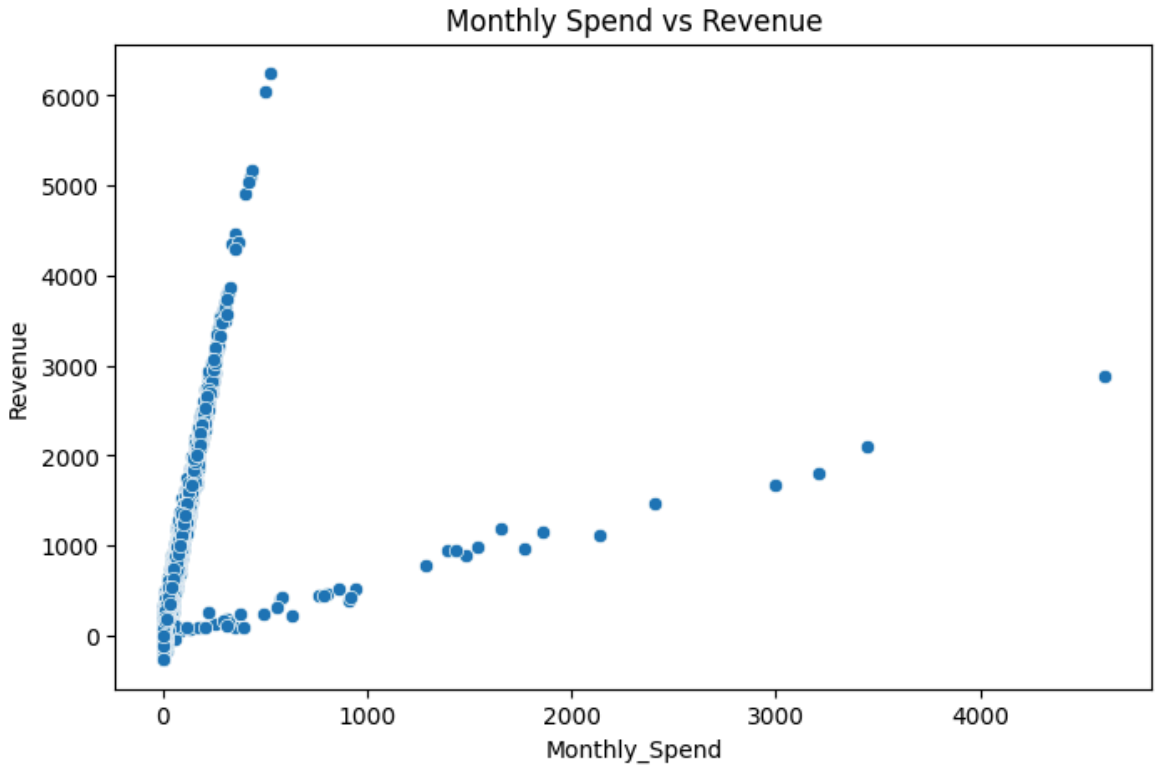
In [30]:
```
plt.figure(figsize=(10,4))

plt.subplot(1,2,1)
sns.histplot(df_csv["Age"], bins=30, kde=True)
plt.title("Age Distribution")

plt.subplot(1,2,2)
sns.boxplot(x=df_csv["Age"])
plt.title("Age Boxplot")

plt.tight_layout()
plt.show()


plt.figure(figsize=(10,4))

plt.subplot(1,2,1)
sns.histplot(df_csv["Monthly_Spend"], bins=30, kde=True)
plt.title("Monthly Spend Distribution")

plt.subplot(1,2,2)
sns.boxplot(x=df_csv["Monthly_Spend"])
plt.title("Monthly Spend Boxplot")
```

```
plt.tight_layout()
plt.show()
```



Bivariate Analysis

In [33]:
```
plt.figure(figsize=(8,5))
sns.scatterplot(x="Monthly_Spend", y="Revenue", data=df_csv)
plt.title("Monthly Spend vs Revenue")
plt.show()


corr = df_csv["Monthly_Spend"].corr(df_csv["Revenue"])
print("Correlation between MonthlySpend and Revenue:", round(corr, 2))
```




```
Correlation between MonthlySpend and Revenue: 0.49
```

In [ ]: