```
import pandas as pd
import re
import matplotlib.pyplot as plt
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.naive_bayes import MultinomialNB

# Sample credit card statement paragraph
paragraph = """
Credit Card Statement:
Date: 01-July-2024
Transaction: Grocery Store Purchase, Amount: ₹500
Date: 03-July-2024
Transaction: Salary Credit, Amount: ₹20000
Date: 05-July-2024
Transaction: Electricity Bill Payment, Amount: ₹1000
Date: 07-July-2024
Transaction: Movie Ticket Purchase, Amount: ₹150
Date: 10-July-2024
Transaction: Restaurant Spend, Amount: ₹700
Date: 12-July-2024
Transaction: Freelance Payment Credit, Amount: ₹5000
Date: 15-July-2024
Transaction: Transfer to Savings Account, Amount: ₹2000
Date: 17-July-2024
Transaction: Mutual Fund Investment, Amount: ₹3000
Date: 18-July-2024
Transaction: Gift Received, Amount: ₹8000
"""

# Step 1: Extract transactions
# Extract descriptions from the paragraph
transactions = re.findall(r'Transaction: (.*?), Amount: ₹[\d,]+', paragraph)

# Step 2: Prepare labeled data for training
data = {
    'description': ['Grocery Store Purchase', 'Salary Credit', 'Electricity Bill Payment',
                     'Restaurant Spend', 'Freelance Payment Credit', 'Transfer to Savings
                     'Gift Received'],
    'category': ['Expense', 'Income', 'Expense', 'Expense', 'Expense', 'Income', 'Transfer
}

df = pd.DataFrame(data)

# Step 3: Convert text data into numerical features
vectorizer = CountVectorizer()
X_train = vectorizer.fit_transform(df['description'])
y_train = df['category']

# Train the model
model = MultinomialNB()
model.fit(X_train, y_train)

# Transform the transactions to be categorized
X = vectorizer.transform(transactions)

# Predict categories for the transactions
predicted_categories = model.predict(X)
```

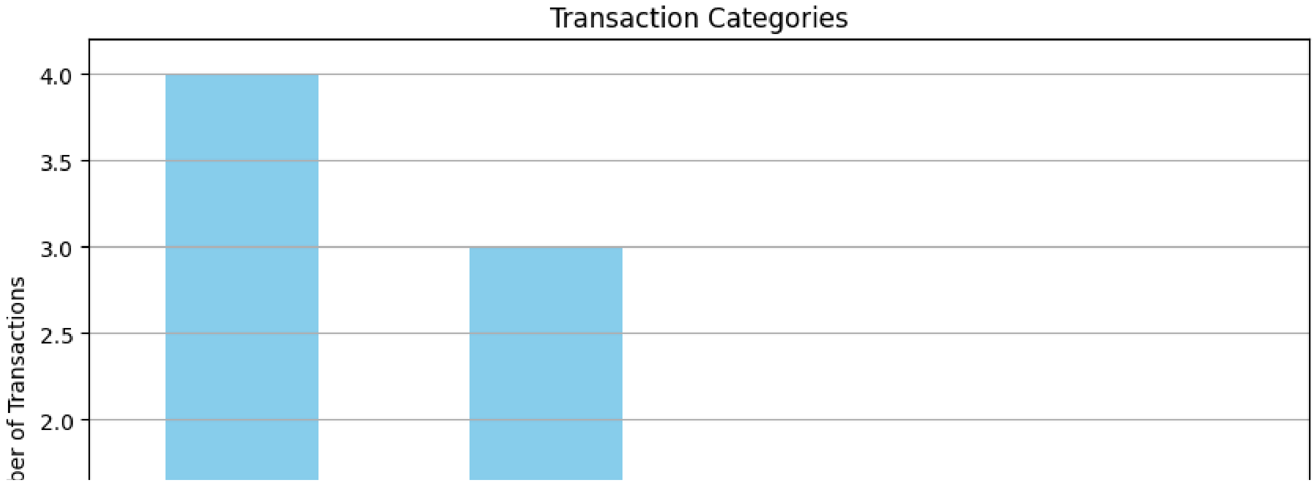
```
# Display the categorized transactions
print("Categorized Transactions:")
for i, transaction in enumerate(transactions):
    print(f"Transaction: {transaction} -> Category: {predicted_categories[i]}")

# Step 4: Visualization
category_counts = pd.Series(predicted_categories).value_counts()

# Create bar chart
plt.figure(figsize=(10, 6))
category_counts.plot(kind='bar', color='skyblue')
plt.title('Transaction Categories')
plt.xlabel('Category')
plt.ylabel('Number of Transactions')
plt.xticks(rotation=45)
plt.grid(axis='y')

# Show plot
plt.show()
```

```
Categorized Transactions:
Transaction: Grocery Store Purchase -> Category: Expense
Transaction: Salary Credit -> Category: Income
Transaction: Electricity Bill Payment -> Category: Expense
Transaction: Movie Ticket Purchase -> Category: Expense
Transaction: Restaurant Spend -> Category: Expense
Transaction: Freelance Payment Credit -> Category: Income
Transaction: Transfer to Savings Account -> Category: Transfer
Transaction: Mutual Fund Investment -> Category: Investment
Transaction: Gift Received -> Category: Income
```


Transaction Categories
ber of Transactions