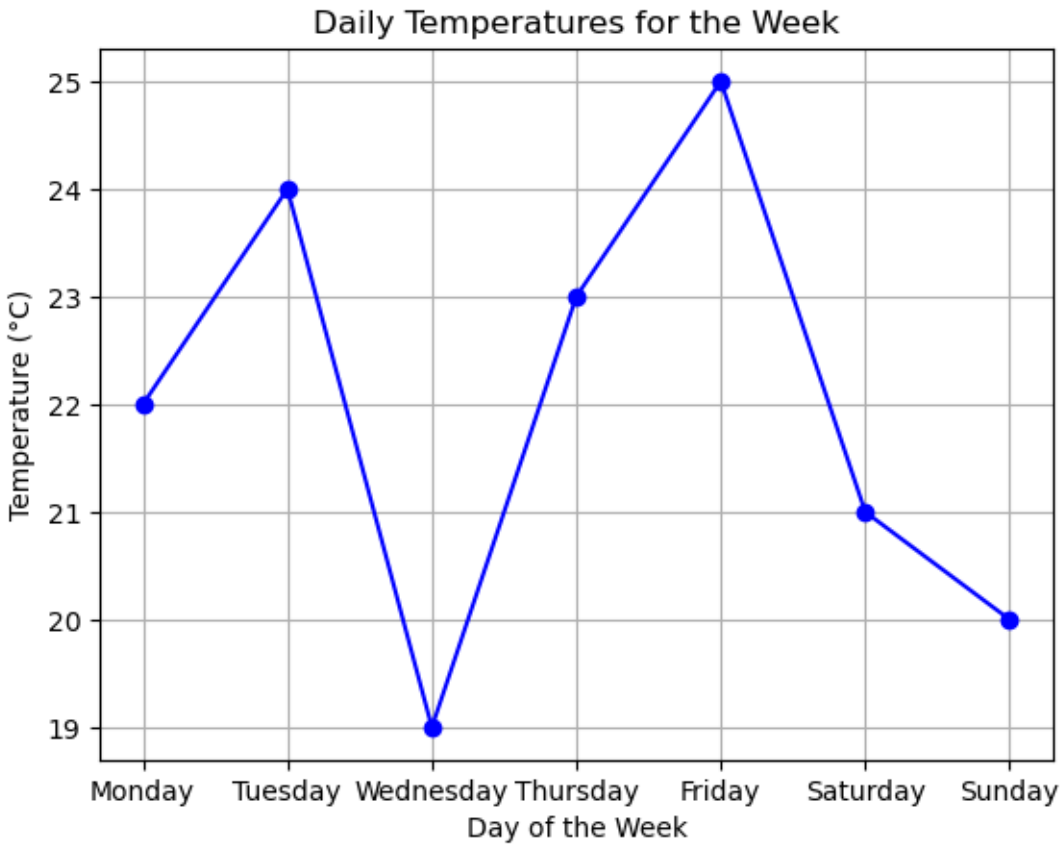
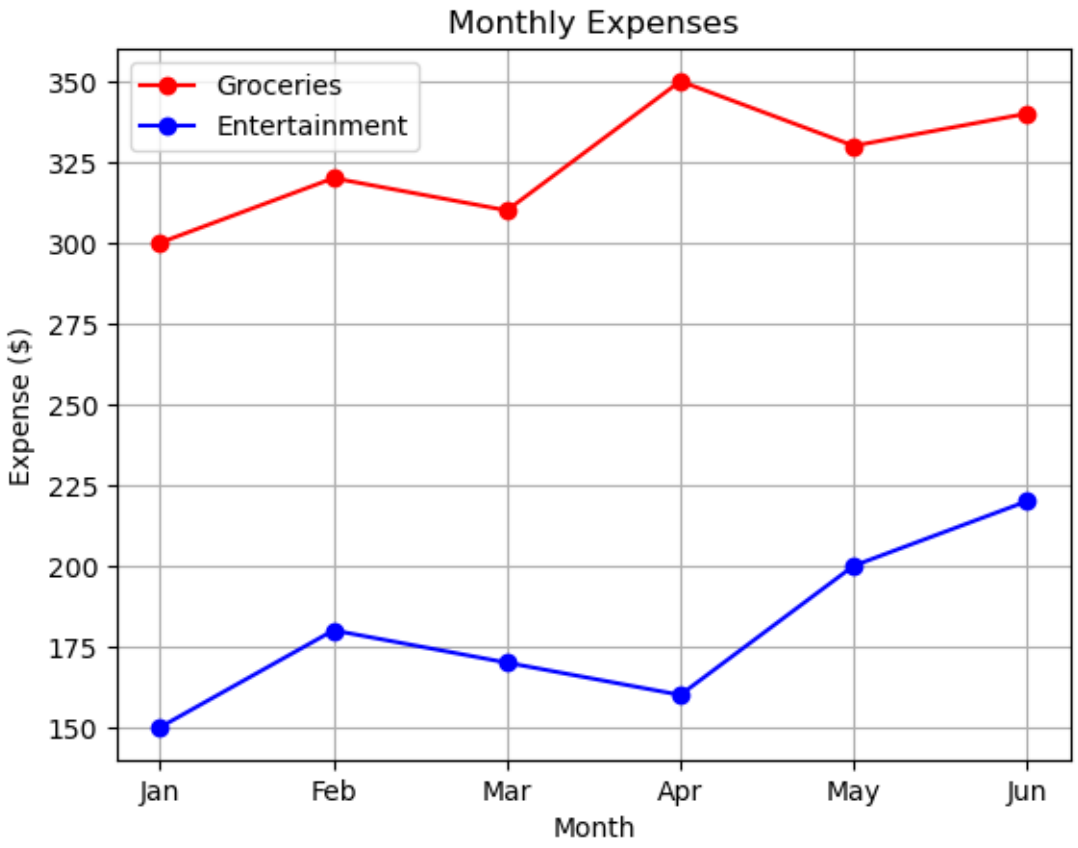
matplotlib2

September 18, 2024

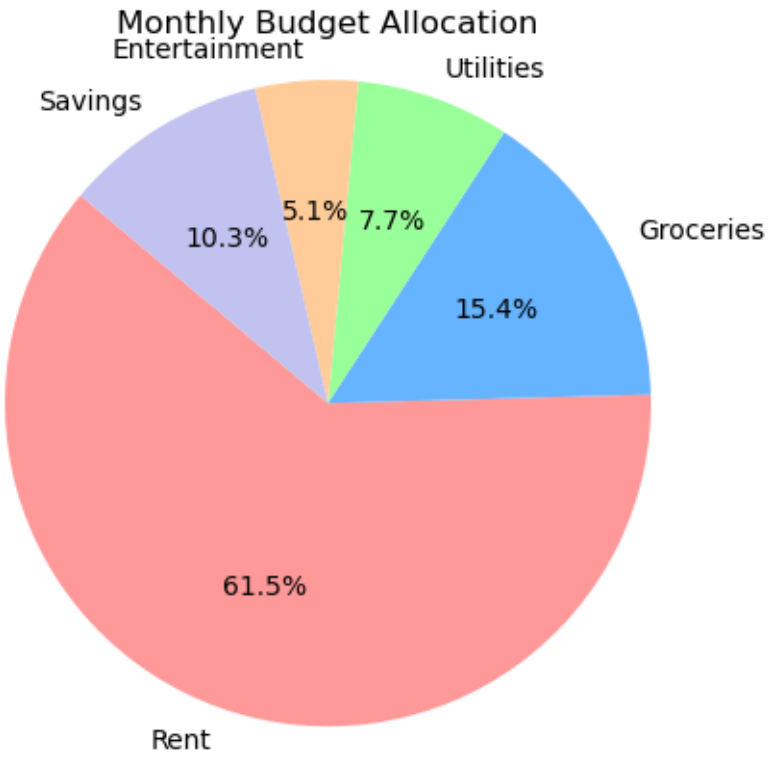
[1]:
```
import matplotlib.pyplot as plt

# Sample data
days = ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday',␣
 ↪'Sunday']
temperatures = [22, 24, 19, 23, 25, 21, 20]

# Create the line chart
plt.plot(days, temperatures, marker='o', linestyle='-', color='b')
plt.xlabel('Day of the Week')
plt.ylabel('Temperature (°C)')
plt.title('Daily Temperatures for the Week')
plt.grid(True)
plt.show()
```



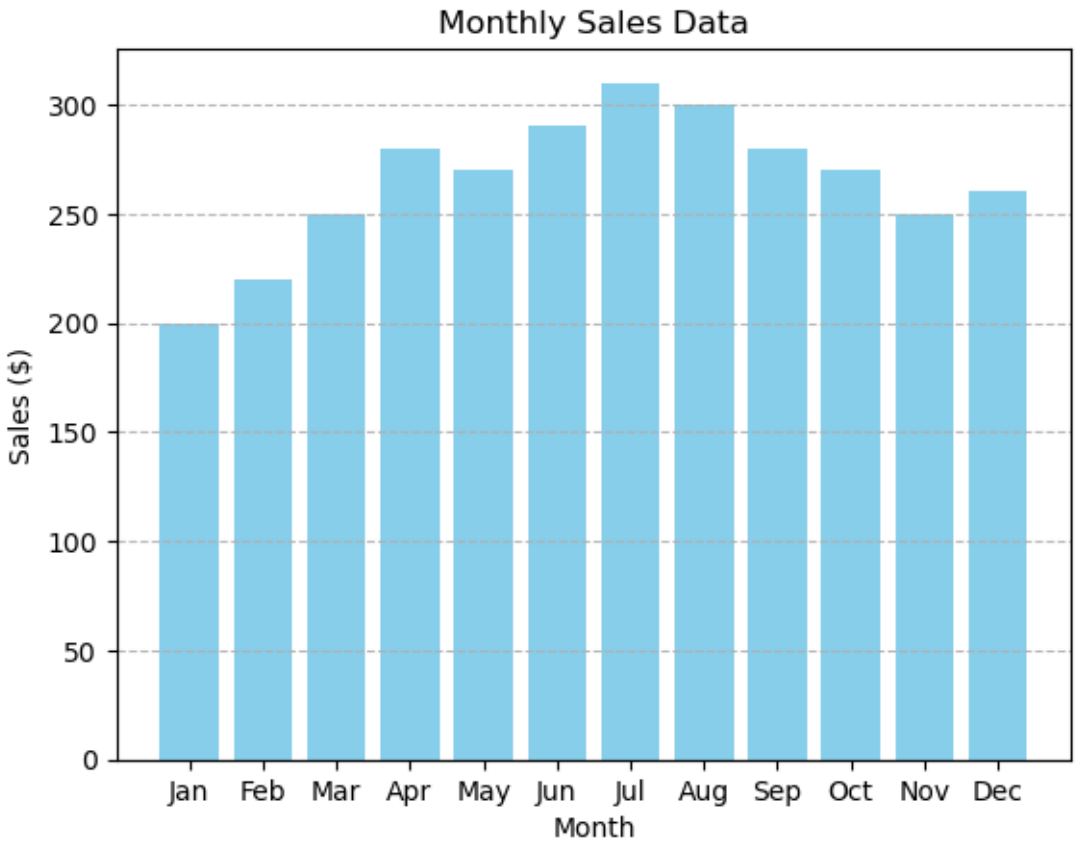
[2]:
```
import matplotlib.pyplot as plt

# Sample data
months = ['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun']
groceries = [300, 320, 310, 350, 330, 340]
entertainment = [150, 180, 170, 160, 200, 220]

# Create the line chart
plt.plot(months, groceries, marker='o', linestyle='-', color='r',␣
 ↪label='Groceries')
plt.plot(months, entertainment, marker='o', linestyle='-', color='b',␣
 ↪label='Entertainment')
plt.xlabel('Month')
plt.ylabel('Expense ($)')
plt.title('Monthly Expenses')
plt.legend()
plt.grid(True)
plt.show()
```



[4]:
```
import matplotlib.pyplot as plt

# Sample data
categories = ['Rent', 'Groceries', 'Utilities', 'Entertainment', 'Savings']
amounts = [1200, 300, 150, 100, 200]

# Simplified color scheme
colors = ['#ff9999','#66b3ff','#99ff99','#ffcc99','#c2c2f0']

# Create the pie chart
plt.pie(amounts, labels=categories, autopct='%1.1f%%', startangle=140,␣
 ↪colors=colors)
plt.title('Monthly Budget Allocation')
plt.axis('equal')  # Equal aspect ratio ensures that pie is drawn as a circle.
plt.show()
```



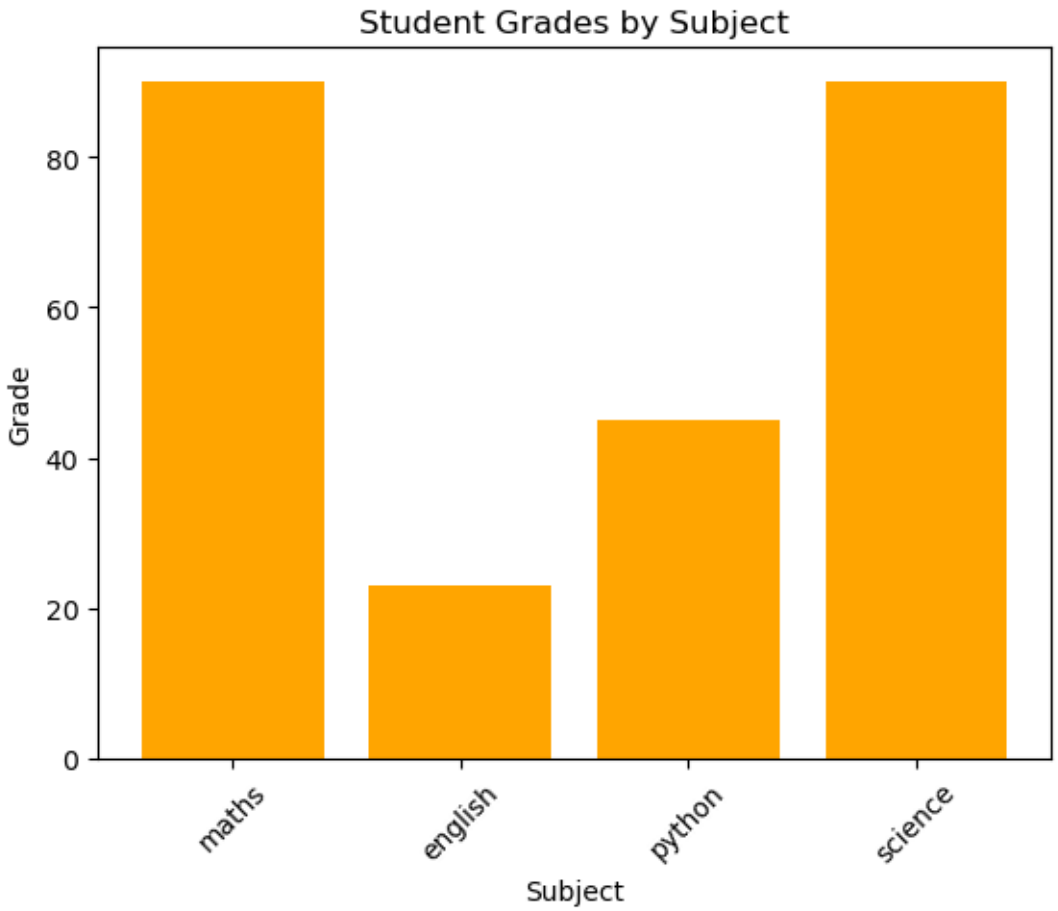
[7]:
```
import matplotlib.pyplot as plt

# Sample data
months = ['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct',␣
 ↪'Nov', 'Dec']
sales = [200, 220, 250, 280, 270, 290, 310, 300, 280, 270, 250, 260]

# Create the bar chart
plt.bar(months, sales, color='skyblue')
plt.xlabel('Month')
plt.ylabel('Sales ($)')
plt.title('Monthly Sales Data')
plt.grid(axis='y', linestyle='--', alpha=0.9)
plt.show()
```



[11]:
```python
import matplotlib.pyplot as plt

subjects = []
grades = []

while True:
    # Get subject input
    subject = input("Enter subject (or '4' to finish): ")
    if subject == '4':
        break
    subjects.append(subject)

    # Get grade input
    while True:
        try:
            grade = int(input(f"Enter grade for {subject}: "))
            if 0 <= grade <= 100:
                grades.append(grade)
                break
            else:
```

```
                print("Grade must be between 0 and 100. Please enter again.")
        except ValueError:
            print("Invalid input. Please enter a numerical value for the grade.
 ↪")

# Create the bar chart
plt.bar(subjects, grades, color='orange')
plt.xlabel('Subject')
plt.ylabel('Grade')
plt.title('Student Grades by Subject')

plt.xticks(rotation=45)  # Rotate x labels for better readability
plt.show()
```

```
Enter subject (or '4' to finish):  maths
Enter grade for maths:  90
Enter subject (or '4' to finish):  english
Enter grade for english:  23
Enter subject (or '4' to finish):  python
Enter grade for python:  45
Enter subject (or '4' to finish):  science
Enter grade for science:  90
Enter subject (or '4' to finish):  4
```

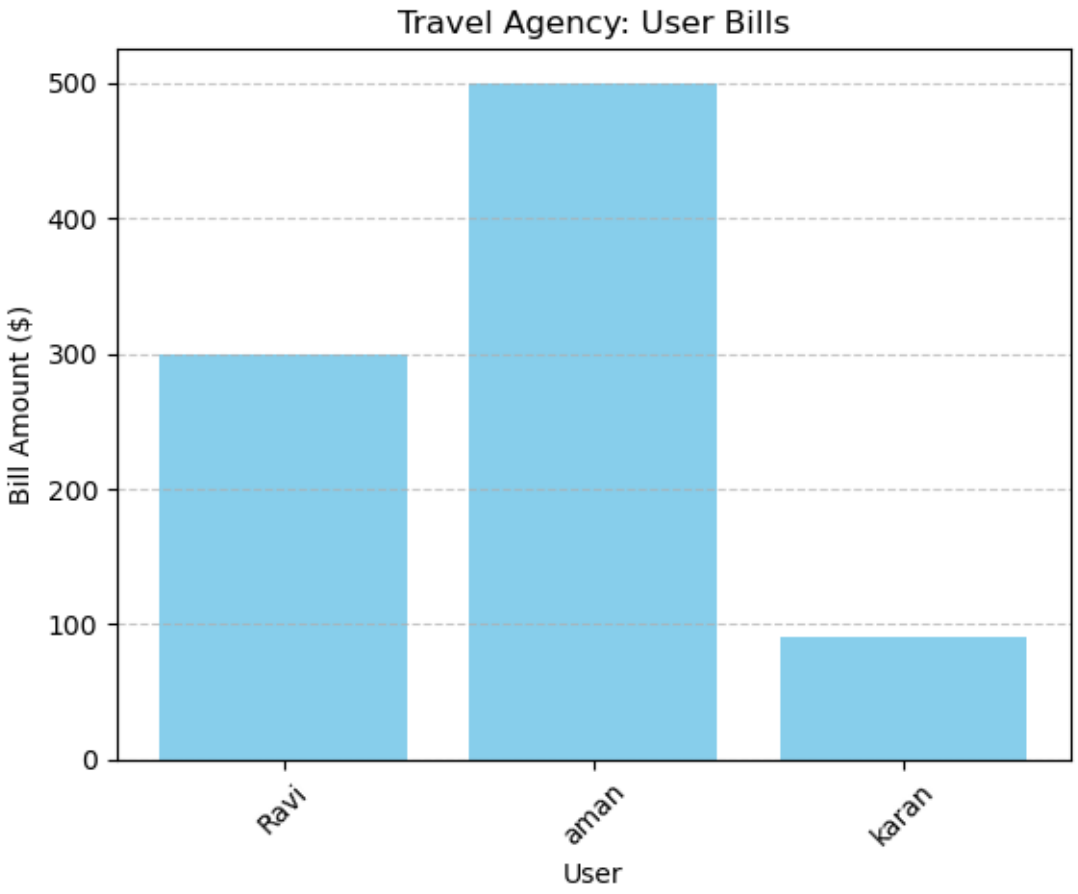


[15]:
```python
import matplotlib.pyplot as plt

# Dictionary to store the rates per kilometer for different car types
car_rates = {
    'Sedan': 10,        # Cost per kilometer
    'SUV': 15,
    'Hatchback': 8,
    'Luxury': 25
}

# Lists to store the names of users and their respective bills
users = []
bills = []

while True:
    print("\nSelect a car type from the following options:")
    for car in car_rates:
        print(f"- {car}")
```

```
    car_type = input("Enter the car type (or 'exit' to finish): ")
    if car_type.lower() == 'exit':
        break
    if car_type not in car_rates:
        print("Invalid car type. Please choose a valid option.")
        continue

    while True:
        try:
            kilometers = float(input("Enter the number of kilometers traveled:␣
↪"))
            if kilometers >= 0:
                break
            else:
                print("Kilometers must be non-negative. Please enter again.")
        except ValueError:
            print("Invalid input. Please enter a numerical value for kilometers.
↪")

    # Calculate the bill
    bill = kilometers * car_rates[car_type]

    # Store the user's name and bill
    user_name = input("Enter your name: ")
    users.append(user_name)
    bills.append(bill)

    print(f"Your bill is: ${bill:.2f}")

# Create the bar chart
plt.bar(users, bills, color='skyblue')
plt.xlabel('User')
plt.ylabel('Bill Amount ($)')
plt.title('Travel Agency: User Bills')
plt.xticks(rotation=45)  # Rotate x labels for better readability
  # Set y-axis limit to make the chart look better
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.show()
```

```
Select a car type from the following options:
- Sedan
- SUV
- Hatchback
- Luxury

Enter the car type (or 'exit' to finish):  Sedan
```

```
Enter the number of kilometers traveled:  30
Enter your name:  Ravi

Your bill is: $300.00


Select a car type from the following options:
- Sedan
- SUV
- Hatchback
- Luxury

Enter the car type (or 'exit' to finish):  Luxury
Enter the number of kilometers traveled:  20
Enter your name:  aman

Your bill is: $500.00


Select a car type from the following options:
- Sedan
- SUV
- Hatchback
- Luxury

Enter the car type (or 'exit' to finish):  SUV
Enter the number of kilometers traveled:  6
Enter your name:  karan

Your bill is: $90.00


Select a car type from the following options:
- Sedan
- SUV
- Hatchback
- Luxury

Enter the car type (or 'exit' to finish):  exit
```


Travel Agency: User Bills
Bill Amount ($)
500
400
300
200
100
0
Ravi
aman
karan
User

[ ]: