Question No-1

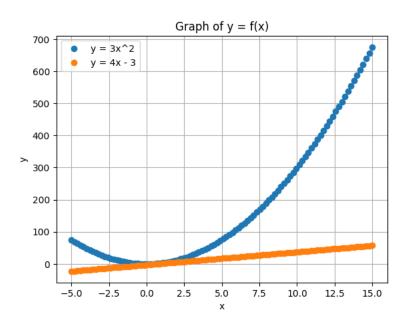
```
#Amit Babu Khatri(2358569)
# define the revised Harris-Benedict formula for calculating BMR
def calculate_bmr(gender, weight, height, age): #The function calculate_bmr takes four arguments: gender, weight, height, and age. It is used
   if gender == "M": #for male
       bmr = 88.362 + (13.397 * weight) + (4.799 * height) - (5.677 * age)
    elif gender == "F": #for female
       bmr = 447.593 + (9.247 * weight) + (3.098 * height) - (4.330 * age)
    return bmr
# prompt the user for their details
gender = input("Enter your gender (M/F): ") #prompt user to enter gender
weight = float(input("Enter your weight in kg: ")) #prompt user to enter weight
height = float(input("Enter your height in cm: ")) #prompt user to enter height
age = int(input("Enter your age in years: ")) #prompt user to enter age
# prompt the user for their level of exercise
print("Exercise Categories:")
print("0: Little to no exercise")
print("1: Light exercise (1-3 days per week)")
print("2: Moderate exercise (3-5 days per week)")
print("3: Heavy exercise (6-7 days per week)")
print("4: Very heavy exercise (twice per day, extra heavy workouts)")
exercise_category = int(input("Enter your exercise category code (0-4): "))
# calculate the BMR and recommended daily kilocalorie intake based on the user's details and exercise category
bmr = calculate_bmr(gender, weight, height, age)
if exercise_category == 0:
   daily_intake = bmr * 1.2
elif exercise_category == 1:
   daily_intake = bmr * 1.375
elif exercise_category == 2:
   daily_intake = bmr * 1.55
elif exercise_category == 3:
   daily intake = bmr * 1.725
elif exercise_category == 4:
   daily_intake = bmr * 1.9
# display the results to the user
print("Your BMR is:", bmr)
print("Your recommended daily kilocalorie intake is:", daily_intake)
     Enter your gender (M/F): M
     Enter your weight in kg: 88
    Enter your height in cm: 179
    Enter your age in years: 23
    Exercise Categories:
    0: Little to no exercise
    1: Light exercise (1-3 days per week)
    2: Moderate exercise (3-5 days per week)
    3: Heavy exercise (6-7 days per week)
    4: Very heavy exercise (twice per day, extra heavy workouts)
    Enter your exercise category code (0-4): 1
     Your BMR is: 1995.748
     Your recommended daily kilocalorie intake is: 2744.1535
Question No-2
#Amit Babu Khatri(2358569)
#Define a recursive function to calculate the sum of a range of numbers
def calculate_sum(low, high): # function calculate_sum takes two arguments, low and high, representing the lower and upper bounds of a range
    if low > high:#if the low number is greater than the high number, return 0
       return 0
    else:#add the current low number to the sum of the rest of the range
       return low + calculate_sum(low + 1, high)
#Get user input for the low and high numbers
   low = int(input("Enter the first low number: ")) #prompt the user to enter low number
   high = int(input("Enter the second high number: ")) #prompt the user to enter high number
except ValueError:
    print("Invalid input, please enter integers.") #print invalid reslut if user doesnot enter the interger number
#Check if the innut is valid (low should not be greater than high)
```

```
else:
    if low > high:
        print("Invalid input, low number should not be greater than high number.") #print if low number is greater than high number else:
        the_sum = calculate_sum(low, high) #calculate the sum
        print("The answer is:", the_sum) #print the sum

Enter the first low number: 2
Enter the second high number: 69
The answer is: 2414
```

Question No-3

```
#Amit Babu Khatri(2358569)
import matplotlib.pyplot as plt #imports the matplotlib.pyplot library as plt
import numpy as np #numpy library as np
def function1(x):
    return 3 * x ** 2 #for function 1
def function2(x):
   return 4 * x - 3 #for function2
# Generate x values between -5 and 15
x = np.linspace(-5, 15, 100)
# Calculate y values for both functions
y1 = function1(x)
y2 = function2(x)
# Plot the functions
plt.plot(x, y1, 'o', label='y = 3x^2')
plt.plot(x, y2, 'o', label='y = 4x - 3')
# Set the title and axes labels
plt.title('Graph of y = f(x)')
plt.xlabel('x')
plt.ylabel('y')
# Enable the grid
plt.grid(True)
# Show the legend
plt.legend()
# Show the plot
plt.show()
```



- -