BMI CALCULATOR

Introduction to Python - Project

ABSTRACT

BMI Calculator to calculator your Body Mass Index using universal formula. This BMI Calculator can recommend average calories intake according to your results. This code allows you to convert your weight measure from Pounds to Kilograms and Height from Feet to Meter.

Kavya Sharawat

The BMI Calculator has been designed through Python to provide the following tasks:

1. **User Input and Conversion Display:**
2. The program prompts the user to input their weight in pounds and height in feet. The program uses the user-inputted weight and height values to calculate and display the converted values in kilograms and meters.
3. **Conversion Functions:**
    - **pounds_to_kg**: Converts weight from pounds to kilograms.
    - **feet_to_meters**: Converts height from feet to meters.
4. **BMI Calculation:**
    - The **calculate_bmi** function calculates the BMI using the formula: BMI = weight (kg) / (height (m))^2.
5. **BMI Category Determination:**
    - The **get_bmi_category** function determines the BMI category based on standard BMI ranges (Underweight, Normal Weight, Overweight, Obese).
6. **Caloric Intake Recommendation:**
    - The **recommend_caloric_intake** function provides recommendations for caloric intake adjustment based on BMI category.
7. **Conversion Table Display:**
    - The **display_conversion_table** function displays a conversion table for weight and height, showing the inputted and converted values.
8. **Main BMI Calculator Function:**
    - The **bmi_calculator** function integrates all the above functionalities. It takes user input, performs conversions, calculates BMI, determines the BMI category, recommends caloric intake, and displays the conversion table.
9. **Exception Handling:**
    - The program includes basic exception handling to handle cases where the user inputs invalid values (non-numeric input).
10. **Executable Code:**
    - The code is structured to be executed as a standalone script. When run, it interacts with the user, performs the necessary calculations, and provides feedback.
11. **Modularity:**
    - The code is modular, with separate functions for different tasks, promoting code readability and reusability.

CODE:

In [6]:
```python
def pounds_to_kg(weight_in_pounds):
 #Convert weight from pounds to kilograms

    return weight_in_pounds * 0.453592

def feet_to_meters(height_in_feet):

    #Convert height from feet to meters

    return height_in_feet * 0.3048

def convert_and_display():
    try:
        weight_pounds = float(input("Enter your weight in pounds: "))
        height_feet = float(input("Enter your height in feet: "))
    except ValueError:
        print("Invalid input. Please enter valid numbers.")
        return

    weight_kg = pounds_to_kg(weight_pounds)
    height_meters = feet_to_meters(height_feet)

    print(f"\nConverted Values:")
    print(f"Weight: {weight_pounds} pounds = {weight_kg:.2f} kilograms")
    print(f"Height: {height_feet} feet = {height_meters:.2f} meters")

if __name__ == "__Main__":
    convert_and_display()
```

Result:

```
Enter your weight in pounds: 40
Enter your height in feet: 50

Converted Values:
Weight: 40.0 pounds = 18.14 kilograms
Height: 50.0 feet = 15.24 meters
```

In [10]:

```python
def calculate_bmi(weight, height):

    #Calculate BMI using the formula: BMI = weight (kg) / (height (m))^2

    if height <= 0:
        return None  # Avoid division by zero
    bmi = weight / (height ** 2)
    return bmi

def get_bmi_category(bmi):

#Determine BMI category based on standard BMI ranges

    if bmi is None:
        return "Invalid input"
    elif bmi < 18.5:
        return "Underweight"
    elif 18.5 <= bmi < 24.9:
        return "Normal Weight"
    elif 25 <= bmi < 29.9:
        return "Overweight"
    else:
        return "Obese"

def recommend_caloric_intake(bmi_category):

    #Recommend caloric intake adjustment based on BMI category

    if bmi_category == "Underweight":
        return "For weight gain: Consume around 500 extra calories per day."
    elif bmi_category == "Normal Weight":
        return "For weight maintenance: Maintain your current caloric intake."
    elif bmi_category == "Overweight":
        return "For weight loss: Consume around 500 fewer calories per day."
```

```
    elif bmi_category == "Obese":
        return "For significant weight loss: Consume around 1000 fewer calories per day
    else:
        return "Invalid BMI category."

def bmi_calculator():
    print("BMI Calculator with Daily Caloric Intake Recommendation")

    while True:
        try:
            weight = float(input("Enter your current weight in kilograms: "))
            height = float(input("Enter your height in meters: "))
            break
        except ValueError:
            print("Invalid input. Please enter a valid number.")

    bmi = calculate_bmi(weight, height)
    category = get_bmi_category(bmi)

    print(f"\nYour BMI is: {bmi:.2f}")
    print(f"Category: {category}")

    if category == "Invalid input":
        print("Please enter valid weight and height values.")
    else:
        recommendation = recommend_caloric_intake(category)

        print(f"\nRecommendation: {recommendation}")

if __name__ == "__main__":
    bmi_calculator()
```

Result:

```
BMI Calculator with Daily Caloric Intake Recommendation
Enter your current weight in kilograms: 38
Enter your height in meters: 1.63

Your BMI is: 14.30
Category: Underweight

Recommendation: For weight gain: Consume around 500 extra calories per day.
```