Internship Project Report – Task 2

BMI Calculator – Task Report Content

Intern Details:

- Name: Sandhiya A
- Domain: Python Programming
- Internship Company: Oasis Infobyte
- College: Podhigai College of Engineering and Technology
- Academic Year: 2022–2026
- CGPA: 8.83

Objective:

To develop a Python program that calculates Body Mass Index (BMI) based on user-provided height and weight, and classifies the result into standard health categories like Underweight, Normal, Overweight, or Obese.

Technologies Used:

- Python 3.x
- input() function
- float() conversion
- Conditional statements (if, elif, else)

- Error handling using try-except
- Python IDLE or VS Code
- Screenshot/recording tools

Logic Used:

1. Take weight (kg) and height (cm) from the user
2. Convert height from cm to meters
3. Calculate BMI using the formula:

python

Copy code

BMI = weight / (height_m ** 2)

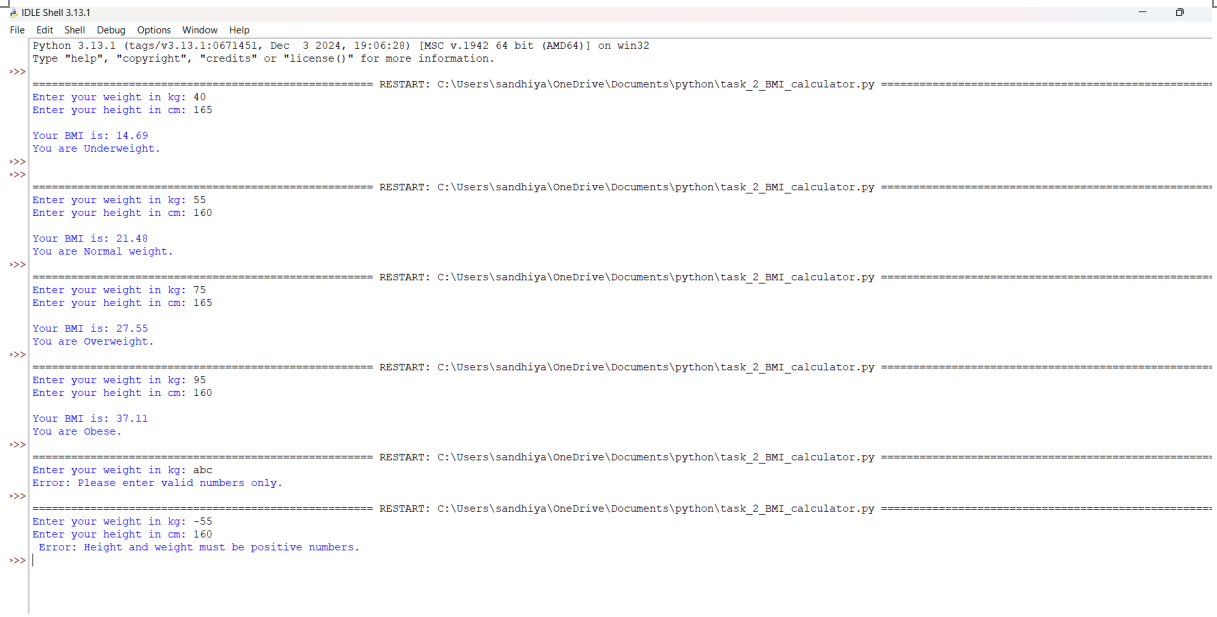
4. Classify the result:
    - Underweight: BMI < 18.5
    - Normal weight: 18.5–24.9
    - Overweight: 25–29.9
    - Obese: 30 and above
5. Handle invalid input using try/except and check for non-positive values

Source Python Code:

try:

```
    weight = float(input("Enter your weight in kg: "))
    height_cm = float(input("Enter your height in cm: "))
    if weight <= 0 or height_cm <= 0:
        print("     Error: Height and weight must be positive
numbers.")
    else:
        height_m = height_cm / 100
        bmi = weight / (height_m ** 2)
        print(f"\nYour BMI is: {bmi:.2f}")
        if bmi < 18.5:
            print("You are Underweight.")
        elif 18.5 <= bmi < 24.9:
            print("You are Normal weight.")
        elif 25 <= bmi < 29.9:
            print("You are Overweight.")
        else:
            print("You are Obese.")
except ValueError:
  print("     Error: Please enter valid numbers only”)
```

Output Screenshot (Both valid and invalid input):



Conclusion:

This project demonstrates the use of Python for real-world health applications. The BMI Calculator not only performs numeric computation but also applies conditional logic and error handling to ensure user-friendly and reliable output.