

PROJECT REPORT

Diet Planner Application Using Python

1. Introduction

Maintaining a balanced diet is important for achieving personal health goals such as weight loss, muscle gain, or simply staying fit. Many people find it confusing to calculate their daily calorie requirements and divide them into proper macronutrients. To address this, I developed a **Python-based Diet Planner** that calculates daily calorie needs and generates a customized food plan based on user details such as age, gender, activity level, goal (bulking or shredding), and food preference (veg/non-veg).

The main objective of this project is to make diet planning simple, accurate, and user-friendly.

2. Objectives

The project aims to:

- Calculate required daily calories based on user profile.
 - Distribute calories across macronutrients (protein, carbs, fats).
 - Suggest food items based on dietary preference.
 - Give a structured diet plan suitable for beginners.
 - Demonstrate practical use of functions, dictionaries, loops, and user input handling in Python.
-

3. System Requirements

- Programming Language: **Python 3.x**
 - Libraries Used: math, random, and collections.Counter
 - Platform: Any system capable of running Python (Windows/macOS/Linux)
-

4. Methodology

4.1 User Inputs Collected

The program asks the user for:

- Age

- Gender
- Weight
- Height
- Activity level (sedentary, light, moderate, very active, athlete)
- Fitness goal (bulking or shredding)
- Food type (veg or non-veg)

4.2 Calculating Calorie Requirements

The program uses the **Mifflin-St Jeor Equation** to calculate BMR (Basal Metabolic Rate).

Then, BMR is multiplied by an **activity factor** to get TDEE (Total Daily Energy Expenditure).

Depending on the goal:

- **Bulking:** Calories are increased.
- **Shredding:** Calories are reduced.

4.3 Macronutrient Breakdown

The program assigns macronutrient ratios based on the user's goal:

- Shredding → Higher protein, lower carbs
- Bulking → Higher carbs, moderate protein
- Maintenance → Balanced

The program then calculates:

- Protein (in grams)
- Carbohydrates (in grams)
- Fats (in grams)

4.4 Food Selection

A custom food database is stored in Python dictionaries.

Based on the user's preference (veg/non-veg), food items are recommended in each meal category:

- Breakfast
- Lunch
- Snacks

- Dinner

Food items are selected to match the recommended protein/carb/fat distribution.

5. Features of the Diet Planner

- Personalized calorie calculation
 - Dynamic macro distribution
 - Supports vegetarian and non-vegetarian diets
 - Beginner-friendly interface
 - Modular structure using functions
 - Easy to modify or extend food database
-

6. Code Overview (Short Explanation)

The program is divided into logical sections:

1. **Constants** – Activity levels, macro ratios, food database
 2. **Functions**
 - BMR calculation
 - Calorie adjustment
 - Macro calculation
 - Food recommendation
 3. **Main Driver Code** – Handles user interaction and prints the final diet plan
-

7. Sample Output (Summary Format)

Daily Calorie Requirement: 2200 kcal

Protein: 150g

Carbohydrates: 250g

Fats: 70g

Recommended Meals:

Breakfast: Oats with milk and fruits

Lunch: Rice, Paneer/Chicken, Vegetables

Snacks: Nuts or protein shake

Dinner: Roti with Dal/Chicken Curry

8. Conclusion

This project successfully demonstrates how Python can be used to solve a real-world problem like diet planning. The program calculates caloric needs accurately and generates personalized meal plans based on the user's goals and food preferences. It also uses key Python concepts such as dictionaries, functions, loops, and user inputs in a practical and meaningful way.

Future enhancements may include GUI support, calorie tracking over time, or integration with online nutritional APIs.

END OF REPORT
