# Reflection Log

```
package mastery;

public class Food {
    private double price;
    private int fat;
    private int carbs;
    private int fiber;
```

**This code defines a Food class that encapsulates the attributes of a food item: its price (as a double), fat, carbohydrates, and fiber (all as integers). By keeping these fields private, the class ensures encapsulation and allows controlled access through getter and setter methods. This design promotes modularity and reusability, making it easier to manage the properties of food items in the application.**

```
  // Constructor to initialize the food item's details
  public Food(double price, int fat, int carbs, int fiber) {
      this.price = price;
      this.fat = fat;
      this.carbs = carbs;
      this.fiber = fiber;
  }
```

**This constructor initializes the Food class by assigning values to its attributes: price, fat, carbs, and fiber. It ensures that every Food object is created with complete and defined details upon instantiation.**

```
    // Getter methods for food properties
    public double getPrice() {
        return price;
    }

    public int getFat() {
        return fat;
    }

    public int getCarbs() {
        return carbs;
    }

    public int getFiber() {
        return fiber;
    }
}
```

**This code defines getter methods to access the private attributes of the Food class: price, fat, carbs, and fiber. Each method returns the corresponding value of the attribute,**

**providing controlled access to the data. This ensures encapsulation while allowing other parts of the program to retrieve the nutritional details of a food item.**