# **🍔 Problem Statement: Restaurant Ordering System**

Write a Python program to simulate a **restaurant ordering system**.

## **✅ Requirements**

1. **Display Menu**
   1. Show a menu with food items and their prices.
   2. Include both **Vegetarian (Veg)** and **Non-Vegetarian (Non-Veg)** items.
   3. Example:

|  |  |  |
| --- | --- | --- |
| **Item** | **Price** | **Type** |
| Paneer Curry | 250 | Veg |
| Chicken Biryani | 300 | Non-Veg |
| Dal Tadka | 150 | Veg |
| Fish Fry | 280 | Non-Veg |
| Salad | 100 | Veg |

1. **Take Order**
   1. User can select **multiple items** in a loop until they finish ordering.
   2. Ask for **quantity** for each selected item.
   3. Maintain a **cart** storing items, type, quantity, and price.
2. **Checkout**
   1. Calculate total bill.
   2. Apply rules for free items or restrictions.

## **💰 Ordering Rules**

1. If **order total > ₹1000**, add a **free dessert item** (e.g., Ice Cream).
2. If customer chooses **Vegetarian mode**, disallow selection of Non-Veg items.
3. Show final order summary including:
   1. List of items ordered with quantities and price
   2. Total bill
   3. Free dessert (if applicable)

## **🔹 Concepts to Use**

* **Functions**
  + show\_menu() → Display menu items.
  + add\_to\_order() → Allow user to select items and quantity, enforce Veg/Non-Veg restrictions.
  + calculate\_bill() → Compute total, check for free dessert.
  + restaurant\_system() → Main function controlling the flow.
* **Loops**
  + To allow users to order multiple items until they choose to stop.
* **Conditions**
  + Veg mode → prevent Non-Veg items.
  + Apply free dessert if total > ₹1000.
* **Data Structures**
  + Use **list of dictionaries** for menu items:

menu = [  
 {"name": "Paneer Curry", "price": 250, "type": "Veg"},  
 {"name": "Chicken Biryani", "price": 300, "type": "Non-Veg"},  
 {"name": "Dal Tadka", "price": 150, "type": "Veg"},  
 {"name": "Fish Fry", "price": 280, "type": "Non-Veg"}  
]

* + Use a **dictionary** for the cart: { "Paneer Curry": {"quantity": 2, "price": 250} }
* **Arithmetic Operations**
  + Multiply quantity × price for each item
  + Sum totals, check for rules (free dessert)
* **String Formatting**
  + Print a clear **final order summary** similar to a restaurant receipt.

## **🚀 Implementation Strategy**

1. **Step 1: Setup Menu Data**
   1. Define list of food items with name, price, type.
2. **Step 2: Ask for Veg/Non-Veg Mode**
   1. Optionally restrict selection to Veg items.
3. **Step 3: Display Menu**
   1. Show only allowed items (depending on Veg mode).
4. **Step 4: Take Orders in Loop**
   1. User selects an item by number or name.
   2. Ask for quantity.
   3. Validate selection (Veg restriction, valid item).
   4. Add to cart dictionary.
5. **Step 5: Calculate Total**
   1. Sum item × quantity for all items.
   2. If total > 1000 → add free dessert to cart.
6. **Step 6: Show Order Summary**
   1. List items with quantity and subtotal
   2. Show total bill, free dessert if added
7. **Step 7: Loop Option**
   1. Optionally, allow another order session until user exits.