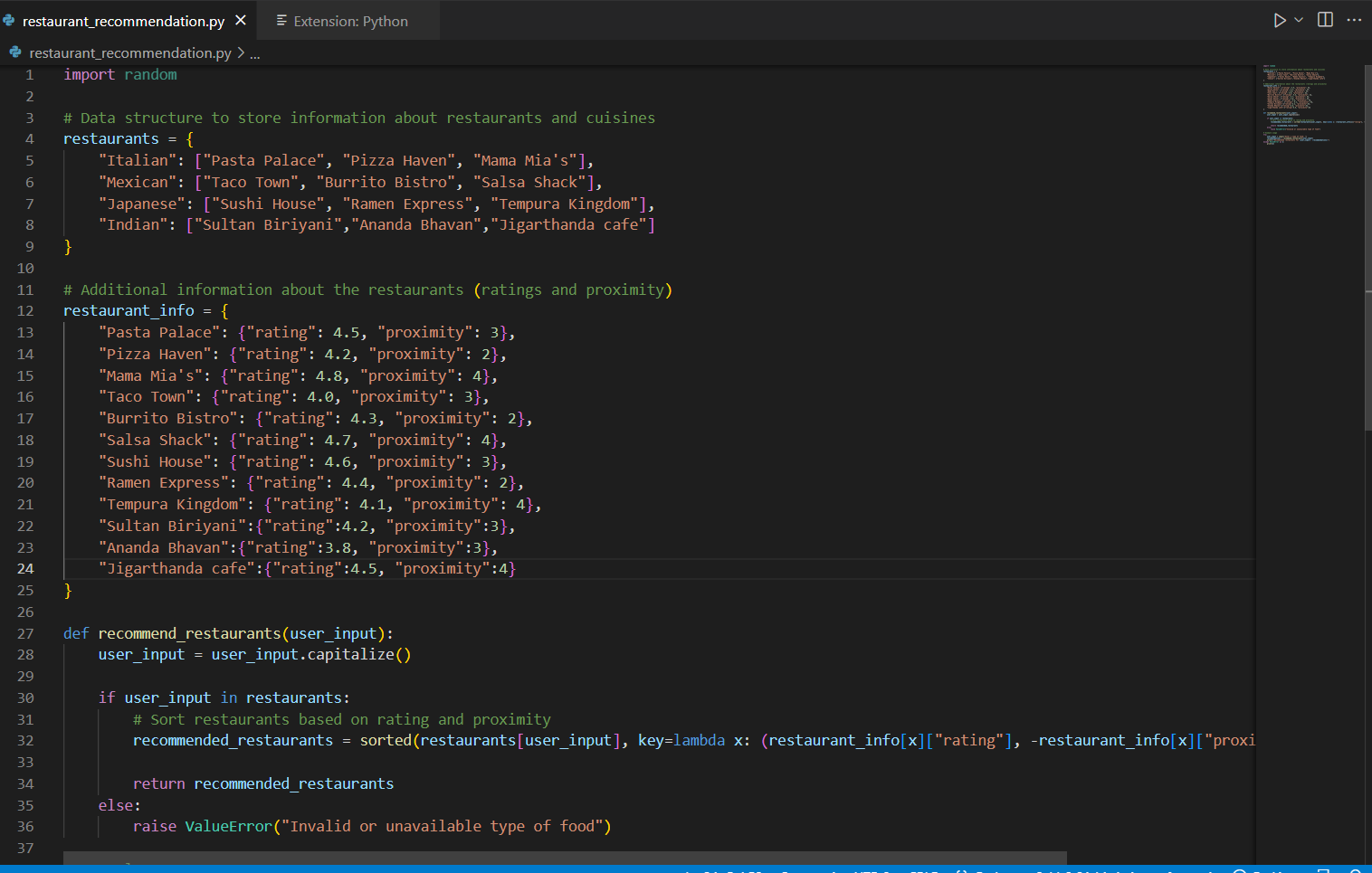
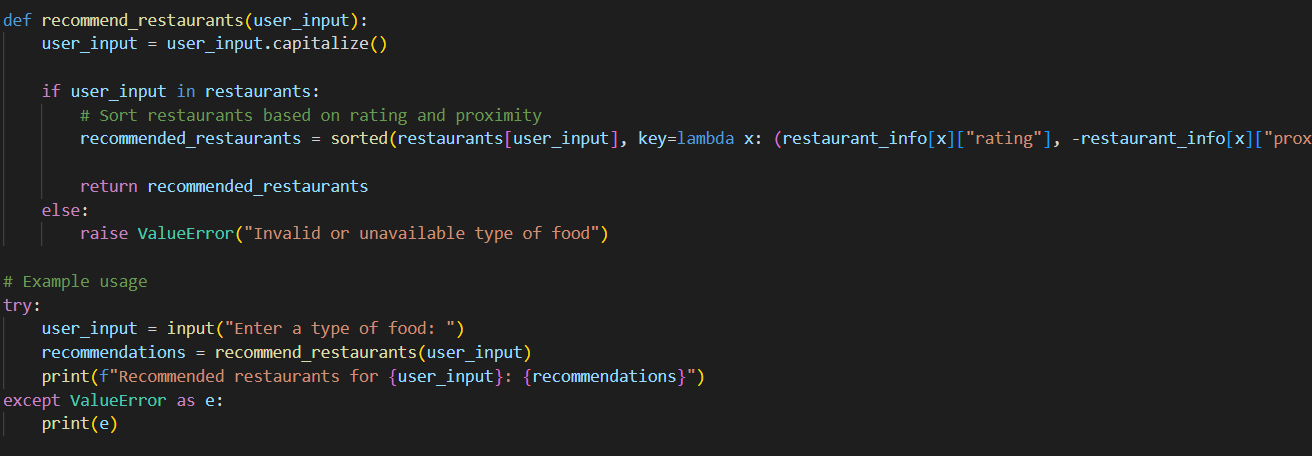
**Restaurant Recommendation System**

**Code:**





import random

# Data structure to store information about restaurants and cuisines

restaurants = {

"Italian": ["Pasta Palace", "Pizza Haven", "Mama Mia's"],

"Mexican": ["Taco Town", "Burrito Bistro", "Salsa Shack"],

"Japanese": ["Sushi House", "Ramen Express", "Tempura Kingdom"],

“Indian”: [“Sultan Biriyani”, “Anandha Bhavan”, “Jigarthanda café”]

}

# Additional information about the restaurants (ratings and proximity)

restaurant\_info = {

"Pasta Palace": {"rating": 4.5, "proximity": 3},

"Pizza Haven": {"rating": 4.2, "proximity": 2},

"Mama Mia's": {"rating": 4.8, "proximity": 4},

"Taco Town": {"rating": 4.0, "proximity": 3},

"Burrito Bistro": {"rating": 4.3, "proximity": 2},

"Salsa Shack": {"rating": 4.7, "proximity": 4},

"Sushi House": {"rating": 4.6, "proximity": 3},

"Ramen Express": {"rating": 4.4, "proximity": 2},

"Tempura Kingdom": {"rating": 4.1, "proximity": 4},

“Sultan Biriyani”: {“rating”:4.2, “proximity”:3},

"Anandha Bhavan": {"rating": 3.8, "proximity": 3},

"Jigarthanda cafe": {"rating": 4.5, "proximity": 4},

}

def recommend\_restaurants(user\_input):

user\_input = user\_input.capitalize()

if user\_input in restaurants:

# Sort restaurants based on rating and proximity

recommended\_restaurants = sorted(restaurants[user\_input], key=lambda x: (restaurant\_info[x]["rating"], -restaurant\_info[x]["proximity"]), reverse=True)

return recommended\_restaurants

else:

raise ValueError("Invalid or unavailable type of food")

# Example usage

try:

user\_input = input("Enter a type of food: ")

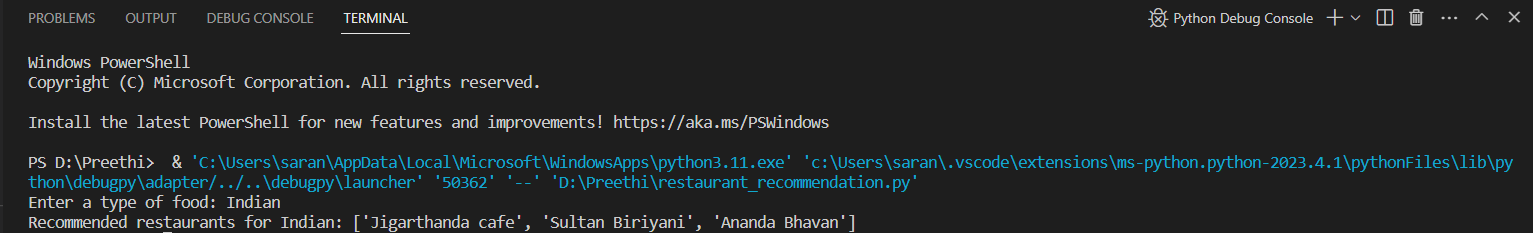
recommendations = recommend\_restaurants(user\_input)

print(f"Recommended restaurants for {user\_input}: {recommendations}")

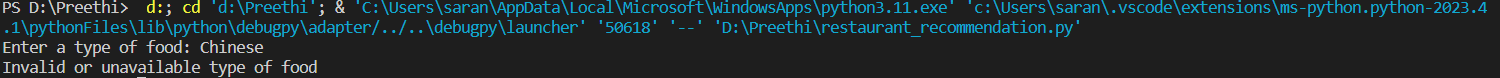
except ValueError as e:

print(e)

**Output:**



****

****

* Here we have designed and developed a python program that allows users to input a type of food and based on which the user gets recommendations as in which place is the best to eat that particular cuisine.
* So here the information about different restaurants and their cuisines are stored in the form of data structure where the recommendations and cuisines are arranged in a stack order.
* Here a “def” function (recommendation\_restaurant) has been developed which takes the user's input (type of food) and recommends restaurants that serve that cuisine which considers the factors such as user preferences, restaurant ratings, and proximity.
* If the cuisine available on the database it shows that particular cuisine and recommendations to eat e.g If Indian is selected it recommends only the restaurants with Indian cuisines.
* If the cuisine entered is not on the system it shows invalid data or unavailable type of food. For e.g If Chinese is entered it shows invalid or unavailable type of food.