

Untitled-3

```
1 def determine_bmi_category(height, weight):
2     """
3     Determine the BMI category based on height and weight.
4
5     :param height: Height in meters
6     :param weight: Weight in kilograms
7     :return: The BMI category as a string
8     """
9     # Calculate BMI
10    bmi = weight / (height ** 2)
11
12    # Determine the BMI category
13    if bmi >= 30:
14        return "Obesity"
15    elif 25 <= bmi < 30:
16        return "Overweight"
17    elif 18.5 <= bmi < 25:
18        return "Normal"
19    else:
20        return "Underweight"
21
22 def main():
23     # Input from the user
24     height = float(input("Enter height in meters: "))
25     weight = float(input("Enter weight in kilograms: "))
26
27     # Determine BMI category
28     category = determine_bmi_category(height, weight)
29
30     # Print the result
31     print(f"Your BMI category is: {category}")
32
33 if __name__ == "__main__":
34     main()
35
```

Untitled-4

```
1 def find_country_by_city(city_name):
2     """
3     Find which country a city belongs to based on predefined lists.
4
5     :param city_name: The name of the city to search for
6     :return: The country where the city is located
7     """
8     # Predefined lists of cities per country
9     australia = ["Sydney", "Melbourne", "Brisbane", "Perth"]
10    uae = ["Dubai", "Abu Dhabi", "Sharjah", "Ajman"]
11    india = ["Mumbai", "Bangalore", "Chennai", "Delhi"]
12
13    # Check the city and return the corresponding country
14    if city_name in australia:
15        return "Australia"
16    elif city_name in uae:
17        return "UAE"
18    elif city_name in india:
19        return "India"
20    else:
21        return "City not found in predefined lists"
22
23 def main():
24     # Input from the user
25     city_name = input("Enter a city name: ")
26
27     # Find the country by city
28     country = find_country_by_city(city_name)
29
30     # Print the result
31     if country == "City not found in predefined lists":
32         print(country)
33     else:
34         print(f"{city_name} is in {country}")
35
36 if __name__ == "__main__":
37     main()
38
```

Untitled-5

```
1 def find_country_by_city(city_name):
2     """
3     Find which country a city belongs to based on predefined lists.
4
5     :param city_name: The name of the city to search for
6     :return: The country where the city is located, or None if not found
7     """
8     # Predefined lists of cities per country
9     australia = ["Sydney", "Melbourne", "Brisbane", "Perth"]
10    uae = ["Dubai", "Abu Dhabi", "Sharjah", "Ajman"]
11    india = ["Mumbai", "Bangalore", "Chennai", "Delhi"]
12
13    # Check the city and return the corresponding country
14    if city_name in australia:
15        return "Australia"
16    elif city_name in uae:
17        return "UAE"
18    elif city_name in india:
19        return "India"
20    else:
21        return None
22
23 def main():
24     # Input from the user
25     city1 = input("Enter the first city: ")
26     city2 = input("Enter the second city: ")
27
28     # Find the countries for both cities
29     country1 = find_country_by_city(city1)
30     country2 = find_country_by_city(city2)
31
32     # Check if both cities belong to the same country
33     if country1 and country2:
34         if country1 == country2:
35             print(f"Both cities are in {country1}")
36         else:
37             print("They don't belong to the same country")
38     else:
39         print("One or both cities are not found in the predefined lists")
40
41 if __name__ == "__main__":
42     main()
43
```