Python Assignment 3: While Loop, For loop and Function

While loop & Control Statements :

(else, break, continue, pass)

Number Guessing Game

Problem Statement: Create a Python program that implements a simple number guessing game using a `while` loop. The program should make use of control statements such as `else`, `break`, and `continue`.

Instructions:

1. **Set Up the Game:**
   - ★ Generate a random number between 1 and 10 that the user has to guess. Import random and use randint function.

```
import random
# Generate random number between 1 and 10
secret_number = random.randint(1, 10)
```

2. **Prompt the User:**
   - ★ Ask the user to guess the number.
   - ★ Set a variable `attempts` to 3, which represents the maximum number of guesses allowed.
3. **Implement the Guessing Logic:**
   - ★ Use a `while` loop to allow the user to keep guessing until they get the correct number or run out of attempts.
   - ★ Provide feedback to the user for each guess:
     - ➢ If the guess is out of the valid range (1 to 10), inform the user.
     - ➢ If the guess is greater than the secret number.
     - ➢ If the guess is lower than the secret number.
     - ➢ If the guess is correct, congratulate the user and end the game.
4. **Control Statements:**
   - ★ Use `continue` to skip the rest of the loop after informing the user if the guess is out of range.

- ★ Use `break` to exit the loop when the guess is correct.
- ★ Use `else` with the `while` loop to provide a message like *"Better luck next time!"* if the user runs out of attempts without guessing the correct number.

Sample Output:

```
Guess the number (between 1 and 10): 2
Too low. Try again.
Guess the number (between 1 and 10): 15
Your guess is out of range. Please guess a number between 1 and 10.
Guess the number (between 1 and 10): 5
Too high. Try again.
Guess the number (between 1 and 10): 3
Congratulations! You guessed the correct number.
```

For Loop:

Multiplication Table Generator

Problem Statement: Create a Python program that generates and prints a multiplication table (from 1 to 10) for a given number using a `for` loop and the `range` function.

Step wise Instructions:

1. Prompt user for Input. Ask the user to enter a number for which they want to generate a multiplication table.
2. Generate the Multiplication Table: Use a `for` loop to iterate through the numbers 1 to 10. In each iteration, calculate the product of the user's number and the current number from the loop.
3. Display the Multiplication Table:
   Print each line of the multiplication table in the format: "number x i = result".

Sample Output:

```
Enter the number for which you want the multiplication table: 5
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
5 x 6 = 30
5 x 7 = 35
5 x 8 = 40
5 x 9 = 45
5 x 10 = 50
```

Function:

BMI Calculator

Problem Statement: Create a Python program that calculates the Body Mass Index (BMI).

Hint: BMI = weight (kg) / [height (m)]²

Instructions:

1. Define a function `calculate_bmi(weight, height)` that returns the BMI.
2. Prompt the user for their weight (in kg) and height (in meters).
3. Use the function to calculate and display the BMI

Sample Output:

```
Enter your weight in kg: 58
Enter your height in meters: 1.62
Your BMI is: 22.10
```