PYTHON FUNCTIONS

QUESTION 1:

What does the len() function do in Python? Write a code example using len() to find the length of a list.

The len() function in Python returns the number of elements in an object, such as a list, string, tuple, or dictionary.
It is commonly used to determine the size of these data structures.

example 1:

```
list_A = [10, 20, 30, 40, 50]

length_of_list = len(list_A)

print("The length of the list is:", length_of_list)
```

OUTPUT:

```
The length of the list is: 5

Process finished with exit code 0
```

example 2:

```
user_input = input("Enter a list of numbers or items, separated by commas: ")
user_list = user_input.split(",")  # Split the input string into list items
list_length = len(user_list)
print("The length of the list is:", list_length)
```

OUTPUT:

```
Enter a list of numbers or items, separated by commas: 11,apple,orange,55,100,56,99
The length of the list is: 7

Process finished with exit code 0
```

QUESTION 2:

Write a Python function greet(name) that takes a person's name as input and prints "Hello, [name]!".

```
def greet():
    name = input("Please enter your name: ")
    print(f"Hello, {name}!")
```

```
greet()
```

OUTPUT:

```
Please enter your name: hiba
Hello, hiba!

Process finished with exit code 0
```

QUESTION 3:

Write a Python function find_maximum(numbers) that takes a list of integers and returns the maximum value without using the built-in max() function. Use a loop to iterate through the list and compare values.

```
def find_maximum(numbers):

    if len(numbers) == 0:
        return None

    max_value = numbers[0]

    for num in numbers:
        if num > max_value:
            max_value = num
    return max_value



user_input = input("Please enter a list of integers separated by commas: ")
user_list = [int(num.strip()) for num in user_input.split(",")]

max_value = find_maximum(user_list)
if max_value is not None:
    print("The maximum value is:", max_value)
else:
    print("No numbers were entered.")
```

OUTPUT:

```
Please enter a list of integers separated by commas: 1,2,67,99,122,556,77
The maximum value is: 556

Process finished with exit code 0
```

QUESTION 4:

Explain the difference between local and global variables in a Python function. Write a program where a global variable and a local variable have the same name and show how Python differentiates between them.

```
"""
Local Variables:
Local variables are defined within a function and can only be accessed inside that
function.
They exist only during the execution of the function.
They are created when the function is called and destroyed when the function exits.

Global Variables:
Global variables are defined outside of any function and can be accessed from any
function within the same module.
They have a broader scope compared to local variables.
They exist for the duration of the program's execution.

"""
#
# Global variable
value = 10
def example():
    # local variable(name same as that of global variable)
    value = 5
    print("Inside the function, local value:", value)

example()

# Accessing the global variable
print("Outside the function, global value:", value)
```

OUTPUT:

```
Inside the function, local value: 5
Outside the function, global value: 10

Process finished with exit code 0
```

QUESTION 5:

Create a function calculate_area(length, width=5) that calculates the area of a rectangle. If only the length is provided, the function should assume the width is 5. Show how the function behaves when called with and without the width argument.

```
def calculate_area(length, width=5):
    area = length * width
    return area
```

```
#with the width argument.
area_of_rectangle_1= calculate_area(10, 3)
print("Area of rectangle with length 10 and width 3:", area_of_rectangle_1)

#without the width argument.
area_of_rectangle_2 = calculate_area(10)
print("Area of rectangle with length 10 and default width 5:", area_of_rectangle_2)

OUTPUT:
  Area of rectangle with length 10 and width 3: 30
  Area of rectangle with length 10 and default width 5: 50

  Process finished with exit code 0
```