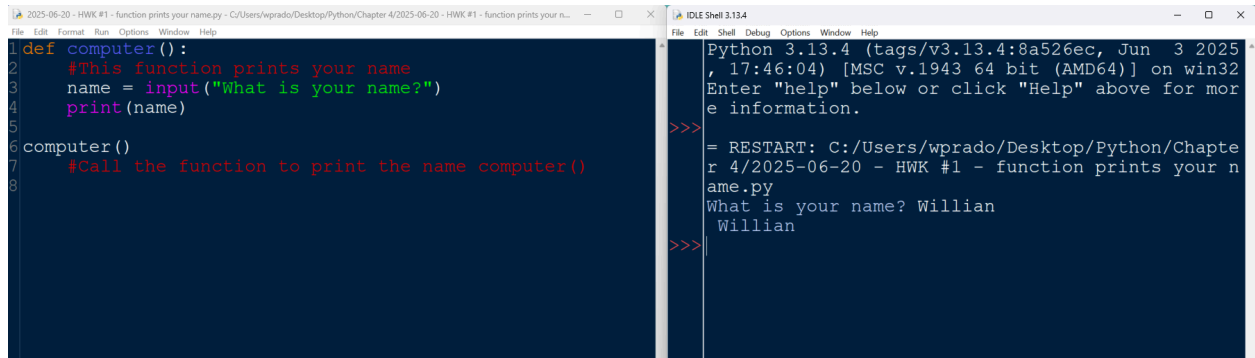


Chapter 5 Functions HW (1)

#1. Write a function named **computer()** that prints your name.



The screenshot shows two windows from a Python IDE. The left window, titled '2025-06-20 - HWK #1 - function prints your name.py', contains the following code:

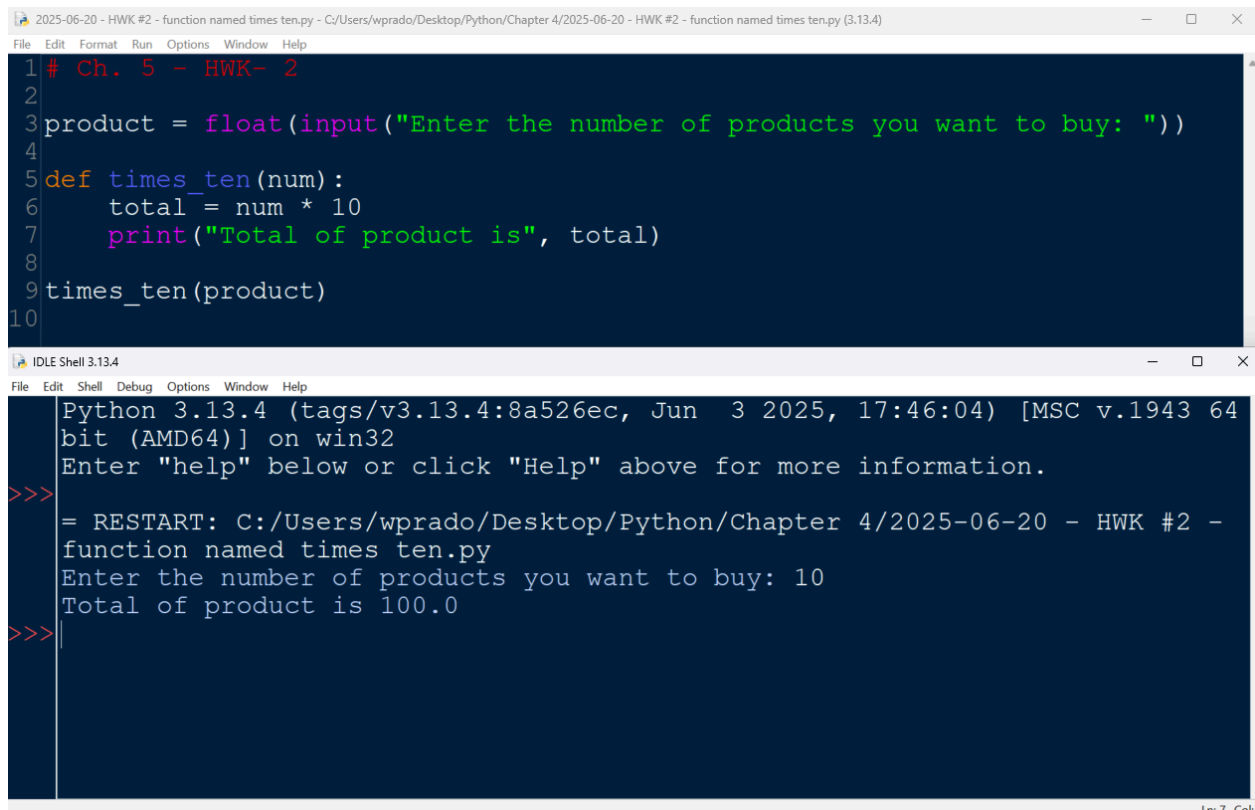
```
1 def computer():
2     #This function prints your name
3     name = input("What is your name?")
4     print(name)
5
6 computer()
7     #Call the function to print the name computer()
8
```

The right window, titled 'IDLE Shell 3.13.4', shows the execution output:

```
Python 3.13.4 (tags/v3.13.4:8a526ec, Jun 3 2025, 17:46:04) [MSC v.1943 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.

>>>
= RESTART: C:/Users/wprado/Desktop/Python/Chapter 4/2025-06-20 - HWK #1 - function prints your name.py
What is your name? Willian
Willian
>>>
```

#2. Write a function named **times_ten()** the function should accept an argument from a user and display the product of its argument multiplied times 10.



The screenshot shows two windows from a Python IDE. The top window, titled '2025-06-20 - HWK #2 - function named times ten.py', contains the following code:

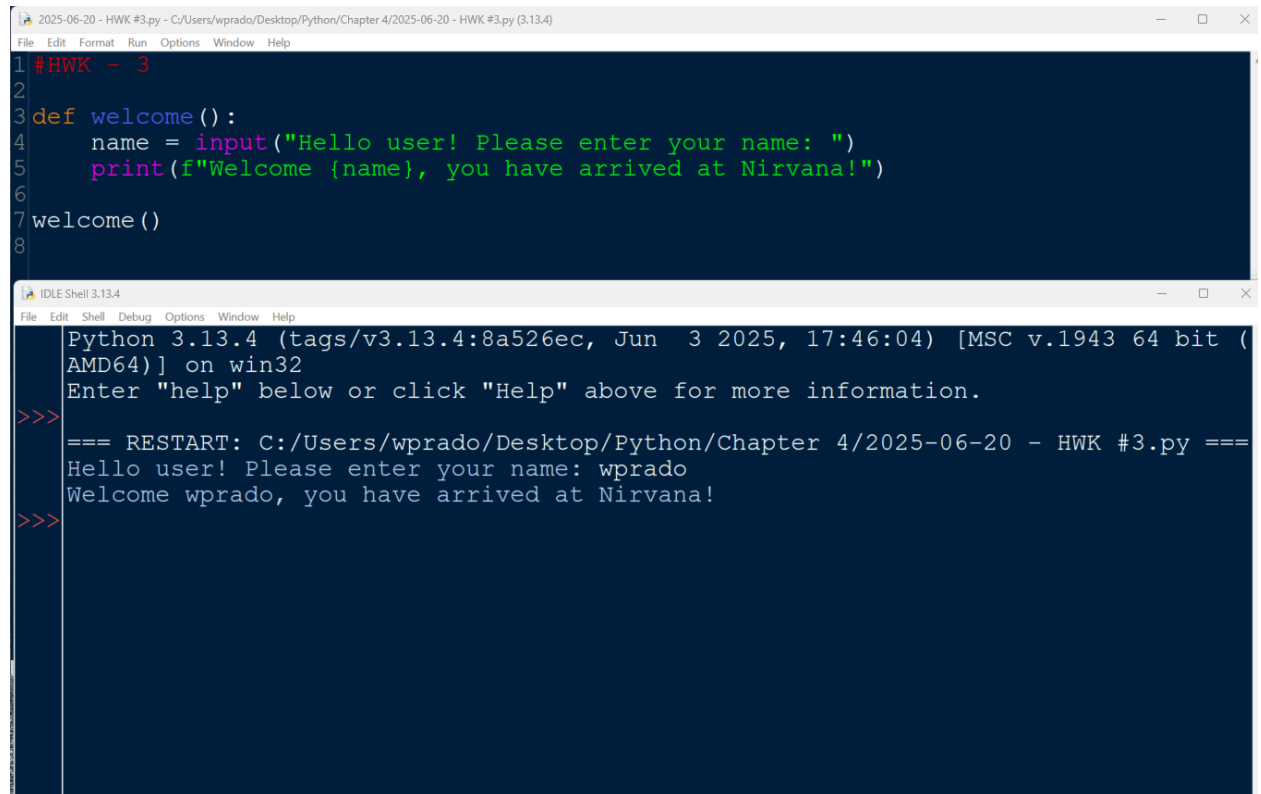
```
1 # Ch. 5 - HWK- 2
2
3 product = float(input("Enter the number of products you want to buy: "))
4
5 def times_ten(num):
6     total = num * 10
7     print("Total of product is", total)
8
9 times_ten(product)
10
```

The bottom window, titled 'IDLE Shell 3.13.4', shows the execution output:

```
Python 3.13.4 (tags/v3.13.4:8a526ec, Jun 3 2025, 17:46:04) [MSC v.1943 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.

>>>
= RESTART: C:/Users/wprado/Desktop/Python/Chapter 4/2025-06-20 - HWK #2 - function named times ten.py
Enter the number of products you want to buy: 10
Total of product is 100.0
>>>
```

#3. Write a function named **welcome()** that asks the user to enter his or her name and displays it followed by a welcome message.



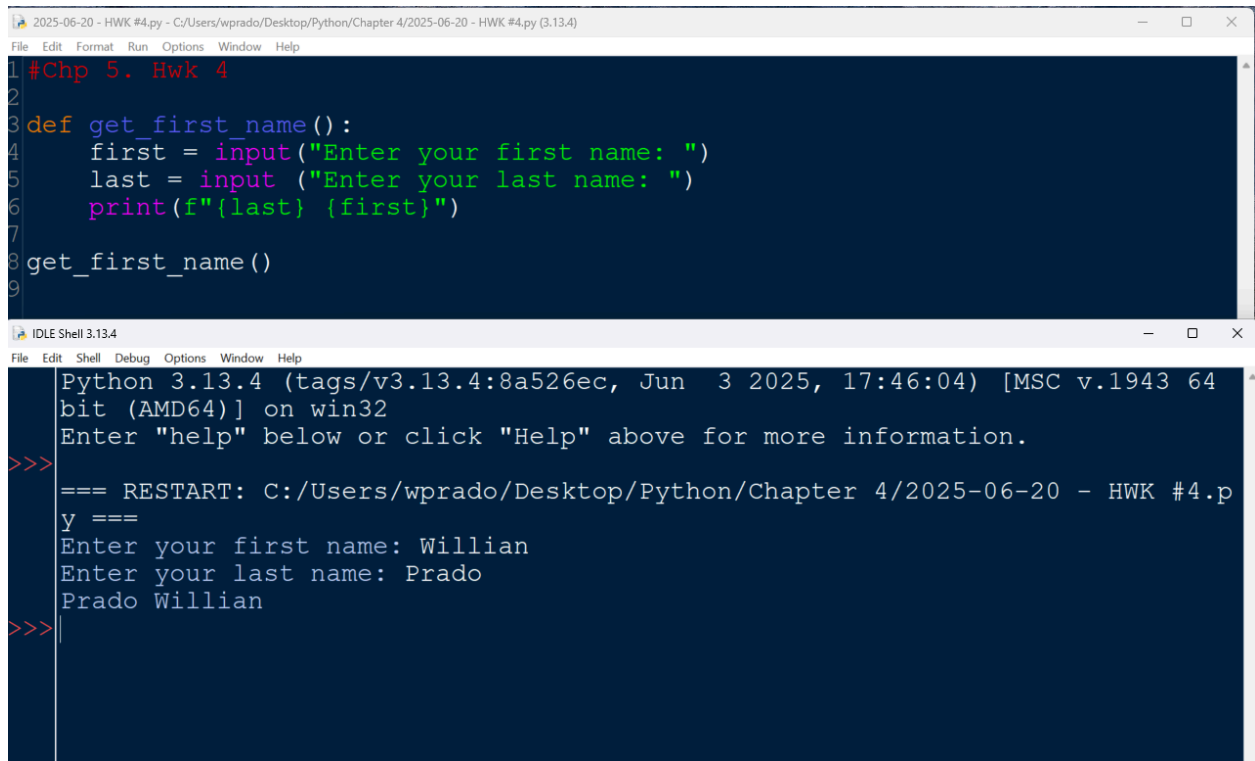
The image shows a screenshot of a Python IDE with two windows. The top window is a script editor titled '2025-06-20 - HWK #3.py' containing the following code:

```
1 #HWK - 3
2
3 def welcome():
4     name = input("Hello user! Please enter your name: ")
5     print(f"Welcome {name}, you have arrived at Nirvana!")
6
7 welcome()
8
```

The bottom window is the IDLE Shell, titled 'IDLE Shell 3.13.4', showing the execution output:

```
Python 3.13.4 (tags/v3.13.4:8a526ec, Jun 3 2025, 17:46:04) [MSC v.1943 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
=== RESTART: C:/Users/wprado/Desktop/Python/Chapter 4/2025-06-20 - HWK #3.py ===
Hello user! Please enter your name: wprado
Welcome wprado, you have arrived at Nirvana!
>>>
```

#4. Write a function named **get_first_name()** that asks the user to enter his or her first name and their last name on two separate lines. Then print last name first, and first name last.



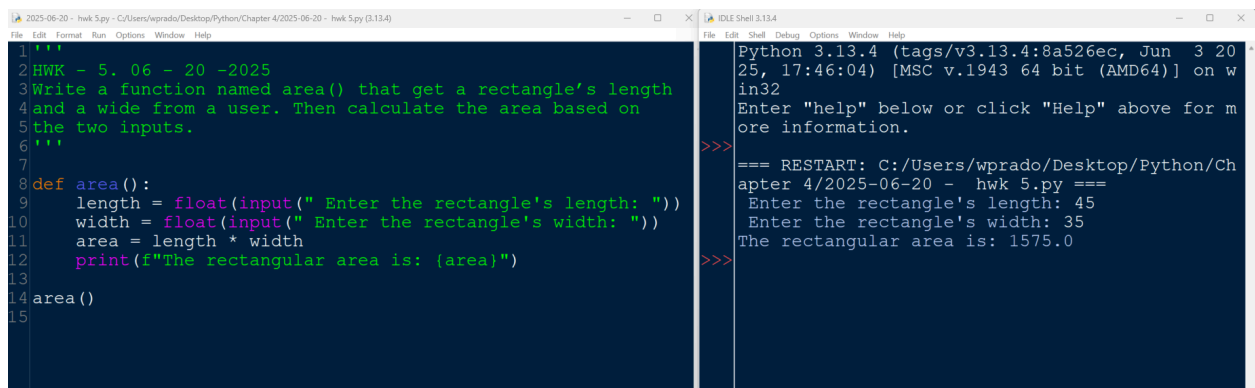
The screenshot shows a Python IDE with two windows. The top window, titled '2025-06-20 - HWK #4.py', contains the following code:

```
1 #Chp 5. Hwk 4
2
3 def get_first_name():
4     first = input("Enter your first name: ")
5     last = input("Enter your last name: ")
6     print(f"{last} {first}")
7
8 get_first_name()
9
```

The bottom window, titled 'IDLE Shell 3.13.4', shows the execution of the program. It displays the Python version and system information, followed by a restart message. The user is prompted to enter their first and last names, and the program outputs the last name followed by the first name.

```
Python 3.13.4 (tags/v3.13.4:8a526ec, Jun 3 2025, 17:46:04) [MSC v.1943 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
=== RESTART: C:/Users/wprado/Desktop/Python/Chapter 4/2025-06-20 - HWK #4.py ===
>>> Enter your first name: Willian
Enter your last name: Prado
Prado Willian
>>>
```

#5. Write a function named **area()** that get a rectangle's length and a wide from a user. Then calculate the area based on the two inputs.



The screenshot shows a Python IDE with two windows. The left window, titled '2025-06-20 - hwk 5.py', contains the following code:

```
1 """
2 HWK - 5. 06 - 20 -2025
3 Write a function named area() that get a rectangle's length
4 and a wide from a user. Then calculate the area based on
5 the two inputs.
6 """
7
8 def area():
9     length = float(input(" Enter the rectangle's length: "))
10    width = float(input(" Enter the rectangle's width: "))
11    area = length * width
12    print(f"The rectangular area is: {area}")
13
14 area()
15
```

The right window, titled 'IDLE Shell 3.13.4', shows the execution of the program. It displays the Python version and system information, followed by a restart message. The user is prompted to enter the rectangle's length and width, and the program outputs the calculated area.

```
Python 3.13.4 (tags/v3.13.4:8a526ec, Jun 3 2025, 17:46:04) [MSC v.1943 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
=== RESTART: C:/Users/wprado/Desktop/Python/Chapter 4/2025-06-20 - hwk 5.py ===
>>> Enter the rectangle's length: 45
Enter the rectangle's width: 35
The rectangular area is: 1575.0
>>>
```

#6. Write a program that asks the user to enter 5 test scores. the program should display a letter grade for each score and the average test score.

Write the following functions in the program:

calc_average – function should accept 5 scores and arguments an return the average of the scores
determine_grade – function should accept a test score as an argument and return a letter grade for the score based on the following grading scale:

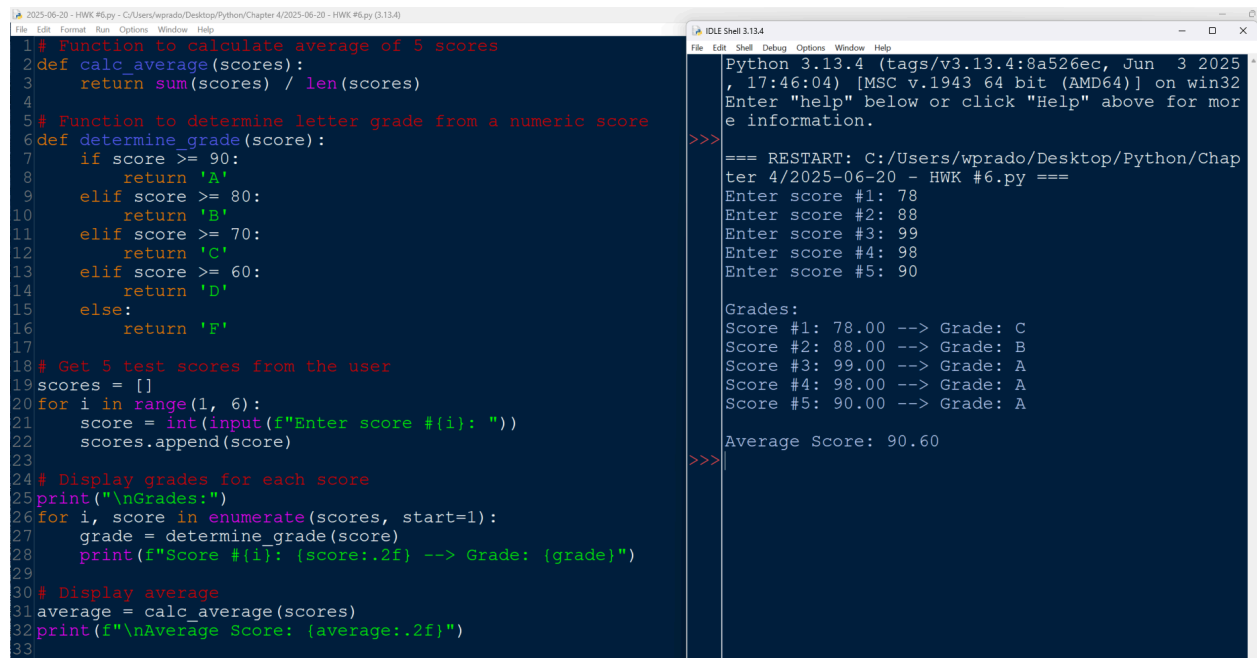
90-100A

80-89B

70-79C

60-69D

Below 60F



```
1 # Function to calculate average of 5 scores
2 def calc_average(scores):
3     return sum(scores) / len(scores)
4
5 # Function to determine letter grade from a numeric score
6 def determine_grade(score):
7     if score >= 90:
8         return 'A'
9     elif score >= 80:
10        return 'B'
11    elif score >= 70:
12        return 'C'
13    elif score >= 60:
14        return 'D'
15    else:
16        return 'F'
17
18 # Get 5 test scores from the user
19 scores = []
20 for i in range(1, 6):
21     score = int(input(f"Enter score #{i}: "))
22     scores.append(score)
23
24 # Display grades for each score
25 print("\nGrades:")
26 for i, score in enumerate(scores, start=1):
27     grade = determine_grade(score)
28     print(f"Score #{i}: {score:.2f} --> Grade: {grade}")
29
30 # Display average
31 average = calc_average(scores)
32 print(f"\nAverage Score: {average:.2f}")
33
```

```
Python 3.13.4 (tags/v3.13.4:8a526ec, Jun 3 2025, 17:46:04) [MSC v.1943 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
=== RESTART: C:/Users/wprado/Desktop/Python/Chapter 4/2025-06-20 - HWK #6.py ===
Enter score #1: 78
Enter score #2: 88
Enter score #3: 99
Enter score #4: 98
Enter score #5: 90

Grades:
Score #1: 78.00 --> Grade: C
Score #2: 88.00 --> Grade: B
Score #3: 99.00 --> Grade: A
Score #4: 98.00 --> Grade: A
Score #5: 90.00 --> Grade: A

Average Score: 90.60
>>>
```