

Python

Module 1 Lesson 9 and Lesson 10



Name: _____

Date: _____

Recap:

1. Local vs Global variables

- Highlight how the different variables interact within and outside the functions
- Highlight the 2 ways of defining a Global variable
- Manipulate the global variable within a function
- Understand how to apply local and global variables in their codes

Learning Outcomes:

2. return of a function

- Difference between Return and Print
- Exiting function upon return
- How to generate readable outputs through the use of print and outputs of a function

Explanation Points:

- Showcase how print and return works
- Explain why a function with no return will give None
- Explain the importance of return, especially for further calculations
- Explain why return will end the function
- Showcase how to implement return and print together to showcase answers

Breakdown of Lesson Plan:

Recap Lesson 8 Quiz <ul style="list-style-type: none">• No help provided. Student must write answer first before being allowed to try typing on computer• This quiz is quite time consuming. It covers all the basic information of the earlier lessons.• Student should score at least 75%	15 min (Test) 5 min (Explanation)
Lesson 9.1 (Return VS Print)	20 min
Lesson 10.1 (Application of Return)	25 min
Lesson 10 Quiz	25 min

**Note: There is a high chance of student not being able to complete on time.*

Python

Module 1 Lesson 9 and Lesson 10



Name: _____

Date: _____

Recap Lesson 8 Quiz

Question 1:

Jane has 4 younger brothers, Tom, Thomas, Kane and Abel. For Christmas, she wants to give each brother some sweets. Each sweet costs \$1. Write a function to calculate how much she needs to spend on each of her 4 brothers. Your answer should be in this format:

“Jane needs to spend \$___ on ___ to buy ___ sweets.”

Create a global variable called *total*. Using this variable, calculate how much Jane has to spend for her Christmas presents. Print your answer in the following format:

“Jane needs to spend \$___ in total.”

Tom: 15 sweets

Thomas: 21 sweets

Kane: 5 sweets

Abel: 27 sweets

Question 2:

Mr Tan owns a manufacturing company. There are a total of 3 types of materials that Mr Tan’s company works with – wood, metal and plastic. The cost of 1 unit of each type of material is \$5, \$11 and \$3 respectively. Create a global variable for **each** material and assign their respective costs.

Currently the company is working on 4 different projects. Each project requires different amounts of each type of material. Create a function that will calculate how much Mr Tan needs to spend on materials for each project. Print your answer in the format:

“Project ___ needs \$___ in total.”

Create another global variable called *total*. Using *total* in your functions to calculate how much Mr Tan needs to spend on materials in total. Print your answer in the following format:

“Mr Tan needs to spend \$___ for all the projects.”

Project	Wood (unit)	Metal (unit)	Plastic (unit)
A	15	21	23
B	0	7	5
C	21	3	24
D	20	0	7

Python

Module 1 Lesson 9 and Lesson 10



Name: _____

Date: _____

Lesson 9.1

1. What is a return function

The statement `return [expression]` exits a function, optionally passing back an expression to the caller. A return statement with no arguments is the same as `return None`.

2a. Python return function syntax:

```
def functionname ():  
    statement(s)  
    return()
```

```
print(functionname ())
```

2b. Difference between print function and return function

The print function writes ie. “prints” a string in the console.

The return function causes the function to exit and hand back a value to the caller.

Recap: Calling a function with “print within a function”

1	<code>def name(x):</code>
2	<code> y=x*5</code>
3	<code> print(y)</code>
4	
5	<code>name(3)</code>

1	15
---	----

Example: Calling a function with a “return within a function”

1	<code>def noname(x):</code>
2	<code> y=x*5</code>
3	<code> return(y)</code>
4	
5	<code>noname(3)</code>

1	
---	--

Example: Difference between “print within a function” and “return within a function”

1	<code>def noreturn(x):</code>
2	<code> y=x*5</code>
3	<code> print(y)</code>
4	
5	<code>print(noreturn(3))</code>

1	15
2	None

1	<code>def withreturn(x):</code>
2	<code> y=x*5</code>
3	<code> return(y)</code>
4	
5	<code>print(withreturn(3))</code>

1	15
---	----

Lesson 9.1

Module 1 Lesson 9 and Lesson 10

Date:



Type or fill in the following to get to/fill in the output after pressing F5

In task 1 and 2, recap on functions with the print function and calling functions with print and calling functions without print

Output

1	
2	

Output

1	
---	--

--

Python

Module 1 Lesson 9 and Lesson 10



Name: _____

Date: _____

Lesson 9.1

In task 3 and 4, we introduce functions with return function and calling functions with print and calling functions without print

Task 3:

Output

1	def stored_numbers():
2	return(5)
3	
4	print(stored_numbers())

1	
---	--

Task 4:

Output

1	def stored_numbers():
2	return(5)
3	
4	stored_numbers()

1	
---	--

Task 3, 4 : Explain the difference and write your Key take-aways.

--

Task 5:

In this task, why is output only 1 when we call the function?

When a *return statement is reached*, it will exit the function and any code within the function below the *return statement* will not be executed.

Output

1	def stored_numbers():
2	return (1)
3	return (2)
4	return (3)
5	
6	print(stored_numbers())

1	
---	--

Python

Module 1 Lesson 9 and Lesson 10



Name:

Date:

Lesson 9.1

In task 6 and 7, we introduce functions with print followed by return function and calling functions without print and calling functions with print.

Task 6:

Output

1	def stored_numbers():
2	print(5)
3	return(4)
4	return(3)
5	
6	stored_numbers()

1	
2	

Task 7:

Output

1	def stored_numbers():
2	print(5)
3	return(4)
4	return(3)
5	
6	print(stored_numbers())

1	
---	--

Task 6, 7 : Explain the difference and write your Key take-aways.

--

Module 1 Lesson 9 and Lesson 10

Date:



In task 8 and 9, we introduce functions with return followed by print function and calling functions without print and calling functions with print.

Output

1	
2	

Output

1	
---	--

--

Output

1	
2	
3	

Python

Module 1 Lesson 9 and Lesson 10

Name: _____

Date: _____



Lesson 9.1

Task 11:

Output

1	def square(x):
2	y=x*x
3	return (y)
4	
5	print(square(10))

1	
---	--

Task 12:

Output

1	def sum(y,z):
2	x = y + z
3	return(x)
4	
5	y=2
6	z=3
7	x=sum(y,z)
8	print(x)

1	
---	--

Task 13: (Recap default arguments)

Output

1	def minus1(y,z):
2	x = y - z
3	return(x)
4	
5	y=9
6	z=7
7	print(minus1(z,y))

1	
---	--

Python

Module 1 Lesson 9 and Lesson 10



Name: _____

Date: _____

Lesson 9.1

Task 14:

There is an **error** in this code. Find the error and correct it!

1	def multiply(y,z):
2	x = y * z
3	return(x)
4	
5	y=6
6	z=7
7	print(multiply(x))

Output

1	
---	--

Task 15:

1	def ans(y,z):
2	x = y - z
3	print(x)
4	
5	y=9
6	z=7
7	x=ans(y,z)
8	print(x)
9	print(ans(10,1))

Output

1	
2	
3	
4	

Task 16:

1	def ans(y,z):
2	x = y - z
3	return(x)
4	
5	y=9
6	z=7
7	x=ans(y,z)
8	print(x)
9	print(ans(10,1))

Output

1	
2	
3	
4	

Question: What is the difference between the outputs of Task 15 and Task 16?

--

Python

Module 1 Lesson 9 and Lesson 10

Name:

Date:



Lesson 10.1

1. Why do we need to use the return expression?

Return changes the value of the function, and outputs the required data type value as needed. This is useful when we need to manipulate the answers further.

2. Manipulation of functions

After using return, we can utilize further operators on the function.

Output

1	def square(x):
2	y=x*x
3	return (y)
4	
5	print(" The result is" + " " + str(square(20)))

1	The result is 400
---	-------------------

Type or fill in the following to get to/fill in the output after pressing F5

**The numbers represent the lines.*

Task 1:

Create a function named **volume** and define it as the multiplication of 3 same numbers, b. Print "The volume is 1000" (HINT: $10 * 10 * 10 = 1000$)

Output

1	def volume(b):
2	
3	
4	
5	print("The volume is", volume(10))
6	

1	The volume is 1000
---	--------------------

Python

Module 1 Lesson 9 and Lesson 10



Name: _____

Date: _____

Lesson 10.1

Task 2:

Create a function named `perimeter` and define it as the sum of 3 same numbers, h. Print "The perimeter is 75 cm" (HINT: $25+25+25 = 75$)

Output

1	
2	
3	
4	
5	
6	

1	The perimeter is 75cm
---	-----------------------

Task 3:

Output

1	<code>def val(x,y):</code>
2	<code> return(x*y)</code>
3	
4	<code>Val1 = val(3,5)</code>
5	<code>Val2 = val(14,2)*2</code>
6	<code>Val3 = val(28,6)*0.5</code>
7	
8	<code>print('The answers are', Val1, Val2, 'and', Val3)</code>

1	The answers are
---	-----------------

Task 4:

Output

1	<code>def funct(a,b,c):</code>
2	<code> return(a*b*c)</code>
3	
4	<code>Val1 = funct(1,2,3)</code>
5	<code>Val2 = funct(4,5,6)*2</code>
6	<code>Val3 = funct(7,8,9)*0.5</code>
7	<code>Val_Final = funct(Val1,Val2,Val3)</code>
8	<code>print('The final value is',Val_Final)</code>

1	
---	--

Python

Module 1 Lesson 9 and Lesson 10



Name: _____

Date: _____

Lesson 10.1

Task 5a:

Create a function to determine the remainder when the following numbers are divided by 7. Print your answer in the format: "The remainder is ____"

- a. Number = 20
- b. Number = 10
- c. Number = 30

Output

1	def funct(number):
2	remainder=(number%7)
3	print("The remainder is", remainder)
4	
5	funct(20)
6	funct(10)
7	funct(30)

1	The remainder is
2	The remainder is
3	The remainder is

Continuing with task 5, return the sum of the remainders of the 3 numbers. Print the sum of the remainders in the format: "The sum of the remainders is ____"

Output

1	def funct(number):
2	remainder = (number%7)
4	return remainder
5	sumofremainders= funct(20) + funct(10) + funct(30)
6	
7	print("The sum of the remainders is", sumofremainders)

1	The sum of the remainders is
---	------------------------------

Python

Module 1 Lesson 9 and Lesson 10



Name: _____

Date: _____

Lesson 10.1

Task 6a: In this task, there are 3 groups with different number of students. There are 10 books in total to be given to each group equally among the number of students in the group. Write a function to show how many books each student in each group would get. Your answer should be in the following format:

Complete line 3,6 and 7

"Each student in Group ____ will receive ____ books."

- a. Group1 = 2
- b. Group2 = 3
- c. Group3 = 4

Output

1	def funct(groupname,numofstudents):
2	Eachstudent=10//numofstudents
3	print(
4	
5	funct("Group 1",20)
6	
7	

1	
2	
3	

Task 6b. In the function, return the total number of books remaining after distributing equally to each student in each group. Print the total books remaining in the following format:

"There are ____ books in total remaining."

Output

1	def funct(groupname,numofstudents):
2	Booksremaining=10%numofstudents
3	return Booksremaining
4	totalbooksremaining= funct("Group1",2) +
	funct("Group2",3) + funct("Group3",4)
5	
6	print("There are " , totalbooksremaining, "books
	in total remaining")

1	
---	--

Python

Module 1 Lesson 9 and Lesson 10



Name: _____

Date: _____

Lesson 10.1

In task 7a, Create a function named area to determine the area of each square with different length. Print your answer in the format: "The area of Square__ is ____"

- a. Square A= 5cm
- b. Square B= 6cm
- c. Square C= 7cm

Complete the missing parts in lines 1, 2, 3 , 5,6)

Output

1	def area(
2	formulaofarea=
3	print
4	
5	funcn("Square A, 5)
6	funcn
7	funcn

1	The area of Square A is
2	The area of Square B is
3	The area of Square C is

In task 7b, Continuing with task 9a, return the sum of the areas of the 3 squares. Print the sum of the areas of the 3 squares in the format: "The sum of area of the 3 squares is____"

Output

1	def area(
2	formulaofarea=
3	return
4	sumofarea =
5	
6	print("The sum of area of the 3 squares is" ,

1	
---	--

Module 1 Lesson 9 and Lesson 10

Date:



In task 8 and 9, key in the numbers to get the output and explain the similarities between task 6 and task 7 and why the outputs are obtained.

Output

1	
---	--

Output

<i>1</i>	
----------	--

--

Python

Module 1 Lesson 9 and Lesson 10



Name: _____

Date: _____

Lesson 10.1

Task 10: (Refer to Task 5, 6 and 7)

Step 1:

Mr Tan wants to reward his workers with some gift cards. His company has 3 departments, as shown below. He has put aside 150 gift cards worth \$10 each for each department. Write a function to show how many gift cards each person in each department should get. Your answer should be in the following format:

“Each personnel in the ____ department will receive ____ gift cards. There will be ____ gift cards remaining.”

Manufacturing Department: 21 personnel

Sales Department: 27 personnel

Technology Department: 19 personnel

Step 2

In the function, return the number of gift cards remaining. Print the value of the remaining gift cards in following format:

“There was \$____ worth of gift cards remaining.”

Python

Module 1 Lesson 9 and Lesson 10



Name: _____

Date: _____

End of Lesson 10 Quiz

Question 1

1	def stored_numbers():
2	print(5)
3	
4	X = (stored_numbers())

What will be the value of X?

- A. 5
- B. 0
- C. None
- D. 10

Question 2

1	def square (y):
2	return(y*y)
3	
4	Z=square(5)*10

What is the value of Z?

- A. 25
- B. 10
- C. 250
- D. 30

Question 3

How many return function can we input per function?

- A. 3
- B. 0
- C. 2
- D. 1

Python

Module 1 Lesson 9 and Lesson 10



Name: _____

Date: _____

End of Lesson 10 Quiz

For the next few questions, decide if the two cases have the same or different outputs.

Question 4:

Case A:

1	def sum(x,y):
2	print(x+y)
3	print(sum(5,4))

Case B:

1	def sum(x,y):
2	return(x+y)
3	print(sum(5,4))

Question 5:

Case A:

1	def words():
2	return('Coding'*2)
3	print(words())

Case B:

1	def words():
2	return('Coding')
3	print(words()*2)

Question 6:

Case A:

1	First=10
2	Second = 20
3	def diff():
4	return(Second-First)
5	
6	print(diff())

Case B:

1	First=10
2	Second = 20
3	def diff(Second,First):
4	return(Second-First)
5	
6	print(diff(First,Second))

Question 7:

Case A:

1	Num=20
2	def quotient(val):
3	return(val//20)
4	
5	print(quotient)

Case B:

1	Num=20
2	def quotient(val):
3	return(val//20)
4	
5	print(quotient(21))

Python

Module 1 Lesson 9 and Lesson 10



Name: _____

Date: _____

End of Lesson 10 Quiz

Question 8:

Case A:

1	Num=20
2	def sum1(x,y):
3	a = num + y + x
4	print (a)
5	
6	sum1(3,2)

Case B:

1	def sum1(x,y):
2	a = num + y + x
3	print (a)
4	Num=20
5	
6	sum1(3,2)

Question 9:

Case A:

1	y=2
2	z=3
3	def sum1():
4	x = y + z
5	return (x)
6	
7	print(sum1())

Case B:

1	y=2
2	z=3
3	def sum1(y,z):
4	x = y + z
5	return (x)
6	
7	print(sum1())

Question 10:

Case A:

1	def sum1 ():
2	y=2
3	z=3
4	x = y + z
5	return (x)
6	
7	print(sum1())

Case B:

1	def sum1(y,z):
2	y=2
3	z=3
4	x = y + z
5	return (x)
6	
7	print(sum1(5,4))

Python

Module 1 Lesson 9 and Lesson 10



Name: _____

Date: _____

End of Lesson 10 Quiz

Question 1

Mrs Tan wants to buy 'tingkat' (food catering) services for her family for 4 weeks. She has written down the number of days per week that she needs 'tingkat' services. For each day of 'tingkat' services, Mrs Tan has to pay \$10. Write a function to calculate how much Mrs Tan needs to spend per week. Your answer should be in the format:

'In week ____ Mrs Tan will need to pay \$____.'

Week A: 7 days

Week B: 2 days

Week C: 5 days

Week D: 3 days

In your function, please return the amount of money Mrs Tan needs to spend per week.

Mrs Tan has set aside \$100 to pay for the "tingkat" services for these 4 weeks. Without using global variables, calculate how much more money Mrs Tan needs in order to pay for all 4 weeks. Print your answer in the following format:

"Mrs Tan needs \$____ more to pay for the service."

Question 2

Define a function `employee` with keyword arguments – name, amount and date. Use the function to print the employees shown in the table below. Your answer should be in the format:

"(name) has been credited (amount) on (date)."

In your function, return the amount of money provided to each employee.

Without using a global variable, calculate how much money the company needs to have to credit all the employees. Print your answer in the following format:

"The company needs to have \$____."

Name	Amount	Date
James	5000	6/6/2020
Alice	6000	5/6/2020
John	3000	3/6/2020
Jacob	2000	8/6/2020
Kenny	4000	1/6/2020
Amy	4000	30/5/2020