

# Python

## Module 1 Lesson 5



Name: \_\_\_\_\_

Date: \_\_\_\_\_

### Recap:

1. String Slicing
  - a. Variations of slice syntax
  - b. Start:Stop:Step
  - c. Positive and Negative indexing
  - d. Inverting string

### Learning Outcomes:

1. Functions: Explain in broad concept what is a function. Link to how it is used in our daily life.
  - We start with a program – A program is the “biggest thing” you are running. It is the umbrella.  
Eg1. A program is a like a day in your life. As a student, packing your bag is what you do every day.
  - Function – Functions are inside the program. It is a repeated action. Function allow you to cut down the repetition. Eg. Packing your bag. You need to pack your bag every day. However, you pack different books for different days
  - Arguments – Part of the function. What books do you need to pack for that day? You need to tell the function the books you need for the day to proceed with the action of packing your bag
    - a. Defining a function
  - You can define a function and not call it.
  - Define functions at the start of the code before calling it
  - Functions will be executed only when you call it. This is when the program starts
    - b. Calling functions
    - c. Arguments
    - d. Output of a function

### Explanation Points:

- Highlight the importance of Functions
  - Different way of handling functions as compared to variables
- Understanding of arguments
  - Importance of naming arguments correctly
  - Highlight difference between arguments and variables

### Breakdown of Lesson Plan:

Recap Lesson 4 Quiz <ul style="list-style-type: none"><li>• No help provided. Student must write answer first before being allowed to try typing on computer</li><li>• Student should score at least 75%</li></ul>	20 min (Test) 5 min (Explanation)
Lesson 5.1 (What is a Function)	20 min
Lesson 5.2 (Arguments)	20 min
Lesson 5 Quiz	25 min

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

# Python

Module 1 Lesson 5

Name:

Date:



---

## Recap Lesson 4 Quiz

### Question 1

Use python string slicing to print "love" from the variable x shown below.

`x='I love coding'`

### Question 2

`y='123453673893013923'`

Use python string slicing and `int()` function to print the integer 333333.

### Question 3

Use python string slicing to print "codingcodingcoding" from the variable z shown below.

`z = 'I love coding'`

### Question 4

Use python string slicing to print "cigoL" from the variable y shown below.

`y='The Logic Coders'`

### Question 5

Jane has 4 oranges. Fred has 2 times the number of oranges as Jane. Tom has 2 less oranges than Fred. Using python, create 3 variables and using arithmetic operators, print out how many oranges each child has, as well as how many oranges they have in total. Use the following format:

"Jane has \_\_\_\_ oranges."

"Fred has \_\_\_\_ oranges."

"Tom has \_\_\_\_ oranges."

"They have \_\_\_\_ oranges in total."

# Python

Module 1 Lesson 5



Name: \_\_\_\_\_

Date: \_\_\_\_\_

## Recap Lesson 4 Quiz

### Question 6

A large rectangle has a length of 25 cm, and a breadth of 10 cm. John draws  $x$  smaller rectangles of size 5 cm by 2 cm. Create 1 variable and using arithmetic operators, how many smaller rectangles is John able to draw in the large rectangle? Print your answer using the following format:

"John draws \_\_\_\_ smaller rectangles."

Create a variable called *large\_rectangle\_area*, assign 25 multiply by 10 to it. Create a variable called *small\_rectangle\_area*, assign 5 multiply by 2 to it. Create a variable called *quotient*, assign *large\_rectangle\_area* divided by *small\_rectangle\_area* to it. Print your answer in the following format:

"John draws \_\_\_\_ smaller rectangles."

(Challenge: can you do this by only creating one variable?)

### Question 7

Kaleb is preparing gifts for his students. He can buy sweets in bags of 20. Kaleb has a total of 73 students. He plans to give each student 3 sweets. Create variables and using arithmetic operators, How many bags of sweets does Kaleb need to buy? How many sweets will be remaining? Print your answer using the following format:

"Kaleb needs to buy \_\_\_\_ bags of sweets. He has a total of \_\_\_\_ sweets remaining."

### Question 8

John has \$20 to spend. He manages to buy a notebook, a pen and 3 sweets, and has \$2 remaining. The notebook costs \$8, while each sweet cost \$2. How much does the pen cost? Create 1 variable and using arithmetic operators, Print your answer using the following format:

"The pen costs \$\_\_\_\_"

### Question 9

A bookshop sells books for \$8 each, pens for \$3 each and pencils for \$2 each. Mary wishes to buy 3 books, 4 pens and 2 pencils. How much money must she prepare for her purchase? Create 1 variable and using arithmetic operators, Print your answer using the following format:

"Mary needs at least \$\_\_\_\_ to purchase her stationery."

# Python

## Module 1 Lesson 5



Name: \_\_\_\_\_

Date: \_\_\_\_\_

### Lesson 5.1

#### 1. What is a function?

- A function is a block of organized, reusable code that is used to perform a single, related action.
- You can pass data, known as parameters, into a function.
- A function can return data as a result.

#### 2a. Rules for creating and defining a function

- Function blocks begin with the keyword `def` followed by the function name and parentheses ( ). The keyword `def` marks the start of function header.
- A function name is used to uniquely identify it.
- The code block within every function starts with a colon (:) and is indented.
- One or more valid python statements that make up the function body. Statements must have same indentation level as the print function

#### 3. Calling a function

- Functions only run when it is called.
- You can define a function but not call it

#### 4a. Rules for calling a function

- To call a function, use the function name followed by parenthesis

#### 4b. Python calling function syntax:

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

`functionname()`

Calling side - Argument

Function side – Parameter

Definition                  Parameter

1	<code>def name(firstandlast):</code>
2	<code>    print(firstandlast)</code>
3	
4	<code>name("Jack Lim")</code>
5	<code>name("Joe Tan")</code>

1	Jack Lim
2	Joe Tan

Python Reading Sequence – Line 1,4,1,2,5,1,2

# Python

Module 1 Lesson 5

Name: \_\_\_\_\_

Date: \_\_\_\_\_



## Lesson 5.1

**Example 1:** You can define a function but not call it. The program has not started. Python will not execute the function.

<b>1</b>	<b>def name(firstandlast):</b>
<b>2</b>	<b>    print(firstandlast)</b>

<b>1</b>	
<b>2</b>	

**Example 2:** You can define a function and call it. The program starts when you call it. It starts from line 4.

<b>1</b>	<b>def name(firstandlast):</b>
<b>2</b>	<b>    print(firstandlast)</b>
<b>3</b>	
<b>4</b>	<b>name("Jack Lim")</b>
<b>5</b>	<b>name("Joe Tan")</b>

<b>1</b>	<b>Jack Lim</b>
<b>2</b>	<b>Joe Tan</b>

**Example 3:** You have to define a function first before calling it

<b>1</b>	<b>name("Jane")</b>
<b>2</b>	
<b>3</b>	<b>def name(firstandlast):</b>
<b>4</b>	<b>    print(firstandlast)</b>
<b>5</b>	

<b>1</b>	<b>"name is not defined"</b>
----------	------------------------------

# Python

## Module 1 Lesson 5



Name: \_\_\_\_\_

Date: \_\_\_\_\_

### Lesson 5.1

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

*\*The numbers represent the lines.*

Task 1:

Output

1	def position(placement):
2	print(placement)
3	
4	position("1st")
5	position("2nd")

1	
2	

Task 2:

Output

1	def standing(number):
2	print (number)
3	
4	
5	

1	1
2	2

Task 3:

Create a function named **salary** and print 2 different salaries

Output

1	
2	
3	
4	salary(1000)
5	salary(5000)

1	\$1000
2	\$5000

Task 4:

Create a function named **children** and print 3 different names

Output

1	
2	
3	
4	
5	
6	

1	Jake
2	Amy
3	Charles

### Lesson 5.1

# Python

Module 1 Lesson 5



Name: \_\_\_\_\_

Date: \_\_\_\_\_

Task 5:

Create a function named `salary` and print "Salary earned = 1000"

Output

1	
2	
3	
4	<code>salary(1000)</code>

1	Salary earned = 1000
---	----------------------

Task 6:

Create a function named `temperature` and print "Temp is = 30 degrees"

Output

1	
2	
3	
4	
5	

1	Temp is = 30 degrees
---	----------------------

Task 7:

Create a function named `mask` and print "Mask supply less than 100 is low"

Output

1	
2	
3	
4	<code>mask(100)</code>

1	Mask supply less than 100 is low
---	----------------------------------

# Python

Module 1 Lesson 5

Name: \_\_\_\_\_

Date: \_\_\_\_\_



## Lesson 5.2

### 1. What are Arguments

The argument is a value that is passed to the function when it's called.

Arguments are specified after the function name, inside the parentheses. You can add as many arguments as you want, just separate them with a comma.

### 2. What are Parameters

A parameter is the variable listed inside the parentheses in the function definition.

Calling side - Argument

Function side – Parameter

	Definition	Parameter
1	def name(firstandlast):	
2	print(firstandlast)	
3		
4	name("Jack Lim")	
5	name("Joe Tan")	

1	Jack Lim
2	Joe Tan

Types of Arguments

Argument

- Required arguments
- Keywords arguments
- Default arguments

**A. Required arguments** - By default, a function must be called with the correct number of arguments. Meaning that if your function expects 2 arguments, you have to call the function with 2 arguments, not more, and not less.

Example of required arguments

1	def sum(x,y):
2	print(x + y)
3	
4	sum(2,8)

Output

1	10
---	----

1	def fullname(first,last):
2	print(first + " " + last)
3	
4	fullname("John","Tan")

Output

1	John Tan
---	----------



# Python

Module 1 Lesson 5

Name: \_\_\_\_\_

Date: \_\_\_\_\_



## Lesson 5.2

**B. Keywords arguments** - Keyword arguments are related to the function calls. When you use keyword arguments in a function call, the caller identifies the arguments by the parameter name.

The way the order of the arguments does not matter

**Example of keywords arguments**

Output

1	def order(child3, child1, child2):
2	print("The youngest is" + " " + child3)
3	
4	order(child1="Ace", child2="Bill", child3="Carl")

1	The youngest is Carl
---	----------------------

**C. Default arguments** - Default argument is an argument that assumes a default value if a value is not provided in the function call for that argument.

**Rules**

- To declare a default value of an argument, assign it a value at function definition.
- You need to declare a default value from right to left

**Example of default arguments**

Output

1	def multiply(x, y=0):
2	print(x*y)
3	
4	multiply(2)

1	0
---	---

Output

1	def multiply(x, y=0):
2	print(x*y)
3	
4	multiply(2,y=5)

1	10
---	----

Output

1	def multiply(x, y=2,z):
2	print(x*y*z)
3	
4	multiply(2,y=5,10)

1	No default argument follows default argument
---	--

# Python

Module 1 Lesson 5



Name: \_\_\_\_\_

Date: \_\_\_\_\_

## Lesson 5.2

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

*\*The numbers represent the lines.*

Task 1:

1	def sum1(a,b):
2	print(a + b)
3	
4	sum1(2,8,4)

Output

1	
---	--

Task 2:

1	def sum2(c,d):
2	print(c * d)
3	
4	sum2(12,4)

Output

1	
---	--

Task 3:

Create a function name **square** and define it as multiplication of the number. Print the result in the format of "The answer is 100 "

1	def square(d):
2	print(
3	
4	square(10)

Output

1	The answer is 100
---	-------------------

Task 4:

Create a function name **total** and define it as sum of 3 numbers. Print the result in the format of "The total is 20 "

1	
2	
3	
4	total(2,10,8)

Output

1	The total is 20
---	-----------------

# Python

Module 1 Lesson 5

Name: \_\_\_\_\_

Date: \_\_\_\_\_



## Lesson 5.2

Task 5:

Output

1	def position(first, second, third):
2	print("The first place is " + " " + first)
3	
4	position(third= "Carl", second= "Bill", first= "Ace")

1	
---	--

Task 6:

Output

1	def icinfo(name,age):
2	print(name)
3	print(age)
4	
5	icinfo(age=50, name="Ash")

1	
2	

Task 7:

Output

1	def minus(a, b=0):
2	print(a - b)
3	
4	minus(10, b=2)

1	
---	--

Task 8:

Output

1	def add(e, d=0):
2	print(e + d)
3	
4	add(d=3)

1	
---	--

# Python

Module 1 Lesson 5

Name: \_\_\_\_\_

Date: \_\_\_\_\_



## Lesson 5.2

Task 9:

Output

1	def numbers(a, b=5, c=10):
2	print("a is " + str(a), "and b is " + str(b) + " and c is " + str(c))
3	
4	numbers(3,7)

1	
---	--

Task 10:

Output

1	def message(msg,num=2):
2	print(msg* num)
3	
4	message("welcome")

1	
---	--

Task 11:

Output

1	def message(msg,num):
2	print(msg* num)
3	
4	message("hello",3)

1	
---	--

Task 12:

Create a function name **greeting**. Print the following messages

Output

1	
2	
3	
4	
5	

1	Morning Morning
2	How do you do

# Python

Module 1 Lesson 5

Name: \_\_\_\_\_

Date: \_\_\_\_\_



## Lesson 5.2

Task 13: Create a function to find the perimeter of a rectangle with breadth= 5 and length=10

\*What are your parameters? Length and Breadth

\*What is the formula to calculate perimeter?

Output

1	def perimeter(length, breadth):
2	Formula=length + breadth + length + breadth
3	print (
4	
5	perimeter (5,10)

1	The perimeter of the rectangle is 30
---	--------------------------------------

Task 14: You have 2 rectangles, A and B with different length and breadth

Create a function to find the perimeter of

Rectangle A: length= 15 and breadth=10

Rectangle B: length= 2 and breadth=1

\*What are your parameters? Length and Breadth and Name of rectangle

\*What is the formula to calculate perimeter?

Output

1	def perimeter(name,length, breadth):
2	Formula =
3	print (
4	
5	perimeter (      )
6	perimeter(      )

1	Rectangle A has perimeter of 50
2	Rectangle B has perimeter of 6

# Python

Module 1 Lesson 5

Name: \_\_\_\_\_

Date: \_\_\_\_\_



## Lesson 5.2

Task 15: Recap default arguments. You have 3 easy summation problems to solve

Create a function to find the answers of the 3 problems and print "The answer of problem \_\_\_\_"

Problem 1: Sum of 10 and 20

Problem 2: Sum of 10,20,30

Problem 3: Sum of 20 and 50

\*What are your parameters?

\*What is the formula?

Output

1	def sum1(problemnumber, a, b, c=0):
2	Formula =
3	print (
4	
5	sum1 (
6	sum1(
7	sum1(

1	The answer of problem 1 is 30
2	The answer of problem 2 is 60
3	The answer of problem 3 is 70

# Python

Module 1 Lesson 5

Name: \_\_\_\_\_

Date: \_\_\_\_\_



## Lesson 5 Quiz

### Question 1

Write a function to calculate the area of a rectangle, given its length and breadth. Using the function, calculate the area of the following rectangles. Your answer should use the following format:

“Rectangle \_\_\_\_ has an area of \_\_\_\_.”

Rectangle A:

Length = 5 cm, Breadth = 3 cm

Rectangle B:

Length = 10 cm, Breadth = 8 cm

Rectangle C:

Length = 5.4 cm, Breadth = 3.8 cm

*HINT: area = length \* breadth*

### Question 2

Write a function to add five values together. Using the function, calculate the sum of the following cases. Your answer should use the following format:

“Case \_\_\_\_ has the sum of \_\_\_\_.”

Case A: 3, 5, 10, 11, 17

Case B: 5278, 8734, 2196, 278, 3045

Case C: 5.24, 67, 9.83, 10.1, 12

Case D: 56, 29, 38

Case E: 98, 70, 29

*HINT: use default argument*

# Python

## Module 1 Lesson 5

Name: \_\_\_\_\_

Date: \_\_\_\_\_



### Lesson 5: EXTRA QUESTIONS AS NEEDED

Question 3:

Mrs Mandy wishes to split her sweets among her students in her classes. She has prepared 219 sweets for each of her 5 classes (each class gets 219 sweets) and wishes to use up as many as possible per class. Write up a function to help inform her how many sweets are to be given to each student, and how many sweets she will have remaining in each class. Your answer should be in the format:

“Mrs Mandy will need to give \_\_\_ sweets to each student and will have \_\_\_ sweets remaining in Class \_\_\_.”

Class A: 20 students

Class B: 31 students

Class C: 29 students

Class D: 35 students

Class E: 34 students

*HINT: To calculate remainder use ‘ % ’, To calculate quotient use ‘ // ’*

Question 4:

In each class, Student B will have twice as much pocket money as Student A. Student C will have \$4 less than Student B. Write a function that prints the amount of money Student C has, as well as how much money the 3 students have. Your answer should be in the format:

“In Class \_\_\_, Student C has \$\_\_\_ and the 3 students have \$\_\_\_ in total.”

Class A: Student A has \$2

Class B: Student A has \$5.50

Class C: Student A has \$7

Class D: Student A has \$4.50

*HINT: For every Student, create a variable and calculate the amount of money each student have i.e.  $A = x$ ,  $B = A * 2$ ,  $C = B - 4$*



# Python

## Module 1 Lesson 5

Name: \_\_\_\_\_

Date: \_\_\_\_\_



### Lesson 5: EXTRA QUESTIONS AS NEEDED

Question 5:

Thomas has  $630 \text{ cm}^3$  of orange juice. He has 5 cups to pour the juice in. Thomas poured equal volume of orange juice per cup. When Thomas pours the orange juice in each cup, he records the percentage that the cup is filled up. His data is shown below. Write a function that prints the capacity of each cup. Your answer should be in the format:

“Cup \_\_\_ has a capacity of \_\_\_.”

Cup A: 50% full

Cup B: 75% full

Cup C: 30% full

Cup D: 63% full

Cup E: 88% full

Question 6:

James was given 4 bags of cherries. He noticed that in each bag, there were some ripe and unripe cherries mixed inside. He counted the number of ripe cherries and wrote his findings below. He also recorded down the total number of cherries per bag below. Write a function that prints the percentage of unripe cherries in each bag. Your answer should follow the following format:

“Bag \_\_\_ has \_\_\_% unripe cherries.”

Bag A: 25 ripe cherries out of 100

Bag B: 29 ripe cherries out of 89

Bag C: 87 ripe cherries out of 92

Bag D: 67 ripe cherries out of 102