CPIT 110

Instructor Manual

For 50 Minutes Lectures

Week 13

24/11/2019 – 28/11/2019

Chapter 6
Functions

This Week Events	− Mid-Term Exam 2 - Part 2 (Writing Code)
Next Week Events	− Lab #9 (Chapter 6) − **Final Exam - Part 1 Writing Code (Week #15)**

FACULTY OF COMPUTING & INFORMATION TECHNOLOGY
KING ABDULAZIZ UNIVERSITY

كلية الحاسبات وتقنية المعلومات
جامعة الملك عبدالعزيز

FCIT
KAU

CPIT 110
Instructor Manual – Lecture #1 in Week 13

Chapter	**6. Functions**
Number of Lectures	3 (50 minutes / Lecture)
Lecture	1 of 6
Slides	1 - 28
Date	Sunday 24/11/2019

Week **13**	**Lecture** **1 of 6**
	Slides **1 - 28**

Topics to Be Covered

❖ **6.1. Introduction**
❖ **6.2. Defining a Function**

Learning Objectives

Learning Outcomes	**Topics**
− To define functions with formal parameters.	6.2. Defining a Function

Exercises

No exercises.

FACULTY OF COMPUTING & INFORMATION TECHNOLOGY

KING ABDULAZIZ UNIVERSITY

كلية الحاسبات وتقنية المعلومات

جامعة الملك عبدالعزيز

FCIT
K A U

CPIT 110
Instructor Manual – Lecture #2 in Week 13

Chapter	6. Functions
Number of Lectures	3 (50 minutes / Lecture)
Lecture	2 of 6
Slides	29 - 75
Date	Tuesday 26/11/2019

Week 13	Lecture 2 of 6
	Slides 29 - 75

Topics to Be Covered

❖ **6.3. Calling a Function**

Learning Objectives

Learning Outcomes	Topics
− To invoke functions with actual parameters.	6.3. Calling a Function

Exercises

No exercises.

FACULTY OF COMPUTING & INFORMATION TECHNOLOGY

KING ABDULAZIZ UNIVERSITY

كلية الحاسبات وتقنية المعلومات

جامعة الملك عبدالعزيز

FCIT
K A U

CPIT 110
Instructor Manual – Lecture #3 in Week 13

Chapter	**6. Functions**
Number of Lectures	3 (50 minutes / Lecture)
Lecture	3 of 6
Slides	76 - 99
Date	Thursday 28/11/2019

Week **13**	Lecture **3 of 6**
	Slides **76 - 99**

Topics to Be Covered

❖ **6.4. Function with/without Return Values**

Learning Objectives

Learning Outcomes	**Topics**
− To distinguish between functions that return and do not return a value	6.4. Function with/without Return Values

Exercises

❖ **6.4. Function with/without Return Values**
1. What are the benefits of using a function?
2. How do you define a function? How do you invoke a function?
3. True or false? A call to a None function is always a statement itself, but a call to a value-returning function is always a component of an expression.
4. Can you have a return statement in a None function? Does the return statement in the following function cause syntax errors?

```
1  def xFunction(x, y):
2      print(x + y)
3      return
```

5. Define the terms function header, parameter, and argument.

FACULTY OF COMPUTING
& INFORMATION TECHNOLOGY
KING ABDULAZIZ UNIVERSITY

FCIT
K A U

كـلـيـة الـحـاسـبـات
وتـقـنـيـة الـمـعـلـومـات
جـامـعـة الـمـلـك عـبـدالـعـزيـز

6. Write function headers for the following functions (and indicate whether the function returns a value):
   - Computing a sales commission, given the sales amount and the commission rate.
   - Printing the calendar for a month, given the month and year.
   - Computing a square root.
   - Testing whether a number is even, and returning true if it is.
   - Printing a message a specified number of times.
   - Computing the monthly payment, given the loan amount, number of years, and annual interest rate.
   - Finding the corresponding uppercase letter, given a lowercase letter.

7. Identify and correct the errors in the following program:

```
1   def function1(n, m):
2       function2(3.4)
3
4   def function2(n):
5       if n > 0:
6           return 1
7       elif n == 0:
8           return 0
9       elif n < 0:
10          return -1
11
12  function1(2, 3)
```

8. Show the output of the following code:

```
1   def main():
2       print(min(5, 6))
3   def min(n1, n2):
4       smallest = n1
5       if n2 < smallest:
6           smallest = n2
7
8   main() # Call the main function
```

FACULTY OF COMPUTING
& INFORMATION TECHNOLOGY
KING ABDULAZIZ UNIVERSITY

كلية الحاسبات
وتقنية المعلومات
جامعة الملك عبدالعزيز

FCIT
KAU

9. What error will occur when you run the following code?

```
1  def main():
2      print(min(min(5, 6), (51, 6)))
3  def min(n1, n2):
4      smallest = n1
5      if n2 < smallest:
6          smallest = n2
7  main() # Call the main function
```