PRG1

W
E
E
K

2

NGEE ANN
SCHOOL OF INFOCOMM TECHNOLOGY

Introduction to Functions

Programming I (PRG1)

Diploma in Information Technology

Diploma in Financial Informatics

Diploma in Information Security & Forensics

Year 1 (2018/19), Semester 1

Objectives

At the end of this lecture, you will ….

- Gain a basic understanding of how functions work.

How to allow others to use your code

- Assuming that Tom has written this code. It works!

```
#CalculateBMI.py

weight = int(input('Please enter your weight: '))
height = float(input('Please enter your height: '))

#Perform BMI calculation
def calculate_bmi(weight, height):
    bmi = weight/height ** 2 #Formula to calculate BMI
    print('BMI: ', bmi)#Modify to display the calculated BMI value
```

- Tom is very proud of his code and he is now thinking of allowing others to use it.

Defining a Function

- In order to do that, Tom inserts his code in a block called a **function**, using the ***def*** keyword. We say that Tom now is ***defining a function***.

```
#Perform BMI calculation
def calculate_bmi(weight,height):
    bmi =   weight/height**2 #Formula to calculate BMI
    print("BMI:",bmi) #Modify to display the calculated BMI value

    return bmi #Do not remove this line
```

Code that performs the task gets inserted here.

- Tom also wants to provide the calculated value to whoever is making use of his code, so he does that using ***return***

Calling a Function

- Now Sam wants to use Tom's function.
- Sam writes some code to get weight and height, then he ***calls*** Tom's ***function***, which is named calculate_bmi.

```
#Perform BMI calculation
def calculate_bmi(weight,height):
    bmi =   weight/height**2 #Formula to calculate BMI
    print("BMI:",bmi) #Modify to display the calculated BMI value

    return bmi #Do not remove this line
```

Tom defines a function that performs task.

```
weight = int(input("Please enter your weight: "))
height = float(input("Pleae enter your height: "))
```

Sam gets input

```
calculate_bmi(weight,height)
```

Sam calls Tom's function to process the inputs.

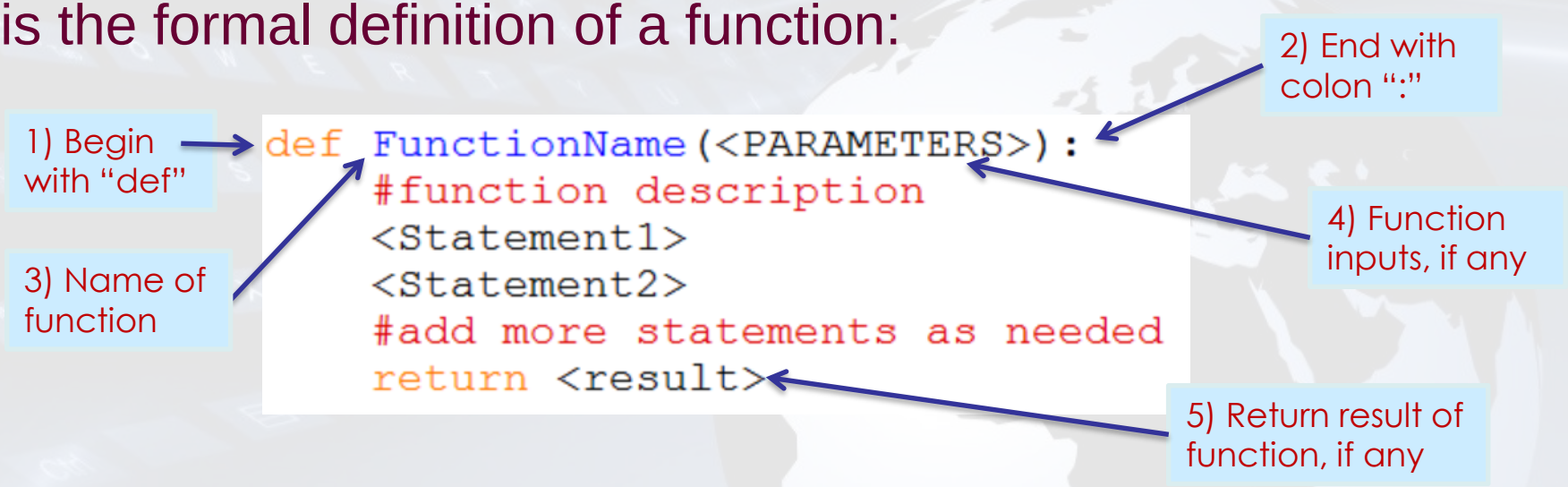
How is this important for us?

- In our practical submissions, Coursemology also works the same way.
- After you have written your code, you need to insert your code in a function and allow Coursemology to make use of it and thus validate your code.

```
#CalculateBMI.py

#weight = int(input('Please enter your weight: '))
#height = float(input('Please enter your height: '))

#Perform BMI calculation
def calculate_bmi(weight, height):
    bmi = weight/height ** 2 #Formula to calculate BMI
    print('BMI: ', bmi)#Modify to display the calculated BMI value

    return bmi #Do not remove this line

#Do not remove the next line
calculate_bmi(weight, height)
```

Function header

- More details on functions will be covered at a later topic. Here is the formal definition of a function:




```
def FunctionName(<PARAMETERS>):
    #function description
    <Statement1>
    <Statement2>
    #add more statements as needed
    return <result>
```

Another example:

```
def multiply_two_values(num1, num2):
    #multiply 2 values
    answer = num1 * num2
    print('Result is', answer)
    return answer

num1 = int(input('Please enter num1: '))
num2 = int(input('Please enter num2: '))

multiply_two_values(num1, num2)
```

Summary

- You can allow others to use your code by defining it in a function.
- The function can then be called anytime to perform the task whenever needed.