

Embarak _Ch01_Introduction_Functions and Modules

September 5, 2018

1 Functions

```
In [20]: def thing():    # function header {def, function name, function arguments }
          print ('Hello ', end='')    # function body
          print ('World')
```

```
thing() # function calling
```

Hello World

```
In [13]: def print_Sms():
          print ("Welcome to Python PD session")
          print ("Jan 2017\n")

          print ("\nPD Session on HCT Dubai")
          print_Sms()
```

PD Session on HCT Dubai
Welcome to Python PD session
Jan 2017

```
In [18]: def sumvalues(x,y):
          print ('The summation of ',x,'+',y,'= ', end='')
          return x+y
```

```
a=5
b=a+2
print (sumvalues(a,b) ) # Function calling
```

The summation of 5 + 7 = 12

```
In [3]: def Details(name, mark):
        if mark>60:
            print ("Congratulation ",name," you pass the course")
        else:
            print ("Unfortunately ",name," you didnt pass the course")
```

```
In [4]: Details("Ossama", 90)
```

```
Congratulation  Ossama  you pass the course
```

```
In [5]: Details( 90,"Ossama")
```

```
-----

TypeError                                Traceback (most recent call last)

<ipython-input-5-095c843850f3> in <module>()
----> 1 Details( 90,"Ossama")

<ipython-input-3-1e8045027d0a> in Details(name, mark)
      1 def Details(name, mark):
----> 2     if mark>60:
      3         print ("Congratulation ",name," you pass the course")
      4     else:
      5         print ("Unfortunately ",name," you didnt pass the course")

TypeError: '>' not supported between instances of 'str' and 'int'
```

```
In [6]: Details( mark=90, name="Ossama")
```

```
Congratulation  Ossama  you pass the course
```

```
In [7]:
```

```
-----

TypeError                                Traceback (most recent call last)

<ipython-input-7-b5eaae6e7476> in <module>()
----> 1 Details( "Ossama")

TypeError: Details() missing 1 required positional argument: 'mark'
```

```
In [9]: def Details(name, mark):
        if mark>60:
            print ("Congratulation ",name," you pass the course")
        else:
            print ("Unfortunately ",name," you didnt pass the course")
```

```
In [10]: Details( "Ossama")
```

```
-----

TypeError                                Traceback (most recent call last)

<ipython-input-10-b5eaae6e7476> in <module>()
----> 1 Details( "Ossama")
```

```
TypeError: Details() missing 1 required positional argument: 'mark'
```

```
In [11]: def Details(name, mark=0):
        if mark>60:
            print ("Congratulation ",name," you pass the course")
        else:
            print ("Unfortunately ",name," you didnt pass the course")
```

```
In [12]: Details( "Ossama")
```

```
Unfortunately  Ossama  you didnt pass the course
```

```
In [1]: max('Welcome to Egypt')
```

```
Out[1]: 'y'
```

```
In [2]: min(3,5,8,9,100,2)
```

```
Out[2]: 2
```

```
In [3]: len('Welcome to Egypt')
```

```
Out[3]: 16
```

```
In [8]: mark=input("Enter your exam mark: ")
        mark=float(mark)
        if (mark>59.5):
            print ("Pass")
        else:
            print ("Fail")
```

```
Enter your exam mark: 55
```

```
Fail
```

2 Convert Celsius to Fahrenheit

2.1 $F = 1.8 C + 32$

```
In [9]: value = input("Enter the Celsius value: ")
        c = int(value)
        f = 1.8 * (c) + 32
        print (c , " Celsius = ", f , " Fahrenheit")
```

```
Enter the Celsius value: 3
3 Celsius = 37.4 Fahrenheit
```

```
In [2]: import random
        for i in range(5):
            x = random.random()
            print (x)
```

```
0.9743073233004959
0.8717800197648119
0.16804156882252586
0.11293371207526814
0.5732777436707432
```

```
In [4]: import random
        for i in range(5):
            x = random.random()
            print (round(x,3))
```

```
0.524
0.109
0.976
0.866
0.278
```

```
In [5]: random.randint(5, 10)
```

```
Out[5]: 9
```

```
In [9]: random.randint(5, 10)
```

```
Out[9]: 5
```

```
In [7]: random.randint(5, 10)
```

```
Out[7]: 10
```

```
In [12]: random.randint(5, 10)
```

```
Out[12]: 10
```

```
In [16]: t = [30, "Omar", 7, 10]
         random.choice(t)
```

```
Out[16]: 10
```

```
In [17]: random.choice(t)
```

```
Out[17]: 'Omar'
```

```
In [18]: random.choice(t)
```

```
Out[18]: 7
```

```
In [23]: import math
         value = 120
         decibels = 10 * math.log10(value)
         print (decibels)
```

```
20.791812460476248
```

```
In [24]: degrees = 45
         radians = degrees / 360.0 * 2 * math.pi
         val= math.sin(radians)
         print (val)
```

```
0.7071067811865475
```

```
In [30]: print (math.sqrt(16))
```

```
4.0
```

```
In [34]: # Anonymous Function Definiton
         summation=lambda val1, val2: val1 + val2

         #Calling summation as a function
         print ("The summation of 7 + 10 = ", summation(7,10) )
```

```
The summation of 7 + 10 = 17
```

```
In [35]: quiz = 50
         def readgrade():
             quiz = input("Enter your quiz mark: ")
             quiz = int(quiz)
             print ("Your quiz score is ", quiz)

         readgrade()
         print ("Your quiz score is ", quiz)
```

Enter your quiz mark: 70
Your quiz score is 70
Your quiz score is 50

```
In [ ]: print ("\n***** Greeting *****")
def greeting(lang):
    if lang=='es':
        print ('Hola')
    elif lang=='fr':
        print ('Bonjour')
    else:
        print ('Hello')

greeting('en')
greeting('es')
greeting('fr')

In [1]: def computepay(hours, rate):
    if hours>40:
        pay= 40 * rate + (rate*1.5) * (hours-40)
    else:
        pay= hours * rate
    return pay

hours = input ('Enter Hours: ')
try:
    hours = int(hours)
except:
    print ("Incorrect hours number !!!!")

try:
    rate = input ("Enter Rate: ")
    rate=float(rate)
except:
    print ("Incorrect rate !!")

fullpay =computepay(hours, rate)

print ("Gross Pay: ", fullpay)
```

Enter Hours: 50
Enter Rate: 10
Gross Pay: 550.0

2.2 Exercises

2.2.1 find the Highest Common Factor of two values.

```
In [5]: def HCF(x, y):
        if x > y:
            smaller = y
        else:
            smaller = x
        for i in range(1, smaller + 1):
            if((x % i == 0) and (y % i == 0)):
                HCF = i
        return HCF

        Number1 = int(input("Enter first number: "))
        Number2 = int(input("Enter second number: "))
        print("The Highest Common Factor of", Number1, "and", Number2, "is", HCF(Number1, Number2))
```

```
Enter first number: 36
Enter second number: 16
The Highest Common Factor of 36 and 16 is 4
```

```
In [6]: #Find Factorial of Number Using Recursion
```

```
In [9]: def RecurFactorial(n):
        if n == 1:
            return n
        else:
            return n*RecurFactorial(n-1)

        # read the value from the user
        Number = int(input("Enter a number: "))

        # check is the number is negative
        if Number < 0:
            print("Sorry, factorial does not exist for negative numbers")
        elif Number == 0:
            print("The factorial of 0 is 1")
        else:
            print("The factorial of", Number, "is", RecurFactorial(Number))
```

```
Enter a number: 5
The factorial of 5 is 120
```

```
In [12]: def RecurFibo(n):
        if n <= 1:
            return n
        else:
```

```

        return(RecurFibo(n-1) + RecurFibo(n-2))

# read input from the user
nlength = int(input("Enter your length? "))
# check if the number of terms is valid
if nlength <= 0:
    print("Plese enter a positive integer")
else:
    print("Fibonacci sequence:")
    for i in range(nlength):
        print(RecurFibo(i), end=' , ')

```

```

Enter your length? 10
Fibonacci sequence:
0 , 1 , 1 , 2 , 3 , 5 , 8 , 13 , 21 , 34 ,

```

2.3 4.6 CREATE PYTHON MODULES

```

In [6]: import addition
        addition.add(10,20)
        addition.add(30,40)

```

ModuleNotFoundError

Traceback (most recent call last)

```

<ipython-input-6-1a2204da0625> in <module>()
----> 1 import addition
      2 addition.add(10,20)
      3 addition.add(30,40)

```

ModuleNotFoundError: No module named 'addition'

```

In [7]: "{1} {0}".format(x, "The")
        "{first} {second}".format(first="The", second=x)

```

NameError

Traceback (most recent call last)

```

<ipython-input-7-e4ee71e1ad54> in <module>()
----> 1 "{1} {0}".format(x, "The")
      2 "{first} {second}".format(first="The", second=x)

```



```
NameError: name 'x' is not defined
```