

```

1 #Lesson 08
2 #The Code Revolution
3 #Advanced Functions
4
5 #We've actually jumped into functions a little early in the last few
6 #lessons to make running and testing our code cleaner and easier to use
7 #In this lesson we will dig further into functions and how they
8 #can be used to do additional things like:
9 # 1. Passing arguments / parameters
10 # 2. Using return values
11 # 3. Optional Arguments
12 # 4. Returning more complex types like Dictionaries & Lists
13 # 5. Using arbitrary arguments
14 # 6. Storing functions in modules
15 # 7. Importing modules and functions
16
17 #####
18 ### ARGUMENTS & PARAMETERS ###
19 #####
20
21 def usingarguments(randomnumber):
22     #we will demonstrate
23     print("The argument passed to this function is: " + str(randomnumber))
24     #the 'randomnumber' is called the parameter
25     #the 'userchoice' VALUE passed is the argument
26
27 def usingpositionalarguments(string1, string2):
28     #when we use positional arguments, that means the position
29     #of the argument in the function call will associate to the
30     #parameter in the function
31     print("The first argument is: " + string1)
32     print("The second argument is: " + string2)
33
34 #when we use keyword arguments we provide the parametername='value' in
35 #the function call. that way we don't have to be concerned about the order
36 #that the arguments are provided
37 def usingkeywordarguments(string1, string2):
38     print("This should demonstrate keyword arguments:")
39     print("We don't have to specify the order if we associate the arguments")
40     print("to the parameter names.")
41     print("The first parameter string1's argument value is: " + string1)
42     print("The second parameter string2's argument value is: " + string2)
43
44 #The 2nd parameter below is optional because we are supplying
45 #a default value when a value is not passed
46 def usingdefaultvalues(string1, string2="string2"):
47     print("Using Default values we can set a parameter value by default")
48     print("In this case our 2nd parameter has a value of 'string2'")
49     print("String1 value: " + string1)
50     print("String2 value: " + string2)
51
52
53
54 #####
55 ### USING RETURN VALUES #####
56 #####
57
58 #To make things easier we can provide a return value to the calling function
59 #so that we can allow the calling function to process the data
60 #we've already done this with the input function
61 #i.e. input("Some question for the user") returns a string that we
62 #do something with. If we do nothing with that value, nothing happens
63 #Using return values is the same because we just return the data to the caller
64 #function and decide what to do with the data at that point

```

```

65
66 def returningavalue(firstname, lastname):
67     #we will concatenate these arguments and title() them
68     #then return the formatted string
69     fullname = firstname.title() + " " + lastname.title()
70     return fullname
71
72
73
74 #####
75 ### OPTIONAL ARGUMENTS #####
76 #####
77
78 #Optional arguments are the same as default values in arguments.
79 #The difference is that we set the default value to be an empty
80 #zero length string. I.e. optionalvalue = ""
81
82 def usingoptionalvalues(optional=""):
83     if len(optional) == 0:
84         print("You did not provide any optional value to the function.")
85     else:
86         print("The value you provided was: " + optional)
87
88 #####
89 ### RETURNING COMPLEX VALUES #####
90 #####
91
92 #in this example we will return a dictionary to the calling function
93 def createaperson(firstname, lastname, age):
94     person = { "first": firstname,
95               "last": lastname,
96               "age": age}
97     return person
98
99 #####
100 ### ARBITRARY ARGUMENTS #####
101 #####
102
103 #Python allows us to pass an arbitrary number of arguments
104 #assuming the data passed can change from call to call, we want to be flexible
105 def makemeapizza(*toppings):
106     for topping in toppings:
107         print("Topping: " + topping)
108
109 #####
110 ### MAIN USER INPUT SECTION #####
111 #####
112 print("Welcome to the Code Revolution Lesson 8 - Functions!")
113 print("Please choose a mini exercise to test!")
114 print("1. Passing arguments and parameters\n2. Using return values")
115 print("3. Optional Arguments\n4. Returning complex values")
116 print("5. Using arbitrary arguments")
117 userchoice = input("=>")
118
119 #let's introduce a new concept, called the try to try "something"
120 #first so we can control what happens if there's an error
121 try:
122     userchoice = int(userchoice)
123 except:
124     print("You did not enter a correct number choice. Please restart.")
125
126 if userchoice == 1:
127     usingarguments(userchoice)
128     usingpositionalarguments("first", "second")
129     usingkeywordarguments(string2="this is string2", string1="this is string1")

```

```

130     # note this function has two parameters but we are only
131     # passing one argumet
132     usingdefaultvalues("just one argument")
133     passdatatodefault = input("Enter some text to pass to the default value: ")
134     # but in this case we are passing two arguments to the same function
135     usingdefaultvalues("just one argument", passdatatodefault)
136 elif userchoice == 2:
137     fname = input("What is your first name? ")
138     lname = input("What is your last name? ")
139     #I changed the variable names here not to match the parameter names
140     #just to reitirate that it's the order that's important
141     #otherwise we have to use the keyword arguments above
142     fullname = returningavalue(fname, lname)
143     print("Your formatted name is: " + fullname)
144 elif userchoice == 3:
145     #calling a function with an optional value passed
146     usingoptionalvalues("Here's an Optional Value")
147     #calling the same function without a value passed
148     usingoptionalvalues()
149 elif userchoice == 4:
150     fname = input("What is your first name?")
151     lname = input("What is your last name?")
152     age = int(input("How old are you in years?"))
153
154     person = createaperson(fname, lname, age)
155     print("The person you created is:")
156     for key, value in person.items():
157         print(key, ": ", str(value))
158
159 elif userchoice == 5:
160     print("Review in your own time Using Arbitrary Arguments in Python.")
161     print("My preference here would be to use a List.")
162     makemeapizza("pepperoni", "cheese", "mushrooms")
163     makemeapizza("cheese")
164
165

```