Hyperion Development

Task: Introduction to Functions - Application

www.hyperiondev.com

Introduction

Welcome to the Introduction to Functions Application Task of the Certified Software Developer Course!
Overview:

This task is aimed to ensure that you have a concrete understanding of Strings, Functions and basic List manipulations, as these will be needed for upcoming more advanced tasks. In example.py, you will see examples that deal with functions and operations that can be applied to elements in lists. The Task also re-introduces Functions and how they can be used to create apps for realistic applications.

In the previous task you've gotten comfortable with the basic aspects of functions. Next we'll look at more application oriented examples, with the end goal being the creation of your own application!

The first step to a more functional program is the use of arguments in your functions. This allows for a function to act on, or carry out on operation using some input variables that you've defined.

Below shows an example of multiple argument inputs to increase the performance of a single function.

```
def mult(a, b):
    if b == 0:
        return 0
    rest = mult(a, b - 1)
    value = a + rest
    return value
result = mult(3, 2)
print "3 * 2 = ", result
```

Another powerful use of functions is a term coined recursion - the repetitive use of a function to achieve some output. An immediate example that should come to mind is a counter or the infamous Fibonacci sequence. You'll see applications of both below.

```
def factorial(n):
    if n <= 1:
        return 1
    return n * factorial(n - 1)

print "2! =", factorial(2)
print "3! =", factorial(3)
print "4! =", factorial(4)
print "5! =", factorial(5)
```

```
def count_down(n):
    print n
    if n > 0:
        return count_down(n-1)

count_down(5)
```

Now it's time to deal with larger applications. Navigate to the example programs folder in your current task folder where you'll locate a python file called Area.py. Make sure you work through the code and understand how it works before you attempt your compulsory task!

Instructions

Feel free at any point to refer back to previous material if you get stuck. Remember that if you require more assistance our tutors are always more than willing to help you!

- We are going to create an app that can calculate the costs required to go on holiday to different locations in South Africa.
- You are not required to read the entirety of Additional Reading.pdf, it is purely for extra reference.
- Complete the compulsory exercise to progress to the next task. It may be helpful to revise your previous task to fully understand how functions work.

Compulsory Task

Follow these steps:

- Create a Python file called "holiday.py" in this folder.
- You will need to to create four functions:
  - Hotel cost - This function will take the number of nights as an argument and return a total cost (You can choose the price per a night)
  - Plane cost - This function will take the city you are flying to as an argument and return a cost for the flight (Hont: use if/else if statements in the function to retrieve a price based on the chose city)
  - Car rental - This function will take the number of days as an argument and return the total cost.
  - Holiday cost - This function will take three arguments, number of nights, city, and days. Using these three arguments, you can call all three of the above functions with respective arguments and finally return a total cost for your holiday.
- Print out the value of your Holiday function to see the result!
- Try using your app with different combinations to show it's compatibility with different options

Things to look out for

1. Make sure that you have installed and setup all programs correctly. You have setup **Dropbox** correctly if you are reading this, but **Python or Notepad++** may not be installed correctly.
2. If you are not using Windows, please ask your tutor for alternative instructions.

Still need help?

Just write your queries in your comments.txt file and your tutor will respond.

Task Statistics

Last update to task: 23/12/2015.
Authors: Jared Ping.
Main trainer: Umar Randeree.
Task Feedback link: Hyperion Development Feedback.