



Image credit: [www.lifeofadventure.com/](http://www.lifeofadventure.com/)

**This is an individual assignment.**

**Due date: Friday February 7 , 11:59 pm**

*LEARNING OBJECTIVES, TOPICS THAT YOU WILL BE SPECIFICALLY WORKING WITH:*

- Practice and strengthen coding of a basic Python program with user interaction
- Variables, assignment statement
- `print(...)` , `input(...)`
- data type conversions (string, int)
- If/else statements
- String manipulations, string built-in methods and functions
- Basic usage of Lists
- Numeric calculations
- Importing and using some functions in the *random* and *math* module
- Defining one simple function

Submit Assignment

**Similarly to the previous assignment (T-shirt greeter),**

- **submit in Canvas two files:**
- **your .py file and your .txt reflection file,**
- **YOU SHOULD NOT SUBMIT JUST THE REPL URL – UPLOAD THE PYTHON .py FILE**

#### *GENERAL PROBLEM DESCRIPTION*

Create a Python program, which implements a simplified “Travel agent bot”.

To start, the bot will greet the user, ask the name , and inform the user the “general traveling options”. Then the bot will ask the user’s the various choices. The bot will confirm choices, and if it does not understand it will choose randomly for the user. Given the various choices, the bot will inform the travel plan, the cost, and a reservation code.

The bot will use a variety of possible responses in some cases, and the user will be able to respond in a variety of ways for some answers as well.

To have different users interact with this bot, the program needs to be run again.

**MORE DETAILED DESCRIPTION:**

Also check the sample runs provided below to further understand how the bot should work. Other possible situations may occur. Your program should work with a variety of responses.

**Traveling options (which will be asked to the user):**

The user will be asked to choose:

- a. to travel to any of 4 destinations
  - As a programmer, you will include the destinations explicitly (“hard coded”)
  - three (3) destinations should start with upper case letters ‘A’, ‘B’, ‘C’ and you should have at least one (maybe 2, up to you) more destinations starting with any uppercase letter after ‘K’ in the alphabet. Destinations should have at least 3 letters.
- b. a travel modality from: plane, cruise, bus, hiking. [the user will select based on a number, and show the word to confirm – see the sample runs]
  - **b.1 BONUS:** The user may choose an additional second travel modality. This will cost extra according to the rules below
- c. If it is a regular or deluxe trip.
  - Restriction: If the user chose a hiking modality, *regular* modality has to be imposed.
- d. the number of days he/she wants to travel. [Assume the user types a number. To test try small numbers]

***If the user types wrongly any of the options a. to c., it will choose randomly one of the options***

**Cost calculation:**

Each day of traveling will cost 400 if the modality is plane or car, and 200 if bus or hiking. [If you do the bonus, a second modality adds that cost]. A deluxe trip adds 1000 to the cost. 50 is added if the destination starts with a letter after ‘K’. The tax is 10% of the result of calculating the name length multiplied by the number pi (math.pi) . The total cost should be the integer part of the sum of all costs. [Your program should show as “TRACE printing” some intermediate calculations. See the sample runs]

**Reservation code:**

The reservation code will have

- the first three(3) letters of the destination repeated  $n$  times (in upper case), where  $n$  is 3 plus the remainder of dividing the number of trip days by 4 .
- then the number  $n$
- then the first two letters of the first (or only) travel modality chosen,
- (part of the bonus: the code will continue with the first two letters of the second travel modality, or one star if the user did not choose another modality of travel) .
- then one or two dollar signs (\$) as follows:
  - If the length of the reservation code generated so far (until the previous step) is odd, then the code will have one dollar sign. If such length is even, the code will have two dollar signs.

### Dialog

The bot should accept a variety of ways that the user may say 'yes' at the end, such as : 'sure', 'of course', 'YES!', etc. The bot should say good bye to the user in a variety of ways (chosen randomly among several possible messages). All other messages are pre-determined. [See sample runs]. Check the assumptions requirements next as well.

### *REQUIREMENTS OF CODE ORGANIZATION AND ASSUMPTIONS*

- Your code needs to follow the [general coding guidelines](#) (See Home website → More LINKS → Python source code general guidelines), including **comments** as described there
- Define (and call) **ONE Python function** informing the general travelling options. (Check the chef bots programs in the lab exercises week #2 to see how basic functions just printing messages are defined and used). No other function definitions are required for this assignment.
- You need to include TRACE printing as included in the sample runs. (Trace printing is information that does not really correspond for the 'real user' to see, but rather useful for you to debug your program and to mark your work.
- The order in which the questions are asked to the user should be the same as in the sample runs
- Assume that when the user is asked a number, he/she provides a number
- In all other choices, if the user does not type as expected, the bot will choose randomly
- Anything that is mentioned in this description or included in the sample runs will be given some points. The exact rubric will be posted later. The submission requirements will also be given points.

### *WHAT YOU ARE REQUIRED TO SUBMIT:*

- Your python file, named [mytravel1.py](#).
- A [reflections.txt](#) file, where you include how much time it took you to do this exercise (approximate, in hours, minutes) and any comments reflecting on what you learned and/or any comment to your marker.

**See SAMPLE RUNS posted together with this description.**  
**The sample runs also describe the problem!**

*End of travel agent assignment description*