**CP1401 SP2 2016 Assignment 2**

**Implement the Tropical Airlines program**

# Task

You are to implement (code in Python) a plane ticket ordering program as described in the following description. Use what you have learned in class, including simple functions, selection, repetition and lists. You should use any feedback that you received from Assignment 1 to improve your program flow.

**See the last section of this document for sample output showing how the program   
should run.**

# Program Features:

**Short description:**  
The program will begin by asking the user for their name, welcoming them and then displaying a menu with three choices, as shown below. Choosing ‘Instructions’ or ‘Order’ will make the program perform the actions described below. If the user chooses to ‘Exit’ the program will display the costs of each order they have made using the program, and then a farewell message as described below. Any invalid value will produce an error message. After every menu selection (other than ‘Exit’) the menu will be displayed again.

The ‘Instructions’ option will display a short paragraph explaining the program, as shown below. The ‘Order’ option will take the user through a series of choices to order a single ticket. Once the choices are made the program will display a summary of the user’s choices and a final total for the ticket. After either of these options, the menu will be displayed again.   
  
When the user chooses (E)xit, a listing of the totals from the previous orders should be displayed with the output message “Your orders are:”, followed by the previous order totals in ascending order (smallest to largest) and a final total of all the orders combined. Then the message “Thank you for visiting Tropical Airlines” will be displayed.

See next page for Detailed Instructions.

**Detailed instructions:**

1. Ask for the user’s name and welcome them to the ordering system, using their name
2. Show the program menu:

(I) nformation  
(O)rder ticket  
(E) xit  
  
If the user chooses (I) display the following text:

“Thank you for choosing Tropical Airlines for your air travel needs. You will be asked questions regarding what type of ticket you would like to purchase as well as destination information. We also offer 50% discounted fares for children.”

If the user chooses (O) then the program will begin the process of ordering a ticket, as described below.

ALL other choices will produce an error. The menu will be displayed after every menu choice, except (E)

1. Once the (O) option has been entered, the program welcomes the user and asks if the ticket is for the user or for another person. If it is for another person, get the traveller’s name to display at the end, otherwise use the user’s name. For the following instructions you will need to keep track of the values and make all needed calculations at the end.

For this assignment you MUST ask the users for the following pieces of information in the order given, and you MUST allow for the specific inputs shown. Real world clients will often have the final say over the user interface, so this is a common restriction.

* 1. Ask if ticket will be one way or return (round trip)
  2. Ask for the destination (choices are Cairns, Sydney or Perth).
  3. Ask for the type of fare (Business, Economy or Frugal)
  4. Ask what type of seat (Window, Aisle or Middle)
  5. Ask age of person in order to check for child fare
  6. Show user or traveller name, choices and total

1. If user chooses (E), the program should exit with the message: “[username], your orders are:” with a list of the ticket costs in ascending numerical order, then the message “Your final total is:” with the combined total and then the message “Thank you for visiting Tropical Airlines”.

Example:  
“[username], your orders are: $250, $300 and $550. Your final total is: $1100. Thank you for visiting Tropical Airlines.”   
  
You also must handle the situation where only one order is placed   
“[username], your order is: $250. Your final total is: $250. Thank you for visiting Tropical Airlines”.

0

# Implementation:

Python file:  
You are to divide your solution into functions, following the principles shown in class. These may involve one for each menu option (other than Exit) as well as functions for parts of the program (e.g. a calculating the total ticket price could be implemented as a function). For each function you are to provide an IPO chart listing all variables which are needed for your solution. In cases where selections are necessary you must include a suitable condition action table directly below the IPO chart. Follow the sample solution provided in Assignment One as a guide.

Planning document:

You also need to provide an updated version of your previous planning document showing an algorithm in pseudocode for each function, being sure to clearly indicate input and output variables for each. This should be based on your submission for Assignment One, with appropriate changes being made for new functionality.

You may show your assignment to your tutor during practical time to get comments or suggestions. It is important to note that you can only get help from staff in practical time after your prac work is finished.

# General Principles:

In this assignment, you will be focusing on basic planning and implementation using selections, repetition, and functions. Lists may be used where they are feasible.

Use the techniques and patterns that you have learned and seen demonstrated in class.

* You are allowed to use the code that Flowgorithm generates as a base for your project, although you will need to make additions to it:
  + Flowgorithm does not create a main function, so you must do this at either the top or the bottom of your code
  + Flowgorithm generated code does not always follow the best practices we have shown in lectures and pracs. Ensure that you do.
* Coding standards:
  + Variables:
    - Wherever possible, variables should be set at the top of the function
    - Although Python does not use constants you may set variables to values that will remain the same for the life of the program. These should be named in all caps
      * You should be able to modify these variables to adjust the various costs of items such as one-way fare, return, discount for child tickets, etc. This is one of the most important aspects of this assignment, so consider carefully how to use constants. Remember to use these variables everywhere you can.
      * You will need to decide the best position to place these variables.
    - You should AVOID the use of global variables unless they are needed in multiple places in the application
    - In other words, you should pass the values to any functions that need them
      * And any constants only needed in one function should be ONLY in that function
    - Variables should be named appropriately using either camelCase or under\_scores
    - No variables should be a single character such as “x” or “y” unless they are used in a loop as an index. Variable names should be meaningful, and describe their use and/or purpose
    - Any list variable should be named with plurals to show multiple data is held (e.g. prices versus price)
  + Comments:
    - You will need to add at LEAST one comment for each function that you define including main
    - The comment should either be placed directly above or below the function header
    - The comment should describe: (a) the purpose of the function, (b) any inputs and (c) any returned values
    - Additional comments should be placed at any position where the functionality of the program may be unclear
    - All code MUST BE properly indented
  + Error handling:
    - Note that menu choice and other character selection should handle upper and lower case letters.
      * Look into the toUpper() or toLower() functions in Python
    - You also must make sure that any functions that get input from the user do error checking to ensure that the input is within the expected range (greater than zero, etc).
  + Output
    - Make your output display as close as possible to the given examples, especially formatting and layout. Use Google to find out how to display tabs and other special characters using Python 3.

# Hints:

* You may find a list variable useful for holding order costs
* You’ll need the cost of all orders to be stored in main, for display when the user quits. This may need to be returned from a function.
* Try to use pre-defined Python functions wherever possible (such as sort, total, len)

# Submission:

Hand in a zip file (named like: **FirstnameLastname)** containing the following:

* Python file (.py)
* Planning document in Microsoft Word format (.docx)

Submit your zip file by uploading it on LearnJCU under Assessment.

# Due:

Submit your assignment bythe date and time specified on LearnJCU.

Submissions received after this date will incur late penalties as described in the subject outline.

# Integrity:

The work you submit for this assignment must be your own. You are allowed to discuss the assignment with other students and get assistance from your peers, but you may not do anyone else’s work for them and you may not get anyone else to do any part of your work. Programs that are detected to be too similar to another student’s work will be dealt with promptly according to University procedures for handling plagiarism.

If you require assistance with the assignment, please ask **general** questions on the itatjcu.slack.com web site, or get **specific** assistance with your own work by talking with your lecturer or tutor.

# Marking Scheme:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Criteria** | **Unacceptable (0%)** | **Limited (40%)** | **Basic (60%)** | **Sound (80%)** | **Excellent (100%)** |
| Planning documents show appropriate use of problem-solving tools (including algorithm, IPO charts and condition/action tables) to communicate the planned solution.  (10% | No use of any problem solving tools | The algorithm provided does not solve the problem, or is not based on other parts of the planning documentation. | The algorithm provided mostly solves the problem, and is based on other parts of the planning documentation; however there are more than 5 significant mistakes. | The algorithm provided solves the problem, and is based on other parts of the planning documentation; however there are more than 3, but no more than 5 significant mistakes. | The algorithm provided solves the problem, and is based on other parts of the planning documentation. There are no more than 3 significant mistakes. |
| Effective and correct use of variables (includes appropriate naming of variables) (20%) | Variables are inappropriately named and used | Variables are sometimes appropriately used, but their naming is unclear | The variables identifies are mostly appropriate and correctly used, but there are more than 5 significant errors | The variables identified are mostly appropriate and correctly used, but there are more than 3, but no more than 5 significant errors | The variables identified are appropriate and correctly used. There are no more than 3 significant errors. |
| Effective and correct use of functions  (20%) | No functions are identified | The functions identified are mostly inappropriate or incorrectly used | The functions identified are mostly appropriate and correctly used, but there are more than 5 significant errors | The functions identified are mostly appropriate and correctly used, but there are more than 3, but no more than 5 significant errors | The functions identified are appropriate and correctly used. There are no more than 3 significant errors |
| Effective and correct use of conditionals [if statements (20%) | No conditions are identified | The conditionals identified are mostly inappropriate or incorrectly used | The conditionals identified are mostly appropriate and correctly used, but there are more than 5 significant errors | The conditional identified are mostly appropriate and correctly used, but there are more than 3 but no more than 5 significant errors | The conditionals identified are appropriate and correctly used. There are no more than 3 significant errors. |
| Effective and correct use of loops  (20%) | No loops are identified | The loops identified are mostly inappropriate or incorrectly used | The loops identified are mostly appropriate and correctly used, but there are more than 5 significant errors | The loops identified are mostly appropriate and correctly used, but there are more than 3 but no more than 5 significant errors | The loops identified are appropriate and correctly used. There are no more than 3 significant errors. |
| Effective and correct use of comments  (10%) | No comments are listed | Comments are mostly inappropriate or incorrectly used | Comments are generally placed appropriately, but there are more than 5 significant errors | The comments are generally placed appropriately but there are more than 3, but no more than 5 significant errors | The comments are appropriate and correctly used. There are no more than 3 significant errors. |

Example of program interaction and output. Program output is in **Bold**. User input is in **Red**

**What is your name? Bob**

**Welcome Bob.**

**Tropical Airlines Ticket Ordering System**

**(I)nstructions**

**(O)rder ticket**

**(E)xit**

U

**Invalid menu choice.**

**Tropical Airlines Ticket Ordering System**

**(I)nstructions**

**(O)rder ticket**

**(E)xit**

I

**Thank you for choosing Tropical Airlines for your air travel needs. You will be asked questions regarding what type of ticket you would like to purchase as well as destination information. We also offer 50% discounted fares for children.**

**Tropical Airlines Ticket Ordering System**

**(I)nstructions**

**(O)rder ticket**

**(E)xit**

O

**Bob, is this ticket for:**

**(Y)ou**

**(S)omeone else**

S

**Please enter the name of the person travelling**

Darth Vader

**Is this a return trip (R) or One-Way (O)**

R

**Please select the destination for your return trip. Fare prices are listed below.**

**(C)airns – $400**

**(S)ydney – $575**

**(P)erth - $700**

S

**Please choose the type of fare. Fees are displayed below and are in addition to the basic fare.   
Please note choosing Frugal fare means you will not be offered a seat choice.**

**(B)usiness - $275**

**(E)conomy - $25**

**(F)rugal - $0**

E

**Please choose the seat type. Choosing the middle seat will deduct 25 from the total fare.**

**(W)indow $75**

**(A)isle $50**

**(M)iddle -$25**

W

**How old is the person travelling? Travellers under 16 years old will receive a 50% discount for the child fare.**

22

**Calculating fare . . .**

**Ticket for: Darth Vader**

**Sydney (return) - $575**

**Economy - $25**

**Window - $75**

**Age - 22 (not eligible for child ticket)**

**Total price: $675**

**Tropical Airlines Ticket Ordering System**

**(I)nstructions**

**(O)rder ticket**

**(E)xit**

O

**Bob, is this ticket for:**

**(Y)ou**

**(S)omeone else**

Y

**Is this a return trip (R) or One-Way (O)**

**O**

**Please select the destination for your return trip. Fare prices are listed below.**

**(C)airns - $250**

**(S)ydney - $420**

**(P)erth - $510**

P

**Please choose the type of fare. Fees are displayed below and are in addition to the basic fare.   
Please note choosing Frugal fare means you will not be offered a seat choice.**

**(B)usiness - $275**

**(E)conomy - $25**

**(F)rugal - $0**

F

**How old is the person travelling. Travellers under 16 years old will receive a 50% discount for the child fare.**

12

**Calculating fare . . .**

**Ticket for: Bob**

**Perth (one-way) - $575**

**Frugal - $0**

**Window - $0**

**Age - 12 (eligible for child ticket, 50% discount applied)**

**Total price: $287.50**

**Tropical Airlines Ticket Ordering System**

**(I)nstructions**

**(O)rder ticket**

**(E)xit**

E

**Bob, your orders are: $287.50 and $675. Your final total is: $962.50. Thank you for visiting Tropical Airlines.**