|  |  |
| --- | --- |
| **Programming 1 (PRG1)**  Diploma in IT / DS / CSF / IM / CICTP  Year 1 (2023/24) Semester 1 | Week **13** |
|  |
| **Exercise 9: Dictionaries & Exception Handling** | |

**OBJECTIVES**

At the end of this exercise, students should be able to:

* understand the use of dictionaries
* understand what are exceptions
* handle exceptions

**IMPORTANT**

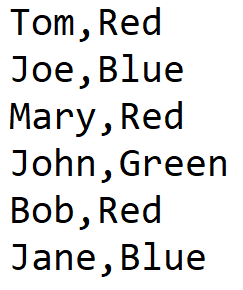
* Create a folder, **Week13**, in your hard disk.
* For programming questions, create Python programs with the given file names in the **Week13** folder created above. Do add the description, your name and student ID as comments at the beginning of each program.
* For non-programming questions, type your answers in the boxes provided below the questions.
* At the end of the session, compress all the files in your **Week13** folder (i.e. the Python program files and this word document) and submit the zip file in POLITEMall.

**Part 1: Dictionaries**

**Activity 1**

Invert dictionary – ( file name: Invert.py )

Create the following text file using Notepad. Save the text file as colors.csv:



Write a Python program Invert.py to

* Read the data from colors.csv and save the name as the key and the color as the value into a dictionary colors\_dict.
* Invert the dictionary colors\_dict to another dictionary new\_colors where the keys are values from color\_dict and the values are lists of keys from color\_dict
* Display new\_colors

**Sample Output:**

****

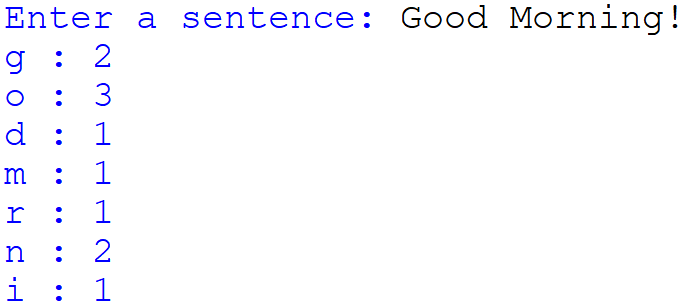
**Activity 2**

Count Letters – ( file name: CountLetters.py)

Write a Python program CountLetters.py to

* Prompt user to input a sentence
* Count the occurrence of each letter in the input sentence
  + Hint: You can use the isalpha() function
* Save the result to a dictionary letters where the key is the letter and the value is the occurrence of the key
* Display letters in alphabetical order of the key.

**Sample output:**



**Activity 3**

Place order in restaurant ( file name: PlaceOrders.py )

* The price list for the items in a restaurant is given in the following dictionary:

prices = {'chicken' : 8.50,\

'steak' : 13.80,\

'fish' : 9.80,\

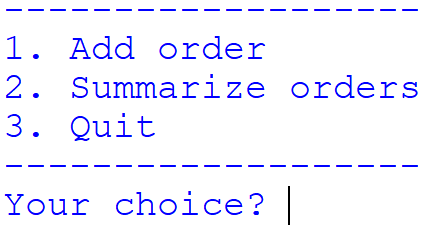
'pasta' : 9.50,\

'coffee' : 2.50,\

'tea' : 1.80,\

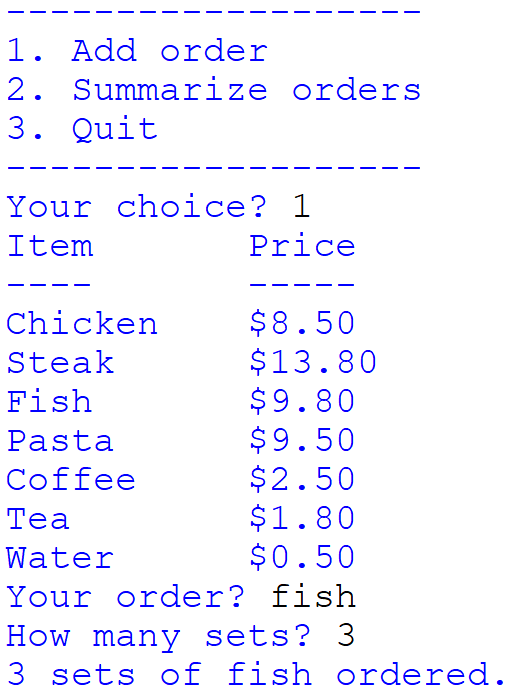
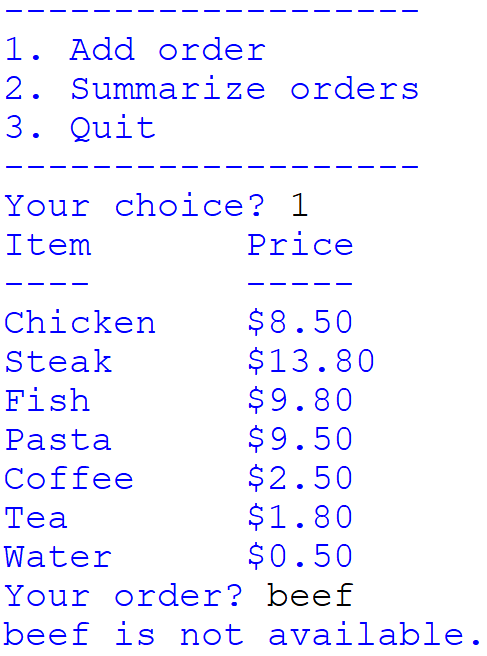
'water' : 0.50}

* Write a Python program that allows users to place orders with the restaurant.
* The program begins by showing the following menu:



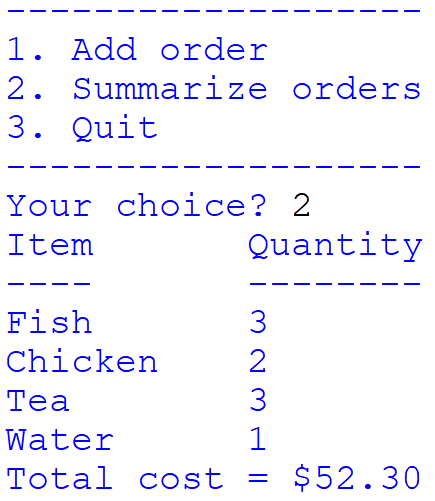
* **Add Order**

If the user selects 1 (Add Order), the program will show the list of food items available and their prices, then prompt the user for the item to order. If the item exists, the program will prompt for the number of sets to order; if it does not exist, an error message will be printed.

* **Summarize Orders**

If the user selects 2 (Summarize Orders), the program shows a summary of the orders made so far.



* **Quit**

If the user selects 3, the program ends.

**Part 2: Exception Handling**

**Activity 4**

Given list = [['John',4],['Mary',6]] and sum = 0, what exception will the following code raise?

|  |  |
| --- | --- |
| **Python code** | **Output** |
| print(list[2]) | IndexError |
| for i in list:  sum += list[i] | TypeError |
| for i in range(len(list)):  sum += 5 % i | ZeroDivisionError |
| for i in list:  Sum += i[1] | NameError |
| import maths  for i in range(5):  sum += maths.sqrt(i) | ModuleNotFoundError |
| import math  for i in range(5):  sum += math.sqrt(i-1) | ValueError |

**Activity 5**

The following program reads a text file containing student-marks pairs separated by commas, and computes the average marks.

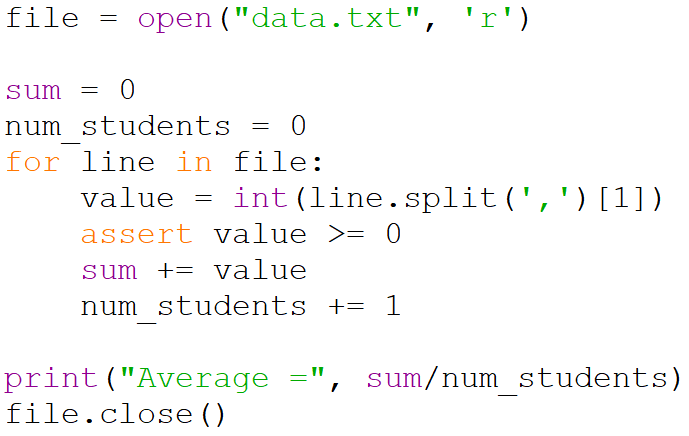
John,80

Mary,56

Bill,87

Samantha,92

Example data.txt:



What are possible reasons for getting the following error messages after running this program?

|  |  |
| --- | --- |
| **Error message** | **Possible reason** |
| Traceback (most recent call last):  File "D:\lecture\_codes.py", line 1, in <module>  file = open("data.txt", 'r')  FileNotFoundError: [Errno 2] No such file or directory: 'data.txt' | File does not exist in working directory |
| Traceback (most recent call last):  File "D:\lecture\_codes.py", line 6, in <module>  value = int(line.split(',')[1])  IndexError: list index out of range | The file line was not properly formatted/is not valid data (empty line) |
| Traceback (most recent call last):  File "D:\lecture\_codes.py", line 6, in <module>  value = int(line.split(',')[1])  ValueError: invalid literal for int() with base 10: 'Mary' | Bad data in data.txt |