Eliana Arias-Dotson
Created: 2020, February 25
Final Editing: March 2nd 2019
Foundations of Programming, Python
Assignment 06

Introduction

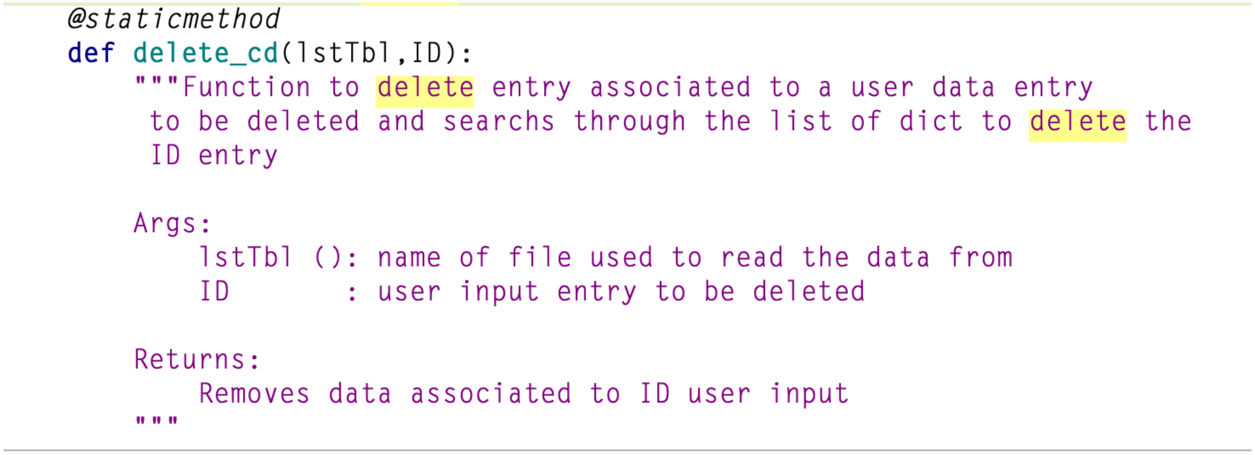
On this module we learned about concepts like functions,parameters (or arguments), variables as arguments, return values and how to incorporate routines into functions that can be used . Documentation of the code was also implemented using docstrings which helps not only as a document for other users of the code but also as a mapping to define the code

Classes and Functions (Using static Methods)

A class is a code template for creating objects. An object is created using the constructor of the class. This object will then be called the `instance` of the class [1]. In programming, a **function** is a self-contained block of code that encapsulates a specific task or related group of tasks where is important to know what **arguments** (if any) it take and what **values** (if any) it returns. For this assignment we learned abotu static methods (@staticmethod) which is one of several methods to implement when defining methods.This approach seemed to be a simpler way to incorporate the use of methods to codes previously created .Static method can neither modify object state nor class state. Static methods are restricted in what data they can access - and they're primarily a way to namespace your methods.[2,3]

Docstrings

In this module we learned about document strings in python (docstrings) which are designed to provide information associated to the classes, functions, and methods.([4])



```
    @staticmethod
    def delete_cd(lstTbl,ID):
        """Function to delete entry associated to a user data entry
         to be deleted and searchs through the list of dict to delete the
         ID entry

        Args:
            lstTbl (): name of file used to read the data from
            ID       : user input entry to be deleted

        Returns:
            Removes data associated to ID user input
        """
```

Figure 1. Example of a docstring used on CD_Inventory.py documenting the static method delete_cd()

[1] https://www.hackerearth.com/practice/python/object-oriented-programming/classes-and-objects-i/tutorial/
[2] https://realpython.com/instance-class-and-static-methods-demystified/#instance-class-and-static-methods-an-overview
[3] https://realpython.com/preview/defining-your-own-python-function/#functions-in-python
[4] https://www.pythonforbeginners.com/basics/python-docstrings

Parts of the Code

When working with classes and methods it is recommended to defined them before the main section of the program so we changed the initial approach of having
Data
Main Processing
Presentation of results
to
Processing - Classes and Functions Definitions
Input/Ouput and Main Processing

The provided script was defined using three classes and several predefined methods

```
# 1 CLASS
    class DataProcessor:”” A set of functions to load, add and delete “””
    # 3.3.1 Ask user for new ID, CD Title and Artist
    # 3.3.2 Add item to the table

# SECOND CLASS
    class FileProcessor: """Processing the data to and from text file”""

@staticmethod
    def read_file -DONE

@staticmethod
    def write_file(file_name, table):
    # TODO Add code here

# 3rd CLASS
    class IO:
    #THIR CLASS FUNCTION 1 - DEFINED
    @staticmethod
    def print_menu():
    #THIR CLASS FUNCTION 2 - DEFINED
    @staticmethod
    def menu_choice():
    #THIR CLASS FUNCTION 3 - DEFINED
    @staticmethod
    def show_inventory(table):
    # TODO add I/O functions as needed
    # Input function
```

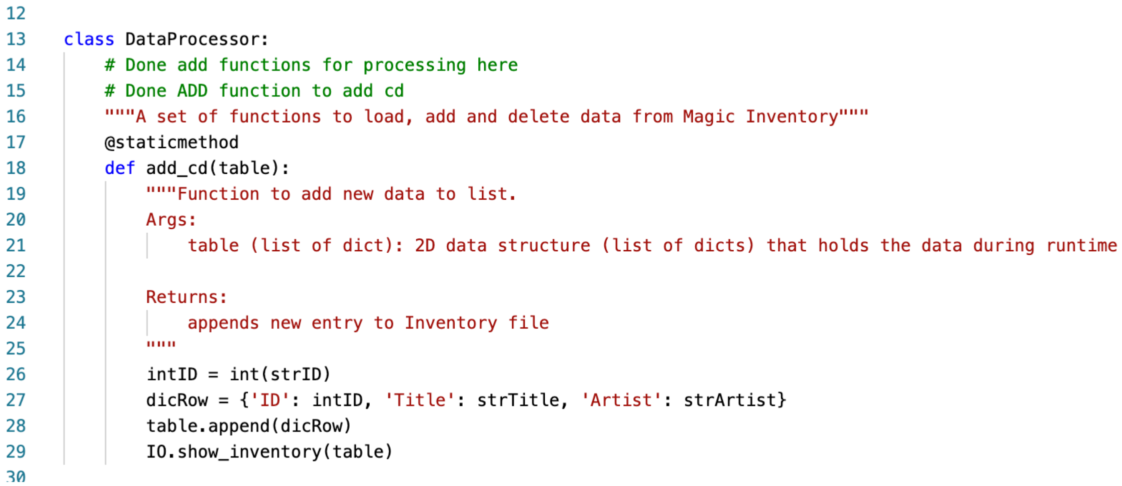
(1) https://www.hackerearth.com/practice/python/object-oriented-programming/classes-and-objects-i/tutorial/
(2) https://realpython.com/instance-class-and-static-methods-demystified/#instance-class-and-static-methods-an-overview
(3) https://realpython.com/preview/defining-your-own-python-function/#functions-in-python
(4) https://www.pythonforbeginners.com/basics/python-docstrings

Defining the static methods to ADD and Delete Entries

# 1 CLASS

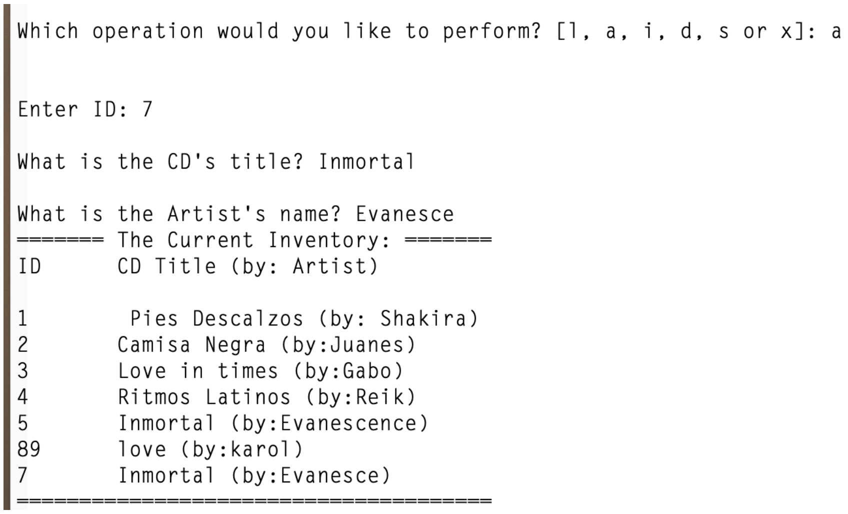
class DataProcessor:"" A set of functions to add and delete """

For this part of the code I thought it was needed to add functions to add the data and delete the data. for this as the other functions I simply defined the function based on the code provided in the processing.



```
class DataProcessor:
    # Done add functions for processing here
    # Done ADD function to add cd
    """A set of functions to load, add and delete data from Magic Inventory"""
    @staticmethod
    def add_cd(table):
        """Function to add new data to list.
        Args:
            table (list of dict): 2D data structure (list of dicts) that holds the data during runtime

        Returns:
            appends new entry to Inventory file
        """
        intID = int(strID)
        dicRow = {'ID': intID, 'Title': strTitle, 'Artist': strArtist}
        table.append(dicRow)
        IO.show_inventory(table)
```

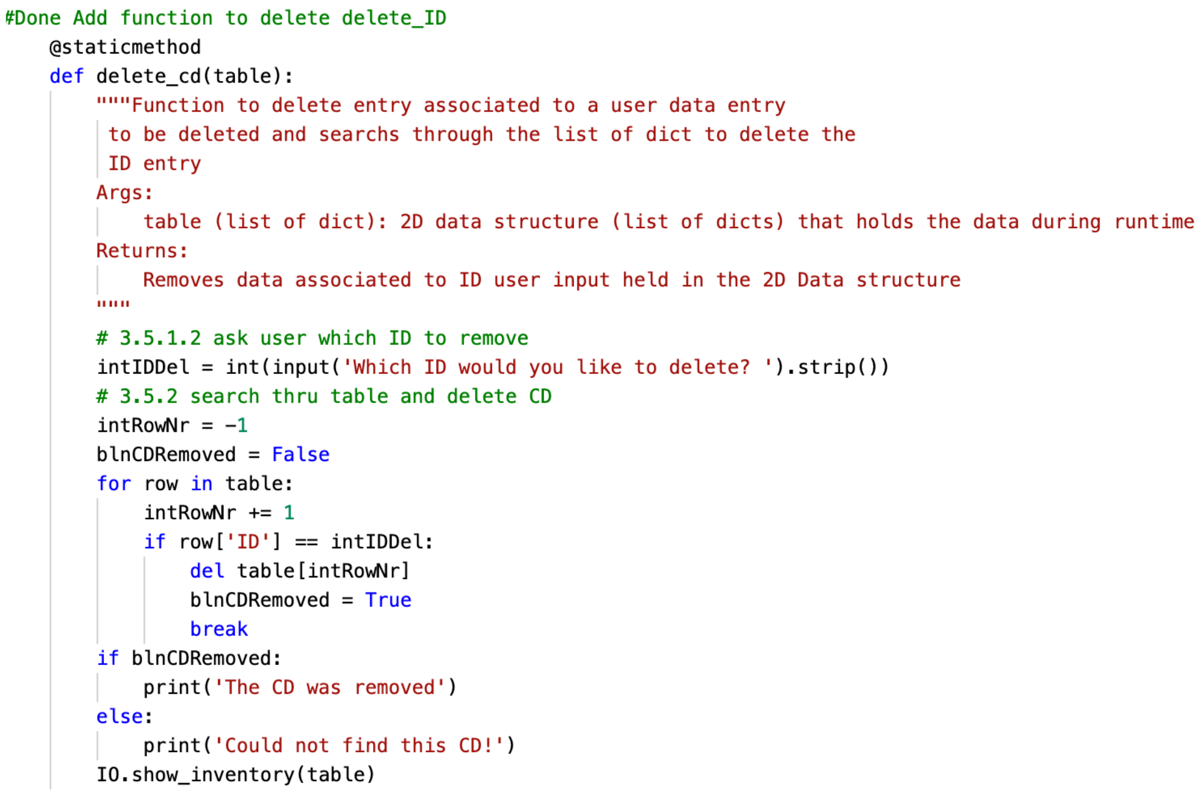
Figure 1a. Definition of function add_cd for the Data Processor Class



```
Which operation would you like to perform? [l, a, i, d, s or x]: a


Enter ID: 7

What is the CD's title? Inmortal

What is the Artist's name? Evanesce
======= The Current Inventory: =======
ID      CD Title (by: Artist)

1        Pies Descalzos (by: Shakira)
2       Camisa Negra (by:Juanes)
3       Love in times (by:Gabo)
4       Ritmos Latinos (by:Reik)
5       Inmortal (by:Evanescence)
89      love (by:karol)
7       Inmortal (by:Evanesce)
======================================
```

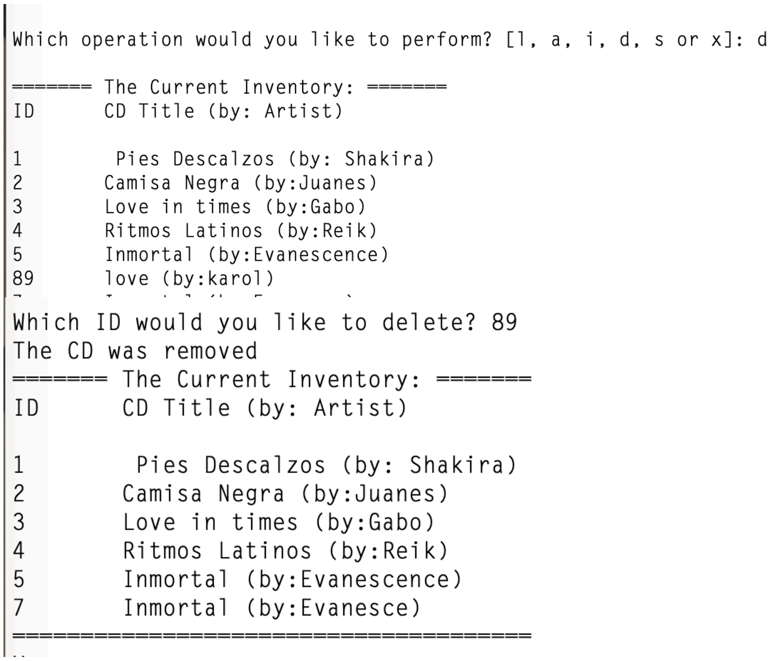
Figure 1a. Output when adding a new entry

(1) https://www.hackerearth.com/practice/python/object-oriented-programming/classes-and-objects-i/tutorial/
(2) https://realpython.com/instance-class-and-static-methods-demystified/#instance-class-and-static-methods-an-overview
(3) https://realpython.com/preview/defining-your-own-python-function/#functions-in-python
(4) https://www.pythonforbeginners.com/basics/python-docstrings



```python
#Done Add function to delete delete_ID
    @staticmethod
    def delete_cd(table):
        """Function to delete entry associated to a user data entry
         to be deleted and searchs through the list of dict to delete the
         ID entry
        Args:
            table (list of dict): 2D data structure (list of dicts) that holds the data during runtime
        Returns:
            Removes data associated to ID user input held in the 2D Data structure
        """
        # 3.5.1.2 ask user which ID to remove
        intIDDel = int(input('Which ID would you like to delete? ').strip())
        # 3.5.2 search thru table and delete CD
        intRowNr = -1
        blnCDRemoved = False
        for row in table:
            intRowNr += 1
            if row['ID'] == intIDDel:
                del table[intRowNr]
                blnCDRemoved = True
                break
        if blnCDRemoved:
            print('The CD was removed')
        else:
            print('Could not find this CD!')
        IO.show_inventory(table)
```

Figure 2a. Definition of function to delete data



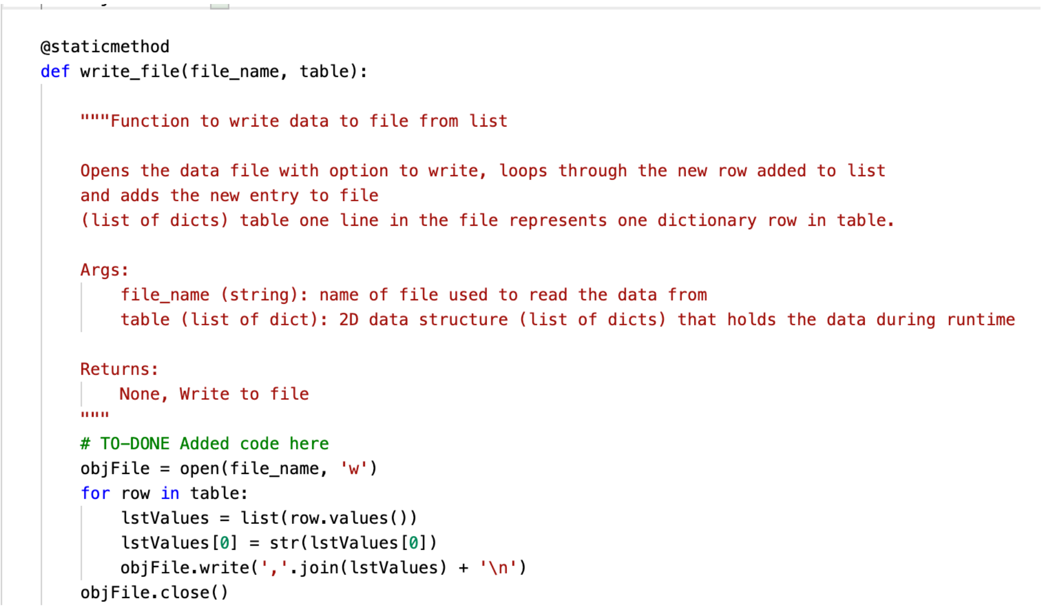
```
Which operation would you like to perform? [l, a, i, d, s or x]: d

======= The Current Inventory: =======
ID      CD Title (by: Artist)

1        Pies Descalzos (by: Shakira)
2       Camisa Negra (by:Juanes)
3       Love in times (by:Gabo)
4       Ritmos Latinos (by:Reik)
5       Inmortal (by:Evanescence)
89      love (by:karol)
Which ID would you like to delete? 89
The CD was removed
======= The Current Inventory: =======
ID      CD Title (by: Artist)

1        Pies Descalzos (by: Shakira)
2       Camisa Negra (by:Juanes)
3       Love in times (by:Gabo)
4       Ritmos Latinos (by:Reik)
5       Inmortal (by:Evanescence)
7       Inmortal (by:Evanesce)
======================================
```

Figure 2b. Output in screen when deleting entry.

(1) https://www.hackerearth.com/practice/python/object-oriented-programming/classes-and-objects-i/tutorial/
(2) https://realpython.com/instance-class-and-static-methods-demystified/#instance-class-and-static-methods-an-overview
(3) https://realpython.com/preview/defining-your-own-python-function/#functions-in-python
(4) https://www.pythonforbeginners.com/basics/python-docstrings

Defining the static methods to Read and Write to File for the Class File Processor

The static method to read file was provided and it serves as an example on how to incorporate the code into a function.



```
@staticmethod
def write_file(file_name, table):

    """Function to write data to file from list

    Opens the data file with option to write, loops through the new row added to list
    and adds the new entry to file
    (list of dicts) table one line in the file represents one dictionary row in table.

    Args:
        file_name (string): name of file used to read the data from
        table (list of dict): 2D data structure (list of dicts) that holds the data during runtime

    Returns:
        None, Write to file
    """
    # TO-DONE Added code here
    objFile = open(file_name, 'w')
    for row in table:
        lstValues = list(row.values())
        lstValues[0] = str(lstValues[0])
        objFile.write(','.join(lstValues) + '\n')
    objFile.close()
```

Figure 3. Definition of write_file method

Class Defining Input/Ouput on the script

The functions to print_menu(), display the menu_choices() and show_inventory() were provided in the starter code. I considered adding a get_UserInput but did had touble getting it to work.

Main code/PROCESSING

This part of the code basically calls of the functions assigning the values defined for the data (Appendix 1 - full code)

Take from this assignment: Coding has been for me like when I have been learning other languages (Russian, portuguess, greek,etc). I used to pass my written test but when it was time to speak them I got completely blocked. It takes for me a lot of practice and building confidence to achieve fluency and it seems like learning python is no different. I need to read, and read again, and split every concept to assimilate the purpose of what needs to be accomplished. I am aware this is perhaps not what is expected but I decided to submit the little bit I understand.

GitHub: https://github.com/elidot/Assignment06.git

(1) https://www.hackerearth.com/practice/python/object-oriented-programming/classes-and-objects-i/tutorial/
(2) https://realpython.com/instance-class-and-static-methods-demystified/#instance-class-and-static-methods-an-overview
(3) https://realpython.com/preview/defining-your-own-python-function/#functions-in-python
(4) https://www.pythonforbeginners.com/basics/python-docstrings

Appendix 1.

```
# MAIN PROCESSING
# -- DATA -- #
strChoice = '' # User input
lstTbl = []  # list of lists to hold data
dicRow = {}  # dictionary of data row
strFileName = 'CDInventory.txt'  # data storage file
objFile = None  # file object
strID =None
strTitle = None
strArtist= None

# 1. When program starts, read in the currently saved Inventory
FileProcessor.read_file(strFileName, lstTbl)

# 2. start main loop
while True:
    # 2.1 Display Menu to user and get choice
    IO.print_menu()
    strChoice = IO.menu_choice()

    # 3. Process menu selection
    # 3.1 process exit first
    if strChoice == 'x':
        break
    # 3.2 process load inventory
    if strChoice == 'l':
        print('WARNING: If you continue, all unsaved data will be lost and the
Inventory re-loaded from file.')
        strYesNo = input('type \'yes\' to continue and reload from file. otherwise
reload will be canceled. ')
        if strYesNo.lower() == 'yes':
            print('reloading...')
            FileProcessor.read_file(strFileName, lstTbl)
            IO.show_inventory(lstTbl)
        else:
            input('canceling... Inventory data NOT reloaded. Press [ENTER] to continue
to the menu.')
            IO.show_inventory(lstTbl)
        continue  # start loop back at top.
```

(1) https://www.hackerearth.com/practice/python/object-oriented-programming/classes-and-objects-i/tutorial/
(2) https://realpython.com/instance-class-and-static-methods-demystified/#instance-class-and-static-methods-an-overview
(3) https://realpython.com/preview/defining-your-own-python-function/#functions-in-python
(4) https://www.pythonforbeginners.com/basics/python-docstrings

```
elif strChoice == 'a':
        # 3.3.1 Ask user for new ID, CD Title and Artist
        # TODO - NOt working moved IO code into function
        strID = input('Enter ID: ').strip()
        strTitle = input('What is the CD\'s title? ').strip()
        strArtist = input('What is the Artist\'s name? ').strip()
        # 3.3.2 Add item to the table
        # Done moved processing code into function
        DataProcessor.add_cd(lstTbl)
        continue # start loop back at top.
    # 3.4 process display current inventory
    elif strChoice == 'i':
        IO.show_inventory(lstTbl)
        continue  # start loop back at top.
    # 3.5 process delete a CD
    elif strChoice == 'd':
        # 3.5.1 get Userinput for which CD to delete
        # 3.5.1.1 display Inventory to user
        IO.show_inventory(lstTbl)
        # 3.5.1.2 ask user which ID to remove
        intIDDel = int(input('Which ID would you like to delete? ').strip())
        # 3.5.2 search thru table and delete CD
        # Done, moved processing code into function
        DataProcessor.delete_cd(lstTbl)
        continue  # start loop back at top.
    # 3.6 process save inventory to file
    elif strChoice == 's':
        # 3.6.1 Display current inventory and ask user for confirmation to save
        IO.show_inventory(lstTbl)
        strYesNo = input('Save this inventory to file? [y/n] ').strip().lower()
        # 3.6.2 Process choice
        if strYesNo == 'y':
            # 3.6.2.1 save data
            # DONE moved processing code into function
            FileProcessor.write_file(strFileName, lstTbl)
        else:
            input('The inventory was NOT saved to file. Press [ENTER] to return to the
menu.')
        continue  # start loop back at top.
    # 3.7 catch-all should not be possible, as user choice gets vetted in IO, but to be save:
    else:
        print('General Error')
```

(1) https://www.hackerearth.com/practice/python/object-oriented-programming/classes-and-objects-i/tutorial/
(2) https://realpython.com/instance-class-and-static-methods-demystified/#instance-class-and-static-methods-an-overview
(3) https://realpython.com/preview/defining-your-own-python-function/#functions-in-python
(4) https://www.pythonforbeginners.com/basics/python-docstrings