INFORMATICS INSTITUTE OF TECHNOLOGY

In collaboration with

UNIVERSITY OF WESTMINSTER

Enhanced Personal Finance Tracker (GUI Implementation with Tkinter and OOP)

Module: Software Development 1

An Individual Assignment by

Mr. G.S.Karnuathilaka

Supervised by

Mr. Guhanathan Poravi

April 29, 2024

IIT Registration No: 20231659 **UOW Student ID:** W2084412

Abstract

Personal Finance Tracker is a python based application designed to assist users in managing their financial transactions efficiently. It is developed with the use of Tkinter Graphical User Interface principles and Objet Oriented Programming principles. The application will support CRUD operations in order to manage financial transactions employing JSON for data storage. This application includes a searching and sorting feature of transactions for ease access. The data storage is handle only by using python Dictionaries

Acknowledgement

This is meant to convey the author's heartfelt gratitude for all the numerous individuals who have provided encouragement & assistance all over every step of this course work.

Special thanks to the module leader (Mr. Guhanathan Poravi) & the rest of academic staff of the IIT who supported to completion of this course work successfully.

Finally this is meant to convey the author's heartfelt gratitude to all the family members & loved ones who provided their immense support to complete this course work successfully

Table of Contents

List of Figures

List of Glossaries

Abbreviation	Acronym
CRUD	Create, Read, Update, Delete
OOP	Object Oriented Programming
GUI	Graphical User Interface
JSON	Java Script Object Notation
I/O	Input / Output

CHAPTER 01 INTRODUCTION

1.1 Project Scope

The scope of this assignment can be categorized into in-scope which contains the inclusions of the assignment and out scope which contains exclusions if the assignment

1.1.1 In-scope

The in-scope of the assignment can be described as follows:

- Developing a personal finance tracker using python programming language.
- Utilizing Tkinter to crate user interface (GUI)
- Implementing OOP principles
- Supporting CRUD operations for managing Financial Transactions.
- Utilizing JSON serialization for data storage.
- File operations for bulk reading of transactions from a text file.
- Introduce searching and sorting transactions

Ensuring program robustness through comprehensive testing and validation.

1.1.2 Out-scope

The out-scope of the assignment can be further defined as follows:

- Implementing advanced python data structures beyond python dictionaries.
- Implementing an advanced GUI features.
- Integrating with external financial systems.
- Incorporating third-party tools for data management.
- Implementing security features for the application.

1.2 Problem Statement

Building on your knowledge of Python, dictionaries, and file I/O, your next challenge is to enhance the Personal Finance Tracker by developing a graphical user interface (GUI) using Tkinter. This advanced version should not only display the information from a provided JSON file but also

incorporate object-oriented programming (OOP) concepts for the GUI components. Additionally, your application will include a search function and a sorting feature, similar to a file explorer, to manage and analyze financial transactions more effectively.

CHAPTER 02 DESIGN

2.1 Design Decisions

- The GUI is designed to be intuitive and easy to use.
- The GUI used clear labels and buttons to keep the user interactions simple to avoid the confusion
- Different colors and font styles were used for increasing user-friendliness and user-attraction.
- This was designed by separating components such as Transaction Search, and transaction sort by each category.
- The personal finance tracker was designed to handle the errors gracefully such as invalid user inputs and file not found errors and designed to display a clear error message to the user indicating the error.
- The transaction data was designed to save permanently in a JSON file using Nested Dictionaries and Nested Lists.

2.2 Class Structures

- The program was put inside a class
- Inside the class, different methods have implemented for,
    1. Graphical user interface components
    2. Searching and Filtering Transactions
    3. Sorting Transactions

CHAPTER 03 IMPLEMENTATION

3.1 System Requirements

To ensure optimal performance smooth operation of the personal finance tracker, ensure that your system meets the following requirements

01. Operating System:
    - The Personal Finance Tracker is compatible with all major operating systems including Windows, MacOS
02. Python Version:
    - The application requires Python 3.x to be installed on your system. Ensure that you have installed the latest version of python to experience the full functionality of the Personal Finance Tracker.
03. Requirements:
    - No additional python libraries are required.
04. Hardware Requirements
    - The personal finance tracker can run smoothly and perform operations on most modern computers and laptops without any hardware constraints

3.2 Installation Instructions

01. Python Installation
    - If python is not already installed on your system, download and install the latest
    - Version of python from the official python website. Follow the installation
    - instructions provided for your operating system

    Download Python

02. Download the personal finance tracker
03. Download the source code for the personal finance tracker from the provided source. Extract the contents of the downloaded zip file to a convenient location on the system
04. Run the application
    - Navigate to the directory where you extracted the source code for the personal finance tracker. Double Click on the source code to start the application.
05. Follow the on-screen instructions to perform financial operations.

06. The entered and recorded transactions will be stored in the extracted JSON file.

3.3 Feature Descriptions and Functionality of GUI

3.3.1 Main Menu

- The main menu presents users with a user-friendly interface that displays the available options for managing transactions
- Users can navigate through different functionalities by entering the corresponding number which represents each functionality from the main menu.
- There are six options for file management including the exit option. They are as follows
  01. Add Transaction – Creates a new record
  02. View Transaction – A User friendly GUI which users are permitted to search through the entire transactions and sorting the transactions
  03. Update Transaction – Update the details of a selected record
  04. Delete Transaction – Delete the details of the selected record
  05. Display Summary - A summary of all the transactions.
  06. Read Bulk Transaction – Reading bulk transactions from a text file
  07. Exit – Exiting the program.
- The main menu feature consists of input validation mechanisms to ensure that user inputs are accurate and follow the specified date formats
- The main menu feature consists of proper error handling mechanisms to ensure that the users are guided in case of an invalid inputs and unexpected errors ensuring the smooth flow of the application.

3.3.2 Adding a Transaction

- This feature enables users to enter their transaction details within the personal finance tracker application

- This feature prompts users to enter the details related to transaction including,
    - i Transaction amount
    - ii Transaction Category
    - iii Transaction Type
    - iv Transaction Date

```
    Personal Finance Tracker

1. Add Transaction
2. View Transactions
3. Update Transaction
4. Delete Transaction
5. Display Summary
6. Read Bulk Transactions
7. Exit

Enter your choice:
```

Figure 1: Main Menu

- Input validation and error handling mechanisms ensure that the users' inputs are accurate and follow the specified date format.
- When users are providing the input for the Transaction Type, the system will only accept one from the two categories; income and expense. This reduces the complexity of the transaction.
- After entering the necessary information, users are prompted to confirm whether they want to save their transactions. This ensure the control over the recording process.
- When the users confirm to save their transaction, the application stores the entered transaction details in its internal data structure which is a nested list.
- Each transaction includes relevant information such as amount, transaction category, transaction type, transaction date and a unique transaction ID which is generated by the application
- The stored transaction details are then writes to an external file for permanent storage
- Sample format of the external file is shown below

Figure 2: JSON file format

- In this format:
    - i The first element is the Key Value (Transaction Category).
    - ii The second element is the Amount.
    - iii The third element is the date (formatted as "YYYY-MM-DD")
    - iv The fourth element is the type ("Income" or "Expense").
    - v The fifth element is the unique transaction ID which starts from one
- Upon successfully adding a transaction, users receive feedback confirming the successful addition of the transaction.

3.3.3 GUI Interface

- The GUI interface presents users with a user-friendly interface that displays the available transactions in a tabular form and options for searching transactions based on different criteria.
- Users can perform searching functionalities by selecting the relevant options from the search menu in the GUI interface.

3.3.3.1 Architecture of the GUI

- The architecture of the main interface is displayed below

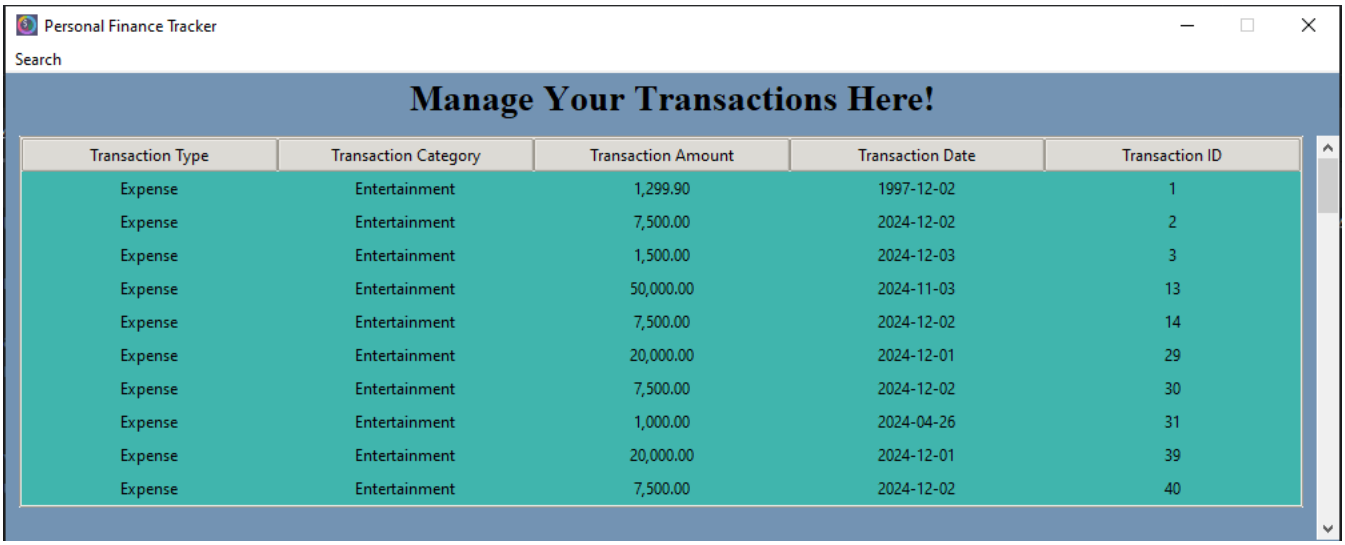

Figure 3: GUI Interface

- All the transaction details were taken from the JSON file and displayed on to the screen in a tabular form. There are five columns in the table. They are listed below.
    1. Transaction Type - This column represents the transaction type (Income or Expense) of all transactions. Transactions belongs to Income and Expenses are represented using two separate colors for ease identification.
    2. Transaction Category – This column represents the transaction category of all transactions.
    3. Transaction Amount – This column represents the values of the relevant transaction category of all transactions.
    4. Transaction Date - This column represents the date of the relevant transaction category of all transactions.
    5. Transaction ID – This column represents the uniquely generated ID for each of the transaction.
- There is one menu in the menu bar. That is Search Menu. This allows users to search transactions by Date, Amount, Transaction Type and Transaction Category.

3.3.4 Update Transaction

- This feature enables users to update their transaction details of the existing transactions within the personal finance tracker application.

- This feature displays the existing transactions to user and prompts the transaction ID which they need to update.
- Upon entering the transaction ID, this prompts users to enter the details related to transaction to update including,
    01. Transaction amount
    02. Transaction Category
    03. Transaction Type
    04. Transaction Date in order
- Input validation and error handling mechanisms ensure that the users' inputs are accurate and follow the specified date format.
- When users are providing the input for the Transaction Type, the system will only accept one from the two categories; income and expense. This reduces the complexity of the transaction.
- After entering the necessary information, users are prompted to confirm whether they want to update their transactions.
- When the users confirm to update their transaction, the application replaces the previous existing details of the transaction with the newly entered transaction details in its internal data structure. But the transaction ID does not change
- The updated transaction details are loaded back again to the external file for permanent storage.

3.3.5 Delete Transaction

- This feature allows users to remove specific transactions from their financial records. This feature is essential for maintaining accurate , up-to-date financial data by removing irrelevant transactions
- Users may need to specify the transaction they wish to delete, usually by providing unique identifiers such as transaction IDs
- The system should validate user inputs to ensure that the selected transaction exists and can be deleted

- Upon confirmation, the selected transaction is removed from the system's records along with the transaction ID
- After successful deletion, the system should provide feedback to the user, confirming that the transaction has been deleted and possibly displaying an updated summary of transactions.
- The updated transaction details are loaded back again to external file for permanent storage

3.3.6. Display Summary

- This feature provides users a comprehensive overview of their financial transactions. This feature enables users to understand their income and expenses which is required for better financial management.
- This feature calculates the category total of the existing transaction categories.
- This feature calculates the total income by adding all the ‘income’ transactions recorded by users.
- This feature calculates the total expense by adding all the ‘expense’ transactions recorded by users
- This feature calculates the net balance by subtracting total expenses from total income. This provides users with an overview of their overall financial health, indicating whether they are saving or overspending

3.3.6 Read Bulk Transactions

- This feature allows user to add bulk transactions to the JSON file from a text file. The bulk transactions are recorded in the text file in a specific format

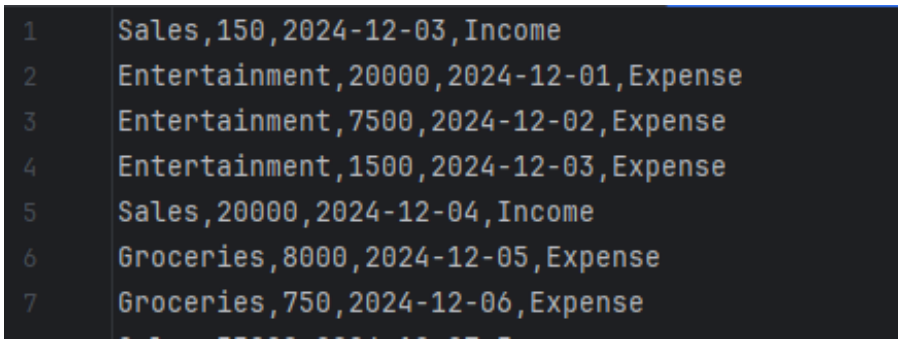

Figure 4: Text File Format

- In this format:

i The first element is the Key Value (Transaction Category).

ii The second element is the Amount.

iii The third element is the date (formatted as "YYYY-MM-DD")

iv The fourth element is the type ("Income" or "Expense").

- These values should be written by the user in a sequential order separated by commas. So in the text file, the details of the transaction should be in the below order

Transaction Category, Transaction Amount, Transaction Date, Transaction Type

- After selecting the relevant option, the system reads from the text file and added to the JSON file.
- After successful data addition to the JSON file, the system provides a message to the user, confirming that the transaction data has been added.

3.3.7 Search Features in GUI

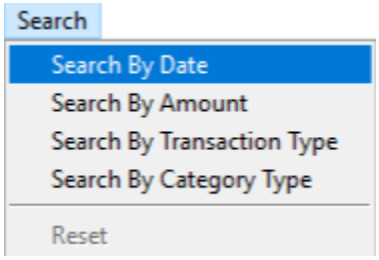

Figure 5: Search Menu

3.3.7.1 Search by Date Feature

- This Feature allows user to search records based on their date.
- The Search by date window pops up after selecting ‘Search By Date’ option from the ‘Search’ Menu.

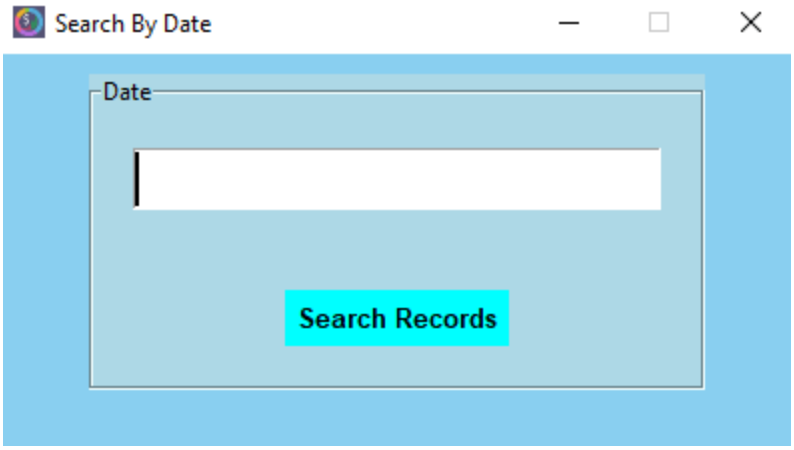

Figure 6: Search By Date Interface

- After entering a date as the input, system will search the records relevant to the entered input and displayed the records of the relevant transactions in the table.

3.3.7.2 Search by Amount Feature

- This Feature allows user to search records based on their date.
- The Search by date window pops up after selecting 'Search By Date' option from the 'Search' Menu.

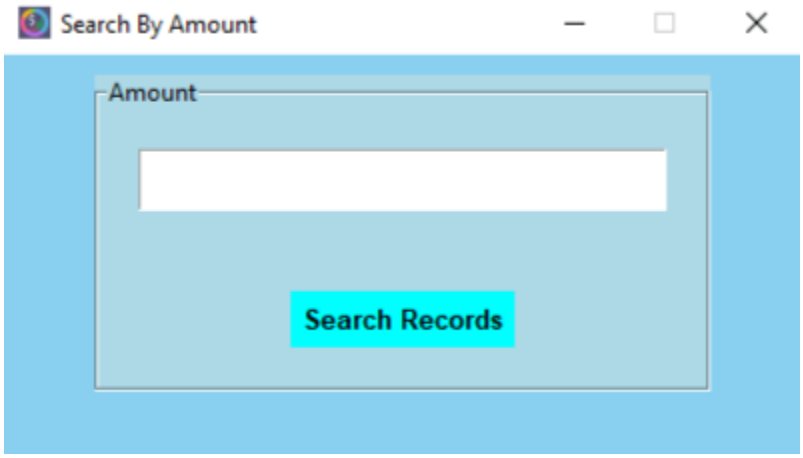

Figure 7: Search By Amount Interface

- After entering an amount as the input, system will search the records relevant to the entered input and displayed the records of the relevant transactions in the table.

3.3.7.3 Search by Transaction Type Feature

- This Feature allows user to search records based on their date.

- The Search by date window pops up after selecting ‘Search By Transaction Type’ option from the ‘Search’ Menu.

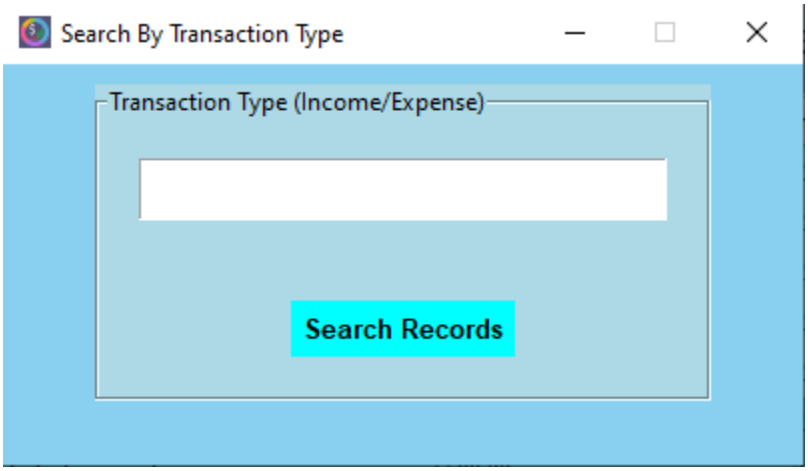

Figure 8: Search By Transaction Type Interface

- After entering a transaction type as the input, system will search the records relevant to the entered input and displayed the records of the relevant transactions in the table.

3.3.7.4 Search by Category Type

- This Feature allows user to search records based on their date.
- The Search by Category Type window pops up after selecting ‘Search By Category Type’ option from the ‘Search’ Menu.

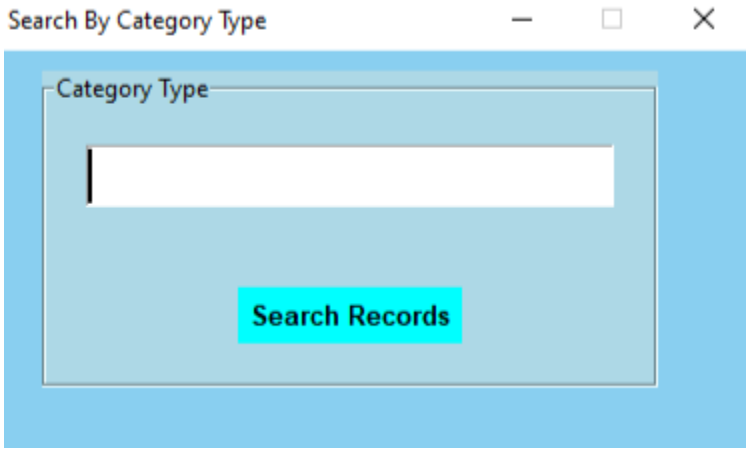

Figure 9: Search By Category Type

- After entering a category type as the input, system will search the records relevant to the entered input and displayed the records of the relevant transactions in the table.

3.3.8 Sorting Feature in GUI

- This allows users to organize the records and view the records in an ascending order and descending order making it easier to analyze.
- In this application, column sorting method has been implemented. By clicking on each column, the data will sort the data in ascending order by default.
- The transaction data will be sorted in descending order by clicking on each column again.