

# Documentation

## Personal\_data\_manager.py

The **`PersonalDataManager`** class offers a simple interface for managing personal data records. It allows you to load, save, add, filter, and display records in various formats.

- Imports: **json**, **yaml**, **glob** in order to handle serialization and be used for filtering syntax inside our filter method
- Explanation of each method and its arguments:
  - The **\_\_init\_\_** method initializes the class and accepts a **file\_name** as an argument, which is the file for storing our records.
  - The **load\_from\_file** method loads records from the specified file using the **json.load** function. In case the file has not been created, it creates a new empty dataset.
  - The **save\_to\_file** method data records to the specified file using the **json.dump** function.
  - The **add\_record** method has the job of adding a new record to the dataset. It accepts the name, address, and phone\_number as arguments.
  - The **filter\_records** method filters the records based on a search query. It returns a list of filtered record and accepts the match\_record as an argument.
  - The **match\_record** receives the search query and check if there is a matching record. Returns True/False based on the result.
  - The **serialize\_records** method serializes the records in the specified format. It supports the three formats ('json', 'yaml', 'csv') which are passed in as an argument and it can be extended to support more formats.
  - The **display\_records** method displays the records in the specified format. It accepts the format as an argument. it has been implemented to support the two formats ('text' and 'html').