**Pharmacy documentation**

The main class:

\* The Main class is the entry point of the Pharmacy Management System program.

\* It provides a menu-driven interface for managing drugs and generating reports.

public class Main

{

* The main method initializes the program and starts the execution.
* It creates an instance of the Drug Methods class and presents a menu to the user.

public static void main(String[] args)

{

// Program execution logic

// which takes the command from the user to start the program

* 1. add drug
* 2. remove drug
* 3. place order
* 4. get total sales
* 5. Exit
* And print them in the files

}

}

The Drug Data class:

\* The DrugData class represents the data structure for a drug.

\* It encapsulates the information related to a drug, such as its name, ID, price, quantity, and category.

public class DrugData

{

* name the name of the drug
* id the ID of the drug
* price the price of the drug
* quantity the quantity of the drug
* category the category of the drug
* Parameterized constructor Returns the initialized values

public DrugData(String name, int id, double price, int quantity, char category) {

* Initialize the DrugData object

}

public String getName() {

* Return the name of the drug

}

public int getId() {

* Return the ID of the drug

}

public double getPrice() {

* Return the price of the drug

}

public char getCategory() {

* Return the category of the drug

}

public int getQuantity() {

// Return the quantity of the drug

}

public void setQuantity(int quantity) {

// Set the quantity of the drug }

The Drug Methods class:

\* The Drug\_Methods class provides methods for managing drugs in a pharmacy.

\* It allows adding drugs, placing orders, getting total sales, and removing drugs.

public class Drug Methods {

* private int capacity;
* private double total\_sales = 0;
* Scanner input = new Scanner(System.in);
* ArrayList<DrugData> drugs = new ArrayList<>();

}

\* Constructs a Drug\_Methods object and prompts the user to input the capacity of the pharmacy.

\* The capacity represents the maximum number of drugs that can be stored in the pharmacy.

public Drug\_Methods()

{ Initialize the Drug\_Methods object and set the capacity of the pharmacy

}

public void add\_drug()

{ Logic for adding a drug

* Adds a drug to the pharmacy.
* It prompts the user to enter the details of the drug and adds it to the drugs list.
* The capacity of the pharmacy is updated, and the operation is recorded in the report.

}

public void place\_order()

{Logic for placing an order

* Places an order for a drug.
* It prompts the user to enter the ID of the drug and quantity.
* If the drug is available, the order is placed, and the total sales and drug quantity are updated.
* The operation is recorded in the report.

}

public void getTotalSales()

{ Logic for getting total sales

* Retrieves the total sales for today.
* It displays the total sales and records the operation in the report.

}

public void remove\_drug()

{ Logic for removing a drug

* Removes a drug from the pharmacy.
* It prompts the user to enter the ID of the drug to be removed.
* If the drug is found, it is removed from the drugs list, and the capacity is updated.
* The operation is recorded in the report.

}

private void writeToTextFile(String operation)

{ Logic for writing to the text file

* Writes an operation to the text file report.txt
* operation the operation to be written to the file

}

private void writeToBinaryFile(String operation)

{ Logic for writing to the binary file

* Writes an operation to the binary file report.dat.
* operation the operation to be written to the file

}