LAB 3 Exercise

TASK – 01:

If you are developing an online shopping cart system. Create a Java class called **CartItem** to represent items that users can add to their shopping cart. Each **CartItem** object should have the following attributes and methods:

Attributes:

- **productName (String)**
- **price (double)**
- **quantity (int)**

Methods:

- **calculateTotalPrice(): Calculates the total price of the items in the cart (price * quantity).**
- **displayCartItemDetails(): Displays the details of the cart item, including the product name, price, quantity, and total price.**

Use **Constructor** to intilize the attributes of class.

Now, in main method demonstrates the use of the CartItem class by creating instances of cart items, adding them to a shopping cart, calculating the total price, and displaying the cart item details.

TASK – 02:

A shopkeeper wants to maintain records for the items in his shop.

1. Create a class named as Item.
2. Public Member variables: Item name, Item ID Private Member Variables: Price, Stock
3. Create Accessor and Mutator for the private member variables.
4. A user will be able to search for items that the user wants to purchase. The price will be displayed for that particular item. Use set function to set the price of an item.
5. Set the items in stock using set function. If the user tries to purchase more items than present in the stock, then display a message that cannot purchase more items than available in stock. Display the total bill for the items purchased.

TASK – 03:

Write a Java class Book with following features:

- Instance variables:
    - **title** for the title of book of type String.
    - **author** for the author's name of type String.
    - **price** for the book price of type double.

- Constructor:
    - **public Book (String title, Author name, double price)**: A constructor with parameters, it creates the Author object by setting the the fields to the passed values.
- Instance methods:
    - **public void setTitle(String title)**: Used to set the title of book.
    - **public void setAuthor(String author)**: Used to set the name of author of book.
    - **public void setPrice(double price)**: Used to set the price of book.
    - **public double getTitle()**: This method returns the title of book.
    - **public double getAuthor()**: This method returns the author's name of book.
    - **public String toString()**: This method printed out book's details to the screen

Write a separate class **BookDemo** with a main () method creates a Book titled "Great Expectations" with author Charles Dickens and price 79.75.

TASK – 04:

Create a class called **Employee** that includes three pieces of information as instance variables

- a first name (type String)
- a last name (type String)
- a monthly salary (double)

If the monthly salary is not positive, set it to 0.0.

Write a test application named EmployeeTest that demonstrates class Employee's capabilities. Create two Employee objects and display each object's yearly salary. Then give each Employee a 10% raise and display each Employee's yearly salary again. Use appropriate methods for this program wherever needed.

TASK – 05:

Let you are developing a simple bank account management system. Create a Java class called BankAccount with the following attributes and methods:

Attributes:

- accountNumber (String)
- accountHolderName (String)
- balance (double)

Methods:

- deposit(double amount): Adds the given amount to the account's balance.
- withdraw(double amount): Subtracts the given amount from the account's balance, but ensures that the balance never goes negative.

Use appropriate setter and getter methods for the attributes of class

Now, create a program that demonstrates the use of the BankAccount class by creating an instance of it, depositing and withdrawing money, and displaying the account details.

TASK – 06:

Create a class Student with the following public fields:

1. Age: integer
2. Name: String
3. Course: String
4. In the Main method, create an object of class Student and set Name to "Your Name", Age to “Your Age” and course to “OOP in JAVA”.
5. Then, output to the screen: "My name is {Name}, I'm {Age} years old., I am studying “OOP in JAVA”" using object fields for Name, Age and Course.

TASK – 07:

An organization keeps a record of all the managers that are part of their organization.

1. The organization stores basic personal details such as the employee number, employee name, employee phone number and employee salary (private).
2. Create Accessor and Mutator for the private member variables.
3. The details such as the employee number, employee name, employee phone number, and salary are required as an input.
4. Display the details for three employees along with the employee with the highest salary.