Experiment - 2

Student Name: Hemant Narain Jha
UID: 23BCS10022
Branch: BE-CSE
Section/Group: KRG-2-B
Semester: 5th
Date of Performance: 23/7/25
Subject Name: PBLJ
Subject Code: 23CSH-304

Aim: : To develop Java programs to manage product details, library systems, andstudent information using classes, inheritance, and abstraction.

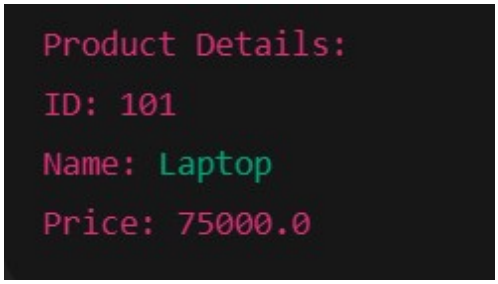
Easy-level Problem

Aim: To write a Java program to create a Product class with attributes id, name and price. The program should demonstrate the use of constructors and methods to display product details.

Code -

```
package exp2;
class Product {
int id;
String name;
double price;
// Parameterized constructor
Product(int id, String name, double price) {
this.id = id;
this.name = name;
this.price = price;
}
// Method to display details
void displayDetails() {
System.out.println("Product Details:");
System.out.println("ID: " + id);
System.out.println("Name: " + name);
System.out.println("Price: " + price);
}
}
public class easy {
public static void main(String[] args) {
// Creating product object
Product p1 = new Product(101, "Laptop", 75000);
p1.displayDetails();
}
}
```

Output -



Medium- Level Problem

Aim : To write a Java program to implement a library management system. The program should Use a base class Book and derived classs Fiction and Non Fiction.

Code :

```
package exp2;
class Book {
String title, author;
double price;
Book(String title, String author, double price) {
this.title = title;
this.author = author;
this.price = price;
}
void displayDetails() {
System.out.println("Book Details:");
System.out.println("Title: " + title);
System.out.println("Author: " + author);
System.out.println("Price: " + price);
}
}
// Derived class Fiction
class Fiction extends Book {
Fiction(String title, String author, double price) {
super(title, author, price);
}
@Override
void displayDetails() {
System.out.println("Fiction Book Details:");
System.out.println("Title: " + title);
System.out.println("Author: " + author);
System.out.println("Price: " + price);
}
}

class NonFiction extends Book {
NonFiction(String title, String author, double price) {
super(title, author, price);
}
@Override
void displayDetails() {

System.out.println("Non-Fiction Book Details:");
System.out.println("Title: " + title);
System.out.println("Author: " + author);
System.out.println("Price: " + price);
}
}
public class medium {
public static void main(String[] args) {
Fiction f1 = new Fiction("Harry Potter", "J.K. Rowling", 500);
NonFiction nf1 = new NonFiction("Sapiens", "Yuval Noah Harari", 700);
```

```
        f1.displayDetails();
        System.out.println();
        nf1.displayDetails();
        }
        }
```

```
Fiction Book Details:
Title: Harry Potter
Author: J.K. Rowling
Price: 500.0

Non-Fiction Book Details:
Title: Sapiens
Author: Yuval Noah Harari
Price: 700.0
```

Hard -level Problem

Aim : To design a student information system using Java with following features: - Use an abstract class Person with attributes name,age and methods like displayDetails(). - Create derived classes Student and Teacher to override displayDetails() and addunique attributes like rollNumber for students and subject for teachers.

Code :

```
package exp2;
abstract class Person {
String name;
int age;
Person(String name, int age) {
this.name = name;
this.age = age;
}
// Abstract method
abstract void displayDetails();
}
// Student class
class Student extends Person {
int rollNumber;
Student(String name, int age, int rollNumber) {
super(name, age);
this.rollNumber = rollNumber;
}
@Override
void displayDetails() {
System.out.println("Student Details:");
System.out.println("Name: " + name);
System.out.println("Age: " + age);
System.out.println("Roll Number: " + rollNumber);
}
}
// Teacher class
class Teacher extends Person {
String subject;
Teacher(String name, int age, String subject) {
super(name, age);
this.subject = subject;
}
```

```
@Override
void displayDetails() {
System.out.println("Teacher Details:");
System.out.println("Name: " + name);
System.out.println("Age: " + age);
System.out.println("Subject: " + subject);
}
}
public class hard {
public static void main(String[] args) {
Student s1 = new Student("Alice", 20, 101);
Teacher t1 = new Teacher("Mr. Smith", 40, "Mathematics");
s1.displayDetails();
System.out.println();
t1.displayDetails();
}
}
```

Output: