**Day 8\_Java Assignment**

**1. Problem description:**

What is Encapsulation?

**2. My solution:**

* Binding code and data together into a single unit and keeps both safes from outside interference and misuse are known as encapsulation. For example, a capsule, it is wrapped with different medicines. Encapsulation is achieved by making instance variables private and defining getter and setter methods to access these instance variables.

Example:

**package** Encapsulation;

**import** java.time.LocalDate;

**public** **class** Encapsul1

{

// private declaration of variables restrict the use to directly accessing

// the variables using objects, and they will need to access variables only through methods

// such as getters and setters

**private** **int** id;

**private** String name;

**private** **int** age;

**private** LocalDate dateOfBirth;

**public** **int** getId() {

**return** id;

}

**public** **void** setId(**int** id) {

**this**.id = id;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **int** getAge() {

**return** age;

}

**public** **void** setAge(**int** age) {

**this**.age = age;

}

**public** LocalDate getDateOfBirth() {

**return** dateOfBirth;

}

**public** **void** setDateOfBirth(LocalDate dateOfBirth) {

**this**.dateOfBirth = dateOfBirth;

}

**public** **static** **void** main(String[] args) {

// Create an object of Encapsul1 class

Encapsul1 obj = **new** Encapsul1();

// Set values using setter methods

obj.setId(1);

obj.setName("Surya V");

obj.setAge(26);

obj.setDateOfBirth(LocalDate.*of*(1997, 11, 17));

// Retrieve values using getter methods

System.***out***.println("ID: " + obj.getId());

System.***out***.println("Name: " + obj.getName());

System.***out***.println("Age: " + obj.getAge());

System.***out***.println("Date of Birth: " + obj.getDateOfBirth());

}

}

**Output:**

ID: 1

Name: Surya V

Age: 26

Date of Birth: 1997-11-17