

JBK Encapsulation Assignment

Example 1)

```
class Area {  
    // fields to calculate area  
    int length;  
    int breadth;  
    // constructor to initialize values  
    Area(int length, int breadth) {  
        this.length = length;  
        this.breadth = breadth;  
    }  
    // method to calculate area  
    public void getArea() {  
        int area = length * breadth;  
        System.out.println("Area: " + area);  
    }  
}  
  
class Main {  
    public static void main(String[] args) {  
        // create object of Area  
        // pass value of length and breadth  
        Area rectangle = new Area(5, 6);  
        rectangle.getArea();  
    }  
}
```

Example 2)

```
class Person {  
    // private field
```

```
private int age;

// getter method
public int getAge() {
    return age;
}

// setter method
public void setAge(int age) {
    this.age = age;
}
}

class Main {
    public static void main(String[] args) {

        // create an object of Person
        Person p1 = new Person();

        // change age using setter
        p1.setAge(24);

        // access age using getter
        System.out.println("My age is " + p1.getAge());
    }
}
```

Example 3)

Student.java

//A Java class which is a fully encapsulated class.

//It has a private data member and getter and setter methods.

```
public class Student{
```

```
//private data member
private String name;
//getter method for name
public String getName(){
return name;
}
//setter method for name
public void setName(String name){
this.name=name
}
}
```

Test.java

```
//A Java class to test the encapsulated class.
class Test{
public static void main(String[] args){
//creating instance of the encapsulated class
Student s=new Student();
//setting value in the name member
s.setName("javabykiran");
//getting value of the name member
System.out.println(s.getName());
}
}
```

Example 4)

```
class EncapsulationDemo{ ESTD 2005
    private int ssn;
    private String empName;
    private int empAge;

    //Getter and Setter methods
    public int getEmpSSN(){
```

```
        return ssn;
    }

    public String getEmpName(){
        return empName;
    }

    public int getEmpAge(){
        return empAge;
    }

    public void setEmpAge(int newValue){
        empAge = newValue;
    }

    public void setEmpName(String newValue){
        empName = newValue;
    }

    public void setEmpSSN(int newValue){
        ssn = newValue;
    }
}

public class EncapsTest{
    public static void main(String args[]){
        EncapsulationDemo obj = new EncapsulationDemo();
        obj.setEmpName("Shalaka");
        obj.setEmpAge(32);
        obj.setEmpSSN(112233);
        System.out.println("Employee Name: " + obj.getEmpName());
        System.out.println("Employee SSN: " + obj.getEmpSSN());
        System.out.println("Employee Age: " + obj.getEmpAge());
    }
}
```

}

Example 5)

// Java program to demonstrate encapsulation

class Encapsulate {

// private variables declared

// these can only be accessed by

// public methods of class

private String Name;

private int Roll;

private int Age;

// get method for age to access

// private variable Age

public int getAge() { return Age; }

// get method for name to access

// private variable Name

public String getName() { return Name; }

// get method for roll to access

// private variable Roll

public int getRoll() { return Roll; }

// set method for age to access

// private variable age

public void setAge(int newAge) { Age = newAge; }

// set method for name to access

// private variable Name

public void setName(String newName)

{

Name = newName;


```
}

// set method for roll to access
// private variable Roll
public void setRoll(int newRoll) { Roll = newRoll; }
}

public class TestEncapsulation {
    public static void main(String[] args)
    {
        Encapsulate obj = new Encapsulate();

        // setting values of the variables
        obj.setName("Harsh");
        obj.setAge(19);
        obj.setRoll(51);

        // Displaying values of the variables
        System.out.println(" name: " + obj.getName());
        System.out.println(" age: " + obj.getAge());
        System.out.println("roll: " + obj.getRoll());

        // Direct access of Roll is not possible
        // due to encapsulation
        // System.out.println("roll: " +
        // obj.Name);
    }
}
```