

Develop a code for the following scenario.

“An encapsulated class contains three variables to store Name, Age and Salary of the employee. Develop getters and setters to set and get values . Develop a test class to test your code.”

```
public class Employee {  
  
    private String name;  
  
    private int age;  
  
    private double salary;  
  
  
    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;  
    }
```

Practical 03 - Encapsulation

```
public double getSalary() {  
    return salary;  
}  
  
public void setSalary(double salary) {  
    this.salary = salary;  
}  
}  
  
public class TestEmployee {  
    public static void main(String[] args) {  
        Employee emp = new Employee();  
        emp.setName("John Doe");  
        emp.setAge(30);  
        emp.setSalary(5000.0);  
  
        System.out.println("Name: " + emp.getName());  
        System.out.println("Age: " + emp.getAge());  
        System.out.println("Salary: " + emp.getSalary());  
    }  
}
```

Now modify the same code by trying to replace the setters using a constructor.

```
public class Employee {
```

Practical 03 - Encapsulation

```
private String name;
```

```
private int age;
```

```
private double salary;
```

```
public Employee(String name, int age, double salary) {
```

```
    this.name = name;
```

```
    this.age = age;
```

```
    this.salary = salary;
```

```
}
```

```
public String getName() {
```

```
    return name;
```

```
}
```

```
public int getAge() {
```

```
    return age;
```

```
}
```

```
public double getSalary() {
```

```
    return salary;
```

```
}
```

```
}
```

```
public class TestEmployee {
```

```
public static void main(String[] args) {  
  
    Employee emp = new Employee("John Doe", 30, 5000.0);  
  
    System.out.println("Name: " + emp.getName());  
  
    System.out.println("Age: " + emp.getAge());  
  
    System.out.println("Salary: " + emp.getSalary());  
  
    }  
  
}
```

Code for the last example has been discussed during the class. We need the following Output. (Use Netbeans code generation option where necessary)

Employee Name: xxxxx (Use setter to set and getter to retrieve)

Basic Salary: xxxxx (Use setter to set and getter to retrieve)

Bonus: xxxxx (You may use the constructor to pass this value)

Bonus Amount: xxxxx (Develop a separate method to calculate Bonus amount. Bonus amount is the total of Bonus and Basic Salary)

E.g.

Employee Name: Bogdan

Basic Salary: 50000

Bonus: 10000

Bonus Amount: 60000

```
public class Employee {  
  
    private String name;  
  
    private double basicSalary;
```

Practical 03 - Encapsulation

```
private double bonus;
```

```
public Employee(double bonus) {  
    this.bonus = bonus;  
}
```

```
public String getName() {  
    return name;  
}
```

```
public void setName(String name) {  
    this.name = name;  
}
```

```
public double getBasicSalary() {  
    return basicSalary;  
}
```

```
public void setBasicSalary(double basicSalary) {  
    this.basicSalary = basicSalary;  
}
```

```
public double getBonus() {  
    return bonus;
```

Practical 03 - Encapsulation

```
}

    public double getBonusAmount() {

        return basicSalary + bonus;

    }

}

public class TestEmployee {

    public static void main(String[] args) {

        Employee emp = new Employee(10000);

        emp.setName("Bogdan");

        emp.setBasicSalary(50000);


        System.out.println("Employee Name: " + emp.getName());

        System.out.println("Basic Salary: " + emp.getBasicSalary());

        System.out.println("Bonus: " + emp.getBonus());

        System.out.println("Bonus Amount: " + emp.getBonusAmount());

    }

}
```