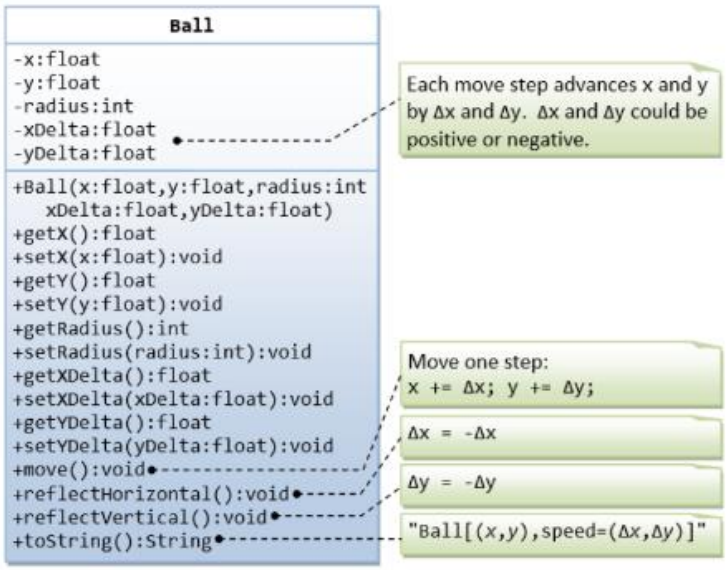
Nama : La Ode Muhammad gazali
NIM : 222212696
Kelas : 2KS2

TUGAS PRA-PERTEMUAN 3

PEMRORAMAN BERBASIS OBJEK

Kerjakanlah menggunakan Netbeans atau IDE lain favorit anda. Deadline adalah sebelum tatap muka dimulai. Tugas ini akan menjadi bahan pembahasan pada tatap muka kelas teori.

1. Sebuah class bernama Ball, yang memodelkan bola yang bisa memantul, didesain dengan class diagram di bawah ini. Terdapat radius, posisi x dan y. tiap pantulan mengubah posisi x dan y sejumlah delta-x dan delta-y. delta-x dan delta-y dapat berupa angka positif atau negative. Method reflectHorizontal() and reflectVertical() dapat digunakan untuk memantulkan bola dari dinding. Buatlah class Ball dan buat class lain berisi method main untuk mengetesnya.



- **Program Ball.java**

```
package ball;
/**
 *
 * @author U53R
 */
public class Ball {
    private float x, y, xDelta, yDelta;
    private int radius;
```

```
public Ball(float x, float y, float xDelta, float yDelta, int radius) {
    this.x = x;
    this.y = y;
    this.xDelta = xDelta;
    this.yDelta = yDelta;
    this.radius = radius;
}

public float getX() {
    return x;
}

public void setX(float x) {
    this.x = x;
}

public float getY() {
    return y;
}

public void setY(float y) {
    this.y = y;
}

public float getxDelta() {
    return xDelta;
}

public void setxDelta(float xDelta) {
    this.xDelta = xDelta;
}

public float getyDelta() {
    return yDelta;
}

public void setyDelta(float yDelta) {
    this.yDelta = yDelta;
}

public int getRadius() {
    return radius;
}
```

```
    public void setRadius(int radius) {
        this.radius = radius;
    }

    public void move(){
        x += xDelta;
        y += yDelta;
    }

    public void reflectHorizontal(){
        xDelta = -xDelta;
    }

    public void reflectVertical(){
        yDelta = -yDelta;
    }

    public String toString(){
        return("Ball[("+x+","+y+"),speed=("+xDelta+","+yDelta+")]");
    }
}
```

- **Program BallMain.java**

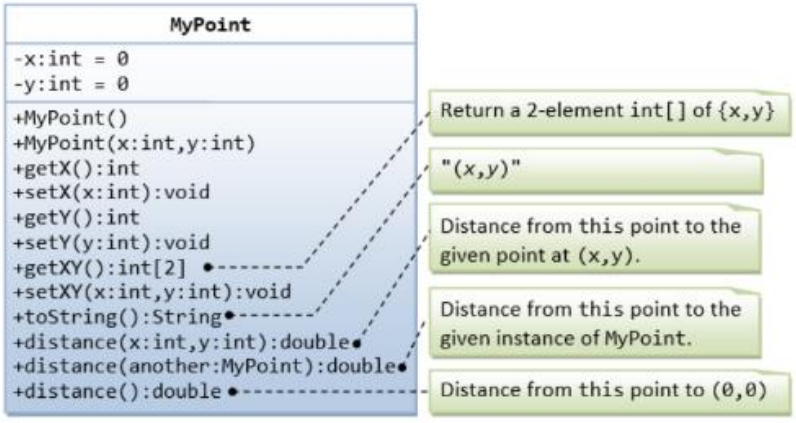
```
package ball;

/**
 *
 * @author U53R
 */
public class BallMain {
    public static void main (String args[]){
        Ball ball = new Ball(20, 20, 5, 3, 5);
        System.out.println("Posisi awal bola : "+ball.toString());

        ball.move();
        System.out.println("Posisi bola setelah bergerak: "+ball.toString());

        ball.reflectHorizontal();
        System.out.println("Posisi bola setelah dipantulkan horizontal: " +
ball.toString());

        ball.reflectVertical();
```

```
        System.out.println("Posisi bola setelah dipantulkan
vertikal:"+ball.toString());

    }
}
```

- **Hasil compile**

```
run:
Posisi awal bola : Ball[(20.0,20.0),speed=(5.0,3.0)]
Posisi bola setelah bergerak: Ball[(25.0,23.0),speed=(5.0,3.0)]
Posisi bola setelah dipantulkan horizontal: Ball[(25.0,23.0),speed=(-5.0,3.0)]
Posisi bola setelah dipantulkan vertikal:Ball[(25.0,23.0),speed=(-5.0,-3.0)]
BUILD SUCCESSFUL (total time: 0 seconds)
```

2. Sebuah class MyPoint, memodelkan titik dalam dua dimensi dengan koordinat x dan y. tuliskan kodenya dan buat kelas lain untuk mengetesnya



- **Program MyPoint.java**

```
package mypoint;

/**
 *
 * @author U53R
 */
public class MyPoint {
    private int x,y;

    public MyPoint() {
        x = 0;
```

```
        y = 0;
    }

    public MyPoint(int x, int y){
        this.x = x;
        this.y = y;
    }

    public int getX() {
        return x;
    }

    public void setX(int x) {
        this.x = x;
    }

    public int getY() {
        return y;
    }

    public void setY(int y) {
        this.y = y;
    }

    public int[] getXY(){
        int[] koordinat = {x,y};
        return koordinat;
    }

    public void setXY(int x, int y){
        this.x = x;
        this.y = y;
    }

    public String toString(){
        return "("+x+","+y+")";
    }

    public double distance(int x, int y){
        int jarakX = this.x - x;
        int jarakY = this.y - y;
        return Math.sqrt(jarakX*jarakX + jarakY*jarakY);
    }
```

```
    public double distance(MyPoint another){
        int jarakX = x - another.getX();
        int jarakY = y - another.getY();
        return Math.sqrt(jarakX*jarakX + jarakY*jarakY);
    }

    public double distance(){
        return Math.sqrt(this.x*this.x + this.y*this.y);
    }
}
```

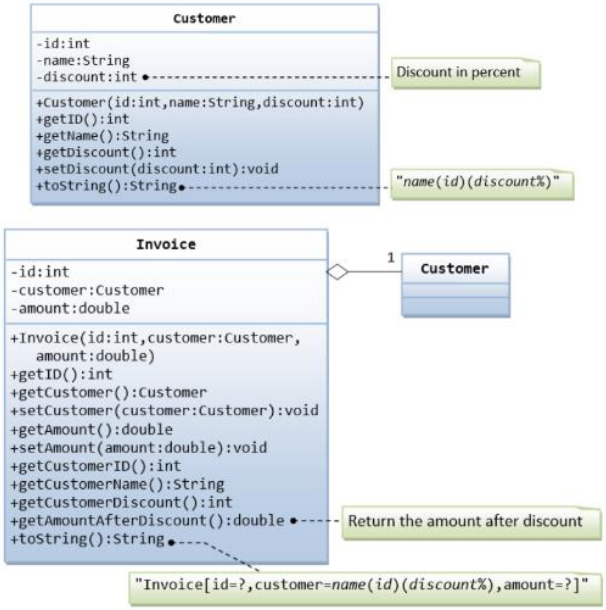
- **Program MyPointMain.java**

```
package mypoint;

/**
 *
 * @author U53R
 */
public class MyPointMain {
    public static void main (String args[]){
        MyPoint point = new MyPoint(3, 4);

        double distanceToSpecificPoint = point.distance(9, 12);
        System.out.println("Jarak dari titik tertentu: " +
distanceToSpecificPoint);

        MyPoint anotherPoint = new MyPoint(3, 6);
        double distanceToAnotherPoint = point.distance(anotherPoint);
        System.out.println("Jarak dari titik lain: " + distanceToAnotherPoint);

        double distanceToOrigin = point.distance();
        System.out.println("Jarak dari titik origin " + distanceToOrigin);
    }
}
```

- **Hasil compile**

```
run:
Jarak dari titik tertentu: 10.0
Jarak dari titik lain: 2.0
Jarak dari titik origin 5.0
BUILD SUCCESSFUL (total time: 0 seconds)
```

3. Terdapat sebuah class Customer, yang memodelkan pembeli dalam suatu transaksi, dan Invoice, yang memodelkan invoice pembeli tertentu. Tuliskan kodenya dan buat kelas untuk mengetesnya.



- **Program Customer.java**

```
package customerinvoice;

/**
 *
 * @author U53R
 */
public class Customer {
    private int id,discount;
    private String name;

    public Customer(int id, String name, int discount){
        this.id = id;
        this.name = name;
        this.discount = discount;
    }

    public int getID(){
        return id;
    }

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

```
    }

    public int getDiscount(){
        return discount;
    }

    public void setDiscount(int discount){
        this.discount = discount;
    }

    public String toString(){
        return name+"("+id+")("+discount+"%)";
    }
}
```

- **Program Invoice.java**

```
package customerinvoice;

/**
 *
 * @author U53R
 */
public class Invoice {
    private int id;
    private Customer customer;
    private double amount;

    public Invoice(int id, Customer customer,double amount){
        this.id = id;
        this.customer = customer;
        this.amount = amount;
    }

    public int getID(){
        return id;
    }

    public Customer getCustomer(){
        return customer;
    }

    public void setCustomer(Customer customer){
        this.customer = customer;
```

```
    }

    public double getAmount(){
        return amount;
    }

    public void setAmount(double amount){
        this.amount = amount;
    }

    public int getCustomerID(){
        return customer.getID();
    }

    public String getCustomerName(){
        return customer.getName();
    }

    public int getCustomerDiscount(){
        return customer.getDiscount();
    }

    public double AmountAfterDiscount(){
        return amount-amount*customer.getDiscount()/100;
    }

    public String toString(){
        return "Invoice[id="+id+", customer =
"+customer.getName()+"("+customer.getID()+")("+customer.getDiscount()+"%), amount
= "+amount+"]";
    }
}
```

- **Program CustomerInvoice.java (Fungsi main)**

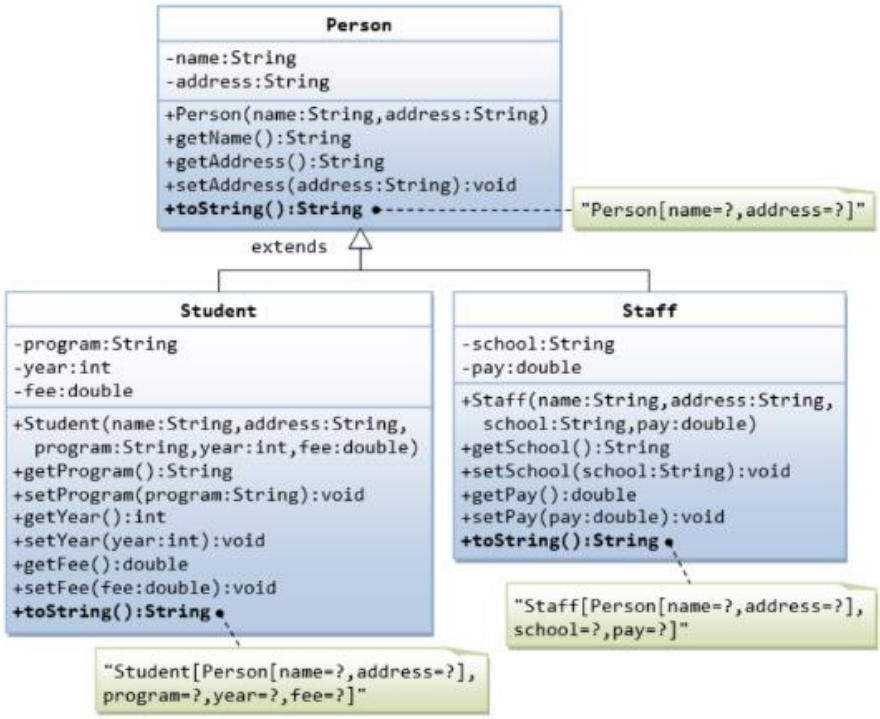
```
package customerinvoice;

/**
 *
 * @author U53R
 */
public class CustomerInvoice {

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

```
        Customer customer = new Customer(1, "Gazali", 50);
        Invoice invoice = new Invoice(101, customer, 1000.0);
        System.out.println("Sebelum diskon : \n"+invoice.toString());

        double setelahDiskon = invoice.AmountAfterDiscount();
        invoice.setAmount(setelahDiskon);
        System.out.println("Setelah Diskon : \n" +  invoice.toString());
    }
}
```

- **Hasil compile**

```
run:
Sebelum diskon :
Invoice[id=101, customer = Gazali(1)(50%), amount = 1000.0]
Setelah Diskon :
Invoice[id=101, customer = Gazali(1)(50%), amount = 500.0]
BUILD SUCCESSFUL (total time: 0 seconds)
```

4. Terdapat sebuah class Person yang menjadi superclass dari Student dan Staff. Tuliskan kodenya dan buat kelas untuk mengetesnya.



- **Program Person.java**

```java
package person;

/**
 *
 * @author U53R
 */
public class Person {
    private String name, address;

    public Person(String name, String address){
        this.name = name;
        this.address = address;
    }

    public String getName() {
        return name;
    }

    public String getAddress() {
        return address;
    }

    public void setAddress(String address) {
        this.address = address;
    }

    @Override
    public String toString(){
        return "Person [name = "+name+", address = "+address;
    }
}
```

- **Program PersonMain.java**

```java
package person;

/**
 *
 * @author U53R
 */
```

```java
public class PersonMain {
    public static void main(String args[]){
        Person person1 = new Person("Gazali","Jl Bonasel");
        System.out.println(person1.toString()+"]");

        Person person2 = new Person("La Ode","Jl Bonasut");
        System.out.println(person2.toString()+"]");
    }
}
```

- **Hasil compile**

```
run:
Person [name = Gazali, address = Jl Bonasel]
Person [name = La Ode, address = Jl Bonasut]
BUILD SUCCESSFUL (total time: 0 seconds)
```

- **Program Student.java (Heritence dari Person.java)**

```java
package person;

/**
 *
 * @author U53R
 */
public class Student extends Person{
    private String program;
    private int year;
    private double fee;

    public Student(String name, String address, String program, int year,double
fee){
        super(name,address);
        this. program = program;
        this.year = year;
        this.fee = fee;
    }

    public String getProgram() {
        return program;
    }

    public void setProgram(String program) {
        this.program = program;
```

```
    }

    public int getYear() {
        return year;
    }

    public void setYear(int year) {
        this.year = year;
    }

    public double getFee() {
        return fee;
    }

    public void setFee(double fee) {
        this.fee = fee;
    }

    @Override
    public String toString() {
        return "Student ["+ super.toString()+", program=" +
                program + ", year=" + year + ", fee=" + fee + "]";

    }
}
```

- **Program StudentMain.java**

```
package person;

/**
 *
 * @author U53R
 */
public class StudentMain {
    public static void main(String args[]){
        Student student1 = new Student("Haris", "Pedati","KS",2020,1250.0);
        System.out.println(student1.toString());
        //update address dan program
        student1.setAddress("Bonsay");
        student1.setProgram("ST");
        System.out.println("\nSetelah diupdate : \n"+student1.toString());

    }
}
```

- **Hasil compile**

```
run:
Student [Person [name = Haris, address = Pedati, program=KS, year=2020, fee=1250.0]

Setelah diupdate :
Student [Person [name = Haris, address = Bonsay, program=ST, year=2020, fee=1250.0]
BUILD SUCCESSFUL (total time: 0 seconds)
```

- **Program Staff.java (Heritence dari Person.java)**

```
package person;

/**
 *
 * @author U53R
 */
public class Staff extends Person {
    private String school;
    private double pay;

    public Staff(String name, String address, String school, double pay){
        super(name,address);
        this.school = school;
        this.pay = pay;
    }

    public String getSchool() {
        return school;
    }

    public void setSchool(String school) {
        this.school = school;
    }

    public double getPay() {
        return pay;
    }

    public void setPay(double pay) {
        this.pay = pay;
    }

    public String toString(){
```

```
        return "Staff ["+ super.toString()+", school=" + school + ", pay=" + pay
+ "]";
    }
}
```

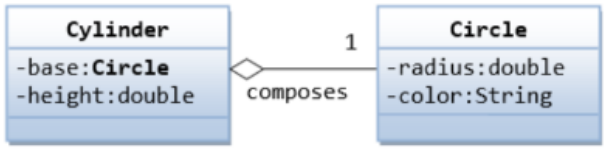
- **Program StaffMain.java**

```
package person;

/**
 *
 * @author U53R
 */
public class StaffMain {
    public static void main(String args[]){
        Staff staff1 =new Staff("Andre", "Pondok Betung","Polstat STIS",9000);
        System.out.println(staff1.toString());

        staff1.setAddress("Pondok bambu");
        staff1.setPay(10000);
        System.out.println("\nSetelah diupdate : \n"+staff1.toString());
    }
}
```

- **Hasil compile**

```
run:
Staff [Person [name = Andre, address = Pondok Betung, school=Polstat STIS, pay=9000.0]

Setelah diupdate :
Staff [Person [name = Andre, address = Pondok bambu, school=Polstat STIS, pay=10000.0]
BUILD SUCCESSFUL (total time: 0 seconds)
```

5. Hubungan antara Cylinder dan Circle selain dapat dimodelkan sebagai hubungan IS-A juga dapat dimodelkan sebagai hubungan HAS-A. contoh hubungan antara Circle and Cylinder di bawah ini sebaga HAS-A. (A Cylinder has a Circle) sesuaikan method dengan rancangan class diagram sebelumnya. Buat kodenya dan buat kelas untuk mengetesnya.



- **Program Circle.java**

```
package cylinder;

/**
 *
 * @author U53R
 */
public class Circle {
    private double radius;
    private String color;

    public Circle(double radius, String color) {
        this.radius = radius;
        this.color = color;
    }

    public double getRadius() {
        return radius;
    }

    public void setRadius(double radius) {
        this.radius = radius;
    }

    public String getColor() {
        return color;
    }

    public void setColor(String color) {
        this.color = color;
    }

    public String toString() {
```

```
        return "Circle[radius=" + radius + ", color=" + color;
    }
}
```

- **Program Cylinder.java**

```
package cylinder;

/**
 *
 * @author U53R
 */
public class Cylinder {
    private Circle base;
    private double height;

    public Cylinder(Circle base, double height) {
        this.base = base;
        this.height = height;
    }

    public Circle getBase() {
        return base;
    }

    public void setBase(Circle base) {
        this.base = base;
    }

    public double getHeight() {
        return height;
    }

    public void setHeight(double height) {
        this.height = height;
    }

    public double volume() {
        return Math.PI * base.getRadius() * base.getRadius() * height;
    }

    public String toString() {
        return "Cylinder[" +base.toString() +", height=" +height +", volume= "
+volume() +"]";
```

```
    }
}
```

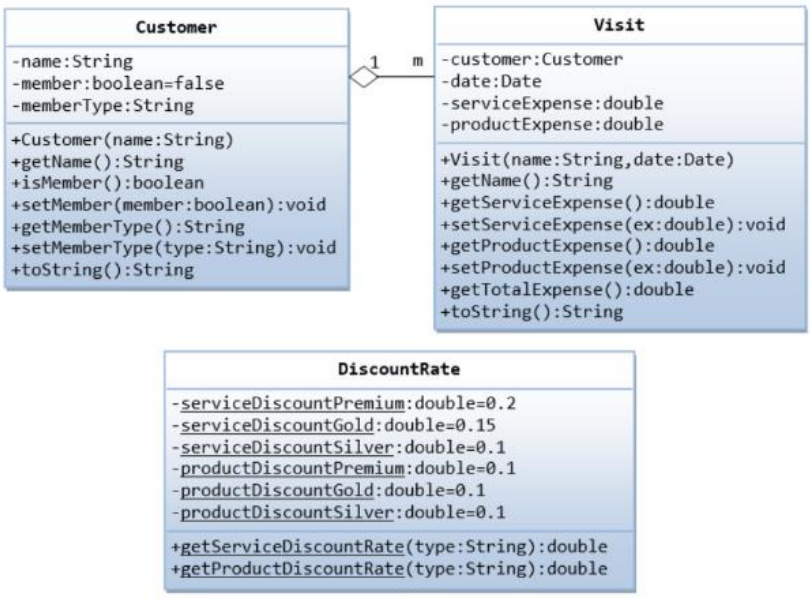
- **Program CylinderMain.java**

```
package cylinder;

/**
 *
 * @author U53R
 */
public class CylinderMain {
  public static void main(String[] args) {
    Circle circle = new Circle(7, "Red");
    Cylinder cylinder = new Cylinder(circle, 12);
    System.out.println(cylinder);

  }
}
```

- **Hasil compile**

```
run:
Cylinder[Circle[radius=7.0, color=Red, height=12.0, volume= 1847.2564803107982]
BUILD SUCCESSFUL (total time: 0 seconds)
```

6. Anda diminta untuk menulis system diskon untuk salon kecantikan, yang menyediakan layanan dan menjual produk kecantikan. Terdapat 3 tipe member: Premium, Gold, dan Silver. Premium, gold, dan silver member menerima diskon masing-masing 20%, 15%, dan 10% untuk layanan kecantikan. Customer non member tidak menerima diskon. Seluruh member menerima diskon 10% flat untuk pembelian produk (mungkin akan berubah di masa yang akan dating). Sistem anda terdiri dari 3 kelas Customer, Discount, dan Visit. Kemudian dapat menghitung total bill jika sorang pelanggan membeli $x produk dan $y layanan. Tulis juga kode pogram untuk mengetes semua kelas. Hint: garis bawah pada variable menandakan variable tersebut dideklarasikan sebagai static final variable. Garis bawah pada method menandakan method tersebut adalah method static. Kelas DiscountRate hanya memiliki variable dan method static.



- **Program Customer.java**

```
package customervisit;

/**
 *
 * @author U53R
 */
public class Customer {
    private String name;
    private boolean member=false;
    private String memberType;

    public Customer(String name) {
        this.name = name;
    }

    public String getName() {
        return name;
    }

    public boolean isMember() {
        return member;
    }

    public void setMember(boolean member) {
```

```
        this.member = member;
    }

    public String getMemberType() {
        return memberType;
    }

    public void setMemberType(String memberType) {
        this.memberType = memberType;
    }


    public String toString(){
        return "[name=" +name +", member=" +member +", memberType=" +memberType
+"]";
    }
}
```

- **Program Visit.java**

```
package customervisit;

/**
 *
 * @author U53R
 */
import java.time.LocalDate;

public class Visit {
    private Customer customer;
    private LocalDate date;
    private double serviceExpense;
    private double productExpense;

    public Visit(Customer customer, LocalDate date) {
        this.customer = customer;
        this.date = date;
    }

    public String getName() {
        return customer.getName();
    }

    public double getServiceExpense() {
```

```
        return serviceExpense;
    }

    public void setServiceExpense(double serviceExpense) {
        this.serviceExpense = serviceExpense;
    }

    public double getProductExpense() {
        return productExpense;
    }

    public void setProductExpense(double productExpense) {
        this.productExpense = productExpense;
    }

    public double getTotalExpense() {
        return getServiceExpense() + getProductExpense();
    }

    public double getTotalExpenseAfterDiscount() {
        return (
          (
            getServiceExpense() *
            (1 - DiscountRate.getServiceDiscountRate(customer.getMemberType()))
          ) +
          (
            getProductExpense() *
            (1 - DiscountRate.getProductDiscountRate(customer.getMemberType()))
          )
        );
    }

    public String toString() {
        String text = "Customer[" + customer + ",\n  date=" + date +
",\n  serviceExpense=" + serviceExpense;

        if (customer.isMember()) {
            text += ",\n  serviceDiscount=" +
(DiscountRate.getServiceDiscountRate(customer.getMemberType()) * 100) + "%";
        }

        text += ",\n  productExpense=" + productExpense;

        if (customer.isMember()) {
```

```
            text += ",\n  productDiscount=" +
(DiscountRate.getProductDiscountRate(customer.getMemberType()) * 100) + "%";
        }

        text += ",\n  totalExpense=" + getTotalExpense();

        if (customer.isMember()) {
            text += ",\n  totalExpenseAfterDiscount=" +
getTotalExpenseAfterDiscount();
        }

        text += "]";

        return text;
    }
}
```

- **Program DiscountRate.java**

```
package customervisit;

/**
 *
 * @author U53R
 */
public class DiscountRate {
    private static final double serviceDiscountPremium = 0.2;
    private static final double serviceDiscountGold = 0.15;
    private static final double serviceDiscountSilver = 0.1;
    private static final double productDiscountPremium = 0.1;
    private static final double productDiscountSilver = 0.1;
    private static final double productDiscountGold = 0.1;

    public static double getServiceDiscountRate(String type) {
        if ("Premium".equals(type)) {
            return serviceDiscountPremium;
        }else if ("Gold".equals(type)) {
            return serviceDiscountGold;
        }else if ("Silver".equals(type)) {
            return serviceDiscountSilver;
        }else {
            return 0;
        }
    }
```

```
    public static double getProductDiscountRate(String type) {
        if ("Premium".equals(type)) {
            return productDiscountPremium;
        }else if ("Gold".equals(type)) {
            return productDiscountGold;
        }else if ("Silver".equals(type)) {
            return productDiscountSilver;
        }else {
            return 0;
        }
    }
}
```

- **Program AllClassMain.java**

```
package customervisit;

/**
 *
 * @author U53R
 */

import java.time.LocalDate;

public class AllClassMain {

  public static void main(String[] args) {
    Customer Gazali = new Customer("Gazali");
    Customer LaOde = new Customer("La Ode");
    Customer Muh = new Customer("dia");

    Gazali.setMember(true);
    Gazali.setMemberType("Premium");
    Muh.setMember(true);
    Muh.setMemberType("Silver");

    Visit Gazali1 = new Visit(Gazali, LocalDate.of(2023, 10, 14));
    Gazali1.setServiceExpense(6.25);
    Gazali1.setProductExpense(8.20);
    System.out.println(Gazali1);

    Visit Gazali2 = new Visit(Gazali, LocalDate.of(2023, 12, 17));
    Gazali2.setServiceExpense(8.0);
    Gazali2.setProductExpense(15.0);
```

```
    System.out.println(Gazali2);

    Visit LaOde1 = new Visit(LaOde, LocalDate.of(2024, 01, 05));
    LaOde1.setServiceExpense(20.25);
    LaOde1.setProductExpense(7.5);
    System.out.println(LaOde1);

    Visit Muh1 = new Visit(Muh, LocalDate.of(2024, 01, 02));
    Muh1.setServiceExpense(12.25);
    Muh1.setProductExpense(6);
    System.out.println(Muh1);
  }
}
```

- **Hasil compile**

```
run:
Customer[[name=Gazali, member=true, memberType=Premium],
  date=2023-10-14,
  serviceExpense=6.25,
  serviceDiscount=20.0%,
  productExpense=8.2,
  productDiscount=10.0%,
  totalExpense=14.45,
  totalExpenseAfterDiscount=12.379999999999999]
Customer[[name=Gazali, member=true, memberType=Premium],
  date=2023-12-17,
  serviceExpense=8.0,
  serviceDiscount=20.0%,
  productExpense=15.0,
  productDiscount=10.0%,
  totalExpense=23.0,
  totalExpenseAfterDiscount=19.9]
Customer[[name=La Ode, member=false, memberType=null],
  date=2024-01-05,
  serviceExpense=20.25,
  productExpense=7.5,
  totalExpense=27.75]
Customer[[name=dia, member=true, memberType=Silver],
  date=2024-01-02,
  serviceExpense=12.25,
  serviceDiscount=10.0%,
  productExpense=6.0,
  productDiscount=10.0%,
  totalExpense=18.25,
  totalExpenseAfterDiscount=16.425]
BUILD SUCCESSFUL (total time: 0 seconds)
```