LAPORAN PRAKTIKUM

PEMOGRAMAN BERBASIS OBJECT

“Pertemuan ke-8”

Disusun oleh :

Alifah Fisalsabilawati

201511035

2B – D3 Teknik Informatika

Jurusan Teknik Komputer dan Informatika

Program Studi D3 Teknik Informatika

Politeknik Negeri Bandung

Latihan 7.1

1. Animal.java

Animal.java | Horse.java

```
1
2  public class Animal {
3      public void sound() {
4          System.out.println("Animal is making a sound");
5      }
6  }
7
```

2. Horse.java

Animal.java | Horse.java

```
1
2  public class Horse extends Animal {
3      @Override
4      public void sound() {
5          System.out.println("Neigh");
6      }
7
8      public static void main (String args[]) {
9          Animal obj = new Horse();
10         obj.sound();
11     }
12 }
13
```

3. Cat.java

Animal.java | Horse.java | Cat.java

```
1
2  public class Cat extends Animal {
3      @Override
4      public void sound() {
5          System.out.println("Meow");
6      }
7
8      public static void main (String args[]) {
9          Animal obj = new Cat();
10         obj.sound();
11     }
12 }
```

Output

Horse.java :

Problems | Javadoc | Declaratio

<terminated> Horse [Java Application] C:\P

```
Neigh
```

Cat.java :

Problems | Javadoc | Declarati

<terminated> Cat [Java Application] C:\Pr

```
Meow
```

Latihan 7.2

1. Overload.java

Overload.java | MethodOverloading.java | Animal.java | Hors

```
1
2  public class Overload {
3      void demo (int a)
4      {
5          System.out.println("a: " + a);
6      }
7      void demo (int a, int b)
8      {
9          System.out.println("a and b: " + a + "," + b);
10     }
11     double demo (double a) {
12         System.out.println("double a: " + a);
13         return a*a;
14     }
15 }
```

2. MethodOverloading.java

Overload.java | MethodOverloading.java | Animal.java | Ho

```
1
2  public class MethodOverloading {
3      public static void main (String args[])
4      {
5          Overload Obj = new Overload();
6          double result;
7          Obj.demo(10);
8          Obj.demo(10, 20);
9          result = Obj.demo(5.5);
10         System.out.println("O/P: " + result);
11     }
12 }
```

Output

Problems | Javadoc | Declaration | Co

<terminated> MethodOverloading [Java Application

```
a: 10
a and b: 10,20
double a: 5.5
O/P: 30.25
```

Latihan 7.3

1. Membuat class commision.java sesuai dengan perintah pada dokumen

Commision.java | Firm.java | Staff.java | StaffMember.java | Volunteer.java

```
1
2  public class Commision extends Hourly {
3      private double totalSale;
4      private double commisionRate;
5
6      public Commision (String eName, String eAddress, String ePhone,
7                        String sosSecNumber, double rate, double commisRate)
8      {
9          super (eName, eAddress, ePhone, sosSecNumber, rate);
10         commisionRate = commisRate;
11     }
12
13     public void addSales (double totalSale)
14     {
15         this.totalSale = totalSale * commisionRate;
16     }
17
18     public double pay ()
19     {
20         double payment = super.pay() + totalSale;
21         totalSale = 0;
22         return payment;
23     }
24
25     public String toString()
26     {
27         String result = super.toString();
28         result += "\n Total Sale : " + totalSale;
29         return result;
30     }
31 }
32
```

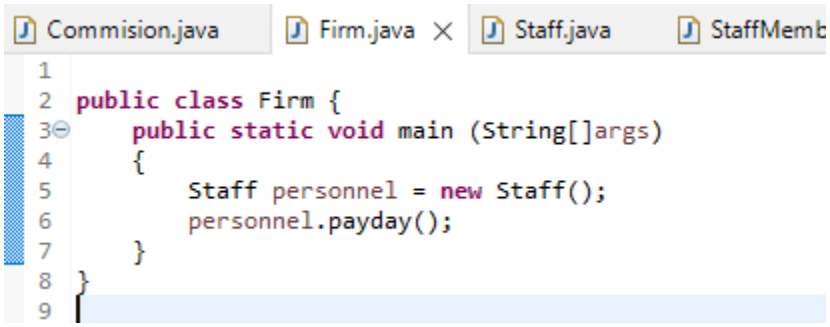
2. Mengubah class Staff.java dengan mengubah indeks array yang asalnya 6 menjadi 8 lalu menambahkan 2 karyawan yang ditugaskan ke daftar staf (buat nama, alamat, nomor telepon, dan nomor jaminan sosial Anda sendiri) . Buat salah satu karyawan mendapatkan $6,25 per jam dan komisi 20% dan yang lainnya mendapatkan $9,75 per jam dan komisi 15%.
   Untuk karyawan tambahan pertama yang Anda tambahkan, masukkan jam kerja pada 35 dan total penjualan $400; untuk yang kedua, letakkan jam di 40 dan penjualan di $950.

Commision.java | Firm.java | Staff.java | StaffMember.java | Volunteer.java

```java
public class Staff {
    StaffMember[] staffList;

    public Staff ()
    {
        staffList = new StaffMember[8];
        staffList[0] = new Executive ("Sam", "123 Main Line",
        "555-0469","123-45-6789", 2423.07);
        staffList[1] = new Employee ("Carla", "456 Off Line",
        "555-0101","987-65-4321", 1246.15);
        staffList[2] = new Employee ("Woody", "789 Off Rocker",
        "555-0000","010-20-3040", 1169.23);
        staffList[3] = new Hourly ("Diane", "678 Fifth Ave",
        "555-0690","958-47-3625", 10.55);

        staffList[4] = new Volunteer ("Norm", "987 Suds Blvd",
        "555-8374");
        staffList[5] = new Volunteer ("Cliff", "321 Duds Lane",
        "555-7282");

        staffList[6] = new Commision ("Alifah", "Cibereum Bandung",
        "555-8789","952-12-0871", 6.25, 0.2);
        staffList[7] = new Commision ("Salsa", "Cimindi Bandung",
        "555-8848","123-32-0879", 9.75, 0.15);

        ((Executive)staffList[0]).awardBonus (500.00);
        ((Hourly)staffList[3]).addHours (40);
        ((Commision)staffList[6]).addHours (35);
        ((Commision)staffList[6]).addSales(400);
        ((Commision)staffList[7]).addHours (40);
        ((Commision)staffList[7]).addSales(950);
    }
```

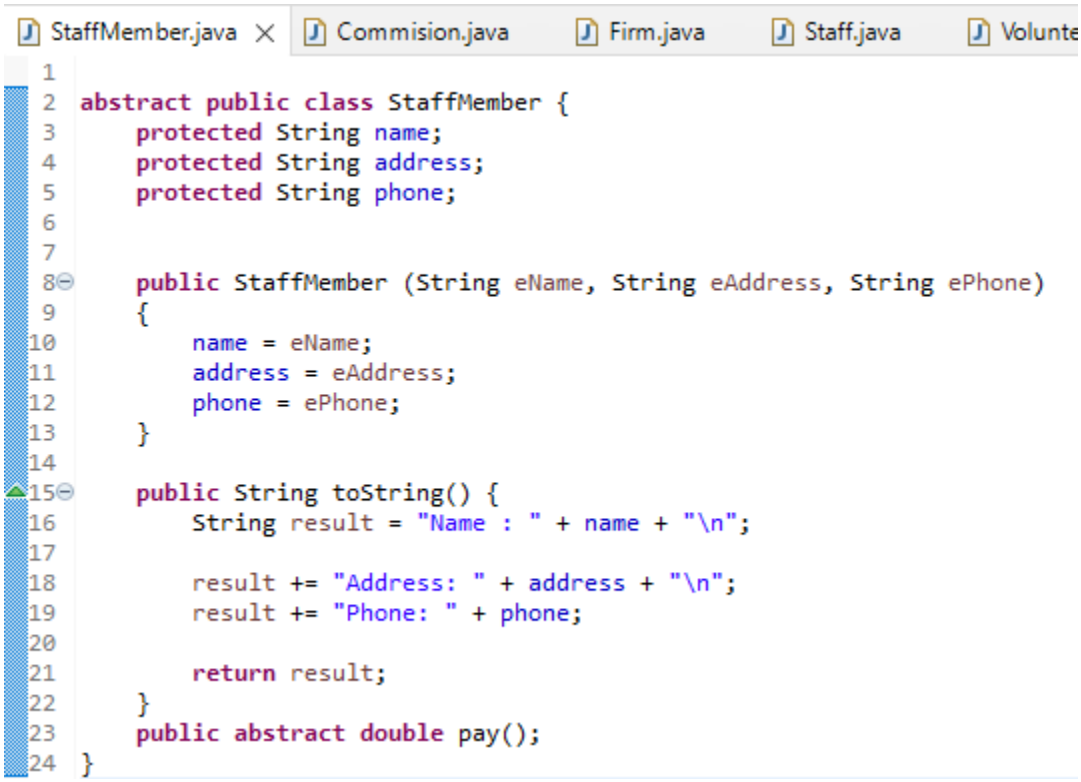
```java
    public void payday()
    {
        double amount;
        for(int count=0;count < staffList.length; count++)
        {
            System.out.println (staffList[count]);

            amount = staffList[count].pay();
            if (amount == 0.0)
                System.out.println ("Thanks!");
            else
                System.out.println ("Paid: " + amount);

            System.out.println("-----------------------------------------------------");
        }
    }
}
```

3. Firm.java



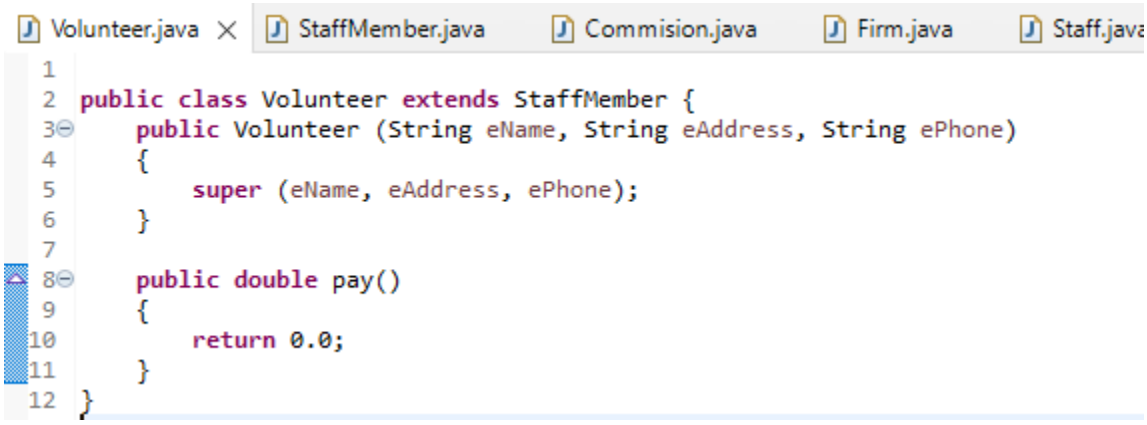

```java
public class Firm {
    public static void main (String[]args)
    {
        Staff personnel = new Staff();
        personnel.payday();
    }
}
```

4. StaffMember.java



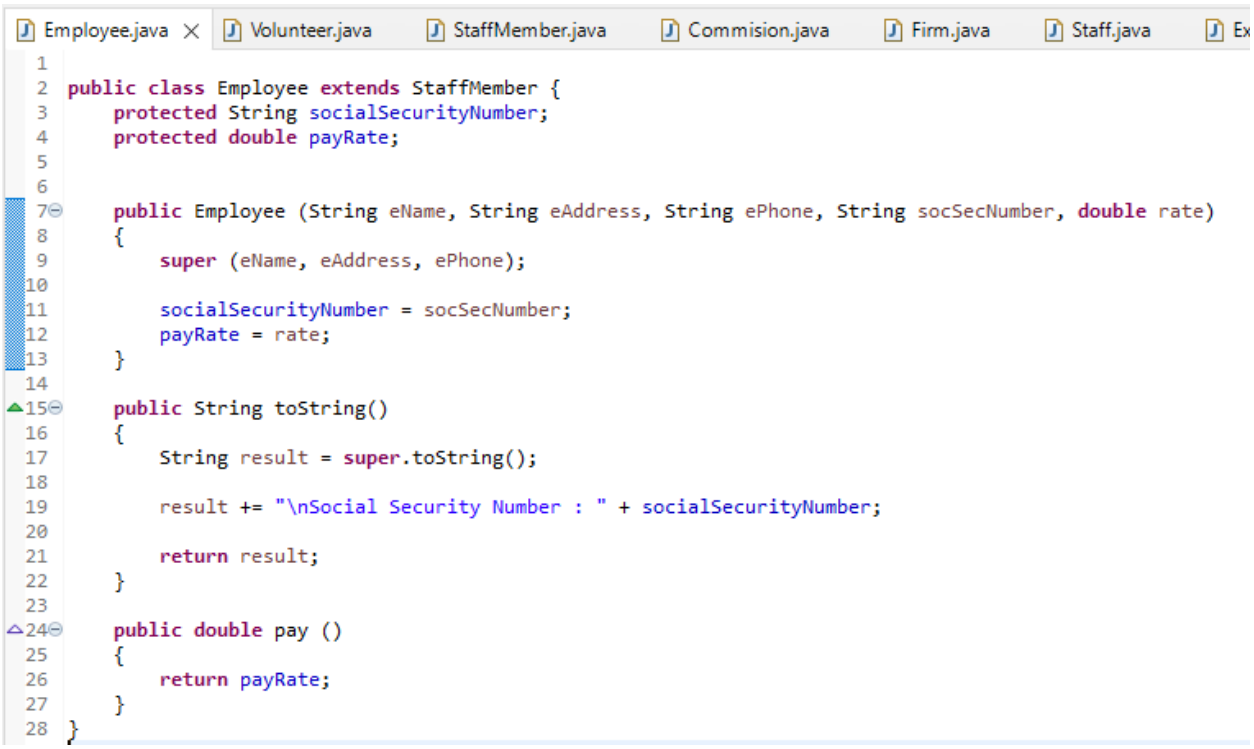

```java
abstract public class StaffMember {
    protected String name;
    protected String address;
    protected String phone;


    public StaffMember (String eName, String eAddress, String ePhone)
    {
        name = eName;
        address = eAddress;
        phone = ePhone;
    }

    public String toString() {
        String result = "Name : " + name + "\n";

        result += "Address: " + address + "\n";
        result += "Phone: " + phone;

        return result;
    }
    public abstract double pay();
}
```

5. Volunteer.java



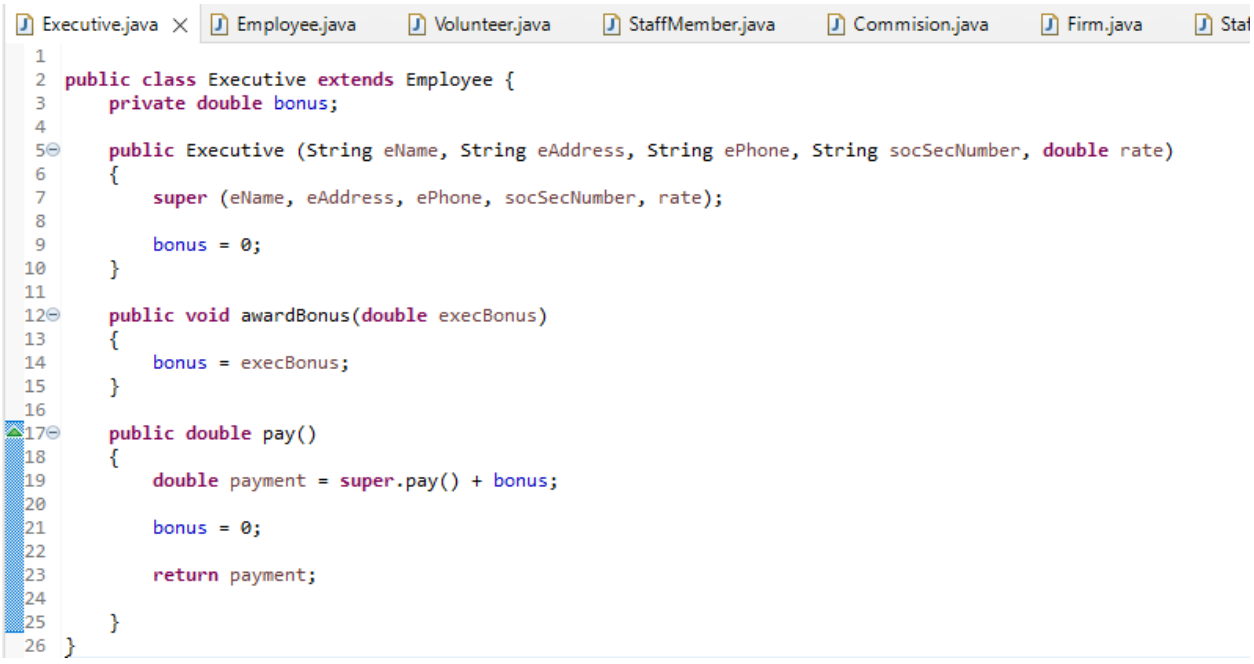

```java
public class Volunteer extends StaffMember {
    public Volunteer (String eName, String eAddress, String ePhone)
    {
        super (eName, eAddress, ePhone);
    }

    public double pay()
    {
        return 0.0;
    }
}
```

6. Employee.java




```java
public class Employee extends StaffMember {
    protected String socialSecurityNumber;
    protected double payRate;


    public Employee (String eName, String eAddress, String ePhone, String socSecNumber, double rate)
    {
        super (eName, eAddress, ePhone);

        socialSecurityNumber = socSecNumber;
        payRate = rate;
    }

    public String toString()
    {
        String result = super.toString();

        result += "\nSocial Security Number : " + socialSecurityNumber;

        return result;
    }

    public double pay ()
    {
        return payRate;
    }
}
```

7. Executive.java



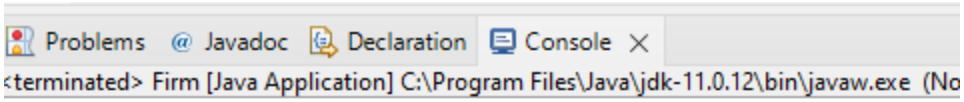

```java
public class Executive extends Employee {
    private double bonus;

    public Executive (String eName, String eAddress, String ePhone, String socSecNumber, double rate)
    {
        super (eName, eAddress, ePhone, socSecNumber, rate);

        bonus = 0;
    }

    public void awardBonus(double execBonus)
    {
        bonus = execBonus;
    }

    public double pay()
    {
        double payment = super.pay() + bonus;

        bonus = 0;

        return payment;

    }
}
```

8. Hourly.java

Hourly.java | Executive.java | Employee.java | Volunteer.java | StaffMember.java | Commision.java | Fi

```
1
2  public class Hourly extends Employee{
3
4      private int hoursWorked;
5
6      public Hourly (String eName, String eAddress, String ePhone, String sosSecNumber, double rate)
7      {
8          super (eName, eAddress, ePhone, sosSecNumber, rate);
9
10         hoursWorked = 0;
11     }
12
13     public void addHours (int moreHours)
14     {
15         hoursWorked +=moreHours;
16     }
17
18     public double pay ()
19     {
20         double payment = payRate * hoursWorked;
21
22         hoursWorked = 0;
23
24         return payment;
25     }
26
27     public String toString () {
28         String result = super. toString();
29
30         result += "\nCurrent hourse " + hoursWorked;
31
32         return result;
33     }
34 }
```

Output




```
Name : Sam
Address: 123 Main Line
Phone: 555-0469
Social Security Number : 123-45-6789
Paid: 2923.07
--------------------------------------------------
Name : Carla
Address: 456 Off Line
Phone: 555-0101
Social Security Number : 987-65-4321
Paid: 1246.15
--------------------------------------------------
Name : Woody
Address: 789 Off Rocker
Phone: 555-0000
Social Security Number : 010-20-3040
Paid: 1169.23
--------------------------------------------------
Name : Diane
Address: 678 Fifth Ave
Phone: 555-0690
Social Security Number : 958-47-3625
Current hourse 40
Paid: 422.0
--------------------------------------------------
Name : Norm
Address: 987 Suds Blvd
Phone: 555-8374
Thanks!

--------------------------------------------------
Name : Cliff
Address: 321 Duds Lane
Phone: 555-7282
Thanks!
--------------------------------------------------
Name : Alifah
Address: Cibereum Bandung
Phone: 555-8789
Social Security Number : 952-12-0871
Current hourse 35
Total Sale : 80.0
Paid: 298.75
--------------------------------------------------
Name : Salsa
Address: Cimindi Bandung
Phone: 555-8848
Social Security Number : 123-32-0879
Current hourse 40
Total Sale : 142.5
Paid: 532.5
--------------------------------------------------
```