

LAPORAN TUGAS

Polymorphism

Laporan ini disusun untuk memenuhi tugas mata kuliah Praktikum Pemrograman Berorientasi Objek



Disusun oleh:

Nurul Anisah 211511052

**PROGRAM STUDI D3 TEKNIK INFORMATIKA
JURUSAN TEKNIK KOMPUTER DAN INFORMATIKA
POLITEKNIK NEGERI BANDUNG
2022**

Studi kasus 1 (Another Type of Employee)

Firm.java

```
1 package SK1;
2
3 public class firm {
4     public static void main (String[] args) {
5         Staff personnel = new Staff();
6
7         personnel.payday();
8     }
9 }
```

Staff.java

```
1 package SK1;
2
3 public class Staff {
4     StaffMember[] staffList;
5     public Staff() {
6         // TODO Auto-generated method stub
7         staffList = new StaffMember[6];
8
9         staffList[0] = new Executive ("Sam", "123 Main Line", "555-0469", "123-45-6789", 2434.07);
10        staffList[1] = new Employee ("Carla", "456 Off Line", "555-0101", "987-65-4321", 1246.15);
11        staffList[2] = new Employee ("Woody", "789 Off Rocker", "555-0000", "010-20-3040", 1169.23);
12        staffList[3] = new Hourly ("Diane", "678 Fifth Ave", "555-0690", "958-47-3625", 10.55);
13        staffList[4] = new Volunteer ("Norm", "987 Suds Blvd", "555-8374");
14        staffList[5] = new Volunteer ("Cliff", "321 Duds Lane", "555-7282");
15
16        ((Executive)staffList[0]).awardBonus (500.00);
17        ((Hourly)staffList[3]).addHours(40);
18    }
19    public void payday() {
20        double amount;
21
22        for (int count=0; count<staffList.length; count++) {
23            System.out.println (staffList[count]);
24
25            amount = staffList[count].pay(); //polymorphic
26
27            if(amount==0.0)
28                System.out.println("Thanks!");
29            else
30                System.out.println("Paid : " + amount);
31
32            System.out.println("-----");
33        }
34    }
35 }
```

staffMember.java

```

1 package SK1;
2
3 abstract public class StaffMember {
4     protected String name;
5     protected String address;
6     protected String phone;
7
8     public StaffMember(String eName, String eAddress, String ePhone) {
9         // TODO Auto-generated method stub
10        name = eName;
11        address = eAddress;
12        phone = ePhone;
13    }
14    public String toString() {
15        String result = "Name: " + name + "\n";
16        result += "Address : " + address + "\n";
17        result += "Phone : " + phone;
18
19        return result;
20    }
21    public abstract double pay();
22
23 }
24

```

Volunteer.java

```

1 package SK1;
2
3 public class Volunteer extends StaffMember {
4
5     public Volunteer(String eName, String eAddress, String ePhone) {
6         super(eName, eAddress, ePhone);
7         // TODO Auto-generated constructor stub
8     }
9     @Override
10    public double pay() {
11        // TODO Auto-generated method stub
12        return 0.0;
13    }
14 }
15

```

Employee.java

```

1 package SK1;
2
3 public class Employee extends StaffMember {
4     protected String socialSecurityNumber;
5     protected double payRate;
6
7     public Employee(String eName, String eAddress, String ePhone, String socSecNumber, double rate) {
8         super(eName, eAddress, ePhone);
9
10        socialSecurityNumber = socSecNumber;
11        payRate = rate;
12        // TODO Auto-generated constructor stub
13    }
14
15    public String toString() {
16        String result = super.toString();
17
18        result += "\nSocial Security Number: " + socialSecurityNumber;
19        return result;
20    }
21    @Override
22    public double pay() {
23        // TODO Auto-generated method stub
24        return payRate;
25    }
26 }
27
28

```

Executive.java

```

1 package SK1;
2
3 public class Executive extends Employee {
4     private double bonus;
5
6     public Executive(String eName, String eAddress, String ePhone, String socSecNumber, double rate) {
7         super(eName, eAddress, ePhone, socSecNumber, rate);
8         // TODO Auto-generated constructor stub
9         bonus = 0;
10    }
11    public void awardBonus (double execBonus) {
12        bonus = execBonus;
13    }
14
15    @Override
16    public double pay() {
17        // TODO Auto-generated method stub
18        double payment = super.pay() + bonus;
19        bonus = 0;
20        return payment;
21    }
22 }
23 }
24

```

Hourly.java

```

1 package SK1;
2
3 public class Hourly extends Employee {
4     private int hoursWorked;
5
6     public Hourly(String eName, String eAddress, String ePhone, String socSecNumber, double rate) {
7         super(eName, eAddress, ePhone, socSecNumber, rate);
8         // TODO Auto-generated constructor stub
9         hoursWorked = 0;
10        // TODO Auto-generated constructor stub
11    }
12    public void addHours(int moreHours) {
13        hoursWorked += moreHours;
14    }
15
16    @Override
17    public double pay() {
18        // TODO Auto-generated method stub
19        double payment = payRate*hoursWorked;
20        hoursWorked = 0;
21        return payment;
22    }
23    public String toString() {
24        String result = super.toString();
25        result += "\nCurrent hours: " + hoursWorked;
26        return result;
27    }
28 }
29 }
30

```

HASILNYA

```

Name: Sam
Address : 123 Main Line
Phone : 555-0469
Social Security Number: 123-45-6789
Paid : 2934.07
-----
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: 950-47-3625
Current hours: 40
Paid : 422.0
-----
Name: Norm
Address : 987 Suds Blvd
Phone : 555-8374
Thanks!
-----
Name: Cliff
Address : 321 Duds Lane
Phone : 555-7282
Thanks!
-----

```

Kendala :

Solusi :

Studi kasus 3 (Polymorphic Sorting)

Sorting.java

```
1 package SK3;
2
3 public class Sorting{
4     public static void selectionSort(Comparable[] list) {
5         int min;
6         Comparable temp;
7
8         for(int index = 0; index<list.length-1; index++) {
9             min = index;
10            for(int scan = index+1; scan<list.length; scan++)
11                if(list[scan].compareTo(list[min])<0)
12                    min = scan;
13
14            temp = list[min];
15            list[min] = list[index];
16            list[index] = temp;
17        }
18    }
19    public static void insertionSort (Comparable[] list) {
20        for(int index = 1; index<list.length; index++) {
21            Comparable key = list[index];
22            int position = index;
23
24            while (position > 0 && key.compareTo(list[position-1])<0) {
25                list[position] = list[position-1];
26                position--;
27            }
28            list[position]=key;
29        }
30    }
31    public static void selectionSort(int[] intlist) {
32        // TODO Auto-generated method stub
33    }
34 }
35 }
```

Number.java

```
1 package SK3;
2 import java.util.Scanner;
3
4 public class Numbers {
5
6     public static void main(String[] args) {
7         // TODO Auto-generated method stub
8         int[] intlist;
9         int size;
10
11         Scanner scan = new Scanner(System.in);
12
13         System.out.print("\nHow many integers do you want to sort? ");
14         size = scan.nextInt();
15         intlist = new int[size];
16
17         System.out.println("\nEnter the numbers . . .");
18         for(int i = 0; i<size; i++)
19             intlist[i] = scan.nextInt();
20         Sorting.selectionSort(intlist);
21
22         System.out.println("\nEnter the numbers . . .");
23         for(int i = 0; i<size; i++)
24             System.out.print(intlist[i] + " ");
25         System.out.println ();
26     }
27 }
28
29 }
```

Salesperson.java

```
1 package SK3;
2
3 public class Salesperson implements Comparable {
4     private String firstName, lastName;
5     private int totalSales;
6
7     public Salesperson(String first, String last, int sales) {
8         // TODO Auto-generated method stub
9         firstName = first;
10        lastName = last;
11        totalSales = sales;
12    }
13    public String toString(){
14        return lastName + ", " + firstName + " : \t" +totalSales;
15    }
16    public boolean equals(Object other) {
17        return (lastName.equals(((Salesperson)other).getLastName()) &&
18            firstName.equals(((Salesperson)other).getFirstName()));
19    }
20    public int compareTo(Object other) {
21        int result = 0;
22        return result;
23    }
24    public String getFirstName() {
25        return firstName;
26    }
27    public void setFirstName(String firstName) {
28        // this.firstName = firstName;
29    }
30    public String getLastName() {
31        return lastName;
32    }
33    public void setLastName(String lastName) {
34        // this.lastName = lastName;
35    }
36    public int getTotalSales() {
37        return totalSales;
38    }
39    public void setTotalSales(int totalSales) {
40        // this.totalSales = totalSales;
41    }
42 }
43
44 }
```

WeeklySales.java

```

1 package SK3;
2
3 public class WeeklySales {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         Salesperson[] salesStaff = new Salesperson[10];
8
9         salesStaff[0] = new Salesperson ("Jane","Jones", 3000);
10        salesStaff[1] = new Salesperson ("Daffy","Duck", 4935);
11        salesStaff[2] = new Salesperson ("James","Jones", 3000);
12        salesStaff[3] = new Salesperson ("Dick","Walter", 2800);
13        salesStaff[4] = new Salesperson ("Don","Trump", 1570);
14        salesStaff[5] = new Salesperson ("Jane","Black", 3000);
15        salesStaff[6] = new Salesperson ("Harry","Taylor", 7300);
16        salesStaff[7] = new Salesperson ("Andy","Adams", 5000);
17        salesStaff[8] = new Salesperson ("Jim","Doe", 2850);
18        salesStaff[9] = new Salesperson ("Walt","Smith", 3000);
19
20        Sorting.InsertionSort(salesStaff);
21        System.out.println("\nRanking of sales for the week\n");
22        for (Salesperson s:salesStaff)
23            System.out.println(s);
24        }
25    }
26 }
27

```

HASILNYA :

```

Ranking of sales for the week

Jones,Jane:      3000
Duck,Daffy:      4935
Jones,James:     3000
Walter,Dick:     2800
Trump,Don:       1570
Black,Jane:      3000
Taylor,Harry:    7300
Adams,Andy:      5000
Doe,Jim:         2850
Smith,Walt:      3000

```

Kendala : awalnya terdapat error, setelah berganti parameter dan yang lainnya ternyata masih terdapat error

Solusi : mencari di google dan merubah parameternya