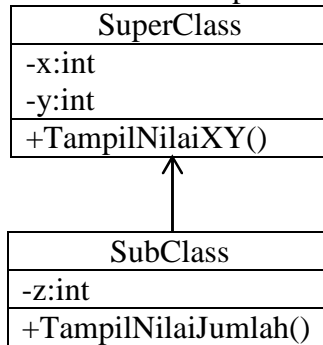


PRAKTIKUM 4 PEWARISAN

LANGKAH PERCOBAAN

1. Permasalahan Inheritance (Pewarisan)

Implementasi UML class diagram untuk class SuperClass dan SubClass



Transformasikan UML class diagram diatas ke dalam bentuk program. Tulislah listing program berikut ini sebagai pengetesan.

```
16 public class TampilInheritance {
17     public static void main(String args[]){
18         SuperClass ObSuper = new SuperClass();
19         SubClass ObSub = new SubClass();
20         System.out.println("SuperClass");
21         ObSuper.x=2;
22         ObSuper.y=3;
23         ObSuper.TampilNilaiXY();
24         System.out.println("SubClass");
25         //member superclass dapat diakses dari subclass-nya
26         ObSub.x=5;
27         ObSub.y=10;
28         ObSub.TampilNilaiXY();
29         //member tambahan yang hanya ada pada subclass
30         ObSub.z=20;
31         ObSub.TampilNilaiJumlah();
32     }
33 }
```

Tampilan yang diharapkan seperti di bawah ini.

```
run:
SuperClass
x = 2 y = 3
SubClass
x = 5 y = 10
z = 20
Jumlah (x+y+z) = 35
BUILD SUCCESSFUL (total time: 0 seconds)
```

2. Permasalahan Penggunaan Penentu Akses

```
class karyawan{
    final double gaji = 1500000;
    protected String nama;
    public void setNama(String nama) {
        this.nama = nama;
    }
    final void cetak(){
        System.out.println("Nama : "+nama+" mendapatkan gaji "+gaji+"(Pada class karyawan)");
    }
}
class honorer extends karyawan{
    public void cetak(){
        System.out.println("Nama : "+nama+" mendapatkan gaji "+gaji+"(Pada class honorer)");
    }
}
public class PenentuAkses {
    public static void main(String args[]){
        karyawan kry = new karyawan();
        kry.setNama("Budi surya wicaksana");
        kry.cetak();
        honorer hnr = new honorer();
        hnr.cetak();
    }
}
```

Jalankan program di atas apabila terjadi error jelaskan penyebabnya dan perbaiki program agar dapat dijalankan. Berdasarkan tampilan program yang didapatkan jelaskan penggunaan penentu akses pada program di atas.

3. Penggunaan Metode Abstract dan Class Abstract

```
1 abstract class X{
2     public abstract void metode_abstract();
3     public void metode_konkrit(){
4         System.out.println("Metode konkrit class X");
5     }
6 }
7 class Y extends X{
8     public void metode_abstract(){
9         System.out.println("Metode abstract yang sudah konkrit pada class Y");
10    }
11 }
12 public class ClassAbstract {
13     public static void main(String args[]){
14         Y obj = new Y();
15         obj.metode_konkrit();
16         obj.metode_abstract();
17     }
18 }
```

Jalankan program di atas sehingga di dapatkan keluaran program seperti di bawah ini.

```
run:
Metode konkrit class X
Metode abstract yang sudah konkrit pada class Y
BUILD SUCCESSFUL (total time: 0 seconds)
```

Jelaskan penggunaan metode abstract dan class abstract seperti pada program di atas.

4. Permasalahan Polymorphisme

```
1 class satu{
2     public void cetak(){
3         System.out.println("Metode class satu diakses");
4     }
5 }
6
7 class dua extends satu{
8     public void cetak(){
9         System.out.println("Metode class dua diakses");
10    }
11 }
12 class tiga extends dua{
13     public void cetak(){
14         System.out.println("Metode class tiga diakses");
15    }
16 }
17 public class ImplementasiPolymorphisme {
18     public static void main(String args[]){
19         satu ob_1;
20         dua ob_2 = new dua();
21         tiga ob_3 = new tiga();
22         ob_1=ob_2;
23         ob_1.cetak();
24         ob_1=ob_3;
25         ob_1.cetak();
26     }
27 }
```

Jalankan program di atas sehingga di dapatkan keluaran program seperti di bawah ini.

run:

Metode class dua diakses

Metode class tiga diakses

BUILD SUCCESSFUL (total time: 0 seconds)

Jelaskan penggunaan polymorphisme pada program di atas.

TUGAS RESMI

1. Buatlah program yang mengimplementasikan inheritance (pewarisan)
 2. Buatlah program yang mengimplementasikan metode abstract dan class abstract
 3. Buatlah program yang mengimplementasikan polymorphisme
- Untuk tugas resmi no 1 - 3, program tidak boleh sama dengan teori dan modul praktikum