

LAPORAN PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK

JOBSHEET 6: INHERITANCE



oleh :
Halim Teguh Saputro
2E
2141762122

**PROGRAM STUDI D-IV SISTEM INFORMASI BISNIS
JURUSAN TEKNOLOGI INFORMASI**

POLITEKNIK NEGERI MALANG
Jl. Soekarno Hatta No .9, Jatimulyo, Kec. Lowokwaru, Kota Malang,
JawaTimur 65141

KOMPETENSI

Setelah melakukan percobaan pada modul ini, mahasiswa memahami konsep:

1. Memahami konsep dasar inheritance atau pewarisan
2. Mampu membuat suatu subclass dari suatu superclass tertentu.

PRAKTIKUM 1 (EXTENDS)

A. Tahapan Percobaan

1. Buatlah sebuah class parent/superclass dengan nama ClassA.java

```
src > Praktikum1 > J ClassA.java > ...
1  package Praktikum1;
2  public class ClassA {
3      public int x;
4      public int y;
5
6      public void getNilai() {
7          System.out.println("nilai x: " + x);
8          System.out.println("nilai y: " + y);
9      }
10 }
11
```

2. Buatlah sebuah class anak/subclass dengan nama ClassB.java

```
src > Praktikum1 > J ClassB.java > ClassB > getJumlah()
1  package Praktikum1;
2  public class ClassB {
3      public int z;
4
5      public void getNilaiZ() {
6          System.out.println("nilai Z: " + z);
7      }
8
9      public void getJumlah() {
10         System.out.println("Jumlah: " + (x + y + z));
11     }
12 }
13
```

3. Buatlah class Percobaan1.java untuk menjalankan program diatas!

```
src > Praktikum1 > J Percobaan1.java > Percobaan1 > main(Str
1  package Praktikum1;
2
3  public class Percobaan1 {
4      Run | Debug
5      public static void main(String[] args) {
6          ClassB hitung = new ClassB();
7          hitung.x = 20;
8          hitung.y = 30;
9          hitung.z = 5;
10
11         hitung.getNilai();
12         hitung.getNilaiZ();
13         hitung.getJumlah();
14     }
}
```

4. Jalankan program diatas, kemudian amati apa yang terjadi!

```

J ClassB.java src\Praktikum1
  x cannot be resolved to a variable Java(33554515) [Ln 11, Col 42]
  y cannot be resolved to a variable Java(33554515) [Ln 11, Col 46]
J Percobaan1.java src\Praktikum1
  x cannot be resolved or is not a field Java(33554502) [Ln 6, Col 16]
  y cannot be resolved or is not a field Java(33554502) [Ln 7, Col 16]
  The method getNilai() is undefined for the type ClassB Java(67108964)

```

Keterangan: terjadi Error pada ClassB dan Percobaan1(main).

B. Pertanyaan

1. Pada percobaan 1 diatas program yang dijalankan terjadi error, kemudian perbaiki sehingga program tersebut bisa dijalankan dan tidak error!

Jawab:

```
src > Praktikum1 > J ClassB.java > ClassB
1  package Praktikum1;
2
3  public class ClassB extends ClassA {
4      public int z;
5
6      public void getNilaiZ() {
7          System.out.println("nilai Z: " + z);
8      }
9
10     public void getJumlah() {
11         System.out.println("Jumlah: " + (x + y + z));
12     }
13 }
```

Keterangan: solusinya yaitu menambahkan extend pada ClassB ke ClassA yang merupakan superclass.

OUTPUT :

```
nilai x: 20
nilai y: 30
nilai Z: 5
Jumlah: 55
```

2. Jelaskan apa penyebab pada percobaan 1 ketika dijalankan terdapat error!

Jawab:

Penyebab error tersebut yaitu pada ClassB, variable x dan y tidak diinisialisasi pada class tersebut. Sehingga tidak bisa dijalankan (terjadi error). Begitupun pada class Percobaan1 (main) nilai x dan y tidak bisa di ambil karena objek yang diinisialisasi berasal dari ClassB sehingga ada method dan variable yang tidak bisa digunakan karena tidak ada di ClassB. Method dan variable yang tidak ada tersebut berasal dari ClassA, sehingga diperlukan inheritance dengan menggunakan sintaks (extends) dari ClassB ke ClassA, agar bisa mengambil nilai x, nilai y, dan method getNilai().

PRAKTIKUM 2 (HAK AKSES)

A. Tahapan Percobaan

1. Buatlah sebuah class parent/superclass dengan nama ClassA.java

```
src > Praktikum2 > J ClassA.java > ClassA > getNilai
1  package Praktikum2;
2
3  public class ClassA {
4      private int x;
5      private int y;
6
7      public void setX(int x) {
8          this.x = x;
9      }
10
11     public void sety(int y) {
12         this.y = y;
13     }
14
15     public void getNilai() {
16         System.out.println("nilai x: " + x);
17         System.out.println("nilai y: " + y);
18     }
19 }
```

2. Buatlah sebuah class anak/subclass dengan nama ClassB.java

```
src > Praktikum2 > J ClassB.java > ClassB > getJuml  
1 package Praktikum2;  
2  
3 public class ClassB {  
4     private int z;  
5  
6     public void setZ(int z) {  
7         this.z = z;  
8     }  
9  
10    public void getNilaiZ() {  
11        System.out.println("nilai Z: " + z);  
12    }  
13  
14    public void getJumlah() {  
15        System.out.println("Jumlah: " + (x + y + z));  
16    }  
17 }
```

3. Buatlah class Percobaan2.java untuk menjalankan program diatas!

```
src > Praktikum2 > J Percobaan2.java > Percobaan2 >  
1 package Praktikum2;  
2  
3 public class Percobaan2 {  
4     Run | Debug  
5     public static void main(String[] args) {  
6         ClassB hitung = new ClassB();  
7  
8         hitung.setX(20);  
9         hitung.setY(30);  
10        hitung.setZ(z: 5);  
11        hitung.getNilai();  
12        hitung.getNilaiZ();  
13        hitung.getJumlah();  
14    }  
15 }
```

4. Jalankan program diatas, kemudian amati apa yang terjadi!

```
ClassB.java src\Praktikum2  
x cannot be resolved to a variable Java(33554515) [Ln 15, Col 42]  
y cannot be resolved to a variable Java(33554515) [Ln 15, Col 46]  
Percobaan2.java src\Praktikum2  
The method setX(int) is undefined for the type ClassB Java(67108964) [Ln 7, Col 16]  
The method setY(int) is undefined for the type ClassB Java(67108964) [Ln 8, Col 16]  
The method getNilai() is undefined for the type ClassB Java(67108964) [Ln 10, Col 16]
```

Keterangan: terjadi error, karena ada beberapa atribut yang tidak diinisialisasi pada ClassB dan method yang tidak terdapat pada ClassB namun di panggil di main.

B. Pertanyaan

1. Pada percobaan 2 diatas program yang dijalankan terjadi error, kemudian perbaiki sehingga program tersebut bisa dijalankan dan tidak error!

Jawab:

a. ClassA

```
src > Praktikum2 > J ClassA.java > ClassA > setY(int)
1  package Praktikum2;
2
3  public class ClassA {
4      private int x;
5      private int y;
6
7      public void setX(int x) {
8          this.x = x;
9      }
10
11  public void setY(int y) {
12      this.y = y;
13  }
14
15  public int getX() {
16      return x;
17  }
18
19  public int getY() {
20      return y;
21  }
22
23  public void getNilai() {
24      System.out.println("nilai x: " + x);
25      System.out.println("nilai y: " + y);
26  }
27 }
```

b. ClassB

```

src > Praktikum2 > J ClassB.java > ClassB > setZ(int)
1  package Praktikum2;
2
3  public class ClassB extends ClassA {
4      private int z;
5
6      public void setZ(int z) {
7          this.z = z;
8      }
9
10     public void getNilaiZ() {
11         System.out.println("nilai Z: " + z);
12     }
13
14     public void getJumlah() {
15         System.out.println("Jumlah: " + (getX() + getY() + z));
16     }
17 }

```

c. Keterangan

Untuk menyelesaikan error tersebut, saya menambahkan getter pada ClassA untuk mengambil nilai X dan nilai Y. kemudian pada ClassB, saya melakukan extends ke ClassA serta mengubah sintaks pada method getJumlah. Tepatnya pada operasi perhitungan yaitu x dan y berubah menjadi getX (mengambil nilai x) dan getY (mengambil nilai Y).

2. Jelaskan apa penyebab program pada percobaan 2 ketika dijalankan terdapat error!

Jawab:

Penyebab terjadinya error pada percobaan 2 mirip dengan percobaan 1 yaitu karena ada beberapa atribut yang tidak diinisialisasi pada ClassB dan method yang tidak terdapat pada ClassB namun di panggil di main. Namun ada penambahan solusi nya berbeda karena pada ClassA dan Class B memiliki atribut yang private sehingga tidak bisa langsung dipanggil pada class lain.

PRAKTIKUM 3 (SUPER)

A. Tahapan Percobaan

1. Buatlah sebuah class parent/superclass dengan nama Bangun.java

```

1  package Praktikum3;
2
3  public class Bangun {
4      protected double phi;
5      protected int r;
6  }

```

2. Buatlah sebuah class anak/subclass dengan nama Tabung.java

```
src > Praktikum3 > J Tabung.java > Tabung
1  package Praktikum3;
2
3  public class Tabung extends Bangun {
4      protected int t;
5
6      public void setSuperPhi(double phi) {
7          super.phi = phi;
8      }
9
10     public void setSuperR(int r) {
11         this.r = r;
12     }
13
14     public void setT(int t) {
15         this.t = t;
16     }
17
18     public void volume() {
19         System.out.println("Volume Tabung adalah: " + (super.phi *
20             super.r *
21             super.r *
22             this.t));
23     }
```

3. Buatlah class Percobaan3.java untuk menjalankan program diatas!

```
src > Praktikum3 > J Percobaan3.java > Percobaan3 > main(Str
1  package Praktikum3;
2
3  public class Percobaan3 {
4      Run | Debug
5      public static void main(String[] args) {
6          Tabung tabung = new Tabung();
7          tabung.setSuperPhi(phi: 3.14);
8          tabung.setSuperR(r: 10);
9          tabung.setT(t: 3);
10         tabung.volume();
11     }
12 }
```

4. Jalankan program diatas

```
bsheet6\bin' 'Praktikum3.Percob
Volume Tabung adalah: 942.0
PS C:\Users\Halim\Downloads\POL
```


B. Pertanyaan

1. Jelaskan fungsi “super” pada potongan program berikut di class Tabung!

Jawab:

Fungsi super pada potongan program tersebut adalah untuk mengakses variable dari super class atau class parentnya.

2. Jelaskan fungsi “super” dan “this” pada potongan program berikut di class Tabung!

Jawab: fungsi super untuk mengakses variable pada superclass, sedangkan this untuk mengakses variable pada class itu sendiri.

3. Jelaskan mengapa pada class Tabung tidak dideklarasikan atribut “phi” dan “r” tetapi class tersebut dapat mengakses atribut tersebut!

Jawab: class Tabung dapat mengakses atribut “phi” dan “r” karena pada program tabung telah dilakukan extends ke Class Bangun (superclass), kemudian atribut “phi” dan “r” menggunakan acces control “super”.

PRAKTIKUM 4 (SUPER CONSTRUCTOR)

A. Tahapan Percobaan

1. Buatlah tiga file dengan nama ClassA.java, ClassB.java, dan ClassC.java, seperti pada code dibawah ini!

a. ClassA.java

```
src > Praktikum4 > J ClassA.java > ...
1  package Praktikum4;
2
3  public class ClassA {
4      ClassA() {
5          System.out.println(x: "Konstruktor A dijalankan");
6      }
7  }
```

b. ClassB.java

```
src > Praktikum4 > J ClassB.java > ClassB > ClassB()
1  package Praktikum4;
2
3  public class ClassB extends ClassA {
4      ClassB() {
5          System.out.println(x: "Konstruktor B dijalankan");
6      }
7  }
```

c. ClassC.java

```
src > Praktikum4 > J ClassC.java > ...
1  package Praktikum4;
2
3  public class ClassC extends ClassB {
4      ClassC() {
5          System.out.println(x: "Konstruktor C dijalankan");
6      }
7  }
```

2. Buatlah class Percobaan4.java untuk menjalankan program diatas!

```
src > Praktikum4 > J Percobaan4.java > ...
1  package Praktikum4;
2
3  public class Percobaan4 {
4      Run | Debug
5      public static void main(String[] args) {
6          ClassC test = new ClassC();
7      }
8  }
```

3. Jalankan program kemudian amati apa yang terjadi!

```
Konstruktor A dijalankan
Konstruktor B dijalankan
Konstruktor C dijalankan
PS C:\Users\Halim\Download
```

B. Pertanyaan

1. Pada percobaan 4 sebutkan mana class yang termasuk superclass dan subclass, kemudian jelaskan alasannya!

Jawab:

Yang termasuk superclass yaitu: ClassA dan ClassB

Yang termasuk subclass yaitu: ClassB dan ClassC

2. Ubahlah isi konstruktor default classC seperti berikut:

```
3  public class ClassC extends ClassB {
4      ClassC() {
5          super();
6          System.out.println(x: "Konstruktor C dijalankan");
7      }
8  }
```

Tambahkan kata super() di baris pertamanya dalam konstruktor defaultnya. Coba jalankan Kembali class Percobaan4 dan terlihat ada perbedaan dari hasil outputnya!

Jawab:

Hasil setelah penambahan super():

```
Konstruktor A dijalankan
Konstruktor B dijalankan
Konstruktor C dijalankan
PS C:\Users\Halim\Downloads
```

Keterangan: tidak terjadi perubahan

3. Ubahlah isi konstruktor default ClassC seperti berikut:

```
3 public class ClassC extends ClassB {
4     ClassC() {
5         System.out.println(x: "Konstruktor C dijalankan");
6         super();
7     }
}
```

Ketika mengubah posisi `super()` dibaris kedua dalam konstruktor defaultnya dan terlihat ada error. Kemudian kembalikan `super()` ke baris pertama seperti sebelumnya, maka errornya akan hilang.

Perhatikan hasil keluaran Ketika class Percobaan4 dijalankan. Kenapa bisa tampil output seperti berikut pada saat instansiasi objek test dari class ClassC

```
Konstruktor A dijalankan
Konstruktor B dijalankan
Konstruktor C dijalankan
PS C:\Users\Halim\Downloads
```

Jelaskan bagaimana urutan proses jalannya konstruktor saat objek test dibuat!

Jawab:

Sintaks `super()` hanya bisa digunakan pada awal baris dalam konstruktor yang berfungsi untuk mengakses constructor dari superclass Class itu sendiri.

Urutan proses program agar menghasilkan output seperti itu adalah:

1. Saat dipanggil pada `main()` (inisialisasi objek ClassC), program melihat bahwa program ini memiliki superclass ClassB (`extends ClassB`)
2. Kemudian melihat lagi pada ClassB, ternyata ClassB memiliki superclass ClassA (`ClassB extend ClassA`).
3. Kemudian melihat bahwa ClassA tidak memiliki superclass, maka program menjalankan constructor ClassA,
4. Setelah selesai. Kembali ke ClassB untuk menjalankan constructor ClassB.
5. Setelah itu Kembali ke classC untuk menjalankan constructor ClassC.

4. Apakah fungsi `super()` pada potongan program dibawah ini di ClassC!

```
public class ClassC extends ClassB {
    ClassC() {
        super();
        System.out.println(x: "Konstruktor C dijalankan");
    }
}
```

Jawab:

Sintaks `super()` hanya bisa digunakan pada awal baris dalam konstruktor yang berfungsi untuk mengakses constructor dari superclass Class itu sendiri. Sehingga pada class tersebut `super()` berfungsi untuk memanggil constructor pada ClassB.

TUGAS

1. Buatlah sebuah program dengan konsep pewarisan seperti pada class diagram berikutini. Kemudian buatlah instansiasi objek untuk menampilkan data nama pegawai dan gaji yang didapatkannya.

Jawab:

a. Pegawai.java

```
src > Tugas > J Pegawai.java > Pegawai > getNama()
1  package Tugas;
2
3  public class Pegawai {
4      protected String nip;
5      protected String nama;
6      protected String alamat;
7
8      Pegawai(String nip, String nama, String alamat) {
9          this.nip = nip;
10         this.nama = nama;
11         this.alamat = alamat;
12     }
13
14     public String getNama() {
15         return nama;
16     }
17
18     public String getNip() {
19         return nip;
20     }
21
22     public String getAlamat() {
23         return alamat;
24     }
25
26     public int getGaji() {
27         return 100000;
28     }
29 }
```

b. Dosen.java

```

src > Tugas > Dosen.java > Dosen > setSKS(int)
1  package Tugas;
2
3  public class Dosen extends Pegawai {
4      private int jumlahSKS;
5      private int tarifSKS = 100000;
6
7      Dosen(String nip, String nama, String alamat) {
8          super(nip, nama, alamat);
9          super.nip = nip;
10         super.nama = nama;
11         super.alamat = alamat;
12     }
13
14     public void setSKS(int sks) {
15         this.jumlahSKS = sks;
16     }
17
18     public int getGaji() {
19         return tarifSKS * jumlahSKS;
20     }
21 }

```

c. DaftarGaji.java

```

src > Tugas > DaftarGaji.java > DaftarGaji > addPegawai(Pegawai)
1  package Tugas;
2
3  public class DaftarGaji {
4      private Pegawai[] listPegawai;
5
6      DaftarGaji(int jmlPegawai) {
7          listPegawai = new Pegawai[jmlPegawai];
8      }
9
10     public void addPegawai(Pegawai pegawai) {
11         for (int i = 0; i < listPegawai.length; i++) {
12             if (listPegawai[i] == null) {
13                 this.listPegawai[i] = pegawai;
14                 break;
15             }
16         }
17     }
18
19     public void printSemuaGaji() {
20         int i = 0;
21         while (listPegawai[i] != null && i < listPegawai.length) {
22             System.out.println("NIP\t: " + listPegawai[i].getNip());
23             System.out.println("Nama\t: " + listPegawai[i].getNama());
24             System.out.println("Alamat\t: " + listPegawai[i].getAlamat());
25             System.out.println("Gaji\t: " + listPegawai[i].getGaji());
26             System.out.println();
27             i++;
28         }
29     }

```

d. MainProgram.java

```
src > Tugas > J MainProgram.java > MainProgram > main(String[])
1  package Tugas;
2
3  public class MainProgram {
4      public static void main(String[] args) {
5          DaftarGaji data = new DaftarGaji(jmlPegawai: 4);
6          Dosen halim = new Dosen(nip: "001", nama: "Halim", alamat: "Malang");
7          halim.setSKS(sks: 5);
8
9          Pegawai halim1 = new Pegawai(nip: "001", nama: "Halim", alamat: "Malang");
10         data.addPegawai(halim1);
11
12         Dosen teguh = new Dosen(nip: "002", nama: "Teguh", alamat: "Surabaya");
13         teguh.setSKS(sks: 3);
14         data.addPegawai(teguh);
15
16         Dosen saputro = new Dosen(nip: "003", nama: "Saputro", alamat: "Bandung");
17         saputro.setSKS(sks: 8);
18         data.addPegawai(saputro);
19
20         data.printSemuaGaji();
21     }
22 }
23 }
```

e. Output

```
NIP      : 001
Nama     : Halim
Alamat   : Malang
Gaji     : 100000

NIP      : 002
Nama     : Teguh
Alamat   : Surabaya
Gaji     : 300000

NIP      : 003
Nama     : Saputro
Alamat   : Bandung
Gaji     : 800000
```

LINK GITHUB: <https://github.com/HalimTeguh/Praktikum/tree/master/06.%20Inheritance>