

**LAPORAN PRAKTIKUM
PEMROGRAMAN II
MODUL 4**



INHERITANCE & CLASS DIAGRAM

Oleh:

Indra Suryadilaga NIM. 2410817310014

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS TEKNIK
UNIVERSITAS LAMBUNG MANGKURAT
OKTOBER 2025**

LEMBAR PENGESAHAN
LAPORAN PRAKTIKUM PEMROGRAMAN II
MODUL 4

Laporan Praktikum Pemrograman II Modul 4: Pewarisan/Inheritance & Diagram Kelas ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman II. Laporan Praktikum ini dikerjakan oleh:

Nama Praktikan : Indra Suryadilaga
NIM : 2410817310014

Menyetujui,
Asisten Praktikum

Mengetahui,
Dosen Penanggung Jawab Praktikum

Galih Aji Sabdaraya
NIM. 2310817210005

Andreyan Rizky Baskara, S.Kom.,
M.Kom.
NIP. 19930703 201903 1 011

DAFTAR ISI

LEMBAR PENGESAHAN	2
DAFTAR ISI.....	3
DAFTAR GAMBAR	4
DAFTAR TABEL.....	5
SOAL 1	6
A. Source Code	6
B. Output Program.....	8
C. Pembahasan.....	8
SOAL 2	9
A. Source Code	10
A. Output Program.....	14
B. Pembahasan.....	14
GITHUB.....	16

DAFTAR GAMBAR

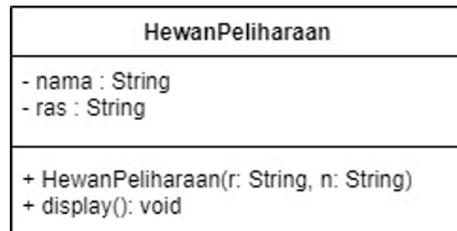
Gambar 1. Screenshot Hasil Jawaban Soal 1	8
Gambar 2. Screenshot Hasil Jawaban Soal 2	14

DAFTAR TABEL

Tabel 1.1 Soal No. 1	6
Tabel 1.2 MainSoal1.java Source Code	6
Tabel 1.3 HewanPeliharaan.java Source Code	7
Tabel 2.1 Soal No. 2	9
Tabel 2.2MianSoal2.java Source Code	10
Tabel 2.3 HewanPeliharaan.java Source Code	12
Tabel 2.4 Anjing.java Source Code	13
Tabel 2.5 Kucing.java Source Code.....	13

SOAL 1

Diberikan class diagram seperti berikut: (isi program harus sesuai dengan class diagram).



Implementasikan class diagram diatas menjadi program menggunakan Bahasa pemrograman java dengan ketentuan seperti berikut:

- Inisiasi objek harus menggunakan konstruktor.
- Output ditampilkan dengan hanya memanggil method display.
- Tidak boleh ada fungsi mencetak di class main. contoh table dibawah.

Tabel 1.1 Soal No. 1

Input
Nama Hewan Peliharaan: Leo Ras: Kucing Anggora
Output
Detail Hewan Peliharaan: Nama hewan peliharaanku adalah : Leo Dengan ras : Kucing Anggora

Simpan coding anda dengan nama package: **soal1**

A. Source Code

Tabel 1.2 MainSoal1.java Source Code

File: MainSoal1.java	
1	package modul_04.soal1;
2	import java.util.Scanner;
3	public class MainSoal1 {
4	public static void main(String[] args) {

```

7         Scanner input = new Scanner(System.in);
8
9         String nama;
10        String ras;
11
12        System.out.print("Nama Hewan Peliharaan:");
13        nama = input.nextLine();
14
15        System.out.print("Ras: ");
16        ras = input.nextLine();
17
18        HewanPeliharaan hewan = new
19 HewanPeliharaan(nama, ras);
20
21        hewan.display();
22        input.close();
23    }
24
25 }
```

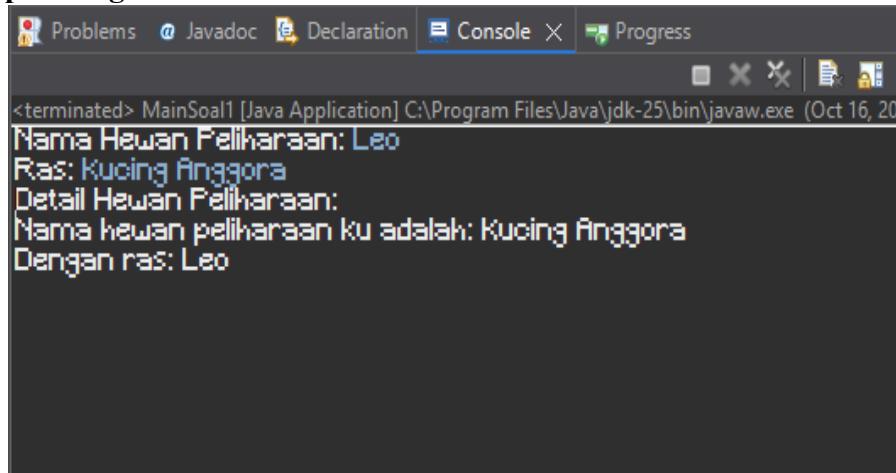
Tabel 1.3 HewanPeliharaan.java Source Code

File: HewanPeliharaan.java

```

1 package modul_04.soall;
2
3 public class HewanPeliharaan {
4     private String nama;
5     private String ras;
6
7     public HewanPeliharaan(String r, String n) {
8         this.ras = r;
9         this.nama = n;
10    }
11
12    public void display() {
13        System.out.println("Detail Hewan
14 Peliharaan:");
15        System.out.println("Nama hewan peliharaan
16 ku adalah: " + this.nama);
17        System.out.println("Dengan ras: " +
18 this.ras);
19    }
20 }
```

B. Output Program



```
<terminated> MainSoal1 [Java Application] C:\Program Files\Java\jdk-25\bin\javaw.exe (Oct 16, 2016)
Nama Hewan Peliharaan: Leo
Ras: Kucing Anggora
Detail Hewan Peliharaan:
Nama hewan peliharaan ku adalah: Kucing Anggora
Dengan ras: Leo
```

Gambar 1. Screenshot Hasil Jawaban Soal 1

C. Pembahasan

Program ini terdiri dari dua kelas: HewanPeliharaan dan MainSoal1. Kelas HewanPeliharaan dirancang sebagai *blueprint* atau cetakan untuk objek hewan. Sesuai dengan Class Diagram, kelas ini memiliki dua atribut, yaitu nama dan ras. Kedua atribut ini dideklarasikan sebagai private.

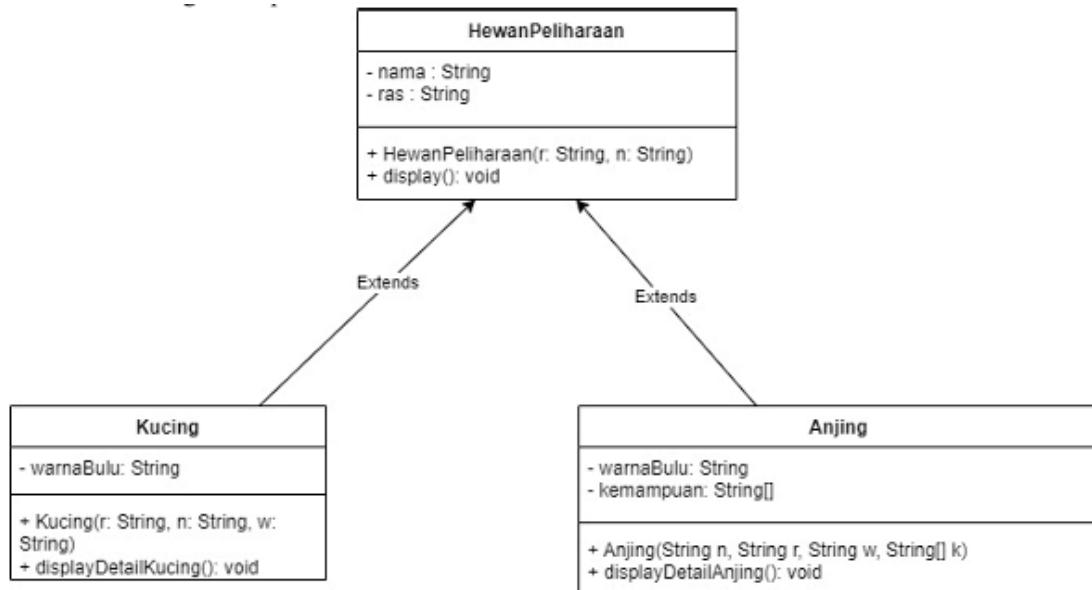
Sesuai ketentuan soal, inisialisasi objek wajib menggunakan konstruktor. Oleh karena itu, kelas HewanPeliharaan menyediakan satu konstruktor publik, yaitu public HewanPeliharaan(String r, String n). Konstruktor ini menerima dua parameter string yang digunakan untuk mengisi nilai atribut ras dan nama saat objek pertama kali dibuat.

Ketentuan soal juga mengharuskan output ditampilkan hanya dengan memanggil *method* display dan tidak boleh ada fungsi mencetak di *class main*. Untuk itu, kelas HewanPeliharaan dilengkapi dengan *method* public void display(). Method inilah yang mencetak seluruh detail informasi hewan peliharaan ke konsol.

Kelas MainSoal1 berperan sebagai *driver class* yang mengelola input pengguna dan alur program. Kelas ini membuat objek dari HewanPeliharaan menggunakan new HewanPeliharaan(nama, ras) dan kemudian memanggil hewan.display() untuk menampilkan output, sehingga memenuhi semua kriteria yang ditetapkan oleh Soal 1.

SOAL 2

Diberikan class diagram seperti berikut:



Implementasikan class diagram diatas menjadi program menggunakan Bahasa pemrograman java dengan ketentuan seperti berikut:

- Inisiasi objek harus menggunakan konstruktor.
- Isi baris program harus sesuai dengan apa yang digambarkan di dalam kelas diagram.
- Menampilkan detail dari class Kucing maupun Anjing harus melalui method yang ada, tidak boleh menggunakan fungsi cetak dari java
- Isi detail dari class Kucing dan Anjing harus memanggil method display pada parent class terlebih dahulu.

Tabel 2.1 Soal No. 2

Input
Pilih jenis hewan yang ingin diinputkan: 1 = Kucing 2 = Anjing Masukkan pilihan: 1

Nama hewan peliharaan: lolo

Ras: Anggora

Warna Bulu: abu-abu

Output

Detail Hewan Peliharaan:

Nama hewan peliharaanku adalah : lolo

Dengan ras : Anggora

Memiliki warna bulu : abu-abu

Input

Pilih jenis hewan yang ingin diinputkan:

1 = Kucing

2 = Anjing

Masukkan pilihan: 2

Nama hewan peliharaan: andi

Ras: cihuahua

Warna Bulu: putih

Kemampuan : berenang, berjabat tangan, melompat

Output

Detail Hewan Peliharaan:

Nama hewan peliharaanku adalah : andi

Dengan ras : cihuahua

Memiliki warna bulu : putih

Memiliki kemampuan : berenang berjabat tangan melompat

Simpan coding anda dengan nama package: **soal2**

A. Source Code

Tabel 2.2 MainSoal2.java Source Code

File: MainSoal2.java

```
1 package modul 04.soal2;  
2  
3 import java.util.Scanner;  
4
```

```

5  public class MainSoal2 {
6      public static void main(String[] args) {
7          Scanner input = new Scanner(System.in);
8          int pilihan = 0;
9
10         do {
11             System.out.println("Pilih jenis
12 hewan yang ingin diinputkan:");
13             System.out.println("1 = Kucing");
14             System.out.println("2 = Anjing");
15             System.out.print("Masukkan pilihan:
16 ");
17
18             String inputStr = input.nextLine();
19
20             if (inputStr.equals("1")) {
21                 pilihan = 1;
22             } else if (inputStr.equals("2")) {
23                 pilihan = 2;
24             } else {
25                 System.out.println("Pilihan
26 tidak valid, silakan masukkan angka 1 atau 2.\n");
27                 pilihan = 0;
28             }
29
30         } while (pilihan != 1 && pilihan != 2);
31
32         if (pilihan == 1) {
33             System.out.print("Nama hewan
34 peliharaan: ");
35             String nama = input.nextLine();
36
37             System.out.print("Ras: ");
38             String ras = input.nextLine();
39
40             System.out.print("Warna Bulu: ");
41             String warnaBulu = input.nextLine();
42
43             Kucing kucing = new Kucing(ras,
44             nama, warnaBulu);
45             kucing.displayDetailKucing();
46
47         } else if (pilihan == 2) {
48             System.out.print("Nama hewan
49 peliharaan: ");

```

```

50         String nama = input.nextLine();
51
52         System.out.print("Ras: ");
53         String ras = input.nextLine();
54
55         System.out.print("Warna Bulu: ");
56         String warnaBulu = input.nextLine();
57
58         System.out.print("Kemampuan: ");
59         String kemampuanStr =
60         input.nextLine();
61         String[] kemampuan =
62         kemampuanStr.split(",\\s*");
63
64         Anjing anjing = new Anjing(nama,
65         ras, warnaBulu, kemampuan);
66         anjing.displayDetailAnjing();
67     }
68
69     input.close();
70 }
71
72 }
```

Tabel 2.3 HewanPeliharaan.java Source Code

File: HewanPeliharaan.java

```

1 package modul_04.soal2;
2
3 public class HewanPeliharaan {
4     private String nama;
5     private String ras;
6
7     public HewanPeliharaan(String r, String n) {
8         this.ras = r;
9         this.nama = n;
10    }
11
12    public void display() {
13        System.out.println("Detail Hewan
14 Peliharaan:");
15        System.out.println("Nama hewan peliharaan
16 ku adalah: " + this.nama);
17        System.out.println("Dengan ras: " +
18 this.ras);
```

19	}
20	}

Tabel 2.4 Anjing.java Source Code

File: Anjing.java

```

1 package modul_04.soal2;
2
3 public class Anjing extends HewanPeliharaan {
4     private String warnaBulu;
5     private String kemampuan[];
6
7     public Anjing(String r, String n, String w,
8     String[] k) {
9         super(r, n);
10        this.warnaBulu = w;
11        this.kemampuan = k;
12    }
13
14    public void displayDetailAnjing() {
15        super.display();
16        System.out.println("Warna bulu : " +
17 this.warnaBulu);
18        System.out.println("Kemampuan : " +
19 String.join(", ", this.kemampuan));
20    }
21 }
```

Tabel 2.5 Kucing.java Source Code

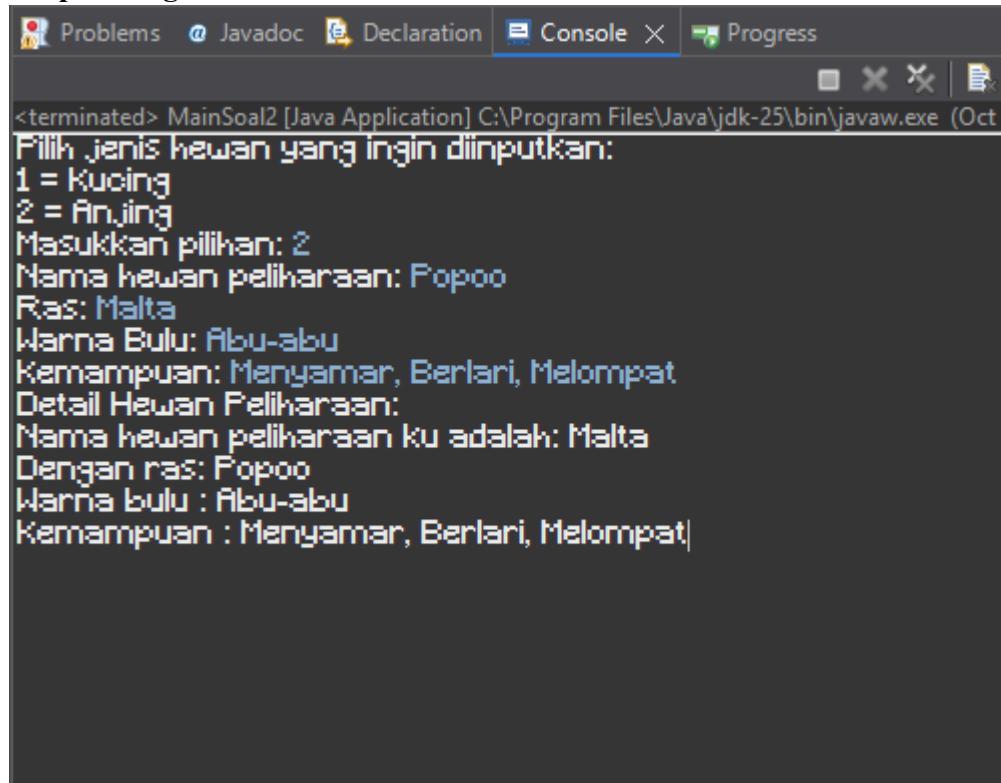
File: Kucing.java

```

1 package modul_04.soal2;
2
3 public class Kucing extends HewanPeliharaan {
4     private String warnaBulu;
5
6     Kucing(String r, String n, String w) {
7         super(r, n);
8         this.warnaBulu = w;
9     }
10
11    public void displayDetailKucing() {
12        super.display();
```

```
14         System.out.println("Warna bulu : " +
15     this.warnaBulu);
16 }
17 }
```

A. Output Program



The screenshot shows a Java application running in an IDE. The console tab is active, displaying the following output:

```
<terminated> MainSoal2 [Java Application] C:\Program Files\Java\jdk-25\bin\javaw.exe (Oct)
Pilih jenis hewan yang ingin diinputkan:
1 = Kucing
2 = Anjing
Masukkan pilihan: 2
Nama hewan peliharaan: Popoo
Ras: Malta
Warna Bulu: Abu-abu
Kemampuan: Menyamar, Berlari, Melompat
Detail Hewan Peliharaan:
Nama hewan peliharaan ku adalah: Malta
Dengan ras: Popoo
Warna bulu : Abu-abu
Kemampuan : Menyamar, Berlari, Melompat
```

Gambar 2. Screenshot Hasil Jawaban Soal 2

B. Pembahasan

Dalam struktur kode ini, kelas `HewanPeliharaan` berperan sebagai superclass. Kelas ini mendefinisikan atribut-atribut dan *method* umum yang dimiliki oleh semua hewan peliharaan, yaitu atribut *private* `nama` dan `ras`, serta *method* `display()` untuk menampilkan data tersebut.

Kelas `Kucing` dan kelas `Anjing` berperan sebagai subclass. Kedua kelas ini menggunakan *keyword* `extends` untuk mewarisi sifat-sifat dari kelas `HewanPeliharaan`. Ini memodelkan hubungan "is-a", `Kucing` adalah

HewanPeliharaan, dan Anjing *adalah* HewanPeliharaan. Tujuannya agar tidak mendefiniskan ulang (code reuse) atribut di setiap kelas hewan.

Selain atribut yang diwarisi, *subclass* juga menambahkan atribut spesifiknya sendiri. Sesuai diagram, kelas Kucing menambahkan atribut warnaBulu. Kelas Anjing menambahkan atribut warnaBulu dan kemampuan. Ini menunjukkan bagaimana *subclass* dapat memperluas (extends) fungsionalitas dari *superclass*.

Sesuai ketentuan soal, inisialisasi objek harus menggunakan konstruktor. Pada *subclass* Kucing dan Anjing, konstruktor mereka memanggil super(r, n). Perintah super() ini berfungsi untuk meneruskan parameter yang relevan ke konstruktor *superclass*. Hal ini penting karena atribut nama dan ras bersifat private di *superclass* dan hanya dapat diinisialisasi melalui konstruktor *superclass* itu sendiri.

Soal Nomor 2d mensyaratkan bahwa detail dari *subclass* harus memanggil *method display* dari *parent class* terlebih dahulu. Ini diimplementasikan dalam *method displayDetailKucing()* dan *displayDetailAnjing()*. Kedua *method* tersebut pertama-tama memanggil super.display() untuk mencetak informasi nama dan ras. Setelah itu, barulah *method* tersebut mencetak detail spesifik milik *subclass* seperti warnaBulu atau kemampuan.

GITHUB

https://github.com/IndraSuryadilaga/Pemrograman_II