**LAPORAN PRAKTIKUM**

**PEMROGRAMAN II**

**MODUL 4**

****

**PEWARISAN/INHERITANCE & DIAGRAM KELAS**

**Oleh:**

**Aulia Az Zahra NIM. 2410817120021**

**PROGRAM STUDI TEKNOLOGI INFORMASI**

**FAKULTAS TEKNIK**

**UNIVERSITAS LAMBUNG MANGKURAT**

**SEPTEMBER 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 Prakitkum ini dikerjakan oleh:

Nama Praktikan : Aulia Az Zahra

NIM : 2410817120021

|  |  |
| --- | --- |
| Menyetujui,  Asisten Praktikum  Jovan Gilbert Natamasindah  NIM. 2310817310002 | Mengetahui,  Dosen Penanggung Jawab Praktikum  Irham Maulani Abdul Gani, S.Kom., M.Kom.  NIP. 199710312025061009 |

# DAFTAR ISI

[DAFTAR ISI 3](#_Toc211562803)

[DAFTAR GAMBAR 4](#_Toc211562804)

[DAFTAR TABEL 5](#_Toc211562805)

[SOAL 1 6](#_Toc211562806)

[A. Pembahasan 7](#_Toc211562807)

[SOAL 2 9](#_Toc211562808)

[A. Source Code 10](#_Toc211562809)

[B. Output Program 11](#_Toc211562810)

[C. Pembahasan 11](#_Toc211562811)

[SOAL 3 13](#_Toc211562812)

[A. Source Code 15](#_Toc211562813)

[B. Output Program 20](#_Toc211562814)

[C. Pembahasan 21](#_Toc211562815)

# DAFTAR GAMBAR

[Gambar 1. Screenshot Hasil Jawaban Soal No.1 8](#_Toc211562816)

[Gambar 2. Screenshot Hasil Jawaban Soal 2 11](#_Toc211562817)

[Gambar 3. Screenshot Hasil Jawaban Soal 3 20](#_Toc211562818)

[Gambar 4. Screenshot Hasil Jawaban Soal 3 20](#_Toc211562819)

# DAFTAR TABEL

[Table 1. Soal 2 9](#_Toc211562820)

[Table 2. Source Code Pet.java Soal No 2 10](#_Toc211562821)

[Table 3 Source Code Soal1Main.java Soal No 2 10](#_Toc211562822)

[Table 4. Soal 3 14](#_Toc211562823)

[Table 5. Source Code Cat.java Soal No 3 15](#_Toc211562824)

[Table 6. Source Code Dog.java Soal No 3 16](#_Toc211562825)

[Table 7. Source Code Pet.java Soal No 3 17](#_Toc211562826)

[Table 8. Source Code Soal2Main.java Soal No 3 18](#_Toc211562827)

# SOAL 1

Sebuah gambar berisi teks, struk, cuplikan layar

Konten yang dihasilkan AI mungkin salah.1. Bengkel Amay baru saja membuat sebuah sistem informasi berbasis online. Di dalam databasenya terdapat 3 table, yaitu: cars, ships, dan airplanes dengan bentuk sebagai berikut

Dari table-table database diatas, kamu diminta untuk:

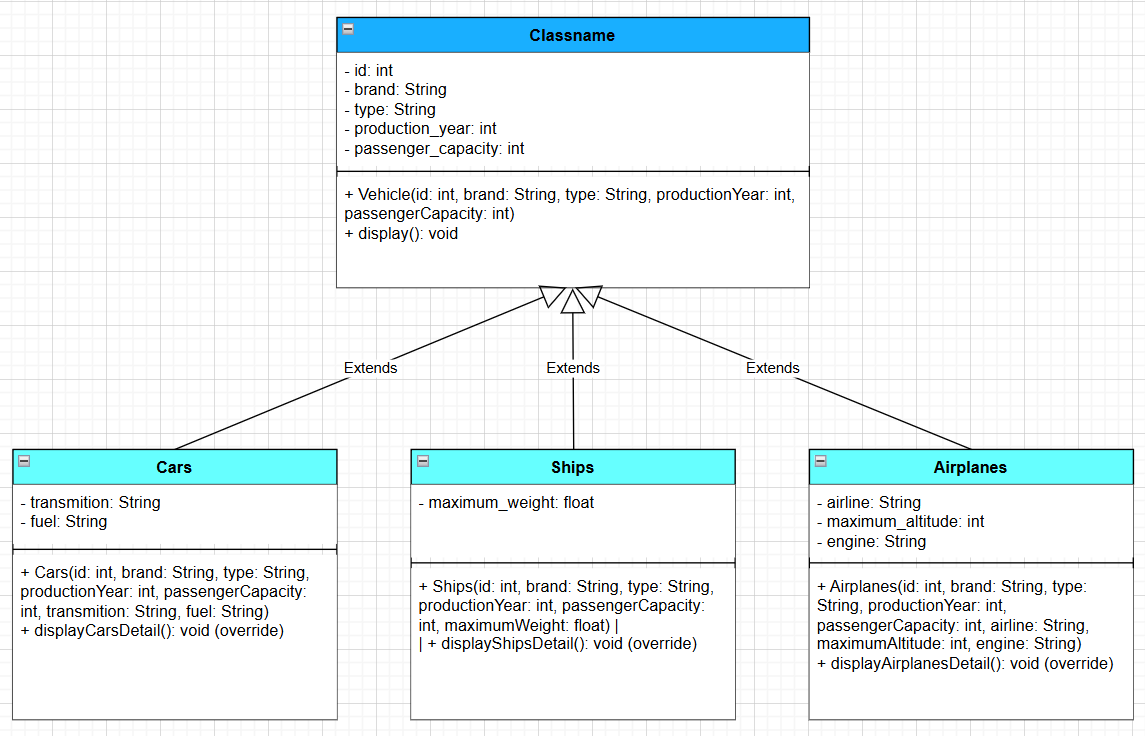
1. Melakukan abstraksi terhadap table-table tersebut dengan mencari atribut umum dan khusus di ketiganya. Jika sudah tulis di bagian pembahasan apa saja atribut hasil abstraksinya
2. Membuat class diagram dan lakukan inheritance (bentuknya kurang lebih seperti class diagram di nomor 2). Jadi nantinya akan terdapat 4 kotak didalam class diagram (Vehicle, Cars, Ships, dan Airplanes)

## Pembahasan

Abstraksi merupakan proses penyederhanaan yang dilakukan dengan mengidentifikasi serta memisahkan atribut atau karakteristik utama dari suatu objek untuk membentuk representasi umum yang dapat digunakan kembali. Dalam konteks pemodelan basis data dan pemrograman berorientasi objek, abstraksi berfungsi menemukan atribut yang bersifat **umum** (dimiliki semua entitas) dan **khusus** (hanya terdapat pada entitas tertentu). Tujuan utamanya ialah mengurangi duplikasi data sekaligus membangun struktur yang efisien dan teratur melalui konsep **inheritance** atau pewarisan.

Dalam proses analisis terhadap tiga tabel, yaitu **Cars**, **Ships**, dan **Airplanes**, dilakukan identifikasi atribut umum dan khusus. Hasilnya menunjukkan bahwa *id*, *brand*, *type*, *production\_year*, dan *passenger\_capacity* merupakan atribut **umum** yang dimiliki oleh semua tabel. Atribut-atribut ini menggambarkan ciri dasar setiap kendaraan, sehingga dikelompokkan ke dalam kelas induk bernama **Vehicle**, yang berfungsi sebagai representasi umum seluruh jenis kendaraan dengan menyimpan informasi pokok seperti identitas, merek, jenis, tahun produksi, serta kapasitas penumpang.

Setiap tabel juga memiliki atribut **khusus** yang hanya berlaku untuk jenis kendaraan tertentu. Pada kelas **Cars**, atribut khususnya adalah *transmission* dan *fuel*, sedangkan pada kelas **Ships** terdapat *maximum\_weight*, dan pada kelas **Airplanes** mencakup *airline*, *maximum\_altitude*, serta *engine*. Dari hasil abstraksi tersebut terbentuk struktur hierarki dengan **Vehicle** sebagai kelas induk dan ketiga kelas lainnya sebagai turunan. Model ini menerapkan prinsip **inheritance**, di mana atribut umum diwariskan, sedangkan atribut khusus tetap dikelola secara mandiri. Dengan demikian, desain sistem menjadi lebih efisien, terstruktur, dan mudah dikembangkan tanpa redundansi data.



Gambar 1. Screenshot Hasil Jawaban Soal No.1

# SOAL 2

Sebuah gambar berisi teks, cuplikan layar, Font, struk

Konten yang dihasilkan AI mungkin salah.2. 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:

a. Inisiasi objek harus menggunakan konstruktor.

b. Output ditampilkan dengan hanya memanggil method display.

c. Tidak boleh ada fungsi mencetak di class main.

Table 1. Soal 2

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

## Source Code

Table 2. Source Code Pet.java Soal No 2

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 | package praktikum;  public class Pet {  private String name;  private String race;   public Pet(String name, String race) {  this.name = name;  this.race = race;  }   public void display() {  System.*out*.println("Detail Hewan Peliharaan:");  System.*out*.println("Nama hewan peliharaanku adalah: " + name);  System.*out*.println("Dengan ras: " + race);  } } |

Table 3 Source Code Soal1Main.java Soal No 2

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18 | package praktikum;  import java.util.Scanner;  public class Soal1Main {  public static void main(String[] args) {  Scanner input = new Scanner(System.*in*);   System.*out*.print("Nama Hewan Peliharaan: ");  String petName = input.nextLine();   System.*out*.print("Ras: ");  String petRace = input.nextLine();   Pet pet = new Pet(petName, petRace);  pet.display();  } } |

## Output Program Sebuah gambar berisi teks, cuplikan layar, software Konten yang dihasilkan AI mungkin salah.

Gambar 2. Screenshot Hasil Jawaban Soal 2

## Pembahasan

**Pet.java:**

* Pada baris 1: *package praktikum;*  Menyatakan bahwa file ini berada dalam package bernama praktikum.
* Pada baris 3 : public class Pet { Mendefinisikan kelas Pet sebagai public, artinya bisa diakses dari luar package.
* Pada baris 4 – 5 : Dua atribut privat: nama hewan peliharaan, ras atau jenis hewan. tribut bersifat private, artinya hanya bisa diakses melalui metode getter/setter
* Pada baris 7 – 9 : menerima dua parameter: *name* dan *race.* Mengisi atribut name dan race menggunakan this, yang merujuk pada objek saat ini. memungkinkan objek Pet dibuat langsung dengan nama dan ras saat instansiasi.
* Pada baris 12 : public void display() { Metode display() digunakan untuk menampilkan informasi hewan peliharaan.
* Pada baris 13 - 15 : Menampilkan header informasi. Menampilkan nama hewan. Menampilkan ras hewan.

**Soal1Main.java:**

* Pada baris 1: *package praktikum;*  Menyatakan bahwa file ini berada dalam package bernama praktikum.
* Pada baris 3 : *import java.util.Scanner;* Mengimpor kelas Scanner dari Java untuk membaca input dari pengguna.
* Pada baris 5 : *public class Soal1Main {* Mendefinisikan kelas utama bernama
* Pada baris 6 : public static void main(String[] args) { Metode main() adalah titik awal eksekusi program.
* Pada baris 7 : *Scanner input = new Scanner(System.in);* Membuat objek Scanner untuk membaca input dari pengguna.
* Pada baris 9 – 13 : Menampilkan dan membaca nama & ras hewan peliharaan dari pengguna.
* Pada baris 17 : *pet.display();* Memanggil metode display() dari objek

# SOAL 3

3. Diberikan class diagram seperti berikut: Sebuah gambar berisi teks, diagram, garis, cuplikan layar

Konten yang dihasilkan AI mungkin salah.

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

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

Table 4. Soal 3

|  |
| --- |
| **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 |

## Source Code

Table 5. Source Code Cat.java Soal No 3

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18 | package praktikum;  public class Cat extends Pet {  private String furColor;   public Cat(String name, String race, String furColor) {  super(name, race);  this.furColor = furColor;  }   @Override  public void display() {  System.*out*.println("Detail Hewan Peliharaan:");  System.*out*.println("Nama hewan peliharaanku adalah: " + getName());  System.*out*.println("Dengan ras: " + getRace());  System.*out*.println("Memiliki warna bulu: " + furColor);  } } |

Table 6. Source Code Dog.java Soal No 3

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24 | package praktikum;  public class Dog extends praktikum.Pet {  private String furColor;  private String[] abilities;   public Dog(String name, String race, String furColor, String[] abilities) {  super(name, race);  this.furColor = furColor;  this.abilities = abilities;  }   @Override  public void display() {  System.*out*.println("Detail Hewan Peliharaan:");  System.*out*.println("Nama hewan peliharaanku adalah: " + getName());  System.*out*.println("Dengan ras: " + getRace());  System.*out*.println("Memiliki warna bulu: " + furColor);  System.*out*.print("Kemampuan: ");  for (String ability : abilities) {  System.*out*.print(ability);  }  } } |

Table 7. Source Code Pet.java Soal No 3

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25 | package praktikum;  public class Pet {  private String name;  private String race;   public Pet(String name, String race) {  this.name = name;  this.race = race;  }   public String getName() {  return name;  }   public String getRace() {  return race;  }   public void display() {  System.*out*.println("Detail Hewan Peliharaan:");  System.*out*.println("Nama hewan peliharaanku adalah: " + name);  System.*out*.println("Dengan ras: " + race);  } } |

Table 8. Source Code Soal2Main.java Soal No 3

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51 | package praktikum;  import java.util.Scanner;  public class Soal2Main {  public static void main(String[] args) {  Scanner input = new Scanner(System.*in*);   System.*out*.println("Pilih jenis hewan yang ingin diinputkan:");  System.*out*.println("1 = Kucing");  System.*out*.println("2 = Anjing");  System.*out*.print("Masukkan pilihan: ");   int choice = input.nextInt();  input.nextLine();   if (choice == 1) {  System.*out*.print("Nama Hewan Peliharaan: ");  String petName = input.nextLine();   System.*out*.print("Ras: ");  String petRace = input.nextLine();   System.*out*.print("Warna Bulu: ");  String furColor = input.nextLine();   Cat cat = new Cat(petName, petRace, furColor);  cat.display();  }  else if (choice == 2) {  System.*out*.print("Nama Hewan Peliharaan: ");  String petName = input.nextLine();   System.*out*.print("Ras: ");  String petRace = input.nextLine();   System.*out*.print("Warna Bulu: ");  String furColor = input.nextLine();   System.*out*.print("Kemampuan: ");  String ability = input.nextLine();  String[] abilities = ability.split(",");   Dog dog = new Dog(petName, petRace, furColor, abilities);  dog.display();  }  else {  System.*out*.print("Input tidak valid");  }  } } |

## Output Program Sebuah gambar berisi teks, cuplikan layar, software, Software multimedia Konten yang dihasilkan AI mungkin salah.Sebuah gambar berisi teks, cuplikan layar Konten yang dihasilkan AI mungkin salah.

Gambar 3. Screenshot Hasil Jawaban Soal 3

Gambar 4. Screenshot Hasil Jawaban Soal 3

## Pembahasan

**Cat.Java:**

* Pada baris 1: *package praktikum;*  Menyatakan bahwa file ini berada dalam package bernama praktikum
* Pada baris 3: *public class Cat extends Pet {* Mendefinisikan kelas Cat yang merupakan turunan (subclass) dari kelas Pet.
* Pada baris 4 : *private String furColor;* Mendeklarasikan atribut privat furColor untuk menyimpan warna bulu kucing.Atribut ini bersifat private, jadi hanya bisa diakses dari dalam kelas Cat
* Pada baris 6: *public Cat(String name, String race, String furColor){* Konstruktor untuk membuat objek Cat dengan parameter name, race, dan furColor.
* Pada baris 7: *super(name, race);* Memanggil konstruktor dari kelas induk Pet menggunakan super(), untuk mengisi atribut name dan race.
* Pada baris 8: *this.furColor = furColor;* Mengisi atribut furColor milik Cat. Subclass memanggil konstruktor superclass untuk menginisialisasi bagian yang diwarisi.
* Pada baris 11: *@Override* Menandai bahwa metode display() di sini meng-override metode display() dari kelas Pet. Overriding memungkinkan subclass memodifikasi perilaku metode yang diwarisi.
* Pada baris 12 : *public void display() {* Mendefinisikan metode display() untuk menampilkan detail hewan peliharaan.
* Pada baris 14 : *System.out.println("Nama hewan peliharaanku adalah: " + getName());* Mengambil nama hewan dari metode getName() yang diwarisi dari Pet.
* Pada baris 15 : *System.out.println("Dengan ras: " + getRace());* Mengambil ras hewan dari metode getRace() yang juga diwarisi.
* Pada baris 16 : *System.out.println("Memiliki warna bulu: " + furColor);* Menampilkan warna bulu dari atribut furColor milik Cat.

**Dog.java:**

* Pada baris 1: *package praktikum;*  Menyatakan bahwa file ini berada dalam package bernama praktikum.
* Pada baris 3 : Mendefinisikan kelas Dog sebagai subclass dari Pet. Dog mewarisi atribut dan metode dari Pet, seperti name, race, dan getName().
* Pada baris 4 – 5 : *furColor*: warna bulu anjing. *abilities:* array berisi kemampuan khusus anjing (misalnya: "berlari", "melompat").
* Pada baris 7 : Konstruktor menerima 4 parameter: nama, ras, warna bulu, dan daftar kemampuan.
* Pada baris 8 : *super(name, race);* Memanggil konstruktor dari kelas induk Pet untuk mengisi name dan race.
* Pada baris 9 – 10 : Mengisi atribut khusus milik Dog. subclass memanggil konstruktor superclass.
* Pada baris 13 – 14 : *@Override* Menandai bahwa metode ini meng-override display() dari kelas Pet.
* Pada baris 15 – 19 : Menampilkan header informasi. Mengambil nama dari metode getName() milik Pet. Mengambil ras dari metode getRace() milik Pet. Menampilkan warna bulu dari atribut furColor. Menampilkan label kemampuan.
* Pada baris 20 – 21 : Menampilkan semua kemampuan dalam satu baris.

**Pet.java**

* Pada baris 1: *package praktikum;*  Menyatakan bahwa file ini berada dalam package bernama praktikum.
* Pada baris 3 : public class Pet { Mendefinisikan kelas Pet sebagai public, artinya bisa diakses dari luar package.
* Pada baris 4 – 5 : Dua atribut privat: nama hewan peliharaan, ras atau jenis hewan. tribut bersifat private, artinya hanya bisa diakses melalui metode getter/setter
* Pada baris 7 – 9 : menerima dua parameter: *name* dan *race.* Mengisi atribut name dan race menggunakan this, yang merujuk pada objek saat ini. memungkinkan objek Pet dibuat langsung dengan nama dan ras saat instansiasi.
* Pada baris 12 – 18 : Mengambil nilai name dan race dari objek
* Pada baris 20 : public void display() { Metode display() digunakan untuk menampilkan informasi hewan peliharaan.
* Pada baris 21 – 25 : Menampilkan header informasi. Menampilkan nama hewan. Menampilkan ras hewan.

**Soal2Main.java**

* Pada baris 1: *package praktikum;*  Menyatakan bahwa file ini berada dalam package bernama praktikum.
* Pada baris 3 : *import java.util.Scanner;* Mengimpor kelas Scanner dari Java untuk membaca input dari pengguna.
* Soal2Main. Pada baris 5 : *public class Soal2Main {* Mendefinisikan kelas utama bernama
* Pada baris 6 : public static void main(String[] args) { Metode main() adalah titik awal eksekusi program.
* Pada baris 7 : *Scanner input = new Scanner(System.in);* Membuat objek Scanner untuk membaca input dari pengguna.
* Pada baris 9 – 12 : Menampilkan menu pilihan jenis hewan peliharaan.
* Pada baris 14 – 15 : Membaca input angka dari pengguna dan menyimpan dalam variabel choice. *input.nextLine()* digunakan untuk membersihkan newline agar input berikutnya tidak terlewat.
* Pada baris 17 : *if (choice == 1) {* Mengecek apakah pengguna memilih kucing.
* Pada baris 18 – 25 : Membaca nama kucing. Membaca ras kucing. Membaca warna bulu kucing.
* Pada baris 27 : *Cat cat = new Cat(petName, petRace, furColor);* Membuat objek Cat dengan data yang dimasukkan.
* Pada baris 28 : *cat.display();* Menampilkan informasi kucing dengan memanggil metode display() dari kelas Cat.
* Pada baris 30 : *else if (choice == 2) {* Mengecek apakah pengguna memilih anjing.
* Pada baris 31 – 38 : Membaca nama, ras, dan warna bulu anjing.
* Pada baris 40 – 42 : Membaca daftar kemampuan anjing dalam format teks dipisahkan koma, lalu diubah menjadi array *String[].*
* Pada bari 44 – 45 : Membuat objek Dog dan menampilkan informasinya.
* Pada baris 47 – 48 : *else { System.out.print("Input tidak valid"); }* Menangani kasus jika pengguna memasukkan angka selain 1 atau 2.

# TAUTAN GIT