**Soal Pertama**

Jelaskan apa itu Inheritance, Bagaimana cara penerapannya serta aturan apa saja yang harus diperhatikan dalam menerapkan konsep ini?

**Jawaban**

Inheritance merupakan suatu konsep yang mewariskan/ menurunkan suatu karakteristik class ke class yang lain. Cara dan aturan penerapannya adalah

* Dalam Inheritance, suatu constructor tidak dapat diwariskan.
* Sub-class dapat mengakses super-class dengan keyword super (Merupakan keyword yang digunakan untuk memanggil constructor dari super-class atau menjadi variabel yang mengacu pada super-class).
* Ketika sub-class dapat mengakses atau mewarisi method dan atribut pada super-class diperlukan menuliskan keyword extends (merupakan keyword yang digunakan untuk menjadikan suatu class sebagai sub-class sehingga dapat mewarisi method dan atribut super-class).

**Soal Kedua**

Perhatikan kode-kode berikut

1. class Kucing extends HewanMamalia
2. class Gajah extend HewanKarnivora
3. class Kelelawar extends HewanMamalia, HewanBersayap
4. class Garuda extends HewanBersayap
5. class Katak extends HewanBersayap

Manakah penggunaan Inheritance yang benar dan memiliki hubungan Generalisasi-Spesialisasi yang masuk akal? berikan penjelasanmu!

**Jawaban**

Yang masuk akal terkait dengan Generalisasi-Spesialisasi dari beberapa kode tersebut adalah opsi a dan d. Penyebabnya adalah pada opsi b penulisan “extend” salah, seharusnya “extends” kemudian pada opsi yang sama Class Gajah berextends dengan HewanKarnivora tidak sesuai karena Gajah merupakan hewan herbivora. Lalu, pada opsi c pemrograman java tidak mendukung multiple inheritances, jika dipaksakan maka akn terjadi error. Untuk solusinya bisa menggunakan interfaces. Lalu, pada opsi e tidak sesuai karena katak merupakan bukan hewan bersayap, melainkan hewan amfibi.

**Soal Ketiga**

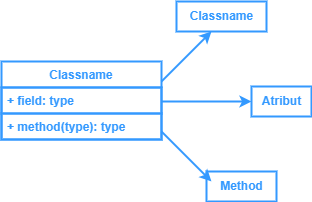
Apa itu Class Diagram dan UML? Serta dalam 1 (satu)Model UML tedapat 3 bagian, gambarkan dan jelaskan apa saja bagian-bagian tersebut!

**Jawaban**

UML merupakan salah satu tools yang paling banyak digunakan untuk membuat model struktur dan perilaku aplikasi.

Class Diagram merupakan model kelas dalam UML berbentuk persegi dengan 3 bagian yaitu nama kelas, properti, dan method.

Gambaran model UML:



Penjelasan bagian dalam model UML:

* Classname: Berisi nama dari class.
* Atribut: berisi nama – nama atribut yang digunakan.
* Method: Berisi proses – proses yang dilakukan dalam program.

**Soal Keempat**

Gambarkan relasi-relasi berikut beserta atribut dan method-methodnya! (pilih minimal 2):

1. **Association Berarah**: Class Pesawat dengan Class Penumpang
2. **Inheritance**: Class Hewan dengan Class Kucing dan Anjing
3. **Realization**: Class Iroda denfan Class Roda
4. **Dependency**: Class Laptop dengan Class Mouse
5. **Aggregation**: Class Kamera dengan Class Lensa
6. **Composition**: Class Buku dengan Class Penulis

**Jawaban**

Relasi yang dipilih:

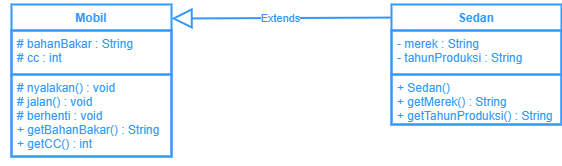
1. Association Berarah

Relasi antara Class Mobil dengan Class Penumpang



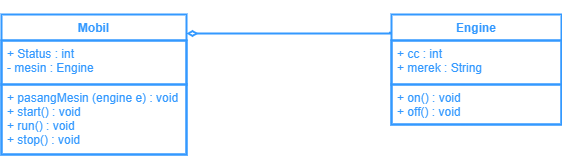
1. Inheritance

Relasi antara Class Mobil dengan Class Sedan



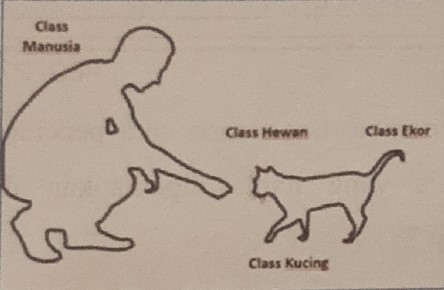
1. Aggregation

Relasi antara Class Mobil dengan Class Engine



**Soal Kelima**

Perhatikan gambar berikut.



(NPM Ganjil)

Sebutkan manakah Class yang memiliki Relasi Association, Relasi Aggregation/ Composite, dan Relasi Inheritance/Realization serta berikan alasannya!

**Jawaban**

(NPM Ganjil)

Hubungan relasi – relasi:

* Relasi **Inheritance**/ Realization

Class Hewan dengan Class Kucing

Karena Class Hewan bertindak sebagai Class induk dan Class Kucing sebagai anakannya. Maksudnya untuk yang induk adalah Kucing masuk dalam kategori hewan.

* Relasi Aggregation/ **composite**

Class Kucing dengan Class Ekor

Karena hubungan antara Class Kucing dengan Class Ekor merupakan hubungan khusus, yakni Class Ekor merupakan bagian dari Class Kucing.

* Relasi Association

Class manusia dengan Class Kucing

Karena hubungan antara Class Manusia dengan Class Kucing merupakan relasi yang logis dan tampak secara lansung dalam kehidupan sehari – hari.

**Soal Keenam**

**Progress Studi Kasus Project Akhir:** Berdasarkan studi kasus kalian coba buatlah Class Diagram (setidaknya mengandung relasi Inheritance) kemudian terapkan relasi inheritance tersebut ke dalam project code kalian!

**Source Code**

Class Transaksi.java

public class Transaksi

{

protected String jenisTransaksi, nominalTransaksi, totalBayar, tanggalTransaksi;

public Transaksi(){

}

public Transaksi(String jenis, String nominal, String total){

jenisTransaksi = jenis;

nominalTransaksi = nominal;

totalBayar = total;

tanggalTransaksi = DateString.now();

}

public void dataTransaksi(){

System.out.println("Jenis Transaksi: "+jenisTransaksi);

System.out.println("Tanggal Transaksi: "+tanggalTransaksi);

System.out.println("Nominal Transaksi: "+nominalTransaksi);

}

}

Class Transfer.java

public class Transfer extends Transaksi{

int nominal;

User userAsal, userTujuan;

public Transfer(int nominal, User userAsal, User userTujuan, String jenisTransaksi, String nominalTransaksi, String totalBayar){

this.nominal = nominal;

this.userAsal = userAsal;

this.userTujuan = userTujuan;

super.jenisTransaksi = jenisTransaksi;

super.nominalTransaksi = nominalTransaksi;

super.totalBayar = totalBayar;

}

}

Class SetorTunai.java

public class SetorTunai extends Transaksi {

int nominalSetor;

public SetorTunai(int nominalSetor, String jenisTransaksi, String nominalTransaksi, String totalBayar){

this.nominalSetor = nominalSetor;

super.jenisTransaksi = jenisTransaksi;

super.nominalTransaksi = nominalTransaksi;

super.totalBayar = totalBayar;

}

}

Class TarikTunai.java

public class TarikTunai extends Transaksi{

int nominalTarik;

public TarikTunai(int nominalTarik, String jenisTransaksi, String nominalTransaksi, String totalBayar){

this.nominalTarik = nominalTarik;

super.jenisTransaksi = jenisTransaksi;

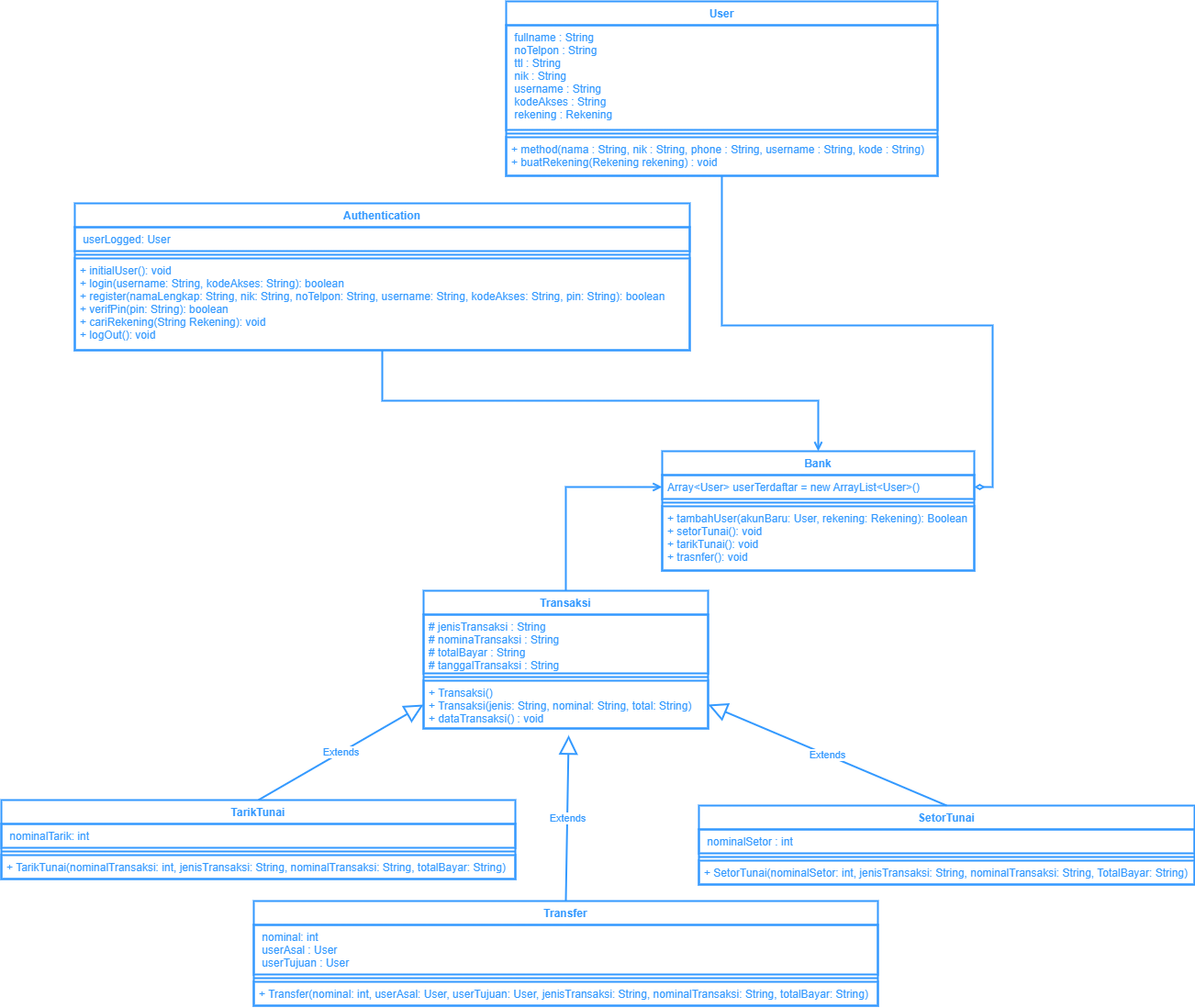
super.nominalTransaksi = nominalTransaksi;

super.totalBayar = totalBayar;

}

}

**Diagram Program**

****