

MODUL & GUIDED

Relasi Kelas

2023



JavaTM

RELASI KELAS

Relasi kelas merupakan sebuah hal yang sangat mendasar di dalam pemrograman berorientasi objek. Relasi kelas mendefinisikan bagaimana kelas-kelas tersebut saling berinteraksi dan berkolaborasi untuk mewujudkan sistem yang utuh. Berikut beberapa jenis relasi yang mungkin terjadi.

1. Pewarisan

Sudah dipelajari di modul 4 Pewarisan.

2. Agregasi

a. Agregasi

Agregasi merupakan sebuah relasi has-a yang dapat dipahami sebagai suatu objek tersusun atau terdiri dari objek lain. Pada relasi agregasi, hubungan antar objek bersifat **lemah**, sehingga objek penyusunnya bisa tetap ada tanpa objek agregat. Dengan demikian, walau objek agregat dihancurkan atau dihilangkan, hal ini tidak akan berdampak pada objek penyusunnya karena objek penyusun ini bersifat independent. Relasi ini digambarkan dengan simbol diamond kosong.

b. Komposisi

Komposisi merupakan sebuah relasi has-a yang dapat dipahami sebagai suatu objek tersusun atau terdiri dari objek lain. Komposisi memiliki karakteristik berbanding terbalik dengan relasi agregasi. Pada relasi komposisi, hubungan antar objek pada relasi komposisi bersifat **kuat**, sehingga objek penyusunnya tidak bisa ada tanpa objek komposit. Artinya objek penyusun dependent terhadap objek kompositnya. sehingga objek penyusun tidak bisa berdiri sendiri tanpa adanya objek komposit. Jika objek komposit dihancurkan atau dihilangkan, maka objek penyusunnya juga akan hancur atau hilang. Relasi ini digambarkan dengan simbol diamond terisi penuh.

3. Asosiasi

Asosiasi adalah hubungan antar objek di mana objek dari suatu kelas dihubungkan ke objek dari kelas lain. Secara konsep, asosiasi berbeda dengan agregasi, namun dalam implementasinya secara sintaks tidak memiliki perbedaan dengan Agregasi. Perbedaannya,

asosiasi bisa bersifat dua arah. Asosiasi dua arah adalah relasi dengan multiplicity many to many.

4. Dependency

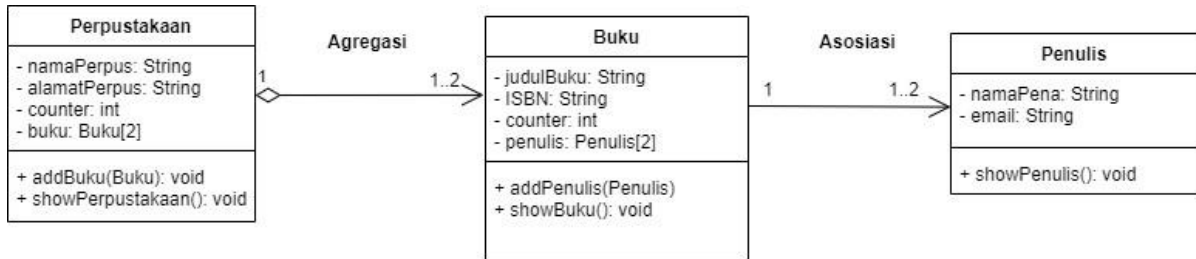
Dependency adalah hubungan antara dua buah kelas, di mana ketika kelas A bergantung pada kelas B, jika kelas B mengalami perubahan maka kelas A akan terkena dampak perubahan tersebut. Contoh:

- Penggunaan kelas B sebagai parameter pada fungsi di kelas A.
- Penggunaan kelas B sebagai nilai balikan pada fungsi di kelas A.
- Penggunaan kelas B sebagai variabel lokal pada fungsi di kelas A.

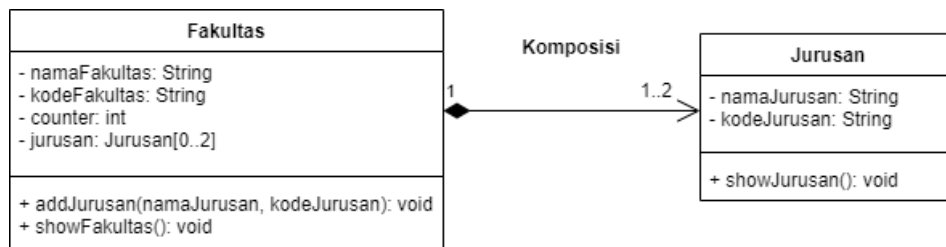
GUIDED

Pada guided kali ini kita akan membuat kelas-kelas yang akan mengimplementasikan 3 jenis relasi yaitu:

- Relasi agregasi antara kelas Perpustakaan dan kelas Buku di mana kelas Perpustakaan sebagai agregat dan kelas Buku sebagai penyusunnya.

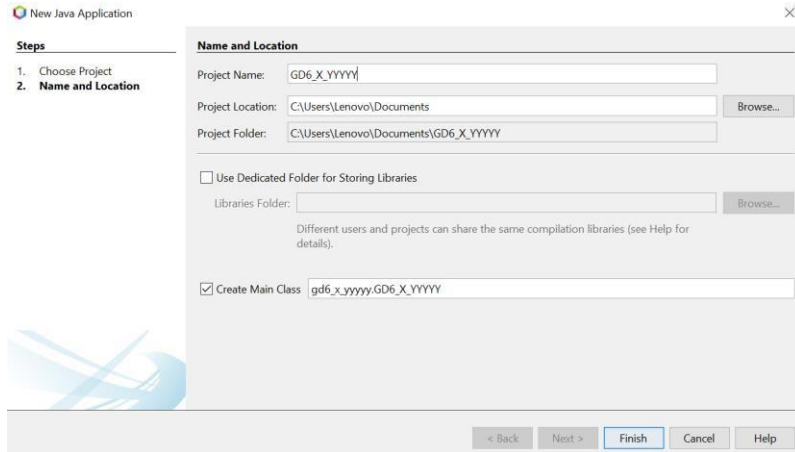


- Relasi asosiasi satu arah antara kelas Buku dan kelas Penulis di mana kelas Buku akan dihubungkan dengan kelas Penulis.
- Relasi komposisi antara kelas Fakultas dan kelas Jurusan di mana kelas Fakultas sebagai komposit dan kelas Jurusan sebagai penyusunnya.

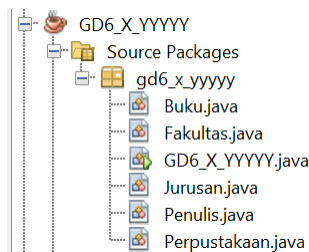


Langkah-langkah:

1. Buatlah sebuah project baru pada netbeans dengan nama GD6_X_YYYYY (X = kelas dan YYYYYY = 5 digit NPM terakhir).



2. Buatlah 5 kelas baru dengan nama **Perpustakaan**, **Buku**, **Penulis**, **Fakultas**, dan **Jurusan**.



3. Pada bagian ini, kita akan menyelesaikan relasi agregasi dan asosiasi terlebih dahulu, kemudian dilanjutkan dengan relasi komposisi. Untuk alur pengerjaannya, pertama kita akan menuliskan code untuk **kelas Penulis**.

```
public class Penulis {  
    private String namaPena, email;  
  
    public Penulis(String namaPena, String email) {  
        this.namaPena = namaPena;  
        this.email = email;  
    }  
  
    public void showPenulis() {  
        System.out.println("\t\tNama Pena Penulis : " + namaPena);  
        System.out.println("\t\tEmail : " + email);  
    }  
}
```

4. Selanjutnya tuliskan code untuk **kelas Buku** yang merupakan penyusun dari kelas Perpustakaan.

```
public class Buku {
    private String judulBuku, ISBN;
    private Penulis penulis[]; // deklarasi array penulis
    private int counter=0;

    public Buku(String judulBuku, String ISBN) {
        this.judulBuku = judulBuku;
        this.ISBN = ISBN;
        penulis = new Penulis[2]; // deklarasi panjang array penulis
    }

    /* digunakan untuk menyalin objek penulis yang dibuat di luar
    kelas Buku ke dalam array penulis yang dideklarasikan di atas*/
    public void addPenulis(Penulis p){
        penulis[counter] = p;
        counter++;
    }

    public void showBuku(){
        System.out.println("\tJudul Buku      : " +judulBuku);
        System.out.println("\tISBN          : " +ISBN);
        System.out.println("\tJumlah Penulis : " +counter);
        for(int i=0; i<penulis.length; i++){
            System.out.println("\n\t\t***** Tampil Penulis *****");
            System.out.println("\t\tData Penulis ke    : " +(i+1));
            penulis[i].showPenulis();
        }
    }
}
```

5. Selanjutnya, tuliskan code berikut untuk **kelas Perpustakaan** yang merupakan agregat dari kelas Buku.

```
public class Perpustakaan {
    private String namaPerpus, alamatPerpus;
    private int counter=0;
    private Buku buku[]; // deklarasi array buku

    public Perpustakaan(String namaPerpus, String alamatPerpus) {
        this.namaPerpus = namaPerpus;
        this.alamatPerpus = alamatPerpus;
        buku = new Buku[2]; // deklarasi panjang array buku
    }

    /* digunakan untuk menyalin objek buku yang dibuat di luar
    kelas Perpustakaan ke dalam array buku yang di dideklarasikan di atas*/
    public void addBuku(Buku b){
        buku[counter] = b;
        counter++;
    }
}
```

```

public void showPerpustakaan(int menu){
    System.out.println("\n***** Tampil Perpustakaan *****");
    System.out.println("Nama Perpustakaan   : " +namaPerpus);
    System.out.println("Alamat Perpustakaan : " +alamatPerpus);
    System.out.println("Jumlah Buku         : " +counter);
    for(int i=0; i<buku.length; i++){
        System.out.println("\n\t***** Tampil Buku *****");
        System.out.println("\tData Buku ke   : " +(i+1));
        buku[i].showBuku();
    }
}
}

```

Catatan: ketika method showPerpustakaan dipanggil, maka data yang ditampilkan adalah data Perpustakaan berserta seluruh data yang ada pada method ShowBuku (data Buku dan data Penulis).

6. Karena relasi asosiasi dan agregasi telah kita selesaikan, maka selanjutnya kita akan mengerjakan relasi komposisi. Berikut merupakan code untuk **kelas Jurusan** yang merupakan penyusun dari kelas Fakultas.

```

public class Jurusan {
    private String namaJurusan;
    private String kodeJurusan;

    public Jurusan(String namaJurusan, String kodeJurusan) {
        this.namaJurusan = namaJurusan;
        this.kodeJurusan = kodeJurusan;
    }

    public void showJurusan() {
        System.out.println("\tNama Jurusan   : "+namaJurusan);
        System.out.println("\tKode Jurusan  : "+kodeJurusan);
    }
}

```

7. Pada **kelas Fakultas**, tambahkan code di bawah ini.

```

public class Fakultas {
    private String namaFakultas, kodeFakultas;
    private Jurusan jurusan[]; // deklarasi array jurusan
    private int counter=0;

    public Fakultas(String namaFakultas, String kodeFakultas) {
        this.namaFakultas = namaFakultas;
        this.kodeFakultas = kodeFakultas;
        jurusan = new Jurusan[2]; // deklarasi panjang array jurusan
    }

    /* digunakan untuk membuat objek jurusan di dalam kelas Fakultas*/
    public void addJurusan(String namaJurusan, String kodeJurusan){
        jurusan[counter] = new Jurusan(namaJurusan, kodeJurusan);
        counter++;
    }

    public void showFakultas() {
        System.out.println("\n***** Tampil Fakultas *****");
        System.out.println("Nama Fakultas : " +namaFakultas);
        System.out.println("Kode Fakultas : " +kodeFakultas);
        System.out.println("Jumlah Jurusan : " +counter);
        System.out.println("\n\t***** Tampil Jurusan *****");
        for(int i=0; i<counter; i++){
            System.out.println("\tData Jurusan ke : " +(i+1));
            jurusan[i].showJurusan();
        }
    }
}

```

8. Pada kelas GD6_X_YYYYY (main class), tuliskan code berikut agar kita dapat menggunakan class bawaan java yang berfungsi untuk menerima inputan user.

```

package gd6_x_yyyyy;
import java.io.BufferedReader;
import java.io.InputStreamReader;

```

9. Selanjutnya, buatlah objek baru BufferedReader dengan nama br.

```

public static void main(String[] args) {
    BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
}

```

10. Berikutnya, tambahkan code berikut pada main class.


```

public static void main(String[] args) {
    BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
    String namaPerpus, alamatPerpus, judul, isbn;
    String namaFakultas, kodeFakultas, namaJurusan, kodeJurusan;
    String namaPenulis, email;
    int menu;

    try{
        do{
            System.out.println("[1]. Relasi Agregasi dan Asosiasi");
            System.out.println("[2]. Relasi Komposisi");
            System.out.println("[0]. Keluar");
            System.out.print("Pilihan : ");
            menu = Integer.parseInt(br.readLine());
            switch(menu){
                case 1:
                    System.out.println("***** Relasi Agregasi dan Asosiasi *****");
                    System.out.print("Nama Perpustakaan : "); namaPerpus = br.readLine();
                    System.out.print("Alamat Perpustakaan : "); alamatPerpus = br.readLine();
                    Perpustakaan perpustakaan = new Perpustakaan(namaPerpus, alamatPerpus);

                    for(int i=0; i<2; i++){
                        System.out.println("\nInput data Buku ke : " + (i+1));
                        System.out.print("Judul Buku : "); judul = br.readLine();
                        System.out.print("Nomor ISBN : "); isbn = br.readLine();
                        Buku buku = new Buku(judul, isbn);
                        perpustakaan.addBuku(buku);
                    }

                    for(int j=0; j<2; j++){
                        System.out.println("\nInput data Penulis ke : " + (j+1));
                        System.out.print("Nama Penulis : "); namaPenulis = br.readLine();
                        System.out.print("Email Penulis : "); email = br.readLine();
                        Penulis penulis = new Penulis(namaPenulis, email);
                        buku.addPenulis(penulis);
                    }
                }
                perpustakaan.showPerpustakaan(menu);
                break;

                case 2:
                    System.out.println("***** Relasi Komposisi *****");
                    System.out.print("Nama Fakultas : "); namaFakultas = br.readLine();
                    System.out.print("Kode Fakultas : "); kodeFakultas = br.readLine();
                    Fakultas fakultas = new Fakultas(namaFakultas, kodeFakultas);
                    for(int i=0; i<2; i++){
                        System.out.println("\nInput Jurusan ke : " + (i+1));
                        System.out.print("Nama Jurusan : "); namaJurusan = br.readLine();
                        System.out.print("Kode Jurusan : "); kodeJurusan = br.readLine();
                        fakultas.addJurusan(namaJurusan, kodeJurusan);
                    }
                    fakultas.showFakultas();
                    break;
            }
        }while(menu != 0);
    }catch(Exception e){}
}

```

11. Selamat! Kita telah menyelesaikan guided ini. Silakan run project, jika benar maka output dari gd ini adalah sebagai berikut.

a. Jika inputan pilihan = 1

Input	Output
<pre> run: [1]. Relasi Agregasi dan Asosiasi [2]. Relasi Komposisi [0]. Keluar Pilihan : 1 ***** Relasi Agregasi dan Asosiasi ***** Nama Perpustakaan : Perpustakaan UAJY Alamat Perpustakaan : Babarsari Input data Buku ke : 1 Judul Buku : Teori Komunikasi Nomor ISBN : 43-213-45-232 Input data Penulis ke: 1 Nama Penulis : Ari Email Penulis : ari@gmail.com Input data Penulis ke: 2 Nama Penulis : Ira Email Penulis : ira@gmail.com Input data Buku ke : 2 Judul Buku : Manajemen Bisnis Nomor ISBN : 28-434-56-257 Input data Penulis ke: 1 Nama Penulis : Angga Email Penulis : angga@gmail.com Input data Penulis ke: 2 Nama Penulis : Anggi Email Penulis : anggi@gmail.com </pre>	<pre> ***** Tampil Perpustakaan ***** Nama Perpustakaan : Perpustakaan UAJY Alamat Perpustakaan : Babarsari Jumlah Buku : 2 ***** Tampil Buku ***** Data Buku ke : 1 Judul Buku : Teori Komunikasi ISBN : 43-213-45-232 Jumlah Penulis : 2 ***** Tampil Penulis ***** Data Penulis ke : 1 Nama Pena Penulis : Ari Email : ari@gmail.com ***** Tampil Penulis ***** Data Penulis ke : 2 Nama Pena Penulis : Ira Email : ira@gmail.com ***** Tampil Buku ***** Data Buku ke : 2 Judul Buku : Manajemen Bisnis ISBN : 28-434-56-257 Jumlah Penulis : 2 ***** Tampil Penulis ***** Data Penulis ke : 1 Nama Pena Penulis : Angga Email : angga@gmail.com ***** Tampil Penulis ***** Data Penulis ke : 2 Nama Pena Penulis : Anggi Email : anggi@gmail.com </pre>

b. Jika inputan pilihan = 2

```
[1]. Relasi Agregasi dan Asosiasi
[2]. Relasi Komposisi
[0]. Keluar
Pilihan : 2
***** Relasi Komposisi *****
Nama Fakultas      : Fakultas Teknologi Industri
Kode Fakultas      : FTI

Input Jurusan ke : 1
Nama Jurusan      : Informatika
Kode Jurusan      : INF

Input Jurusan ke : 2
Nama Jurusan      : Sistem Informasi
Kode Jurusan      : SI

***** Tampil Fakultas *****
Nama Fakultas      : Fakultas Teknologi Industri
Kode Fakultas      : FTI
Jumlah Jurusan    : 2

      ***** Tampil Jurusan *****
      Data Jurusan ke : 1
      Nama Jurusan    : Informatika
      Kode Jurusan    : INF
      Data Jurusan ke : 2
      Nama Jurusan    : Sistem Informasi
      Kode Jurusan    : SI
```

Kumpulkan dalam bentuk zip dengan nama **GD6_X_YYYYYY.zip** (X = Kelas, YYYYYY = 5 digit NPM terakhir).