

RELASI KELAS

PERTEMUAN 4 PEMROGRAMAN BERORIENTASI OBJEK



JURUSAN
TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG

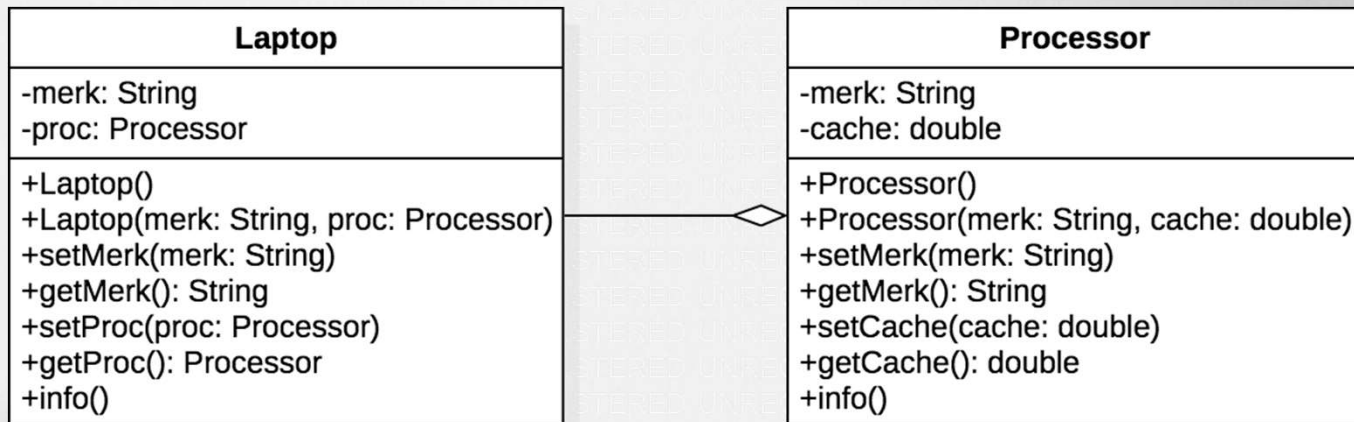


KOMPETENSI

- Memahami konsep relasi kelas;
- Memahami jenis-jenis relasi kelas dalam program.

PENDAHULUAN

Pada kasus yang lebih kompleks, dalam suatu sistem akan ditemukan lebih dari satu *class* yang saling memiliki keterkaitan antara *class* satu dengan yang lain, contoh:





Jenis-jenis Relasi Kelas :

- ✓ Dependence (*uses-a*)
- ✓ Aggregation (*has-a*)
- ✓ Inheritance (*is-a*)

(Horstmann dan Cornell, 2008)

DEPENDENCE (“USES-A”)

Relasi yang paling jelas dan umum.

Contoh: Kelas `Pesan` menggunakan kelas `Akun` karena objek `Pesan` membutuhkan akses objek `Akun` untuk mengecek status kredit. Tetapi, kelas `Barang` tidak bergantung pada kelas `Akun`, karena objek `Barang` tidak berkaitan dengan akun pelanggan.

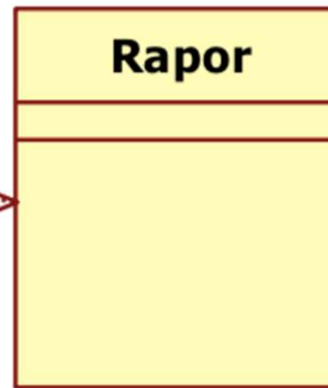
Jadi, inti relasi ini adalah sebuah kelas yang bergantung pada kelas lain berupa *method* atau manipulasi objek dari kelas itu.

Dependency

1. Relasi yang menggambarkan penggunaan suatu class pada class yang lainnya.
2. Contoh: class A memiliki dependency dengan kelas B maka jika kelas B berubah maka kelas A pun akan berubah.

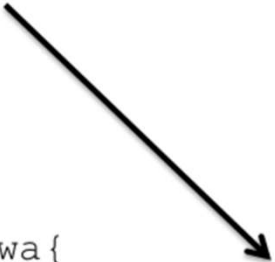
Bentuk Dependency

1. Penggunaan kelas B sebagai **parameter method** di kelas A.
2. Penggunaan kelas B sebagai **return value** di kelas A.
3. Penggunaan kelas B sebagai **variabel lokal** di kelas A.



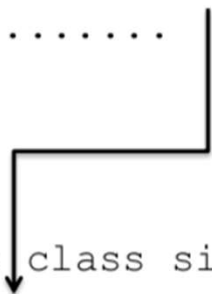
Dependency Parameter (JAVA)

```
public class Rapor{  
    .....  
}  
  
public class siswa{  
    boolean kenaikanKelas(Rapor rapor,int kelas){  
        .....  
    }  
}
```



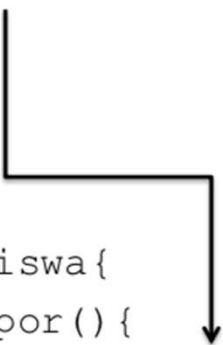
Dependency Return Value (JAVA)

```
public class Rapor{  
    .....  
}  
  
public class siswa{  
    Rapor getRapor(int kelas, int sem){  
        .....  
    }  
}
```



Dependency Variabel Lokal (JAVA)

```
public class Rapor{  
    .....  
}  
  
public class siswa{  
    void isiRapor(){  
        Rapor rapor=new Rapor();  
    }  
}
```



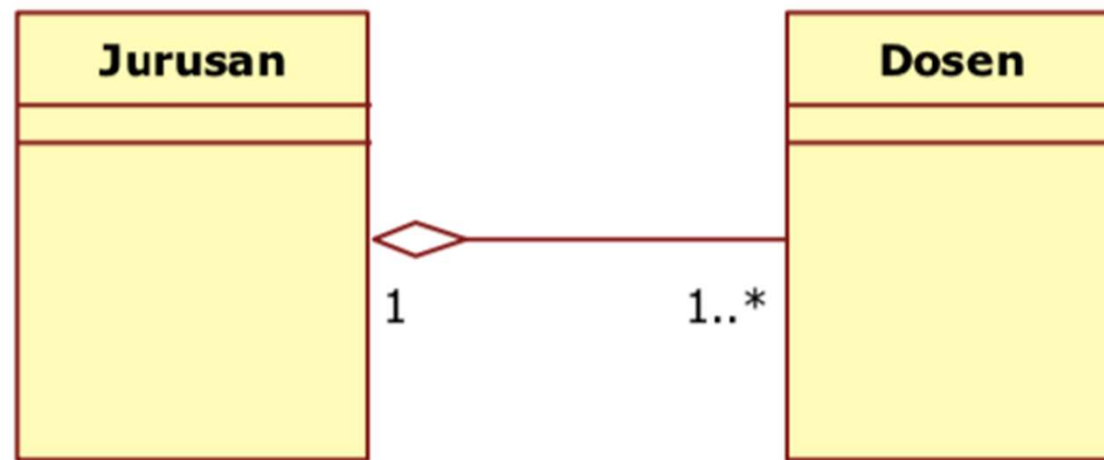
AGGREGATION ("HAS-A")

Jenis relasi ini mudah dipahami, karena bersifat konkrit.

Contoh: Sebuah objek Pesan berisi objek Barang. Berisi maksudnya objek dari kelas A didalamnya ada objek dari kelas B.

Contoh Aggregation

1. Dosen-dosen berkumpul membentuk suatu jurusan.
2. Jurusan berkumpul membentuk fakultas.
3. Fakultas-fakultas dikumpulkan menjadi universitas.



Contoh Aggregation (JAVA)

```
public class Dosen {  
    private String nip;  
    private String nama;  
  
    public Dosen(String nim, String nama) {  
        this.nip = nim;  
        this.nama = nama;  
    }  
  
    public String getNama() {  
        return nama;  
    }  
  
    public String getNim() {  
        return nip;  
    }  
}
```

Contoh Aggregation (JAVA)

```
public class Jurusan {  
    private String namaJurusan;  
    private Dosen dosen;  
  
    public Jurusan(String namaJurusan) {  
        this.namaJurusan = namaJurusan;  
    }  
  
    public void setDosen(Dosen dosen) {  
        this.dosen = dosen;  
    }  
  
    public Dosen getDosen() {  
        return dosen;  
    }  
  
    public String getNamaJurusan() {  
        return namaJurusan;  
    }  
}
```


Contoh Aggregation (JAVA)







```
public class TesterAgregasi {  
  
    public static void main(String[] args) {  
        Dosen dosen=new Dosen("1", "Adam");  
        Jurusan jurusan=new Jurusan("Teknik Informatika");  
  
        jurusan.setDosen(dosen);  
  
        System.out.println("NIP Dosen      : "+jurusan.getDosen().getNim());  
        System.out.println("Nama Dosen   : "+jurusan.getDosen().getNama());  
        System.out.println("Jurusan     : "+jurusan.getNamaJurusan());  
    }  
}
```

INHERITANCE ("IS-A")

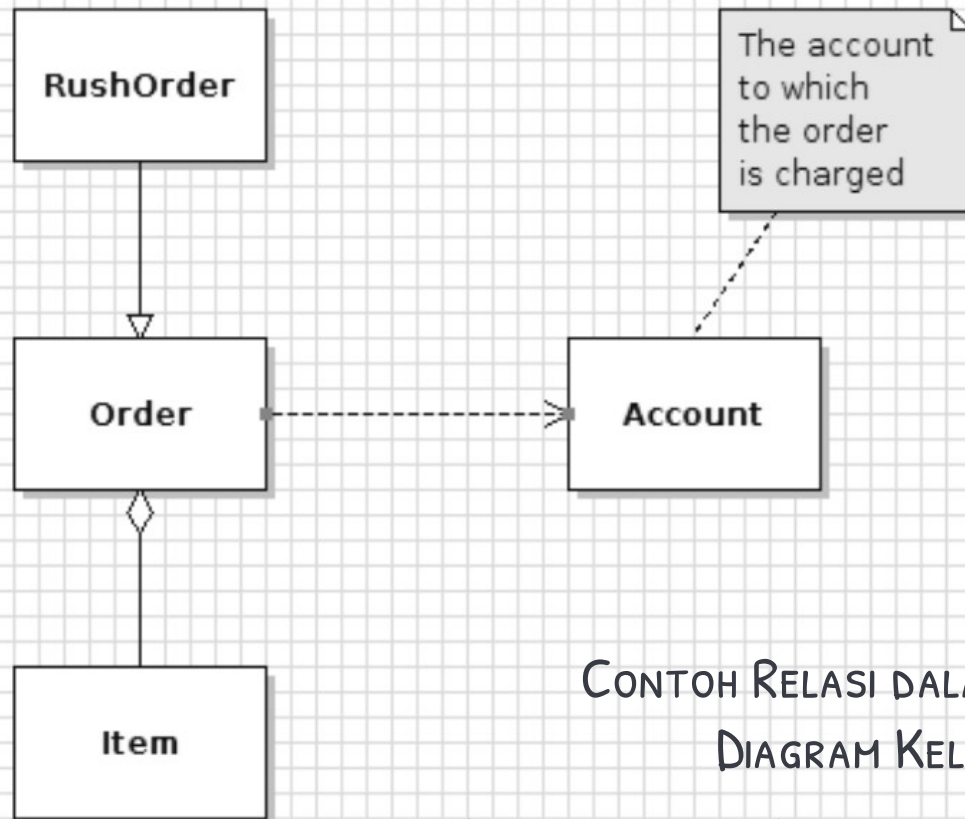
Relasi yang mengekspresikan antara kelas yang lebih umum dengan kelas yang lebih khusus.

Contoh: Sebuah kelas `PesanKilat` turunan dari kelas `Pesan`. Kelas khusus `PesanKilat` mempunyai *method* khusus untuk prioritas pelanggan dan mempunyai *method* khusus dalam perhitungan ongkir, tetapi *method* untuk menambah barang dan kasir diturunkan dari kelas `Pesan`.

NOTASI UML RELASI KELAS

Relationship	UML Connector
Inheritance	
Interface inheritance	
Dependency	
Aggregation	
Association	
Directed association	

(Horstmann dan Cornell, 2008)



CONTOH RELASI DALAM
DIAGRAM KELAS
(Horstmann dan Cornell, 2008)

UML Tools

Kakas untuk membantu
dalam merancang
diagram UML



Visual Paradigm + NetBeans

[Visual-
paradigm.com/tutorials
/modelinginnetbeans.jsp](http://visual-paradigm.com/tutorials/modelinginnetbeans.jsp)

UML Tools

Kakas untuk membantu
dalam merancang
diagram UML



ArgoUML

argouml.trigis.org

UML Tools

Kakas untuk membantu
dalam merancang
diagram UML

 **Violet** UML Editor

Violet

violet.sourceforge.net



THANKS!

Any questions?