



PROGRAM STUDI
TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS DIAN NUSWANTORO

Mata Kuliah
Analisis dan Perancangan
Berorientasi Obyek



Hubungan Antar Objek

Tim Pengampu Mata Kuliah

Analisis dan Perancangan Berorientasi Obyek

Capaian Pembelajaran

- Menjelaskan hubungan antar obyek

Hubungan Antar Objek (Kelas)

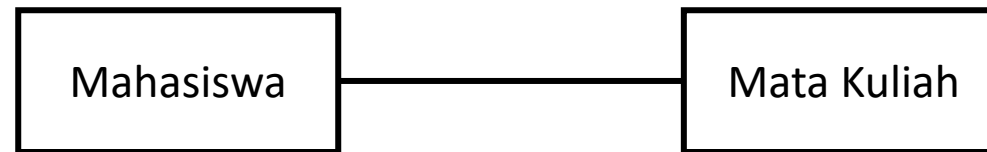
- Struktur perangkat lunak terdiri dari blok – blok yang berkaitan satu dengan yang lain
- Perilaku sebuah system juga didasarkan oleh perilaku objek yang berkaitan dengan objek lainnya
- Dasar pengembangan system dalam perangkat lunak adalah bagaimana membuat hubungan antar objek.

Hubungan Antar Objek

- Asosiasi
- Inheritance
- Komposisi
- Agregasi
- Dependensi

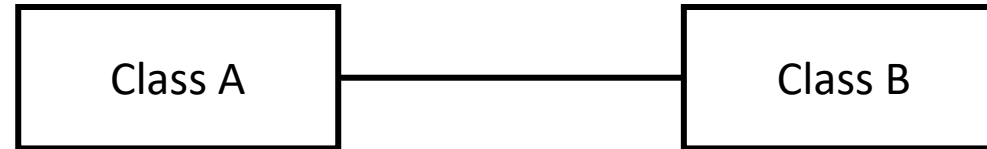
Asosiasi

- Asosiasi menggambarkan hubungan antar dua kelas atau lebih dengan struktur yang sama
- Satu kelas **menggunakan** kelas lain untuk memenuhi kebutuhan fungsionalitanya
- Contoh: Mahasiswa melakukan input KRS



- Mahasiswa dapat mengambil banyak matakuliah
- Object Mata Kuliah dapat diakses oleh Object Siswa

Asosiasi



Secara konsep, asosiasi menghubungkan dua kelas, seperti:

- Class A menyimpan atribut kunci yang unik mengidentifikasi objek kelas B
- Class A menyimpan referensi ke Class B
- Karena Class A dan Class B bersifat independent (berdiri sendiri), jadi Class A dapat memutuskan relationship dengan Class B
 - Masing – masing Class dapat dihancurkan tanpa terikat atau terpengaruh dengan kelas objek lain
- Dalam relasi Asosiasi, tidak ada kepemilikan atau hirarki yang ditetapkan

Contoh

- Di dalam class Mahasiswa menggunakan Class Matakuliah untuk menampilkan nama matakuliah dan sks

```
class Matakuliah
{
    public:
    string nama_makul;
    int sks;

    Matakuliah(string makul, int s)
    :nama_makul(makul),sks(s){}

    void CetakMakul()
    {
        cout << "mata kuliah: " << nama_makul << endl;
        cout << "Jumlah sks: " << sks << endl;
    }
};
```

```
class Mahasiswa
{
    public:
    string nama;
    Matakuliah M;

    Mahasiswa(string nm, Matakuliah mhs)
    :nama(nm),M(mhs){}

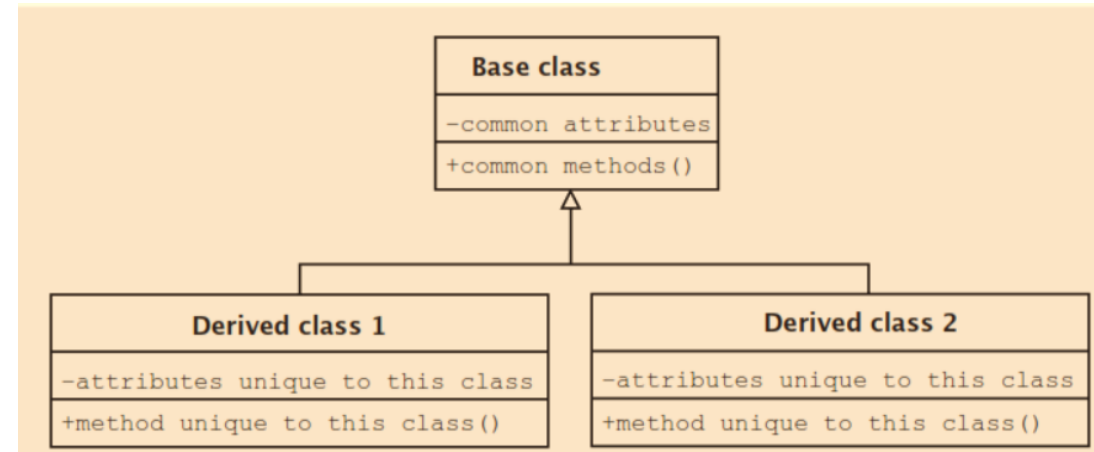
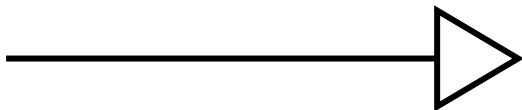
    void Cetak()
    {
        cout << "nama:" << nama << endl;
        M.CetakMakul();
    }
};
```

Multiplicity

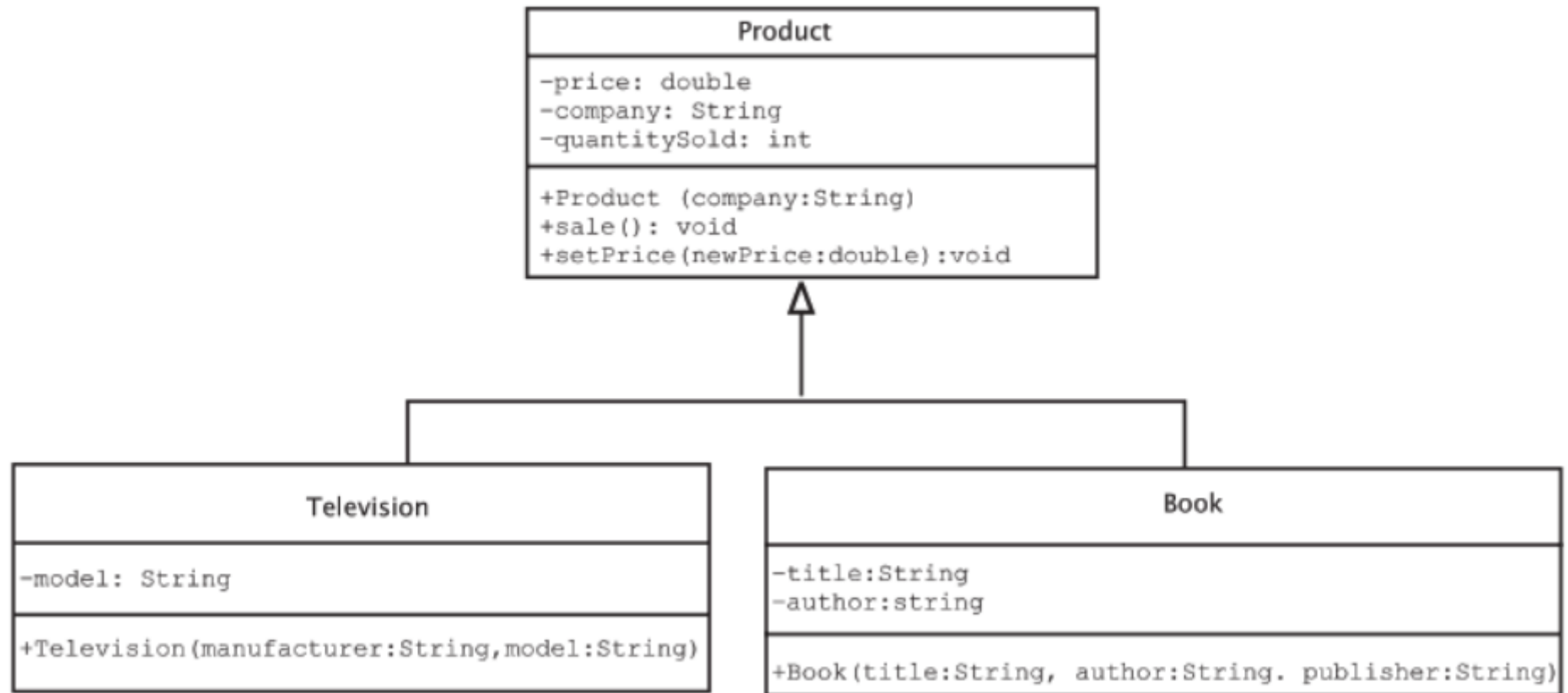
- Multiplicity adalah derajat kardinalitas atau jumlah elemen dalam suatu relasi
- Terdapat tiga multiplicity:
 - One to one
 - Contoh: satu mahasiswa hanya memiliki satu id card, satu id card hanya untuk satu mahasiswa
 - One to many
 - Contoh: setiap mahasiswa dapat mengambil banyak matakuliah, matakuliah hanya terkait dengan satu mahasiswa
 - Many to many
 - Contoh: setiap mahasiswa dapat mengambil banyak matakuliah, setiap matakuliah dapat diambil banyak mahasiswa

Inheritance

- Inheritance menggambarkan pewarisan dari suatu kelas.
- Kelas mewarisi atribut dan/atau operasi dari kelas lain yang dirujuk.
- Terdiri dari:
 - Base class atau superclass
 - Derived class atau subclass
- Notasi inheritance disebut generalisasi (generalisation)

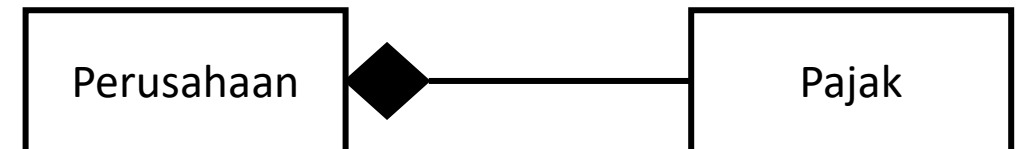


Contoh



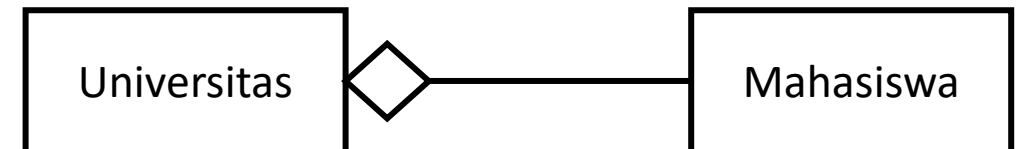
Composition (Komposisi)

- Komposisi menggambarkan sebuah kelas memiliki referensi ke kelas lain dalam variable instance
- Konsepnya adalah 'Has A' (memiliki) atau dependen. Bertanggung jawab pada object yang dirujuk
- Sebuah kelas dijalankan menjadi bagian dari kelas lain. Sebagai contoh
 - Perusahaan dengan pajak. Perusahaan memiliki pajak, pajak mengikat pada Perusahaan dan akan hilang jika Perusahaan hilang
- Composition lebih kuat dari Agregasi



Aggregation (Agregasi)

- Agregasi menggambarkan hubungan objek memiliki referensi ke objek lain sebagai: bagian darinya /part of
- Objek dapat bersifat independent dari objek yang dirujuk
- Contoh: Universitas dengan Mahasiswa
 - Universitas memiliki mahasiswa
 - Mahasiswa menjadi bagian dari universitas, namun mahasiswa dapat berdiri sendiri atau memutuskan hubungan ke universitas lainnya



Dependency

- Dependency digunakan untuk menggambarkan hubungan antar objek yang membutuhkan objek lain dalam melaksanakan fungsionalitasnya
- Suatu objek bergantung pada objek lain
- Notasi:



Contoh Dependency

- Atribut NILAI dengan MATAKULIAH
 - Atribut nilai memiliki ketergantungan pada Matakuliah
 - Nilai diinputkan sesuai dengan matakuliah
 - Namun, bisa jadi matakuliah tanpa NILAI



TERIMA KASIH

ANY QUESTIONS?