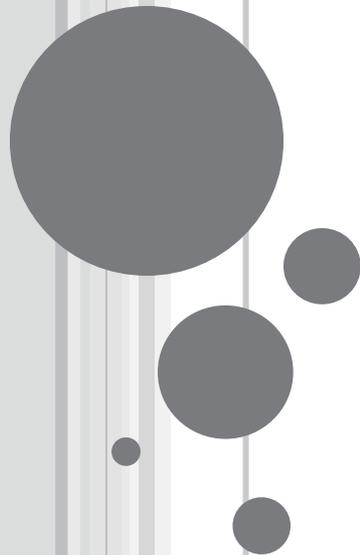


CLASS DIAGRAM

(ANALISIS DAN PERANCANGAN BERORINTASI OBJEK)

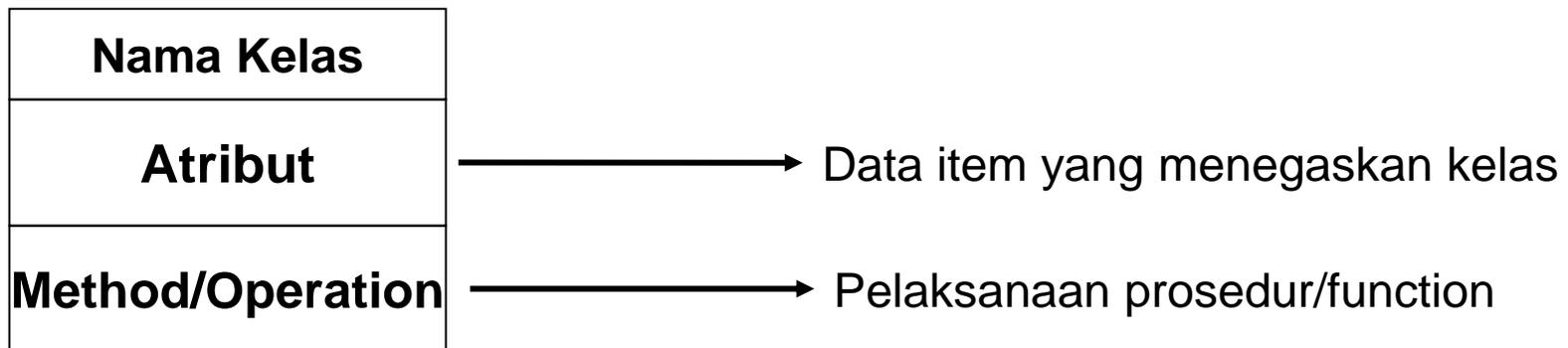
Annisa Paramitha F., S.Kom.,M.Kom.



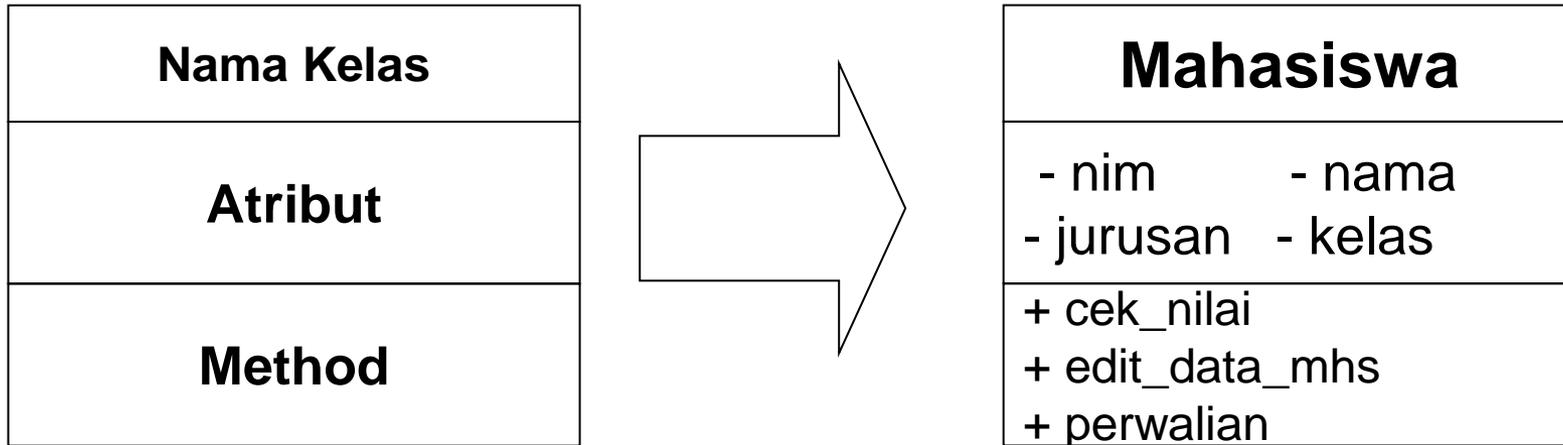
CLASS DIAGRAM

Diagram Kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas – kelas yang akan dibuat untuk membangun sistem.

Kelas memiliki apa yang disebut atribut dan metode atau operasi.



KONSEP DASAR KELAS



Visibility

- + Public : Able to access from any classes
- # Protected : Able to access from same class and its inherited classes
- ~ Package : Able to access within same package
- Private : Able to access from same class



PART OF CLASS

- **Atribut** merupakan variabel data, yang dapat memberikan informasi keadaan dimana tiap objek dari suatu kelas mempunyai nilai tersendiri.
- **Operation/method** atau sering disebut layanan (service) atau operasi adalah prosedur atau fungsi yang menjadi perilaku kelas-&-objek dan menjadi tanggung jawab objek tersebut. Dalam bentuk pemrograman merupakan bentuk subprogram yang digunakan terhadap atribut kelas-&-objek.

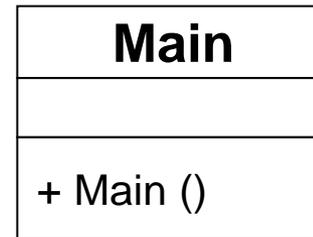


STRUKTUR KELAS

Susunan kelas yang baik pada diagram kelas sebaiknya memiliki jenis- jenis kelas berikut:

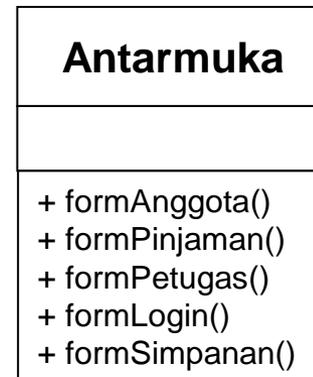
- **Kelas Main**

Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.



- **Kelas yg menangani tampilan sistem (*view*)**

Kelas yang mendefinisikan dan mengatur tampilan ke pengguna/*user*



STRUKTUR KELAS

- **Kelas yg diambil dari pendefinisian usecase (*controler*)**

Kelas yg menangani fungsi – fungsi yg diambil dari pendefinisian *usecase*, kelas ini biasanya disebut dengan kelas proses yang menangani proses bisnis pada sistem.

- **Kelas yg diambil dari pendefinisian data (*model*)**

Kelas yg digunakan untuk memegang atau membungkus data menjadi kesatuan yang diambil maupun akan disimpan ke basis data.

Simpanan
+ idSimpanan + JenisSimpanan + tglSimpanan + TotalSimpanan
+ tambahSimpanan() + editSimpanan() + hapusSimpanan()

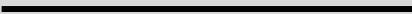
Pegawai
+ idPegawai + nama + alamat + Pendidikan
+ setNama() + getNama() + setIdPegawai() + getIdPegawai() + setPendidikan() + getPendidikan()

Pegawai
+ idPegawai + nama + alamat
+ setNama() + getNama()



Pendidikan
+ idPegawai + Pendidikan
+ setIdPegawai() + getIdPegawai() + setPendidikan() + getPendidikan()



Simbol	Deskripsi
Asosiasi/association 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai <i>multiplicity</i>
Asosiasi/association 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai <i>multiplicity</i>
Generalisasi 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum-khusus)
Kebergantungan/dependency 	Relasi antarkelas dengan makna kebergantungan antarkelas
Agregasi/aggregation 	Relasi antarkelas dengan makna <i>whole-part</i>

Catatan

Arah panah relasi pada diagram kelas mengarah pada diagram kelas yang lebih besar kontrolnya atau yang dipakai



HUBUNGAN ANTAR KELAS

○ Association

Hubungan statis antar class. Pada umumnya menggambarkan class yang memiliki atribut berupa class lain, atau class yang harus mengetahui eksistensi class lain

Menggambarkan hubungan antar class dengan ditandai dengan anak panah dan seringkali ditambahkan label dan multiplicity untuk memperjelas hubungan

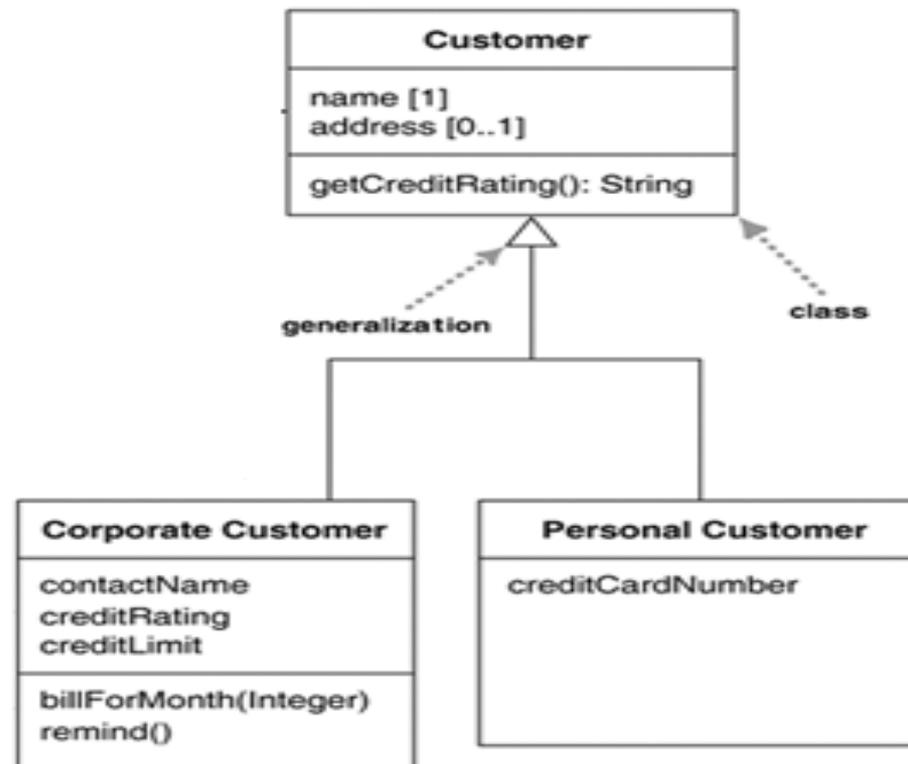
Multiplicity	Arti
N (default)	Banyak
0..0	Nol
0..1	Nol atau satu
0..n	Nol atau banyak
1..1	Tepat satu
1..n	Satu atau banyak



HUBUNGAN ANTARKELAS

Generalization

- Generalization adalah inheritance pada UML dimana sub class mewarisi feature dari super classnya.
 - Hubungan pewarisan yang menyatakan satu class adalah superclass dari class lain
 - Generalization dinotasikan dengan anak panah mengacu ke super class.
- Inheritance adalah hubungan hirarkis antar class.
 - Class dapat diturunkan dari class lain dan mewarisi semua atribut dan metode class asalnya dan menambahkan fungsionalitas baru, sehingga ia disebut anak dari class yang mewarisinya.



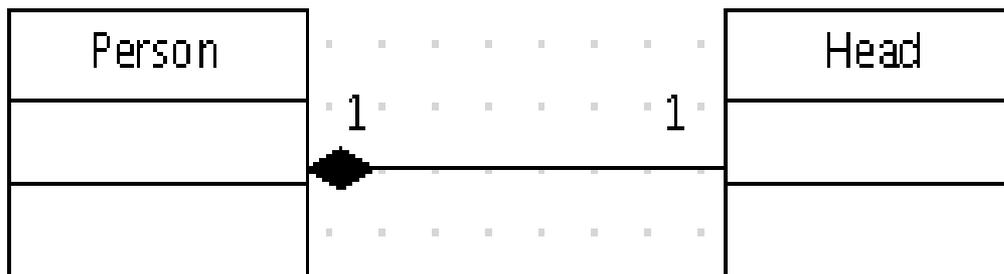
HUBUNGAN ANTAR KELAS

Aggregation

- Sebuah aggregation adalah bentuk khusus association yang memodelkan hubungan whole-part antara sebuah aggregation dengan bagiannya.
- Hubungan secara keseluruhan antara aggregate class dengan component class.

Ilustrasi :

- Class Person dan Class Head
- Menghapus person berarti juga menghapus kepalanya dan orang tidak bisa hidup tanpa kepala
- Orang dan Kepala harus ada bersamaan



HUBUNGAN ANTARKELAS

Dependency

Dependency adalah perubahan pada salah satu elemen yang mengakibatkan perubahan pada elemen yang lain. Semakin kompleks sistem, maka dependency menjadi sesuatu yang harus dipertimbangkan. Dependency hanya berlaku satu arah.

