



Modul Praktikum **RPL**



CLASS DIAGRAM

PENGERTIAN CLASS DIAGRAM

Class Diagram adalah jenis diagram struktur statis dalam UML yang menggambarkan struktur sistem dengan menunjukkan sistem *class*, atributnya, metode, dan hubungan antar objek. *Class diagram* disebut jenis diagram struktur karena menggambarkan apa yang harus ada dalam sistem yang dimodelkan dengan berbagai komponen. Berbagai komponen tersebut dapat mewakili *class* yang akan diprogram, objek utama, atau interaksi antara *class* dan objek. *Class* sendiri merupakan istilah yang mendeskripsikan sekelompok objek yang semuanya memiliki peran serupa dalam sistem. Sekelompok objek ini terdiri atas fitur struktural yang mendefinisikan apa yang diketahui *class* dan fitur operasional yang mendefinisikan apa yang bisa dilakukan oleh *class*. Ada beberapa karakteristik dasar dari class diagram.

a. Atribut

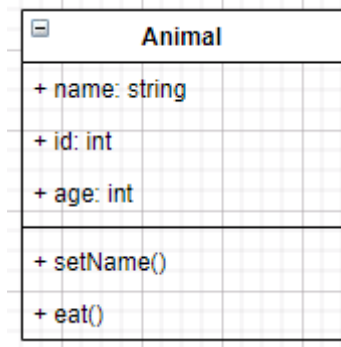
Atribut adalah salah satu properti yang dimiliki oleh class yang menggambarkan batasan dari nilai yang dapat dimiliki oleh property tersebut. Sebuah class mungkin memiliki beberapa atribut atau tidak memilikinya sama sekali. Sebuah atribut merepresentasikan beberapa properti dari sesuatu yang kita modelkan, yang dibagi dengan semua object dari semua class yang ada. Contohnya, setiap tembok memiliki tinggi, lebar dan ketebalan. Atribut dalam implementasinya akan digambarkan sebagai sebuah daftar (list) yang diletakkan pada kotak dibawah nama class. Seperti halnya nama class, atribut merupakan teks. Biasanya huruf pertama dari tiap kata merupakan huruf kapital, terkecuali untuk huruf awal.

b. Method

Method adalah sebuah implementasi dari layanan yang dapat diminta dari beberapa object dari class yang mempengaruhi behaviour. Dengan kata lain method adalah abstraksi dari segala sesuatu yang dapat kita lakukan pada sebuah object dan ia berlaku untuk semua object yang terdapat dalam class tersebut. Class mungkin memiliki beberapa method atau tanpa method sama sekali. Contohnya adalah sebuah class "kotak", methodnya dapat



dipindahkan, diperbesar atau diperkecil. Biasanya (namun tidak selalu), memanggil method pada sebuah object akan mengubah data atau kondisi dari object tersebut. Method ini dalam implementasinya digambarkan dibawah atribut dari sebuah class.



Gambar diatas merupakan contoh suatu class. Class tersebut kita beri nama Animal kemudian untuk kotak ditengah adalah atribut dari class tersebut. Dengan atribut ini kita bisa membedakan antara animal yang satu dengan yang lainnya. Kemudian kotak yang paling bawah adalah method atau bisa disebut operasi atau fungsi dari sebuah class. Dengan method ini kita bisa menentukan perilaku atau apa saja yang bisa dilakukan oleh class tersebut. contoh hasil akhir dari class tersebut seperti gambar dibawah ini.

VISIBILITY

Jika teman-teman perhatikan gambar diatas sebelum kita menuliskan atribut atau method ada tanda + yang dimana ini disebut dengan visibility. Visibility ini yang menentukan aksesibilitas untuk sebuah atribut atau method. Terdapat 3 visibility yang biasa dipakai yaitu:

1. Private(-)

Jika atribut atau method yang diberikan visibility tersebut maka, tidak dapat diakses oleh class atau subclass lainnya dan hanya bisa di class itu sendiri.

2. Protected(#)

Jika atribut atau method yang diberikan visibility tersebut maka, hanya dapat diakses oleh subclass atau class itu sendiri.

3. Public(+)

Jika atribut atau method yang diberikan visibility tersebut dapat diakses oleh class lainnya

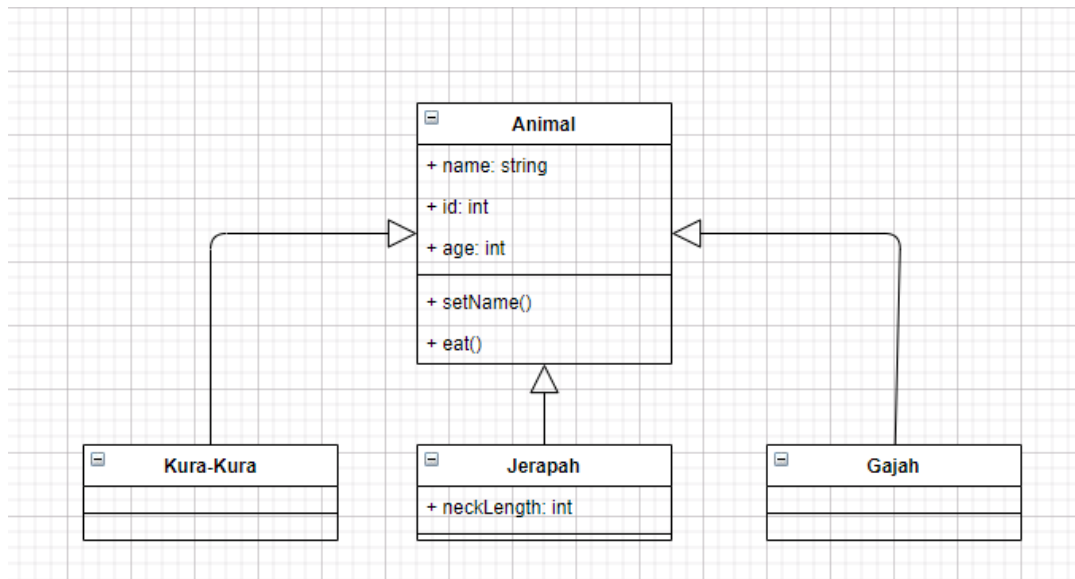


RELATIONSHIP

Dalam class diagram terdapat juga relasi karena pada dasarnya setiap class memiliki hubungan atau relasi terhadap class lain yang saling terkait. Berikut beberapa relasi yang ada dalam class diagram:

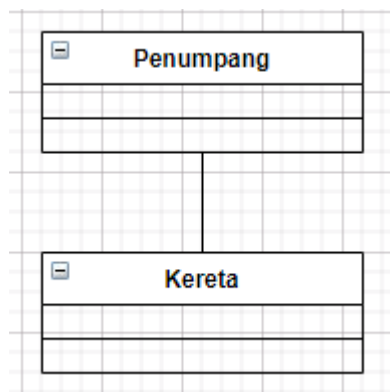
- Inheritance

Inheritance atau biasa disebut dengan pewarisan dimana suatu class mewariskan atribut atau methodnya kepada sub class contohnya seperti dibawah ini:



- Association

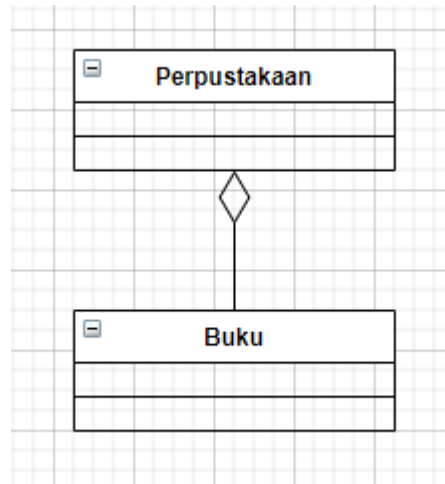
Asosiasi dapat diartikan sebagai hubungan antara dua class yang bersifat statis. Biasanya asosiasi menjelaskan class yang memiliki atribut tambahan seperti class lain.





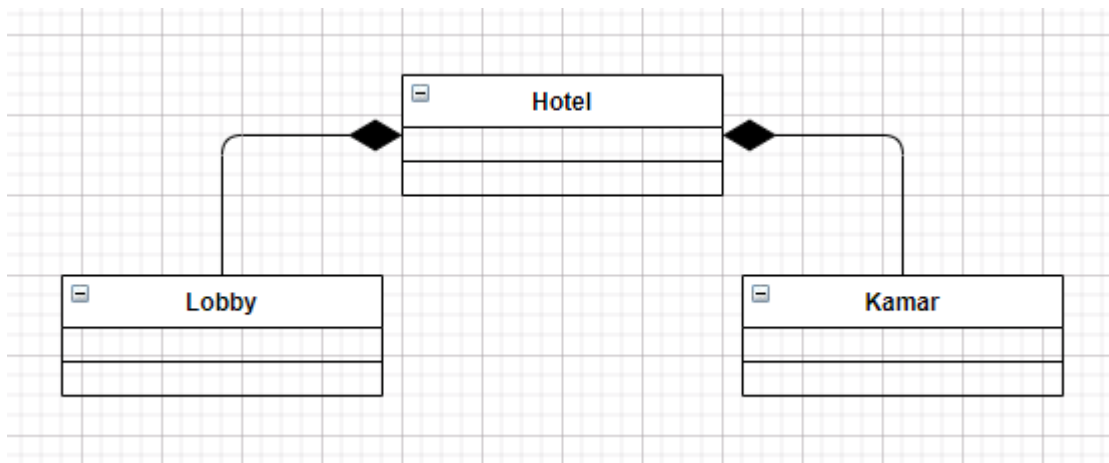
- Aggregation

Agregasi adalah hubungan antara dua class di mana salah satu class merupakan bagian dari class lain, tetapi dua class ini dapat berdiri masing-masing.



- Composition

Composition adalah jenis aggregation khusus yang menunjukkan kepemilikan yang kuat antara dua class ketika satu class menjadi bagian dari class lain.

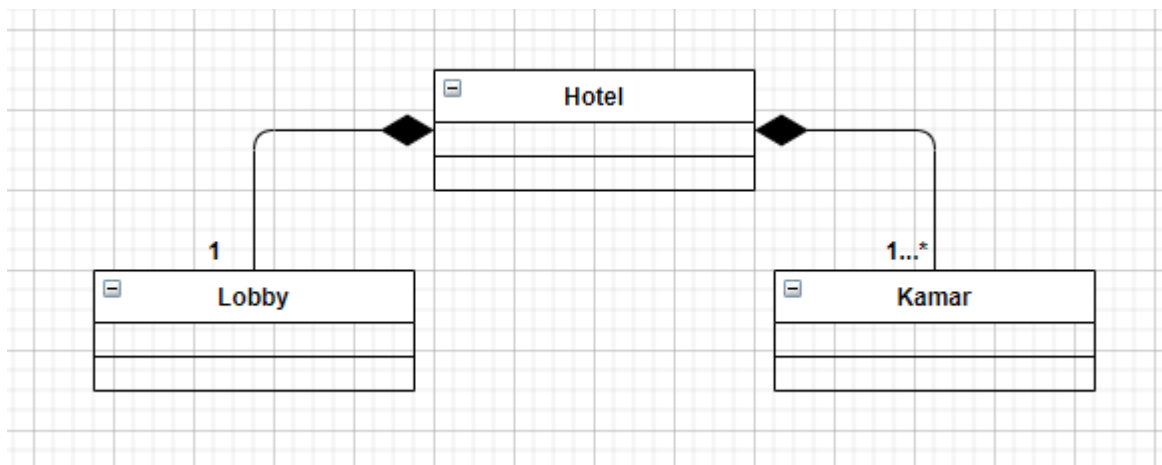




MULTIPLISITAS

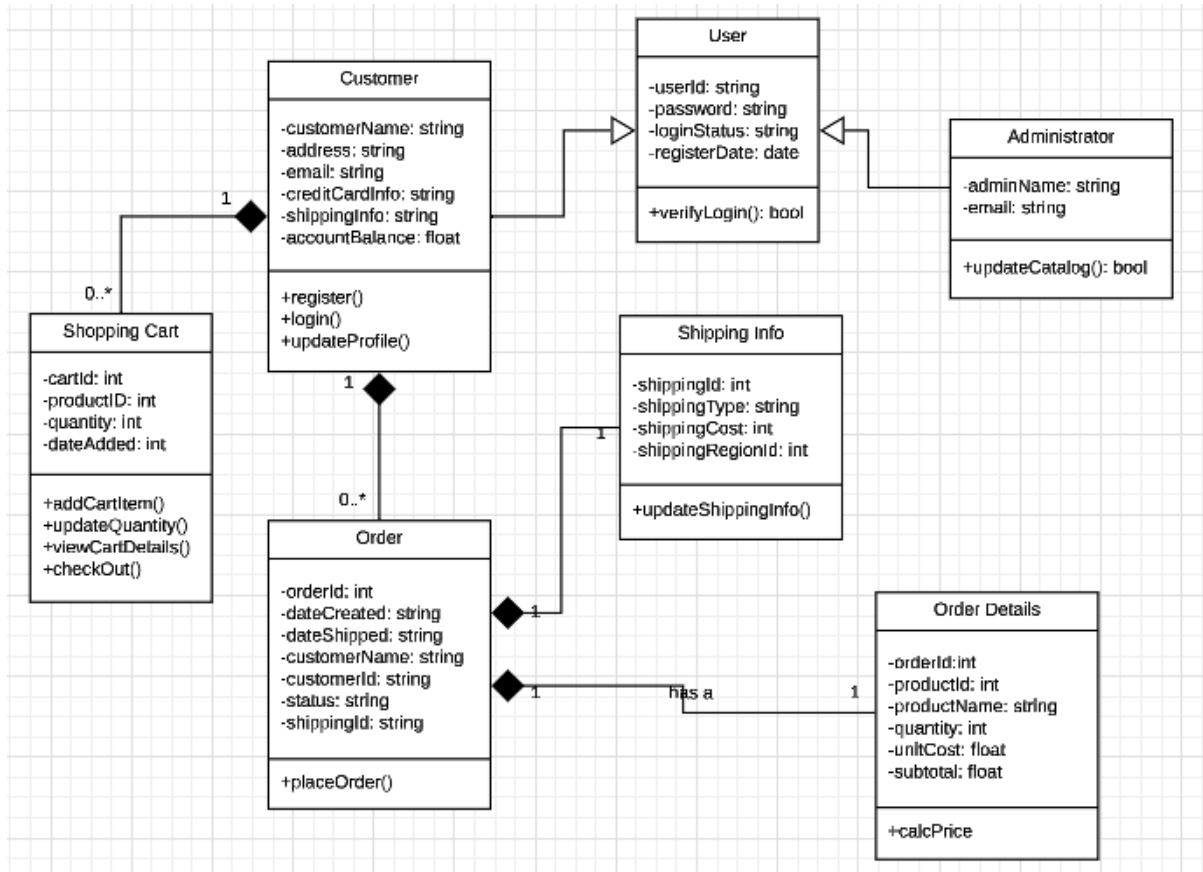
Multiplicity atau multiplisitas adalah jumlah banyaknya objek sebuah class yang berelasi dengan sebuah objek lain pada class lain yang berasosiasi dengan class tersebut. Untuk menyatakan multiplisitas anda dapat meletakkannya diatas garis asosiasi berdekatan dengan class yang sesuai. Ada banyak multiplisitas yang mungkin untuk dipakai. Tabel berikut menjabarkan multiplisitas yang dapat digunakan.

Multiplisitas	Arti
*	Banyak
0	Nol
1	Satu
0...*	Nol atau Banyak
1...*	Satu atau Banyak
0...1	Nol atau Banyak
1...1	Hanya satu



Terdapat 2 multiplisitas diatas yaitu:

- Setiap Hotel hanya memiliki 1 lobby
- Setiap Hotel memiliki 1 atau banyak kamar



Berdasarkan gambar diatas coba teman-teman identifikasi class-class apa saja yang ada, relasinya seperti apa dan multiplisitasnya seperti apa!

Sumber:

https://www.youtube.com/watch?v=UI6lqHOVHic&t=358s&ab_channel=LucidSoftware

<https://www.dicoding.com/blog/memahami-class-diagram-lebih-baik/>