1. 4 access modifier in Java Programming Language

* Private = semua variable, method maupun attribute yang menggunakan access modifier ini hanya dapat digunakan dalam satu class yang sama.
* No- access modifier = semua variable, method maupun attribute yang menggunakan access modifier ini hanya dapat digunakan dalam class dan package yang sama.
* Protected = semua variable, method maupun attribute yang menggunakan access modifier ini hanya dapat digunakan dalam class, package, dan subclass yang sama.
* Public = semua variable, method maupun attribute yang menggunakan access modifier ini dapat digunakan baik pada class, package, subclass, maupun project yang berbeda.

Dari segi visibilitynya urutannya dari yang paling aman adalah : private-> no access modifier -> protected -> public.

1. Overloading = menggunakan nama yang sama dengan nama pada class parent (superclass) tapi parameter atau tipe data yang digunakan berbeda. Overloading dicek pada saat compile, static. Contohnya, pada saat suatu class menggunakan nama yang sama untuk fungsi yang berbeda (contoh ada fungsi dengan parameter a(int,string) dan fungsi dengan parameter a(int,int,string)).

**public** Salesman() {

**this**.address=address;

**this**.jobName= "Salesman";

**this**.vProduct = **new** Vector <Product> ();

**this**.name = name;

}

**public** Salesman(String name, String address) {

**super**();

}

Overriding = menggunakan nama dengan nama yang sama dengan nama pada class parent (superclass) dengan tipe data yang sama tapi fungsi yang dijalankan dimodifikasi. Overriding dicek pada saat runtime, dynamic. Contohnya, pada saat suatu class melakukan extends dari class lainnya, maka class tersebut menurunkan variable, method, dan attribute dari class parentnya (superclass). Method dari superclass tersebut dapat di tulis ulang pada sub class ( class anaknya) dengan tipe data yang sama (cth: sama- sama memilikin string nama, int umur) tapi fungsi yang dijalankan berbeda.

@Override

**public** String toString() {

**return** "Salesman [vProduct=" + vProduct + ", name=" + name + ", address=" + address + ", jobName=" + jobName

+ "]";

}

1. Size() = untuk mengetahui ukuran dari array list yang kita miliki. Contoh array list tersebut memiliki 5 data maka sizenya ada 5.

Add() = untuk menambahkan suatu data baru kedalam array list yang sudah dibuat sebelumnya. Contoh kita memasukan data nama Andi dan data nama Budi kedalam suatu array list nama yang sudah kita buat sebelumnya.

Get() = untuk mengambil suatu data dari array list yang telah kita buat sebelumnya. Kita bisa mengambil data tersebut dari indexnya. Contoh dalam sebuah array list nama, pada index 0 terisi nama Andi dan pada index 1 terisi nama Budi, maka saat kita get index ke 0, akan terambil data nama Andi.