

# STK372 Komputasi Statistik 2

## Pemrograman Berorientasi Objek

Dr. Agus M Soleh



**IPB University**  
— Bogor Indonesia —

*agusms@apps.ipb.ac.id*

- 1 Prinsip Pemrograman Berorientasi Objek
  - Abstraksi
  - Enkapsulasi
  - Inheritance (pewarisan)
  - Polymorphism
- 2 Pemrograman Berorientasi Objek di R

# Prinsip Pemrograman Berorientasi Objek

Pemrograman Berorientasi Objek merupakan sebuah paradigma dalam pembuatan sebuah program

PBO menitikberatkan pada identifikasi objek-objek yang terlibat dalam sebuah program dan bagaimana objek-objek tersebut berinteraksi

Pada PBO, program yang dibangun akan dibagi-bagi menjadi objek-objek

[Meyer98]: Sebuah sistem yang dibangun berdasarkan metoda berorientasi objek adalah sebuah sistem yang komponennya di-**enkapsulasi** menjadi kelompok data dan fungsi, yang dapat **mewarisi atribut dan sifat** dari komponen lainnya, dan komponen-komponen tersebut saling berinteraksi satu sama lain.

# Prinsip Pemrograman Berorientasi Objek

PBO adalah metode pemrograman berdasarkan hirarki **class**, dimana class-class tersebut didefinisikan dengan baik dan bisa saling bekerja sama untuk memecahkan masalah

Beberapa prinsip dari PBO adalah:

- Abstraksi
- Enkapsulasi
- Inheritance (pewarisan)
- Polymorphism

# Abstraksi

Menunjukkan karakteristik atau sesuatu yang khas dari sebuah objek yang bisa membedakannya dengan objek yang lain

Dengan demikian akan terlihat batasan konseptual dari sebuah objek dari sudut pandang tertentu

Contoh: Pada sistem informasi perpustakaan, apakah membuat class Mahasiswa dan class Dosen atau membuat class Member saja?

Dalam pengembangan aplikasi, abstraksi difokuskan pada:

- Hal atau sesuatu yang mana saja yang akan menjadi object?
- Apa ciri-ciri pembeda (property) yang dimiliki object tersebut?
- Bagaimana kelakuan (method) dari object tersebut?

## Enkapsulasi

Merupakan proses untuk menutupi semua detail dari sebuah object yang tidak menyokong karakteristik khas dari object

Artinya, data property dan method-method dari object hasil abstraksi dibungkus oleh struktur class agar terlindungi

Rincian implementasi internal suatu object dibuat tersembunyi sehingga tidak bisa diketahui dan diakses oleh object lain

Object lain hanya bisa berkomunikasi dengan object tersebut melalui antarmuka (interface) yang disediakan oleh object tersebut

Misal pada class Car, data variabel `current_gear` dibuat private sehingga object lain tidak bisa mengakses variabel tersebut. Object lain hanya bisa menjalankan method `changeGear(int gear)` tanpa mengetahui posisi gear saat sebelum diganti.

## Inheritance (pewarisan)

Sebuah class bisa mewariskan atribut property-nya dan method-nya ke class yang lain

Class yang mewarisi disebut superclass (kelas induk)

Class yang diberi warisan disebut subclass (kelas turunan)

Pewarisan memungkinkan kita untuk menciptakan kelas baru (kelas turunan) yang merupakan perluasan atau spesialisasi dari kelas lain yang telah ada (kelas induk)

Kelas turunan akan memiliki property dan method yang sama dengan kelas induk, dimana pada kelas turunan memungkinkan untuk menambahkan property atau method-method baru

## Inheritance (pewarisan)

Contoh: membuat class ModernCar yang merupakan turunan dari class Car

Artinya, class ModernCar memiliki property dan method yang sama dengan induknya dan mungkin tanpa perlu dideklarasikan ulang

Agar menjadi unik, class ModernCar menambahkan method retreat() pada definisi class-nya



# Polymorphism

Merupakan kemampuan suatu object untuk melakukan hal yang berbeda-beda dengan menggunakan interface yang sama

Hal ini tercermin dengan menggunakan teknik overriding dan overloading

Overriding adalah penulisan ulang method warisan dari kelas induk

Overloading adalah penulisan method yang sama dengan parameter input yang berbeda

# Class (Kelas)

- Merupakan definisi statik (kerangka dasar) dari objek yang akan diciptakan.
- Berupa struktur yang mendefinisikan data (property) dan method dari objek.
- Bisa disebut cetak biru dari objek.
- Bisa disebut gambaran umum dari benda
- Contoh penamaan kelas: Mobil, Laptop, Anggota, Buku, dll

# Class (Kelas)

Suatu Class dibagi menjadi:

- Property Merupakan data atau state yang dimiliki oleh class.  
Contoh pada class Mobil, memiliki property:  
Warna, Model, Produsen
- Method Merupakan behavior (perilaku) sebuah class  
Bisa dikatakan sebagai aksi atau tindakan yang bisa dilakukan oleh suatu class  
Contoh pada class Mobil, memiliki method: Start, Stop, Change Gear, Turn

# Object

Merupakan abstraksi dari sesuatu yang mewakili sesuatu pada dunia nyata

Pada bahasa pemrograman, object adalah komponen yang diciptakan dari class (*instance of class*)

Merupakan entitas pada saat RUN TIME

Memiliki siklus creation, manipulation, dan destruction

Satu class bisa menghasilkan banyak object

# Pemrograman Berorientasi Objek di R

R telah mengimplementasi pemrograman berorientasi objek

Semua dalam R adalah objek

Implementasi objek dalam R berbeda dengan pemrograman berorientasi objek lain, seperti: C++, python, dll

Pengembangan awal objek di R menggunakan Class System S3 yang tidak terlalu ketat

Pendefinisian yang ketat secara formal, R menggunakan Class System S4

## Ilustrasi: Pemrograman Berorientasi Objek di R

Sebuah class `coords` dirancang untuk digunakan dengan menyimpan data koordinat titik pada dua buah vektor `x` dan `y`.

Metode pada class ini terdiri dari metode `print`, `length`, `bbox`, dan `plot`.

Class lain dirancang sebagai turunan dari class `coords` dengan menambahkan data nilai (`value`) untuk setiap titik pada koordinat `x` dan `y`.

Metode pada class `vcoords` merupakan pewarisan dari class `coords` dan operasi-operasi aritmetik terhadap nilai (`value`)-nya.

Lanjut ke materi 3a...