Nama : Rahmad Setiya Budi

NIM : 222011623

Kelas : 2KS3

Kode : 01N446I

PBO Materi Pertemuan 1

Pemrograman -> Suatu cara penulisan urutan perintah/instruksi kepada komputer untuk melakukan tugas tertentu.

Paradigma pemrograman adalah suatu cara untuk mengkonseptualisasikan suatu masalah di dunia nyata dan solusinya dalam bentuk model komputasi yang diterapkan sebagai fitur di bahasa pemrograman. Terkadang suatu Bahasa pemrograman memiliki fitur yang bisa mendukung lebih dari satu paradigma.

Manfaat :

1. Mengetahui motivasi dibalik dibangunnya Bahasa pemrograman tersebut.
2. Memberi tahu kita bagaimana bahasa tersebut memodelkan skenario dunia nyata. Dengan begitu akan lebih mudah untuk memilih alat dan teknik yang sesuai untuk digunakan dalam menyelesaikan permasalahan tertentu/spesifik.
3. Memahami bagaimana menerapkan pendekatan bahasa yang sesuai untuk menyusun atau mengatur kode program yang akan kita tulis.

Pada Paradigma Pemrograman Berorientasi Objek, program terdiri dari interaksi antar objek. Sebuah objek akan membungkus (encapsulate) data dan algoritma.

Catatan :

• Paradigma Pemrograman ≠ Bahasa Pemrograman

• Bahasa Pemrograman memiliki satu atau lebih fitur paradigma pemrograman.

Objek adalah entitas apapun yang memiliki properties dan behaviours tertentu baik dalam bentuk fisik ataupun logical. Class adalah sebuah template atau cetak biru untuk menciptakan

satu atau lebih objek dengan properties dan behaviours yang serupa

Properties = Variabel ; Behaviour = Method

❖ Abstraksi

✓ Proses atau cara untuk mengekspose/memunculkan APA detail utama dari sebuah

entitas, dan menyampingkan hal yang tidak relevan untuk mengurangi kompleksitas bagi

penggunanya untuk Tujuan tertentu.

❖ Encapsulation

✓ Proses Bagaimana membungkus data dan operasi terhadap data, secara bersamaan

dalam sebuah entitas.

❖ Inheritance

✓ Kemampuan untuk merancang tipe entitas baru yang menurunkan sifat dari entitas yang

sudah ada, sehingga membentuk hubungan parent-child.

❖ Polymorphism

✓ Konsep dimana entitas bisa memiliki banyak perilaku yang berbeda meskipun nama

perilakunya sama tetapi berbeda konteks.

Proses atau teknik untuk Menyaring Apa detail utama dari sebuah entitas, dan menyampingkan hal yang tidak relevan untuk mengurangi kompleksitas bagi penggunanya untuk Tujuan tertentu.

Interface dan Abstract Class merupakan salah satu cara eksplisit menerapkan konsep

Abstraksi dalam Bahasa Java.

Enkapsulasi adalah membungkus data (variabel) dan operasi (mehod) dalam sebuah entitas yang disebut dengan Class.

Data hiding adalah proses untuk menyembunyikan implementasi detail yang kemungkinan besar akan berubah.

Perbedaan abstraction dan encapsulation

1. Abstraksi hanya menyediakan dan memunculkan informasi APA yang penting kepada Pengguna, Sementara Encapsulation membungkus method dan variable untuk informasi yang diperlukan.
2. Abstraksi menyembunyikan kompleksitas bagi pengguna dengan *interface* atau Gambaran umum, sementara Encapsulation menyembunyikan detail pekerjaan internal sehingga memungkinkan perubahan nantinya.
3. Abstraksi membantu untuk mempartisi program ke dalam subprogram yang independent, Sementara Encapsulation menyediakan kemudahan pengembangan bila terdapat *requirement* baru.
4. Abstraction memecahkan masalah di level desain, sedangkan Encapsulation di level implementasi.
5. Abstraksi menyembunyikan detail yang tidak relevan pada kode, sementara Encapsulation membantu developer untuk mengorganisasi source code secara lebih mudah.