**TUGAS API (Application Program Interface)**



**NAMA : MUHAMAD HUSEIN**

**N.P.M : 06.2010.1.05249**

**FAKULTAS TEKNIK INFORMATIKA**

**INSTITUT TEKNOLOGI ADHI TAMA**

**SURABAYA**

Java API

API (Application Program Interface)

Dalam suatu program, dibutuhkan setidaknya ribuan *system calls* per detik. Oleh karena itu Kebanyakan *programmer* membuat aplikasi dengan menggunakan *Application Programming Interface* (API). Dalam API itu terdapat fungsi-fungsi / perintah-perintah untuk menggantikan bahasa yang digunakan dalam *system calls* dengan bahasa yang lebih terstruktur dan mudah dimengerti oleh *programmer*. Fungsi yang dibuat dengan menggunakan API tersebut kemudian akan memanggil *system calls* sesuai dengan sistem operasinya. Tidak tertutup kemungkinan nama dari *system calls* sama dengan nama di API.

API

• Seperangkat fungsi standar yang disediakan oleh OS atau Bahasa

• Dalam Java, API dimasukkan ke dalam package-package yang sesuai dengan fungsinya

• Java mengandung ratusan kelas standar

– J2SE: Edisi standar

– J2EE: Edisi enterprise (lebih banyak kelas)

– J2ME: Subset kelas standar

• Kelas-kelas ini memungkinkan pembuatan

program dengan mudah

• API Java cukup lengkap

– Mulai dari yang sederhana (misalnya struktur data

Stack)

– Sampai yang kompleks (seperti enkripsi dan akses

file ZIP)

Dilakukan dengan mengimpor

package/kelas

import java.util.Stack;

• Ada beberapa kelas bernama sama di

package yang berbeda

– import salah satu dan gunakan nama

lengkap untuk yang lain, atau:

– gunakan nama lengkap semua kelas

**Platform Specific API**

* Operating System
  + Windows 🡪 WinAPI, ShellAPI, dll
  + Linux/Unix 🡪 kernel, socket, dll
  + Mobile device

Disediakan melalui SDK

(*Software Development Kit*)

* + - Nokia 🡪 Symbian OS SDK, UIQ
    - Windows CE / Pocket PC 🡪 winceos core
    - Blackberry 🡪 Blackberry SDK, Blackberry JDE
    - Java Based 🡪 J2ME SDK

Droid 🡪 Google Android SDK

* Graphic API
  + DirectX 🡪 game, multimedia streaming
  + OpenGL 🡪 fractal, 3D modeling
  + Modem 🡪 TAPI (Telephony API)
* Database connector
  + ADODB 🡪 via ODBC
  + OLEDB 🡪 via provider library
  + Direct Connection
    - MySQL
    - PostgreSQL
    - SQLite

**Java Sebagai Suatu Platform**

Java adalah bahasa pemrograman yang dapat berjalan di semua platform tanpa harus mengubah kode sedikitpun dengan syarat pada sistem yang digunakan sudah terdapat JRE (Java Runtime Environment). Hal inilah yang menjadi kekuatan Java sebagai bahasa pemrograman multiplatform write once run anywhere. Aplikasi menggunakan Java sangatlah luas dan dibagi menjadi tiga sub bagian yaitu J2EE (Java 2 Enterprise Edition), J2SE (Java 2 Standard Edition), dan J2ME (Java 2 Micro Edition). Dari ketiga sub bagian tersebut yang membedakan adalah cakupan aplikasi yang ingin dibuat. J2EE digunakan untuk aplikasi yang bersifat enterprise dan dalam sekala yang besar seperti sistem terdistribusi, J2SE digunakan untuk aplikasi standard pada desktop sedangkan J2ME lebih dikonsentrasikan untuk aplikasi yang bersifat embedded seperti mobile device. Dari ketiga sub bagian yaitu J2EE, J2SE, dan J2ME masih dibagi lagi menjadi bagian-bagian yang spesifik dan disesuaikan dengan kebutuhan. J2SE adalah framework yang sering digunakan, karena semua platform Java menggunakan J2SE sebagai base development.

*Penanganan String optimal di Java*

**String**

• Merupakan kelas khusus di Java

(ditangani secara transparan)

• Sifatnya immutable (tidak bisa diubah)

• Memiliki berbagai method untuk memiliki

manipulasi String

Literal String

• Harap diingat lagi bahwa:

– Setiap Literal String adalah sebuah objek

String. Contoh:

String teks = “Hello”;

System.out.println(teks.length());

– Sama dengan

System.out.println(“Hello”.length());

Sifat Immutable String

• String sebenarnya tidak dapat diubah,

namun Java memungkinkan String

seolah-olah diubah, Contoh:

– Jika dilakukan ini:

String judul = "Judul :";

judul += "Eyes On Me";

– Maka sebenarnya yang dilakukan adalah ini

String judul = "Judul :";

judul = judul.concat("Eyes On Me");

Method yang penting

• Beberapa method String dan

StringBuffer yang penting adalah:

– length(): panjang string

– replace(): mengganti suatu karakter

– charAt(): mengakses karakter di posisi tertentu

– trim(): menghilangkan spasi di awal dan di akhir

string

• Perhatikan bahwa meskipun namanya sama,

sifat keduanya berbeda

– String menciptakan objek baru, sedangkan

StringBuffer tidak

Membandingkan String

• Method equals() membandingkan

string untuk memeriksa kesamaan

• Method equalsIngnoreCase()

melakukan hal yang sama, tapi besar

kecil huruf tidak diperhatikan

• Method compareTo() menghasilkan 0

jika string sama, >0 jika String1>String2

dan <0 jika String1<String2

*API untuk aneka macam hal*

Utility

• Berbagai macam kelas yang tidak cocok

dimasukkan ke package tertentu

• Beberapa yang akan dibahas

– Hashtable

– Stack

– Vector

– Calendar

• Kelas lain yang cukup berguna untuk

dipelajari: List, Tree

Hashtable

• Digunakan untuk menyimpan data

dengan assosiasi tertentu

– Misal nama panggilan diassosiasikan dengan

nama lengkap

• Memetakan suatu nilai String dengan

suatu Objek tertentu (Objek apa saja,

termasuk juga Objek String)

• Method yang dipakai adalah put dan

Get