

MK. Pemrograman Berorientasi Objek



PENGENALAN JAVA

KARMILASARI

APA itu JAVA ?

- ▶ Dibuat pertama kali oleh **James Goslig** dkk (1991) sebagai bagian dari **Sun Microsystem Java Platform**
- ▶ Awalnya diberi nama "Oak" kemudian berganti nama menjadi "**Java**" (1995)
- ▶ Termasuk ke dalam Bahasa Pemrograman Berorientasi Objek
- ▶ Digunakan untuk beragam aplikasi seperti : aplikasi embedded, aplikasi keuangan, desktop, simulasi pesawat, pemrosesan citra, game, dll
- ▶ Sintaks Java banyak diturunkan dari C dan C++ tetapi lebih sederhana

APA itu JAVA ?

- ▶ Java sebagai :
 - ▶ Bahasa Pemrograman
 - ▶ Platform
 - ▶ Java Virtual Machine (Java Framework Runtime)
 - ▶ Java Application Programming Interface (Java API)
- ▶ Pada dasarnya terdapat banyak platform OS, namun demikian aplikasi yang berjalan pada suatu platform OS tidak dapat dijalankan pada platform OS lain. Java platform merupakan perangkat lunak yang menjadi mesin virtual bagi aplikasi Java untuk dieksekusi .
 - ▶ Saat proses kompilasi, program diubah menjadi **kode byte**, saat eksekusi JVM membaca kode byte dan mengubahnya menjadi bahasa mesin yang dimengerti OS tempat program tersebut dijalankan
 - ▶ JVM sangat tergantung platformnya, sedangkan kode byte dibuat untuk terbebas dari lingkungan platform tertentu. Dengan demikian kode byte yang dihasilkan dalam proses kompilasi Java akan selalu sama untuk setiap OS dan jenis mesinnya, tapi JVM akan mengubah kode byte tersebut menjadi bahasa mesin tujuannya.

Teknologi JAVA

Java Application Programming Interface (Java API) menyediakan fitur-fitur yang menarik untuk mengembangkan aplikasi, antara lain :

▶ ***Applet***

Java Applet merupakan program Java yang berjalan di atas browser. Dengan menggunakan Java Applet, maka halaman HTML akan lebih dinamis dan menarik, sangat cocok untuk pengembangan aplikasi-aplikasi berbasis web.

▶ ***Java Networking***

Java Networking merupakan sekumpulan Application Programming Interface (API) yang menjadikan fungsi-fungsi untuk aplikasi-aplikasi jaringan. Java Networking menyediakan akses untuk TCP, UDP, IP Address dan URL, tetapi tidak untuk ICMP dikarenakan alasan sekuriti.

▶ ***Java Database Connectivity (JDBC)***

JDBC menyediakan sekumpulan Application Programming Interface (API) yang dapat digunakan untuk mengakses database seperti Oracle, MySQL, PostgreSQL, Microsoft SQL Server.

▶ ***Java Security***

Java Security menyediakan sekumpulan Application Programming Interface (API) yang dapat digunakan untuk mengatur *security* dari aplikasi Java baik secara *high level* maupun *low level*, seperti *public/ private key management* dan *certificates*.

Teknologi JAVA

▶ *Java Swing*

Java Swing menyediakan sekumpulan Application Programming Interface (API) yang dapat digunakan untuk membangun aplikasi-aplikasi GUI (Graphical User Interface) dan model GUI yang diinginkan bisa bermacam-macam.

▶ *Java RMI*

Java RMI menyediakan sekumpulan Application Programming Interface (API) yang dapat digunakan untuk membangun aplikasi-aplikasi Java yang mirip dengan model RPC (Remote Procedure Call), jadi objek-objek Java bisa di-*call* secara *remote* pada jaringan komputer.

▶ *Java 2D/ 3D*

Java 2D/ 3D menyediakan sekumpulan Application Programming Interface (API) yang dapat digunakan untuk membangun grafik-grafik 2D/ 3D yang menarik dan juga akses ke printer

▶ *Java Server Pages (JSP)*

Berkembang dari Java Servlet yang digunakan untuk mengganti-kann aplikasi-aplikasi CGI, JSP (Java Server Page) yang mirip ASP dan PHP merupakan alternatif terbaik untuk solusi aplikasi internet

Teknologi JAVA

▶ **Java Native Interface (JNI)**

JNI menyediakan sekumpulan Application Programming Interface (API) yang dapat digunakan untuk mengakses fungsi-fungsi pada library (*.dll atau *.so) yang dibuat dengan bahasa pemrograman yang lain seperti C++, dan Basic.

▶ **Java Sound**

Java Sound menyediakan sekumpulan Application Programming Interface (API) yang dapat digunakan untuk memanipulasi *sound*.

▶ **Java IDL + COBR**

Java IDL (*Interface Definition Language*) menyediakan dukungan Java untuk implementasi COBRA (*Common Object Request Broker*) yang merupakan model objek terdistribusi untuk solusi aplikasi besar di dunia jaringan.

▶ **Java Card**

Java Card utamanya digunakan untuk aplikasi-aplikasi pada *smart card*, yang sederhana wujudnya seperti SIM card pada *handphone*

▶ **Java Telephony Application Programming Interface (JTAPI)**

JTAPI menyediakan sekumpulan Application Programming Inter-face (API) yang dapat digunakan untuk memanfaatkan alat-alat telepon, sehingga akan cocok untuk aplikasi-aplikasi CTI (*Compu-ter Telephony Integration*) yang dibutuhkan seperti ACD (*Auto-matic Call Distribution*), PC-PBX dan lainnya.

Kelebihan JAVA

Semboyan JAVA , “ “write once run anywhere” “

Kelebihan JAVA :

▶ ***Bersifat portable dan platform independent***

Program Java dapat dijalankan di sembarang platform asalkan Java Virtual Machine (JVM) untuk platform tersebut tersedia

▶ ***Pembuang Sampah (Garbage Collection)***

Program Java membuang sendiri “sampah-sampah” yang tak berguna, artinya program tidak perlu menghapus objek-objek yang dialokasikannya di memori dan dealokasi memori secara otomatis. Ini menjadikan program Java bebas dari masalah-masalah pengelolaan memori.

Kelebihan JAVA

▶ *Bebas Arsitektur (Architecture Neutral)*

Pada dasarnya, program Java tidak dirancang untuk prosesor ataupun sistem operasi tertentu. Ia dirancang untuk bekerja pada berbagai arsitektur prosesor dan berbagai sistem operasi. Oleh karena itu, ia bekerja dengan interpreter Java untuk mengubah program Java menjadi *byte-code* sebelum dijalankan oleh suatu mesin tertentu

▶ *Tangguh (Robust)*

Karena interpreter Java memeriksa seluruh akses sistem yang dilakukan program, maka program Java tidak akan membuat sistem menjadi crash. Bila terjadi masalah serius, program Java membuat pengecualian (exception). Exception ini dapat ditangani dan dike-lola oleh program tanpa beresiko memacetkan sistem.

▶ *Aman (Secure)*

Sistem Java sangat handal dalam mengelola memori, tidak hanya memverifikasi seluruh akses ke memori, tetapi juga menjamin tidak ada virus yang "membonceng" pada program yang sedang berjalan. Karena Java tidak mendukung pointer, program tidak mendapat akses ke area sistem dimana ia tidak mendapat otorisasi.

▶ *Dapat Diperluas*

Program Java mendukung *metode native code*, yaitu memungkinkan pemrogram menulis fungsi dalam bahasa lain, biasanya C++ yang bisa dieksekusi secara lebih cepat karena langsung berjalan di atas perangkat keras yang bersangkutan, dari pada yang ditulis dalam Java yang berjalan di atas *Java Virtual Machine (JVM)*. *Metode native code* dikaitkan secara dinamis (*dynamically linked*) ke program Java, yakni dikaitkan dengan program saat runtime. Pada perkembangannya, bila Java berkembang lebih jauh dan mampu mengeksekusi fungsi-fungsi lebih cepat dari bahasa lain, *metode native* mungkin tidak diperlukan lagi

Karakteristik JAVA

▶ *Sederhana (Simple)*

Java dirancang untuk mudah dipelajari, terutama bagi pemrogram yang telah mengenal C++ akan mudah sekali untuk berpindah ke Java. Selain itu pengembang Java banyak membuang fitur-fitur yang tidak diperlukan seperti bahasa pemrograman tingkat tinggi, seperti *pointer* matematis, *casting tipe implicit*, struktur atau *union*, *operator overloading*, *template*, *header filer*, maupun pewarisan majemuk (*multiple inheritance*). Pemrograman Java bisa dikatakan semudah C dan seampuh C++

▶ *Berorientasi Objek (Object Oriented)*

Java menggunakan kelas-kelas untuk mengelola kode ke dalam modul-modul logika. Pada saat *runtime*, program menciptakan objek dari kelas-kelas yang ada. Kelas-kelas Java dapat merupa-kan warisan dari kelas lain, tetapi pewarisan majemuk tidak diper-kenankan.

▶ *Bertipe Statik*

Semua objek-objek yang digunakan dalam program harus didekla-rasikan sebelum digunakan. Cara ini memungkinkan kompilator Java melaporkan dan melokalisir terjadinya konflik-konflik tipe

▶ *Terkompilasi*

Sebelum program Java dapat berjalan, maka harus dikompilasi oleh kompilator Java. Hasil kompilasinya berupa file kode byte (*byte-code*), yang mirip dengan kode mesin, yang dapat dieksekusi di bawah sistem operasi apapun asal memiliki intepreter Java. Jadi dapat dikatakan program Java merupakan bahasa yang di-kompilasi sekaligus diintepretasi

▶ *Multi-threaded*

Program Java dapat berisi eksekusi multiple threads, yang me-mungkinkan program mengulas citra di layar pada satu thread, sementara tetap dapat menerima masukan dari keyboard pada thread utamanya. Semua aplikasi memiliki paling tidak satu thread yang menggambarkan alur eksekusi utama program

Pertanyaan dan Diskusi ???