

TPI4202
e-tp.ub.ac.id

Lecture 3

Introduction to Java

Sekilas Java

- Bahasa pemrograman berorientasi objek (*object-oriented programming/OOP*)
 - Tidak mungkin membuat program java tanpa kita mendefinisikan class, data dan method
- Bersifat netral, tidak tergantung suatu platform dan mengikuti prinsip WORA (*Write Once and Run Anywhere*)
 - Spesifikasi bahasa dan implementasi Java sudah ditetapkan dan tidak tergantung pada mesin dan platform yang akan digunakan
- Dipelopori James Gosling, dkk dari Sun Microsystems, Inc pada tahun 1991



Sekilas Java

- Java adalah pemrograman yang lahir untuk menjawab teknologi jaringan yg interaktif
- Program java dapat dibuat dengan dua pendekatan, sebagai **aplikasi** yang berdiri sendiri layaknya program C atau Pascal, atau sebagai **applet** yang disisipkan dalam dokumen HTML
- Open Source
- Java dirancang menjadi bahasa yang sederhana, meminimalkan kesalahan, namun tangguh



Sekilas Java

- Suatu aplikasi Java dapat memanfaatkan Java API (Application Programming Interface)
 - Java API berisi koleksi class-class siap pakai yang mempermudah dalam penulisan aplikasi
- Java tidak mempergunakan compiler-directive pada source code
- Java memiliki fasilitas garbage-collector yang bekerja secara otomatis
 - menjamin ketersediaan memory
- Java sangat portabel, baik file sumbernya maupun hasil kompilasinya
- Java tidak mengenal multiple inheritance



Pemrograman Prosedural vs Pemrograman Berorientasi Objek

- Pemrograman prosedural menggabung semua data dan kode menjadi satu dalam satu program
- Pemrograman berorientasi objek secara konseptual memecah komponen-komponen penyusun program menjadi bagian-bagian tersendiri yang disebut “objek”
 - Setiap objek memiliki *data* (sifat berupa variabel atau konstanta) dan *method* (perilaku berupa fungsi)
 - Objek berupa entitas yang memiliki *data* dan *method*



Sifat Pemrograman Java

- Berorientasi Objek
 - Penulisan program harus dalam bentuk objek-objek kemudian memodelkan sifat masing-masing objek
- Terdistribusi
 - Dapat menyebarkan/mendistribusikan data melalui jaringan
- Multiplatform
 - Program java yang dikompilasi dengan benar dapat dijalankan pada platform yang berbeda
- Multithread
 - Menjalankan beberapa pekerjaan dalam waktu yang hampir bersamaan → client-server



Penulisan Program Java (1)

1. Setiap program Java harus disimpan sebagai file yang diakhiri dengan ekstensi **.java**
2. Nama file harus sama dengan nama dari class **public**.
Sebagai contoh, apabila nama dari class public pada program adalah **Hello**, maka nama filenya adalah **Hello.java**
3. Comment sebaiknya ditulis untuk menjelaskan perilaku dari sebuah class, atau perilaku dari sebuah method.



Penulisan Program Java (2)

- Struktur program java adalah :

```
(Modifier 1) class Nama_Kelas (Modifier 2)
{
    class body
}
```

- Struktur program pengembangan adalah semata-mata mengembangkan class-class inti java menjadi solusi atas kebutuhan
- Identifier (nama, variabel, konstanta, method) tidak boleh menggunakan keyword Java



Jenis Modifier dalam Java

Public	Method/field dapat diakses oleh semua objek
private	Method/field hanya dapat dipergunakan oleh class tersebut
protected	Method/field tersebut hanya dapat dipergunakan oleh class tersebut dan turunannya
Static	Method/field tersebut dapat diakses tanpa harus menciptakan objeknya terlebih dahulu. Semua objek yang dibuat dari class yang memiliki field static akan memiliki nilai yang sama untuk field tersebut.
Final	Menyatakan bahwa definisi dari field/method tersebut tidak dapat berubah lagi.



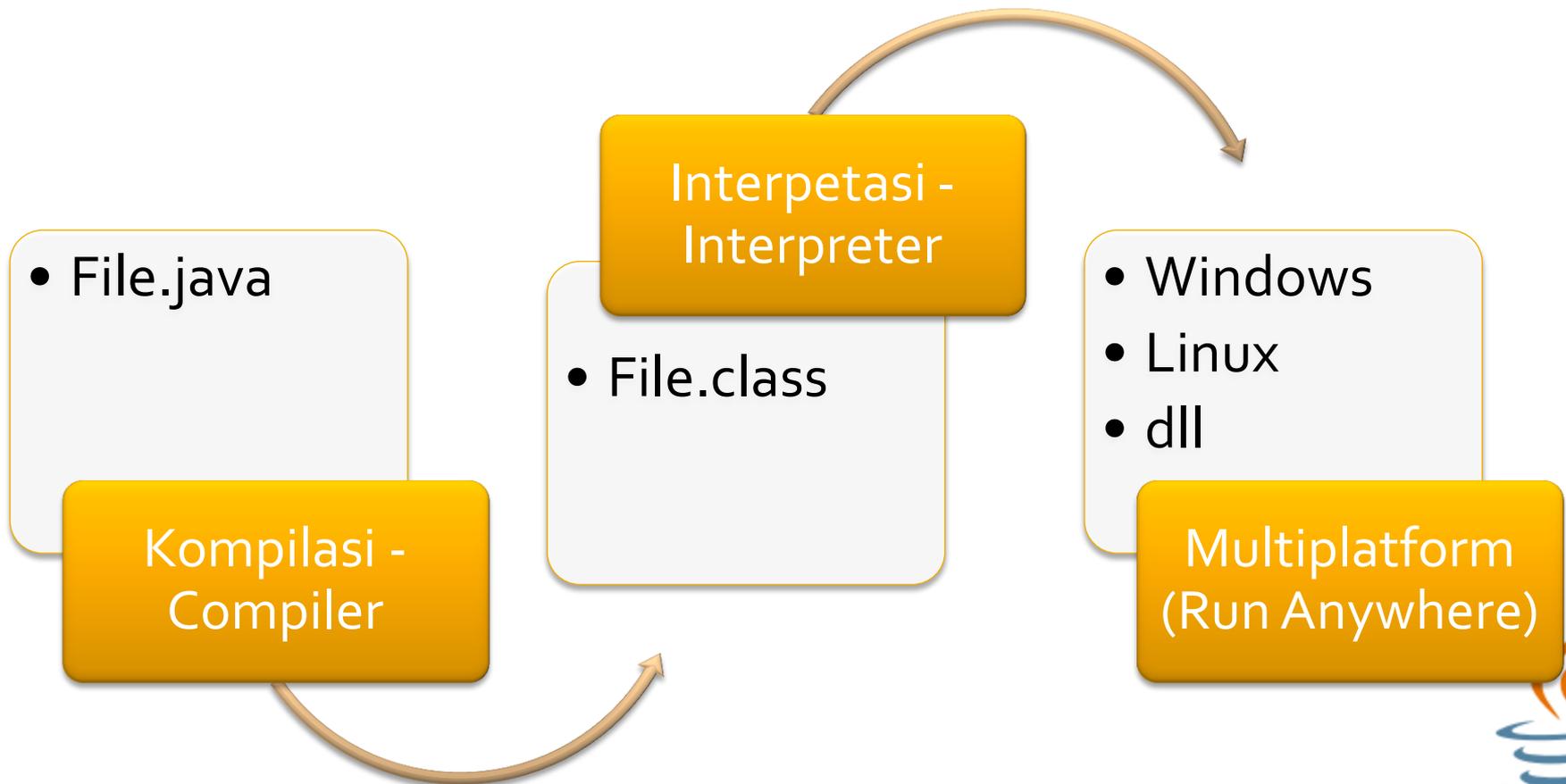
Java Keywords

<code>abstract</code>	<code>continue</code>	<code>for</code>	<code>new</code>	<code>switch</code>
<code>assert^{***}</code>	<code>default</code>	<code>goto[*]</code>	<code>package</code>	<code>synchronized</code>
<code>boolean</code>	<code>do</code>	<code>if</code>	<code>private</code>	<code>this</code>
<code>break</code>	<code>double</code>	<code>implements</code>	<code>protected</code>	<code>throw</code>
<code>byte</code>	<code>else</code>	<code>import</code>	<code>public</code>	<code>throws</code>
<code>case</code>	<code>enum^{****}</code>	<code>instanceof</code>	<code>return</code>	<code>transient</code>
<code>catch</code>	<code>extends</code>	<code>int</code>	<code>short</code>	<code>try</code>
<code>char</code>	<code>final</code>	<code>interface</code>	<code>static</code>	<code>void</code>
<code>class</code>	<code>finally</code>	<code>long</code>	<code>strictfp^{**}</code>	<code>volatile</code>
<code>const[*]</code>	<code>float</code>	<code>native</code>	<code>super</code>	<code>while</code>

* not used, ** added in 1.2, *** added in 1.4, **** added in 5.0



Proses Pembuatan Program Java



Contoh Program Java

```
public class Hello
{
    /**
     * Program Java pertamaku
     */
    public static void main( String[] args ) {
        //menampilkan String Hello world pada
        layar
        System.out.println("Hello world");
    }
}
```



Penjelasan Contoh (1)

Hello - deklarasi nama class

Pada Java, semua code harus berada di dalam deklarasi sebuah class.

Class menggunakan akses **public**, sehingga class **Hello** dapat diakses oleh class lain yang berada di luar package (package adalah sekumpulan dari class).

Baris berikutnya merupakan tanda kurung kurawal
{ yang merupakan awal dari sebuah blok.



Penjelasan Contoh (2)

Tiga baris selanjutnya merupakan **comment** pada Java.

Comment

- Catatan yang ditulis sebagai dokumentasi pada suatu bagian code.
- Comment bukanlah bagian dari program, akan tetapi digunakan untuk tujuan dokumentasi.
- Menambahkan comment pada source code merupakan cara yang baik untuk berlatih pemrograman.



Penjelasan Contoh (3)

method main() - nama dari sebuah method pada class Hello

method **main()** merupakan *starting point* pada program Java.

Semua program Java kecuali Applet dijalankan pada method **main()**.

Baris selanjutnya merupakan *comment* pada Java

Perintah **System.out.println()** digunakan untuk menampilkan ke layar berupa teks yang ada antara tanda petik ganda



Penjelasan Contoh (4)

- Pada dua baris berikutnya, terdapat dua tanda kurung kurawal yang digunakan untuk menutup method **main()** dan deklarasi class.



Terima kasih

