



Pendahuluan Dasar Pemrograman Java

Oleh:
Mike Yuliana
PENS-ITS



Tujuan Instruksional Khusus

- Memahami tentang lingkungan java
- Memahami tentang aplikasi program java sederhana
- Memahami tentang proses input dan output



Sejarah Java

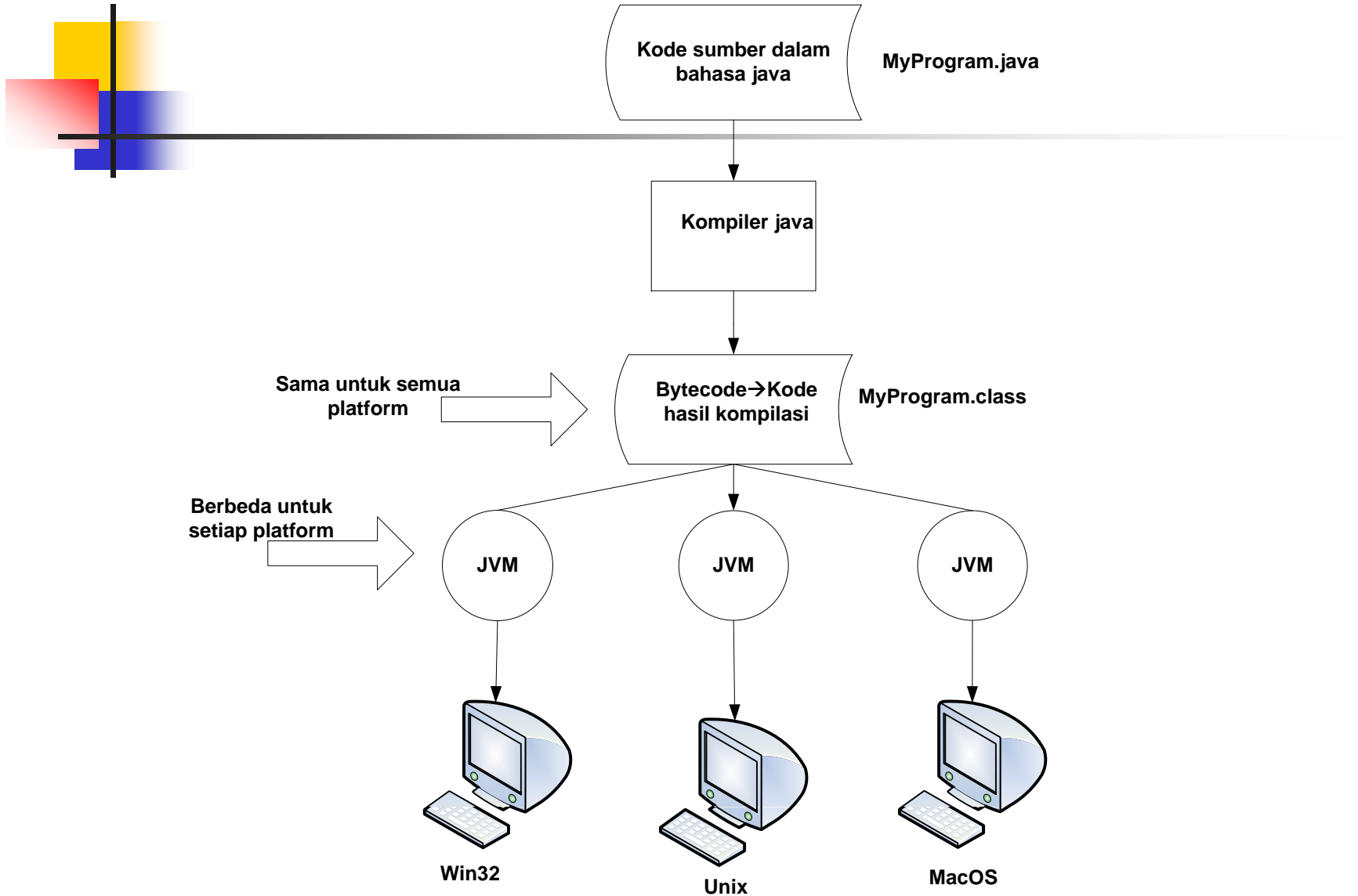
- Bahasa pemrograman Java lahir pertama kali dari The Green Project (James Gosling dkk dari sun microsystems) pada tahun 1991 → Teknologi “*7” (Star Seven)
- Mula-mula James Gosling menyebut bahasa pemrograman yang dihasilkannya dengan OAK → Sun mengubahnya menjadi Java
- Sun merilis versi awal java secara resmi pada awal 1996 yg kemudian terus berkembang hingga muncul JDK 1.1 kmd JDK 1.2 shg mulai versi ini java disebut java 2
- Pada tahun 1998-1999 lahirlah teknologi J2EE (Java 2 Enterprise Edition)
- Terakhir Teknologi Java melahirkan J2ME (Java 2 Micro Edition) yang sudah diadopsi oleh Nokia, Siemens, SonyEricsson, Motorola untuk menghasilkan aplikasi mobile baik games maupun software lainnya yang dpat dijalankan diperalatan mobile seperti handphone



Kelebihan Java

- Multiplatform → dapat dijalankan di beberapa *platform* / sistem operasi komputer
- OOP (Object Oriented Programming) → semua aspek yang terdapat di Java adalah Objek.
- Perpustakaan Kelas yang lengkap → Java terkenal dengan kelengkapan *library*/ perpustakaan (kumpulan program yang disertakan dalam pemrograman java) yang sangat memudahkan dalam penggunaan oleh para pemrogram untuk membangun aplikasinya.
- Bergaya C++ → memiliki sintaks seperti bahasa pemrograman [C++] sehingga menarik banyak pemrogram C++ untuk pindah ke Java.

Lingkungan Java



Kompiler dan Interpreter

- **J2SDK: seperangkat tool (kit) yang berisi kompiler dan interpreter java yang direlease oleh sun dan bersifat free → www.sun.com**

Nama Program	Keterangan
javac	Kompiler
java	Interpreter
jdb	Debugger
javap	Disassembler
appletviewer	Penampil applet
javadoc	Pembangkit dokumentasi
javah	Pembangkit berkas header C

Program pada J2SDK



Editor

- **Notepad** → editor standard yang disediakan oleh windows. Tidak dilengkapi dengan utilitas jumlah baris (berpengaruh pada pelacakan baris program yang menyebabkan error kompilasi)
- **Netbeans** → salah satu editor yang dilengkapi dengan pelacakan baris program yang menyebabkan error kompilasi
- **Editor-editor lain: Eclipse, Gel, JBuilder, Jcreator dll**

Aplikasi Sederhana dari Java

Contoh program 1:

Masing-masing program berada dalam sebuah class

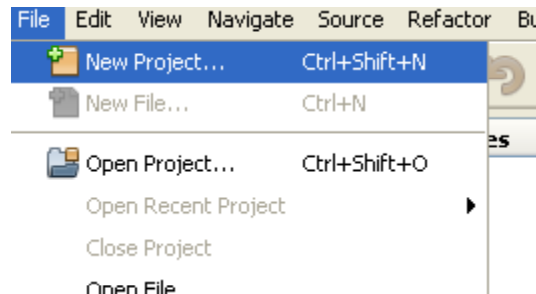
main() adalah method pertama yang dijalankan.

```
class coba {  
    public static void main (String[] args) {  
        System.out.println("Hello World!");  
    }  
}
```

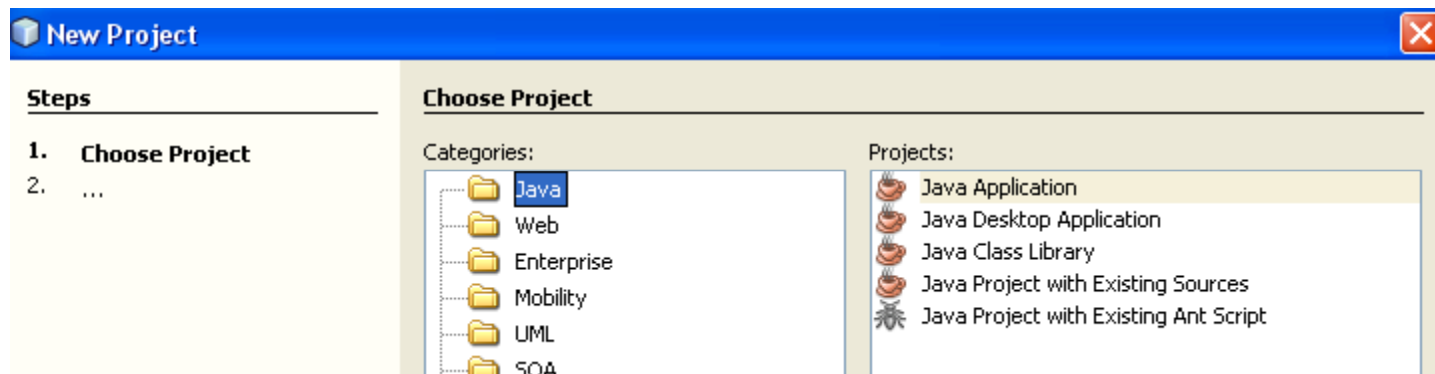
Syntax hampir sama dengan C –
kurung kurawal untuk blok, titik
koma setelah pernyataan.

Langkah-langkah pembuatan project:

1. Start NetBeans IDE.
2. Pada IDE, pilih File > New Project (Ctrl-Shift-N), seperti yang ditunjukkan oleh gambar dibawah ini



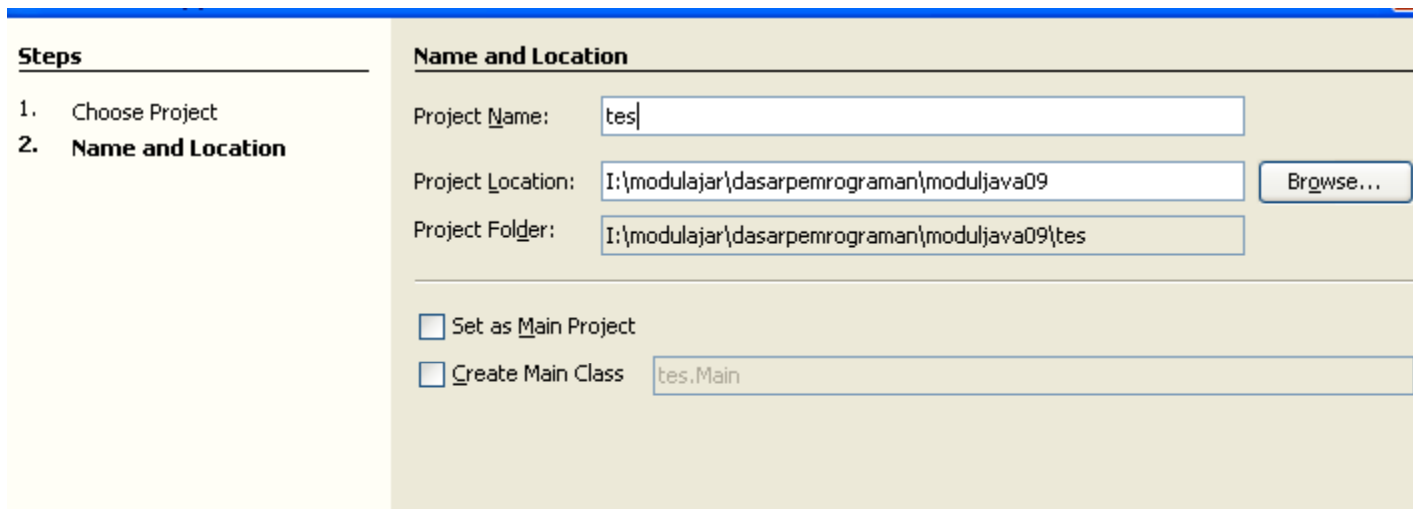
3. Pada New Project wizard, Pilih categories Java dan pilih Java Application seperti yang ditunjukkan oleh gambar dibawah ini. Kemudian pilih Next.



4. Pada halaman Name and Location dari wizard, pilih:

-- Nama dari project.

-- Lokasi dari project



The screenshot shows a wizard window with a 'Steps' pane on the left and a 'Name and Location' pane on the right. The 'Steps' pane lists two steps: '1. Choose Project' and '2. Name and Location', with the second step being the active one. The 'Name and Location' pane contains three text input fields: 'Project Name' with the value 'tes', 'Project Location' with the value 'I:\modulajar\dasarpemrograman\moduljava09', and 'Project Folder' with the value 'I:\modulajar\dasarpemrograman\moduljava09\tes'. There is a 'Browse...' button next to the Project Location field. Below the input fields, there are two checkboxes: 'Set as Main Project' (unchecked) and 'Create Main Class' (unchecked). The 'Create Main Class' checkbox has a text input field next to it containing the value 'tes.Main'.

Kemudian klik Finish.

5. Buat file .java, kemudian klik finish

Steps

- Choose File Type
- **Name and Location**

Name and Location

Class Name:

Project:

Location:

Package:

Created File:

Tampilan yang dihasilkan

The screenshot displays an IDE interface. On the left, the 'Projects' view shows a tree structure for a project named 'ChatApplication'. Underneath, there are sub-projects: 'cobaku', 'cobalagi', 'soket', and 'tes'. The 'tes' project is expanded to show 'Source Packages' containing a '<default package>' folder with a 'tes.java' file. Below this, there are 'Test Packages', 'Libraries', and 'Test Libraries' sections. A 'tes.java - Navigator' window is also visible at the bottom left.

On the right, the code editor shows the content of 'tes.java'. The code is as follows:

```
...va | Log.java x | cobaku.java x | cobalagi.java x | cobalagi.java x
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
/**
 *
 * @author user
 */
public class tes {
}
```

Contoh program 2:

Untuk menampilkan kalimat:

Halo apa kabar PENS ITS tercinta



```
System.out.println("halo apa kabar PENS ITS tercinta");
```

atau

```
System.out.println("halo apa kabar" + "PENS ITS tercinta");
```

atau

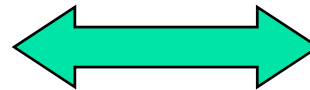
```
System.out.print("halo apa kabar");
```

```
System.out.println("PENS ITS tercinta");
```

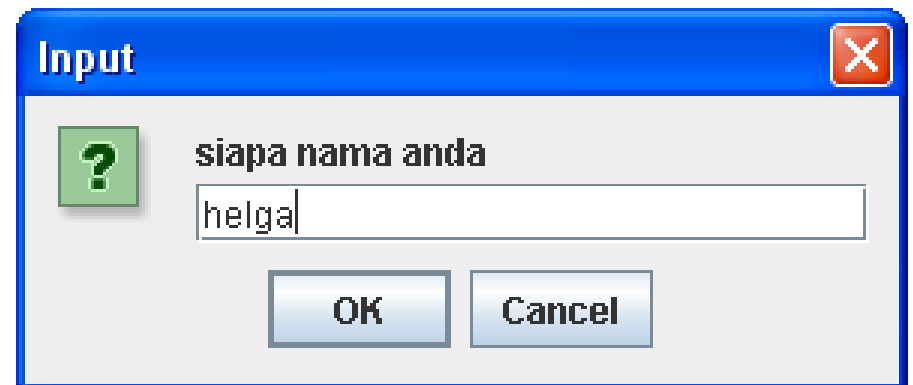
Contoh Program 3:

- Menggunakan JOptionPane yang dapat menampilkan window kecil dengan textfield untuk menerima input dari keyboard
- JOptionPane merupakan bagian dari package javax.swing → import javax.swing
- JOptionPane menerima input berupa String

```
import javax.swing.*;  
public class hallo {  
    public static void main(String args[ ]) {  
        String nama=JOptionPane.showInputDialog("siapa  
        nama anda");  
        System.out.println(nama);  
        System.exit(0);  
    }  
}
```



Membaca Input data dari keyboard



Contoh Program 4:

Untuk menghendaki data yang bertipe integer, float, dan double → bisa digunakan kelas Integer, Float, Double dan memanggil metode parseInt, parseDouble, parseFloat

```
import javax.swing.*;
```

```
public class Hallo {
```

```
    public static void main(String args[ ]) {
```

```
        final double PI = 3.14;
```

```
        double Luas;
```

```
        String jari=JOptionPane.showInputDialog("masukkan  
                jari-jari");
```

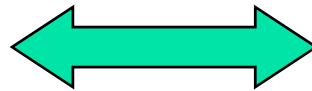
```
        double jari_jari=Double.parseDouble(jari);
```

```
        Luas = PI * jari_jari * jari_jari;
```

```
        System.out.println(Luas);
```

```
    }
```

```
}
```



Mengkonversi String ke Type Yang lain

