



# 2

# COMPUTER PROGRAMMING



**Execution Environment  
for JAVA Development**



# CONTENTS

- General Architecture of JAVA
- JAVA Integration Environment
- Application Program and Applet Program
- JAVA Application
- Download Netbean and installation



# Main Structure

01: **package** javaapplication1; /\* 패키지 정의: \*/

02: **public class** Main { /\* 클래스 정의\*/

    // 메소드 정의

03:     **public** Main( ) { // 생성자 메소드 정의 :  
                          //

04:     }

05:     **public static void** main(String[] args) { // Main() 메소드 정의

06:         System.out.println("Hello World"); // 프로그램 문장 기술

07:     }

08: }

# Process Flow of JAVA

## □ JAVA Execution Structure



# Process Flow of JAVA

## □ Binary Code



# Process Flow of JAVA

- **JAVA Compiler Command : javac**
  - That is a tool that turns to Java source code (filename.java) into byte code (filename.class)
  -
- **JAVA Interpreter Command : java**
  
- **SDK (software development kit) tools**
  - Java Compiler: javac
  - Java Interpreter : java
  - Applet program runner : appletviewer

# Process Flow of JAVA



# Java Configuration to Execute

## □ JDK installation

- <http://java.sun.com/javase/downloads/index.jsp>

## □ Check

- Java compiler file and execution file in folder C:\Program Files\Java\jdk1.6.0\bin whether installed.



# Java Configuration to Execute(not Netbean)

## □ Check Configuration

- Open a command window to check whether Java execution environment operates properly after installing JDK
  - Start – command line – write "cmd"
- Write "java -version" in **JDK installed directory**(~/bin)
  - move: c~>cd \ or c~>cd directory name

# Java Configuration to Execute(not Netbean)

## □ JAVA in Notepad (Memo)

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

# Java Configuration to Execute(not Netbean)

## □ Java Compile before configuration

- C:\javawork> **javac** Hello.java  
>> error occurred, therefore need configuration (path)



# Java Configuration to Execute(not Netbean)

- **System Variable and Environment Variable**
- **OS has the both values.**
  - For instance, Window environment variables
  
  - Check each content of windows environment variables



# Java Configuration to Execute(not Netbean)

## □ Environment variable : PATH

- If Javac.exe file runs in everywhere, configure PATH

## □ Environment variable : CLASSPATH

### ■ JRE ?

- JRE abbreviates words **J**AVA **R**UNTIME **E**NVIRONMENT that means Java execution environment.
- A kind of tools(kit) to make programmer develop SW using JDK

# Java Configuration to Execute(not Netbean)

- JRE(java runtime environment)?



# Java Configuration to Execute(not Netbean)

□ To use classes provided by JAVA, configure CLASSPATH.

■ CLASSPATH = .; C:\Program Files\Java\jre 1.6.0\lib



# Java Configuration to Execute(path,classpath)

□ 내컴퓨터속성-[시스템 등록 정보] 창의 [고급] 탭에서 [환경 변수] 설정

- Copy string C:\Program Files\Java\jdk1.6.0\bin  
javac.exe의 마우스 우측버튼-속성에서 위치를 복사  
내컴퓨터의 속성-시스템등록정보-고급탭-환경변수
  - Path: “C:\Program Files\Java\jdk1.6.0\_11\bin;”
  - Classpath: “C:\Program Files\Java\jre6\lib”



# Java Configuration to Execute

- Steps of Java compiling after configuration



# Java Configuration to Execute

## □ Running java source program

■ Caution : Uppercase, lowercase (class name)

➤ Ex) > java Hello vs >java hello



## □ <follow>- Java compiler configuration in EditPlus

- 1. Select menu [도구]-[기본설정]
- 2. [도구]-[사용자 도구]에서 그룹에 가서 [그룹 이름..] 버튼을 눌러서 그룹의 이름을 “자바”로 바꾼다.
- 3. [추가] 버튼을 누른 후 [프로그램]을 선택한다.

# JAVA in EditPlus Tool

- 4. [메뉴제목:] 항목의 내용을 “자바컴파일러”로 바꾼다.
  - [명령:]의 [...] 버튼을 클릭 javac.exe 파일을 선택.
  - [인수:]에서 “파일 이름”을 선택.
  - [디렉토리:]에서 “파일 디렉토리” 선택
  - 출력내용 캡처 선택 및 적용 클릭
- 5. 상기에서처럼 “자바실행”을 추가
  - 단, [인수:]에서 “확장자를 뺀 파일이름”을 선택

## □ EditPlus에서 자바 실행

- [파일] – [새 파일] – [Java] 선택
  - 새로운 자바예제(“Hello World” 출력) 파일(Class이름은 제외된상태)
- Class 이름에 “firstApp”입력 – [파일] – [저장]
  - (클래스이름과 동일한이름. 단, 확장자는 제외)
- [보기] – [도구모음/창] - [사용자도구모음]
  - 작성한 “자바” 그룹 표시

# Download NetBean and Installation

## □ 선(sun) 사이트 접속

- <http://java.sun.com/javase/downloads/index.jsp>

- JDK & with NetBean 5.5 선택
- JDK 6 Update 11 with NetBeans 6.5 선택(web 개발도구 제외)

- 추천 : <http://www.netbeans.org/downloads/index.html>

(All 선택 download)

- Download
- Installation

## □ Net Bean 6.5 실행

# Download NetBean and Installation

- A execution snapshot of Net Bean 6.5



# Print out “Hello World”

## □ Menu – File – New Project

- Select Categories(JAVA) – Projects (Java Application)
- Choice default
- Click Finish

## □ Main 메소드에 다음 문장(“Hello World”)출력해보기

- “run”의 “compile \_ main.java” 클릭하여 실행
- 결과 확인(“Hello World”) 출력



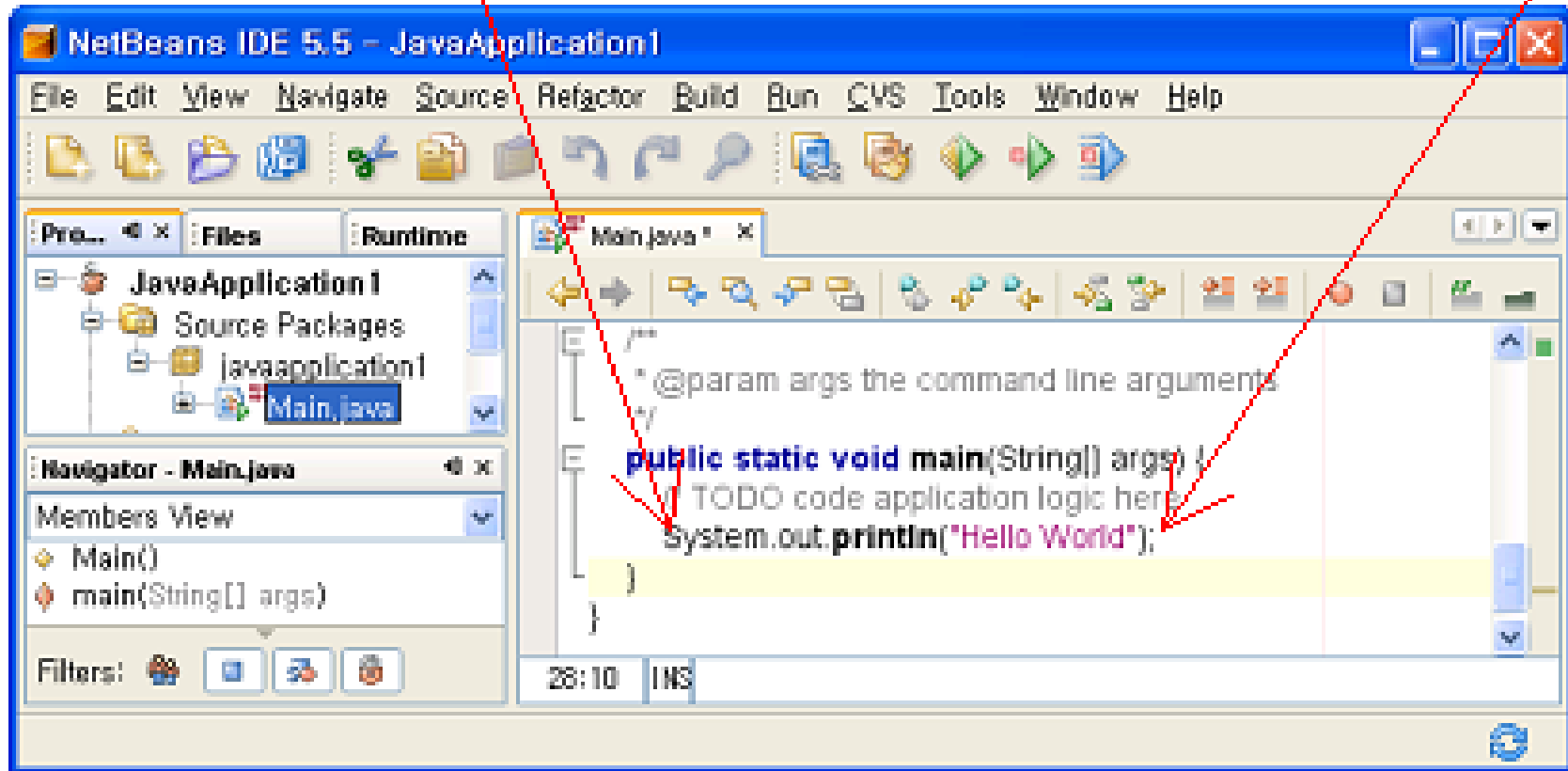
# JAVA in Net Bean

외부에서 Main사용시:  
`import javaapplication1.Main;`



# JAVA in Net Bean

```
System.out.println("Hello World");
```



# Code Insight function



# JAVA Compiling

## □ Menu [Build]-[Compile XXX.java]

The screenshot shows the NetBeans IDE 5.5 interface. The 'Build' menu is open, displaying the following options:

- Build Main Project (F11)
- Clean and Build Main Project (Shift+F11)
- Generate Javadoc for "JavaApplication1"
- Compile "Main.java" (F9)** (highlighted)
- Compile "Main.java" (highlighted)
- Stop Build/Run

The 'Main.java' file is open in the editor, showing the following code:

```
public static void main(String[] args) {  
    // TODO code application logic here  
    System.out.println("Hello World");  
}
```

The 'Output' window at the bottom shows the compilation results:

```
init:  
deps-jar:  
Created dir: C:\J\JavaApplication1\build\classes  
Compiling 1 source file to C:\J\JavaApplication1\build\classes  
compile-single:  
BUILD SUCCESSFUL (total time: 4 seconds)
```

At the bottom of the IDE, a status bar message reads: "Finished building JavaApplication1 (compile-single)." The cursor is positioned at line 29, column 1.

# JAVA Run

## □ Menu [Run]-[Run Main Project]

The screenshot displays the NetBeans IDE 5.5 interface for a Java project named 'JavaApplication1'. The 'Run' menu is open, showing various options for running and debugging the project. The 'Run Main Project' option is highlighted, with a keyboard shortcut of F6. Other options include 'Debug Main Project' (F5), 'Test "JavaApplication1"' (Alt+F6), 'Run File', 'Attach Debugger...', 'Finish Debugger Session' (Shift+F5), 'Pause', 'Continue' (Ctrl+F5), 'Step Over' (F8), 'Step Into' (F7), 'Step Out' (Ctrl+F7), 'Run to Cursor' (F4), 'Run Into Method' (Shift+F7), 'Apply Code Changes', 'Stack', 'Toggle Breakpoint' (Ctrl+F8), 'New Breakpoint...', 'New Watch...', and 'Evaluate Expression...' (Ctrl+F9).

The IDE interface includes a menu bar (File, Edit, View, Navigate, Source, Refactor, Build, Run, CVS, Tools, Window, Help), a toolbar, a Project Explorer showing the project structure (Source Packages, javaapplication1, Test Packages, Libraries), a Navigator for 'Main.java' showing 'Main()' and 'main(String[] args)', a code editor with a snippet of Java code, and an Output window showing the following text:

```
init :
deps-jar :
compile :
run :
Hello World
BUILD SUCCESSFUL (total time: 0 seconds)
```

The status bar at the bottom indicates 'Finished building JavaApplication1 (run)'.

# CONCLUDE

- General Architecture of JAVA
- Configuration
- Execution Environment of JAVA using EditPlus tool
- JAVA Development Environment
- JAVA Integration Environment
- Application Program and Applet Program
- JAVA Application
- Download Netbean and installation

