



*Power Java*

## 제2장 자바 개발 도구



© 2009 인피니티박스 All rights reserved



## 자바를 사용하려면 무엇이 필요한가?

- 명칭: JDK (Java Development Kit)
- 설명: 자바 개발 도구
- 다운로드 위치: [java.sun.com](http://java.sun.com)
- 비용: 무료

© 2009 인피니티박스 All rights reserved



# JDK와 JRE

- JRE(Java Runtime Environment)
  - JRE는 자바 프로그램을 실행하기 위한 라이브러리, 자바 가상 기계, 기타 컴포넌트들을 제공한다.
- JDK(Java Development Kit)
  - JDK는 JRE에 추가로 자바 프로그램을 개발하는데 필요한 컴파일러, 디버거와 같은 명령어행 개발 도구를 추가한 것이다.

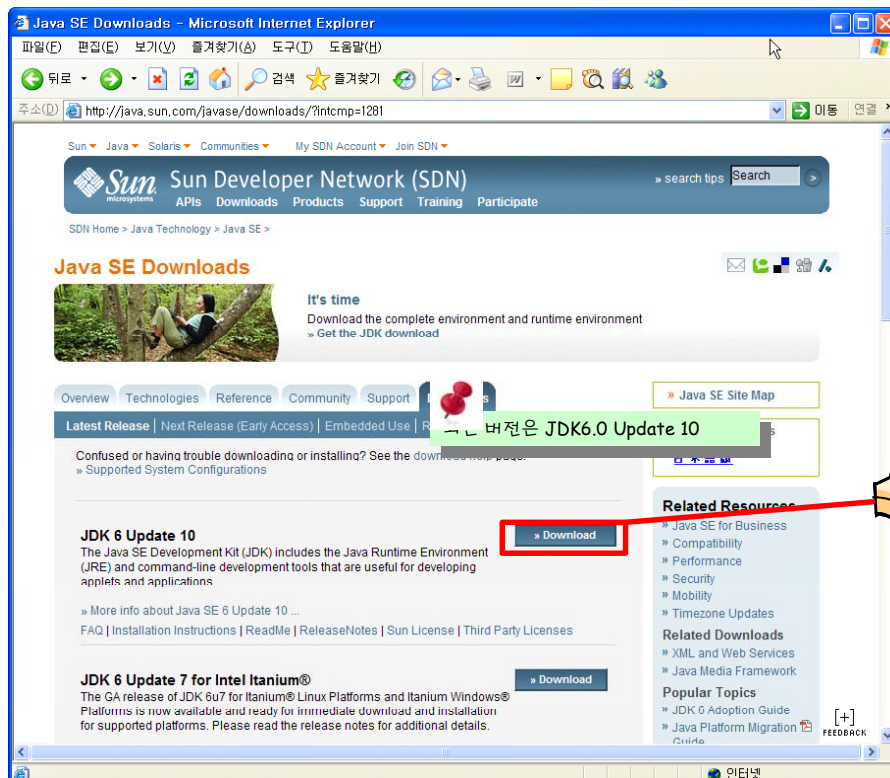


*JDK = JRE + 컴파일러 + 디버거 + ...*

© 2009 인피니티북스 All rights reserved



# 다운로드



© 2009 인피니티북스 All rights reserved



# JDK의 구조

표 2.1 JDK의 디렉토리 구조

폴더	설명
bin	자바 언어를 이용하여 프로그램을 개발하고 실행하며 디버깅, 주석 작업을 도와주는 도구.
db	Java DB, 아파치 Derby 데이터베이스 기술의 선 마이크로 시스템의 배포판 포함.
demo	데모 애플릿과 애플리케이션이 소스와 함께 제공, Swing, Java Foundation Classes, Java Platform Debugger Architecture 등이 포함되어 있다.
include	네이티브 코드 프로그래밍을 지원하는 헤더 파일들이다. 이들 파일들은 자바와 C를 동시에 사용하는 프로그램 개발시에 쓰인다.
jre	자바 실행 환경. 자바 가상 기계, 클래스 라이브러리들, 기타 자바 프로그램의 실행을 지원하는 파일들로 이루어져 있다.
lib sample	개발 도구들이 필요로 하는 추가적인 클래스 라이브러리와 지원 파일들이다. 특정한 자바 API를 사용하는 프로그램 소스 샘플
src	자바 API 클래스들의 소스 파일 포함, JDK 폴더 안에 존재하는 src.zip의 압축을 풀은 경우에만 생성. 자바 핵심 API를 이루는 클래스 라이브러리 소스이다. 이들 소스는 자바 언어를 배우고 사용하는 것을 도와주기 위하여 필요한 정보 제공 차원에서 제공된다.

© 2009 인피니티박스 All rights reserved



## 중간 점검 문제

1. 자바 프로그램을 개발하는 데 개발 도구를 JDK라고 한다.
2. JDK와 JRE의 차이점은 무엇인가?

**JRE는 자바 프로그램을 실행만 시킬 수 있는 환경이고 JDK는 개발도 할 수 있는 환경이다.**

© 2009 인피니티박스 All rights reserved



## 자바 프로그램 개발 단계

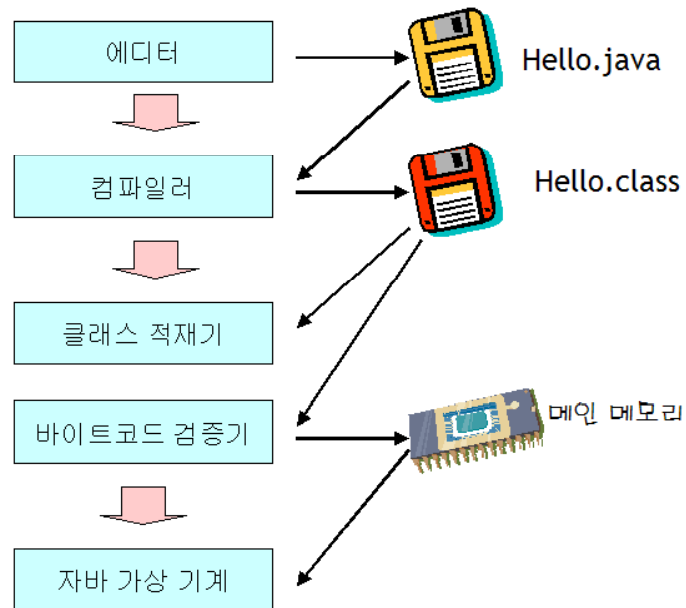


그림 2-1 자바 프로그램 개발 단계

© 2009 인피니티박스 All rights reserved



## 자바 프로그램 개발 단계

- 소스 파일의 생성
  - 에디터를 사용
  - 소스 파일은 .java 확장자
  - 메모장, 이클립스 ([www.eclipse.org](http://www.eclipse.org)), 넷빈([www.netbeans.org](http://www.netbeans.org)), JBuilder ([www.borland.com](http://www.borland.com)) 사용 가능
- 컴파일
  - 컴파일러로 컴파일
  - 컴파일러는 자바 소스 코드를 바이트 코드로 변환
  - 바이트 코드는 확장자가 .class로 끝나는 파일에 저장
- 클래스 적재
  - 바이트 코드 파일을 메모리로 적재
  - 네트워크를 통하여 적재될 수도 있다.

© 2009 인피니티박스 All rights reserved



## 자바 프로그램 개발 단계

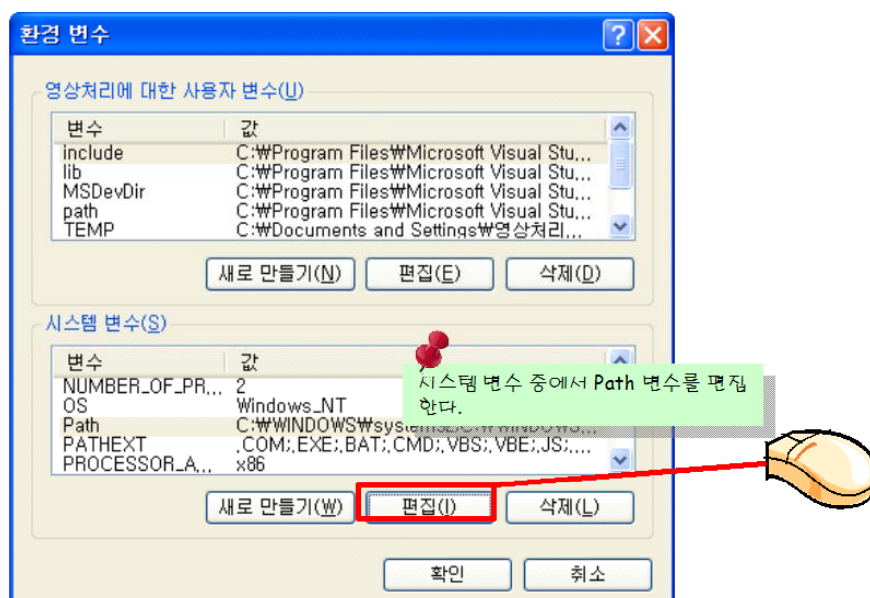
- 바이트 코드 검증
  - 바이트 코드들이 이상이 없으며 자바의 보안 규칙을 위배하지 않는지를 검사
- 실행
  - 자바 가상 기계가 바이트 코드를 실행
  - 가장 빈번하게 실행되는 부분(HotSpot)에서는 JIT 컴파일러가 바이트 코드를 실제 컴퓨터의 기계어로 직접 변환하여 빠르게 실행

© 2009 인피니티박스 All rights reserved



## JDK 사용하기

- 어디에서나 컴파일러를 실행할 수 있도록 경로(path) 설정하기



© 2009 인피니티박스 All rights reserved



## 소스 파일 작성

- 윈도우 보조 프로그램의 메모장 사용
- 파일의 확장자는 .java로 한다.

Hello.java

```
public class Hello {  
    public static void main(String args[]) {  
        System.out.println("안녕하세요. 자바 프로그래머 여러분!");  
    }  
}
```

© 2009 인피니티박스 All rights reserved



## 컴파일

```
C:\java\examples>javac Hello.java  
C:\java\examples>
```

```
C:\java\examples>dir  
...  
2009-06-05 오후 04:06 454 Hello.class  
2009-06-05 오후 02:53 144 Hello.java  
2개 파일 598 바이트
```

© 2009 인피니티박스 All rights reserved



## 실행

```
C:\java\examples>java Hello  
안녕하세요. 자바 프로그래머 여러분!  
C:\java\examples>
```

*java 명령어가 바로 “자바 가상 기계”를 구현한 것입니다.*



© 2009 인피니티박스 All rights reserved



## 통합 개발 환경

- 통합 개발 환경(integrated development environment): 소스 에디터, 컴파일러, 디버거가 결합
- 비주얼 컴포넌트를 마우스로 드래그하고 드롭하여 응용 프로그램을 생성할 수 있는 비주얼 개발 도구를 포함하는 것도 있다.
- (예) 이클립스(Eclipse), 넷빈(NetBeans)
- 통합 개발 환경도 내부적으로는 명령어 도구들을 이용

© 2009 인피니티박스 All rights reserved



## 중간 점검 문제

1. 컴파일하는 명령어 버전 이름은 javac이다.
2. Hello.java를 컴파일하면 Hello.class 파일이 생성된다.
3. 자바 가상 기계는 java 프로그램으로 구현한다.



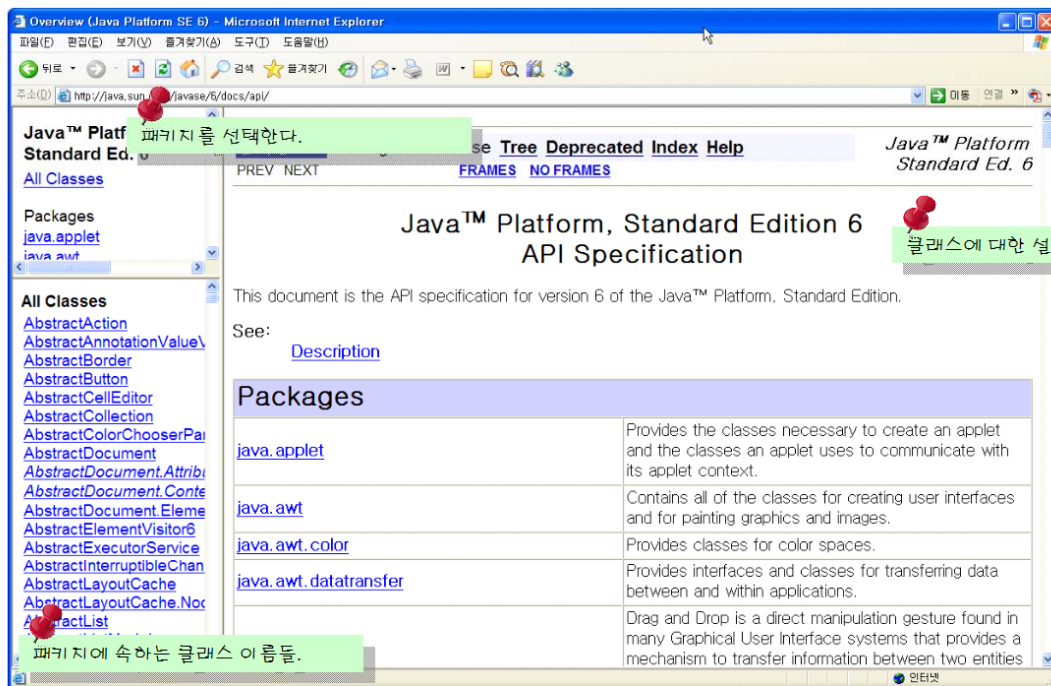
## 자바 문서 참조 하기

- 자바 API 문서
  - 자바 라이브러리에 대하여 설명하는 문서
- 자바 튜토리얼
  - 자바 언어에 대한 강좌





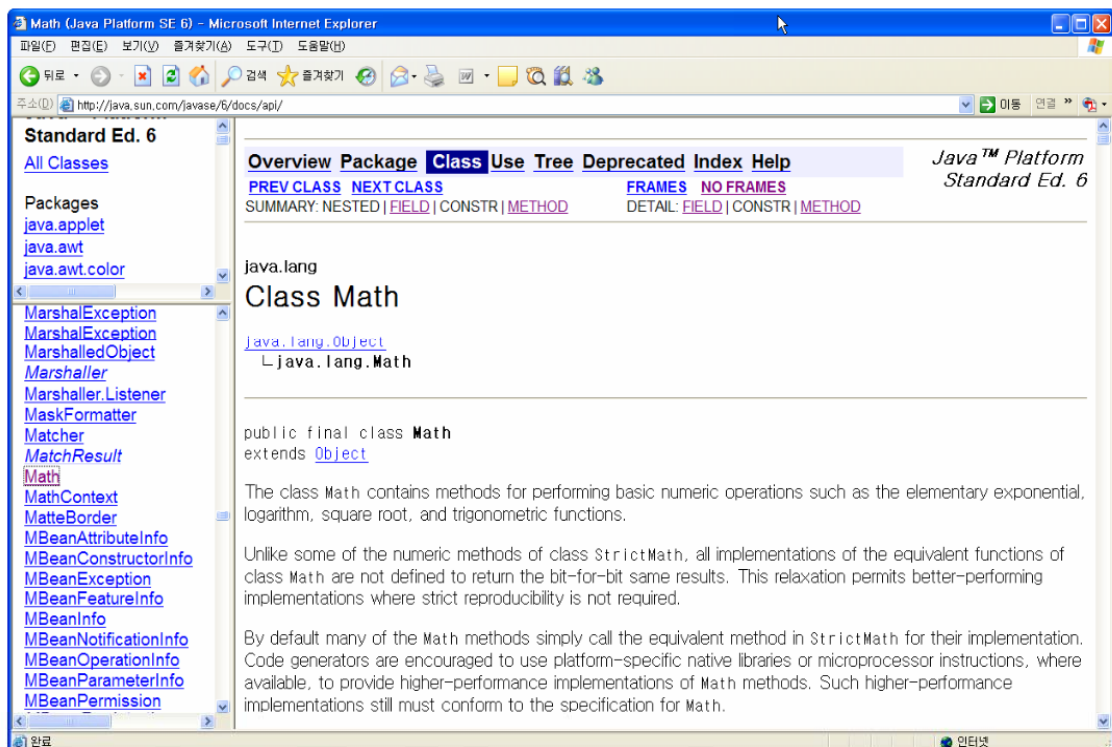
# 자바 API 문서



© 2009 인피니티북스 All rights reserved



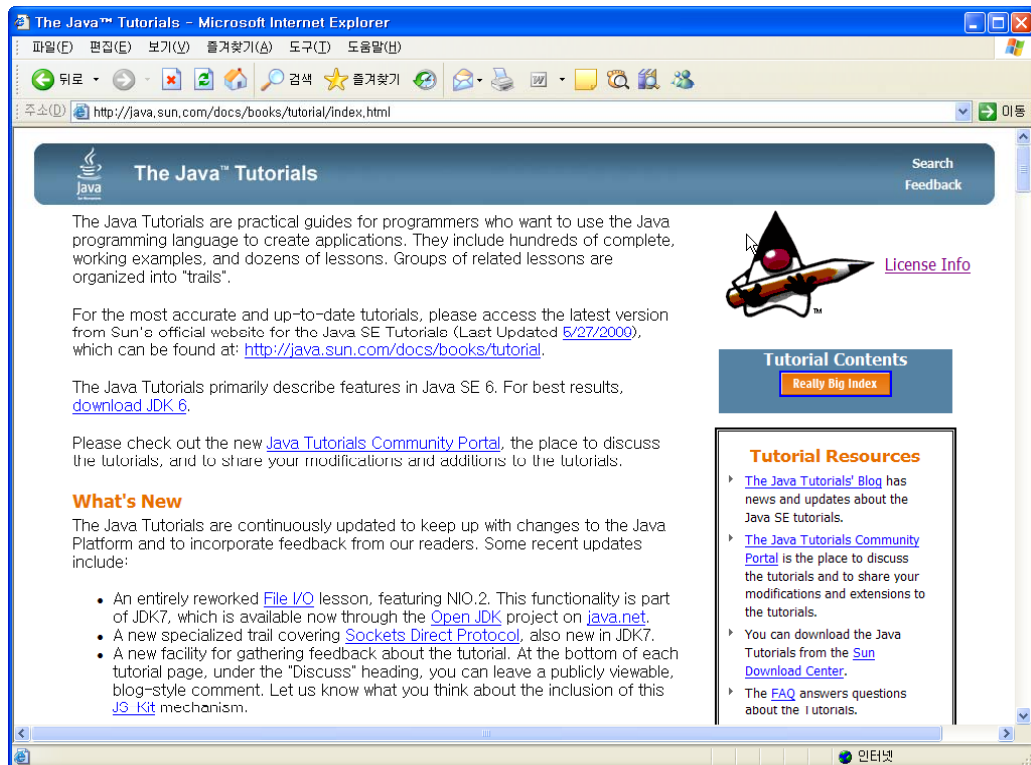
# 자바 API 문서



© 2009 인피니티북스 All rights reserved



# 자바 튜토리얼



© 2009 인피니티북스 All rights reserved



## 중간 점검 문제

1. 자바에서 제공하는 기본적인 클래스 라이브러리를 자바 API 라고 한다.
2. Sun사에서 제공하는 자바 초보자를 위한 문서를 자바 튜토리얼 라고 한다.
3. 자바 API 문서에서 System 클래스를 찾아서 어떤 내용들이 포함되어 있는지를 조사하여 보자.

© 2009 인피니티북스 All rights reserved



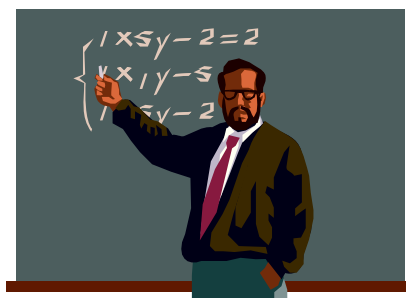
# System 클래스에 대한 설명

The screenshot shows the Java Platform SE 6 API documentation for the `System` class. The browser window is titled "System (Java Platform SE 6) - Microsoft Internet Explorer". The address bar shows the URL `http://java.sun.com/javase/6/docs/api/`. The left sidebar lists the "All Classes" and "Packages" for the Java Platform SE 6. The main content area shows the "Overview" tab selected, displaying the class hierarchy: `java.lang.Object` extends `java.lang.System`. The class is described as a "public final class System" that "extends Object". It states that the `System` class contains several useful class fields and methods, and it cannot be instantiated. It also mentions that the `System` class provides facilities for standard input, standard output, and error output streams, as well as access to externally defined properties and environment variables. The "Field Summary" section shows the `static PrintStream err` field, which is the "standard" error output stream. The "Since" section indicates that the class is available since JDK 1.0.

© 2009 인피니티북스 All rights reserved



## Q & A



© 2009 인피니티북스 All rights reserved