

# 개발 도구 설치와 환경설정

## \* 학습목표

- 에디트플러스를 설치하고 환경을 설정하는 방법을 알아본다.
- 이클립스를 설치하고 환경을 설정하는 방법을 알아본다.

## 01. 개발 도구 설치와 환경설정



## 개발 도구 설치와 환경설정

일반 문서나 편지 등을 작성하려면 한글과 같은 워드프로세서가 필요한 것처럼 프로그램을 만들려면 편집기 또는 개발 도구가 있어야 한다. 물론, 윈도우에서는 메모장, 리눅스에서는 vi 등과 같은 기본적인 편집기를 제공하지만 초보자에게는 불편한 점이 너무 많다. 그러므로 저자는 FTP 기능 등 프로그래밍에 적합한 국산 편집기인 에디트플러스(Editplus)와 개발에 필요한 여러 기능을 내장하면서도 비교적 심플한 도구인 이클립스(Eclipse)를 적극 추천한다.

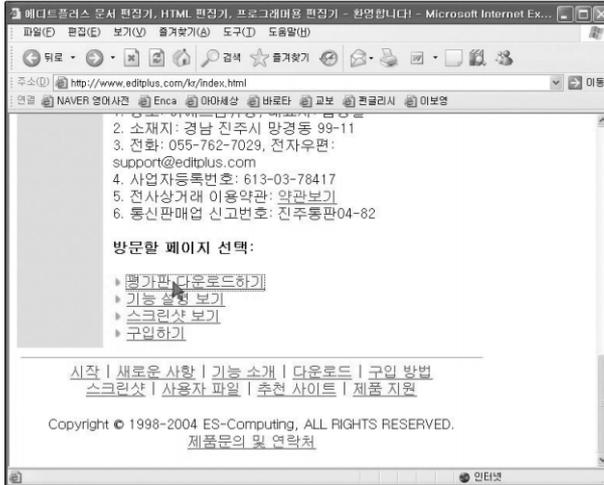
### ① 에디트플러스 설치 및 환경설정

에디트플러스는 웹 프로그래밍뿐만 아니라 여러 언어들을 지원하는 통합된 편집기로 가벼우면서 다양한 기능들을 쉽고 편리하게 제공하고 있다. 이제 'EditPlus 30일 평가판'을 내려받아 설치하고 환경을 설정하는 방법부터 알아보자.

## 실습하기

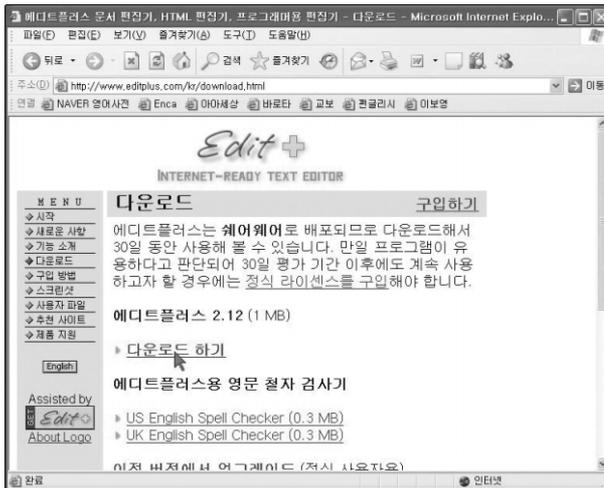
## 에디트플러스 설치

- 1 <http://www.editplus.com/kr/index.html>에 접속한 후 '평가판 다운로드하기'를 클릭한다.



[그림 A-1] 에디트플러스 사이트 초기 화면

- 2 웨어웨어에 대하여 간단한 설명을 읽은 후 '다운로드 하기'를 클릭한다. 그러면 내려받기 대화상자가 나오는데, 적당한 위치에 에디트플러스를 내려받은 후 실행시킨다.



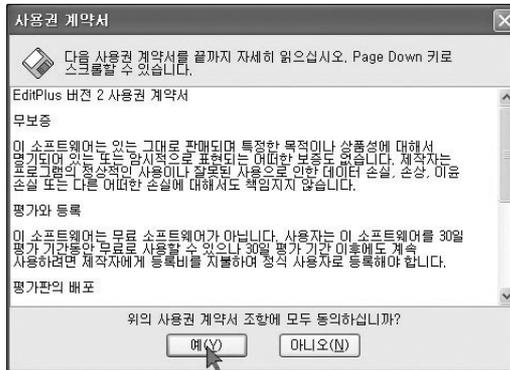
[그림 A-2] 에디트플러스 평가판 다운로드 페이지

- 3 아래와 같이 언어를 선택하는 대화상자에서 '한글(Korean)'을 선택한 후 <확인> 버튼을 선택하여 다음 화면으로 넘어간다.



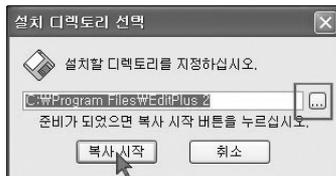
[그림 A-3] 에디트플러스 설치 초기화면

- 4 평가판과 사용권에 대한 설명이 나온다. <예> 버튼을 클릭하자.



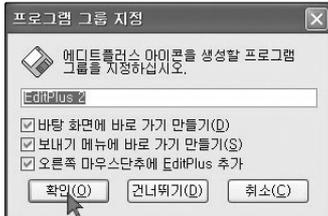
[그림 A-4] 에디트플러스 설치 사용권에 대한 설명

- 5 사용권에 동의하면 설치할 디렉토리를 묻는다. 기본으로 'C:\Program Files\EditPlus 2' 라는 위치로 지정되어 있다. 여기서는 기본 위치는 이 위치로 설치하지만 혹 다른 곳에 설치하고 싶다면 오른쪽의 <...> 버튼을 눌러서 변경할 수 있다.



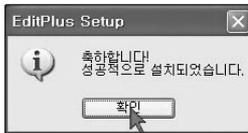
[그림 A-5] 에디트플러스 설치 경로 지정화면

- 6 에디트플러스 설치 시 그룹명을 지정해주고 단축 아이콘들의 생성여부를 선택하여 설치를 마무리한다. 물론 그룹명은 사용자가 임의로 정할 수 있다. 여기서는 기본값인 'EditPlus 2'를 그대로 두었다.



[그림 A-6] 에디트플러스 그룹명 지정

- 7 <확인> 버튼을 눌러 설치를 완료하자.



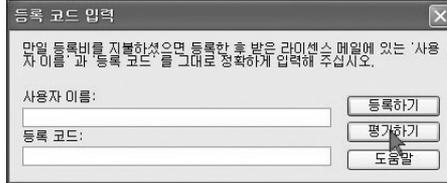
[그림 A-7] 에디트플러스 설치 종료화면

- 8 다음 그림처럼 [시작]--[프로그램]에서 [EditPlus 2]를 실행해 보자.



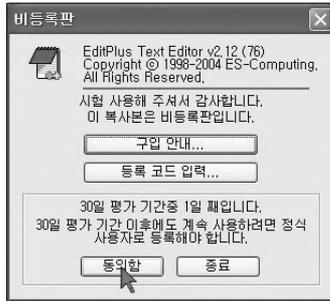
[그림 A-8] 에디트플러스 시작

- 9 마지막으로 에디트플러스를 시작하면 등록 코드를 입력하는 대화상자가 나오는데, 이 책에서는 평가판을 사용하므로 <평가하기>를 클릭한다.



[그림 A-9] 에디트플러스 사용권 검증 화면

- 10 에디트플러스를 실행할 때마다 30일 이내에서 앞으로 남은 평가 기간을 알려준다.



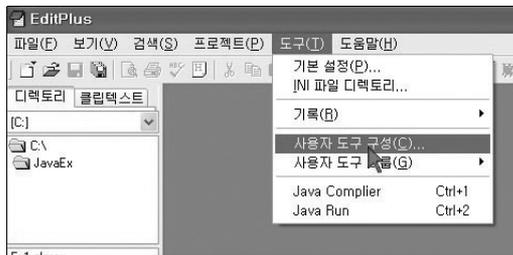
[그림 A-10] 에디트플러스 평가판 날짜 카운트



에디트플러스를 잘 설치했다면 이제 자바프로그래밍을 할 때 에디트플러스를 효율적으로 사용할 수 있도록 몇 가지 환경을 설정해주어야 한다. 잘 보고 따라해보자.

**실습하기**      **에디트플러스 환경설정**

- 1 [도구] → [사용자 도구 구성] 순으로 선택한다.

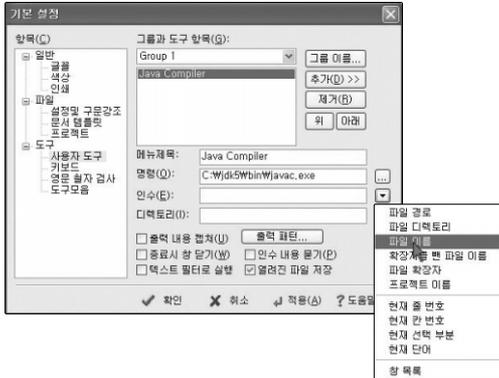


[그림 A-11] 에디트플러스 환경설정하기

- 2 자바 컴파일러를 연결하기 위하여 <그룹 이름> 버튼을 눌러서 'Java Compiler' 라고 입력하고 [프로그램] 으로 등록한다. 물론, '메뉴제목' 은 사용자 정의이므로 사용자 임의로 다른 이름을 기입해도 된다.

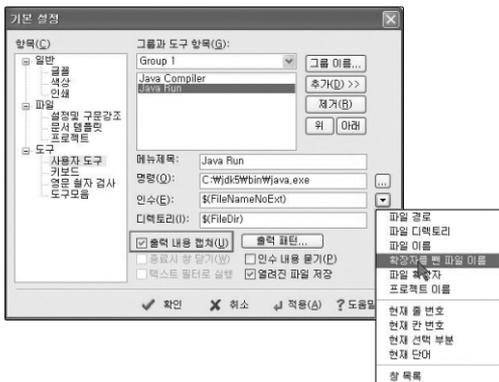


[그림 A-12] 자바 컴파일러 등록하기 1



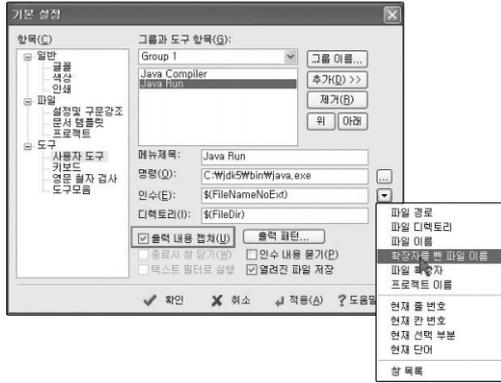
[그림 A-13] 자바 컴파일러 등록하기 2

- 3 '명령' 항목에는 앞서 JDK를 자동 경로로 지정한 'C:\jdk5\bin\javac.exe'로 하고 '인수' 항목에는 '\$(filename)', '디렉토리' 항목에는 '파일 디렉토리'로 설정한다. 그리고 에디트플러스 내에서 출력된 결과를 확인할 수 있도록 '출력 내용 캡처' 항목도 체크한다.



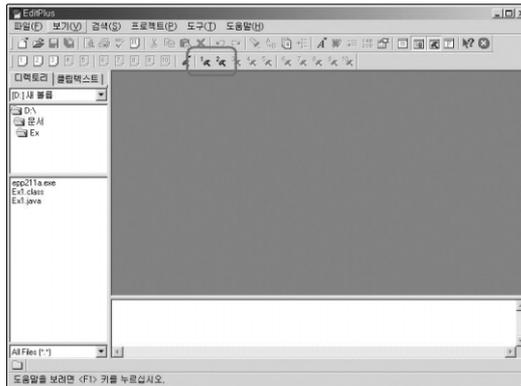
[그림 A-14] 자바 컴파일러 등록하기 3

- 4 다음은 바이트코드를 번역하는 Java Run 명령을 설정해야 하는데, 컴파일러를 설정할 때와 유사하다. 다음 그림을 보고 설정하자.



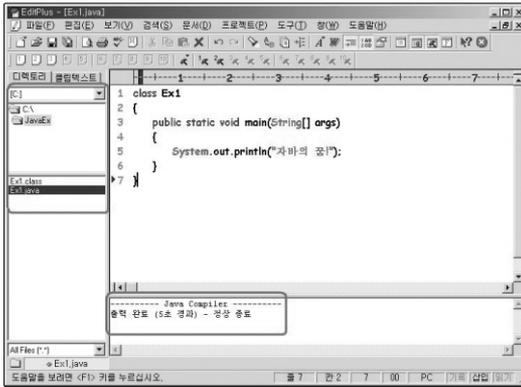
[그림 A-15] 에디트플러스에서 단축키를 등록한 화면

- 5 지금까지의 과정으로 환경설정은 끝났다. 마지막으로 바로 위의 [그림 A-15]에서 [보기] → [사용자 도구 모음]을 선택하면 다음 그림과 에디트플러스를 실행시켰을 때 망치그림이 도구 모음 줄이 나타나는데, 앞에 있는 것이 컴파일러(단축키는 <Ctrl> + <1>), 그 다음 것이 인터프리터(<Ctrl> + <2>).

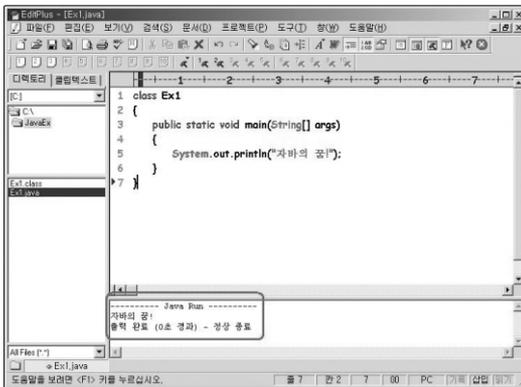


[그림 A-16] 자바 컴파일 화면

- 6 그림 이제 우리가 앞서 작업했던 C:\Java\Ex1.java를 불러 컴파일러와 인터프리터를 직접 실행해보자.



[그림 A-17] 자바 컴파일 화면



[그림 A-18] 자바 실행 화면

## 2 이클립스 설치

자바 개발자 입장에서 이클립스를 처음 접했을 때의 그 감동은 잊지 못할 정도다. 이클립스는 JSP의 모델 2 작업을 할 때나 또는 빈즈, 그리고 자바 응용프로그램을 작성하는 것까지 모든 면에서 간편하게 사용할 수 있는 아주 강력한 통합환경의 개발 도구(IDE)다.

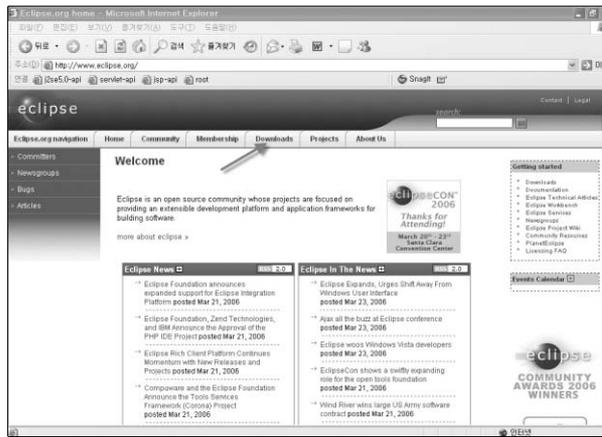
무엇보다 중요한 것은 간단한 플러그인(Plug-in) 작업으로 원하는 기술력을 지원받도록 도구 자체가 공개되어 있으며 비록 국내 제품이 아니지만 이런 간단한 플러그인 작업이 지원

됨으로써 모든 메뉴와 도움말까지 우리 한글로 지원받을 수 있다. 하지만 처음 자바를 배우는 사람들에게는 권하고 싶지 않다. 처음 자바를 배우는 사람일수록 이클립스를 사용하기 보다는 에디트플러스와 자바 API를 사용하여 자바 API를 좀더 익히고 이해하는 것이 더 바람직하다는 생각이 저자의 마음이자 바람이다.

실습하기

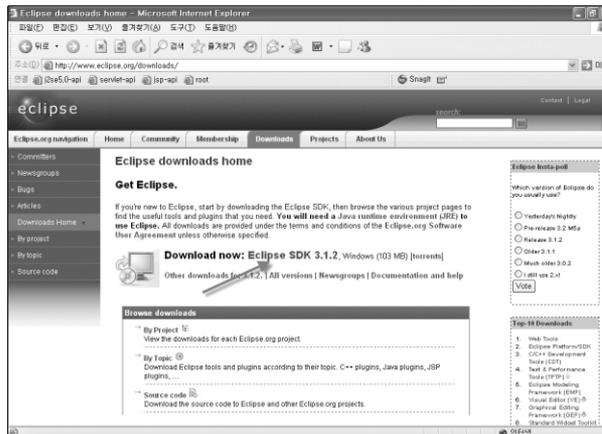
이클립스 설치하기

- 1 <http://www.eclipse.org>에 접속해서 상단에 있는 메뉴 중에서 [downloads]를 선택한다.



[그림 A-19] 이클립스 설치 (1)

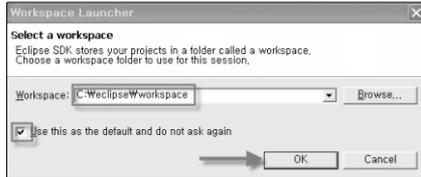
- 2 그러면 최근에 배포된 버전을 확인할 수 있다. 'Download now: Eclipse SDK 3.1.2'를 클릭한다.



[그림 A-20] 이클립스 설치 (2)

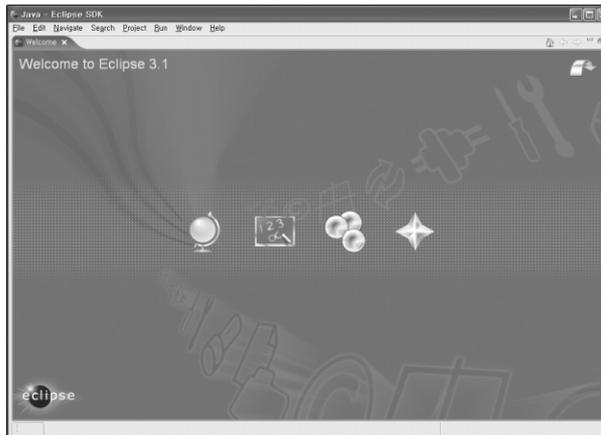


- 6 설치 끝나면 'workspace' 를 지정해야 한다. 'workspace' 란 이클립스에서 생성된 프로젝트를 관리하는 디렉토리다. 이클립스에서는 프로젝트를 생성해야 파일을 작성할 수 있다(필자는 C:\eclipse\workspace로 지정했다). 아래의 체크박스는 이클립스를 실행할 때 다음부터는 이 창을 띄우지 않겠다는 의미다. 여기도 체크한다.



[그림 A-24] 이클립스 설치 (6)

- 7 이클립스의 실행 화면은 다음과 같다.



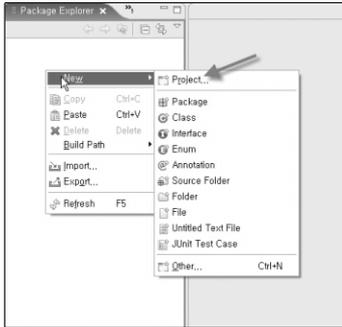
[그림 A-25] 이클립스 설치 (7)

### 3 이클립스 환경설정

이클립스의 설치가 끝났다면 이제 프로젝트를 생성해야 한다. 이클립스에서 프로젝트를 생성해야만 자바 파일을 생성할 수 있다.

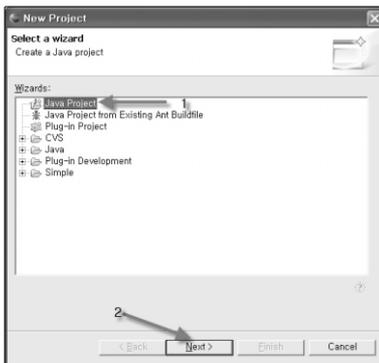
## 실습하기 이클립스에서 프로젝트 생성하기

- 1 이클립스를 실행한 후 'Package Explorer'에서 마우스 오른쪽 버튼을 클릭한 후 [New] → [Project]를 지정한다.



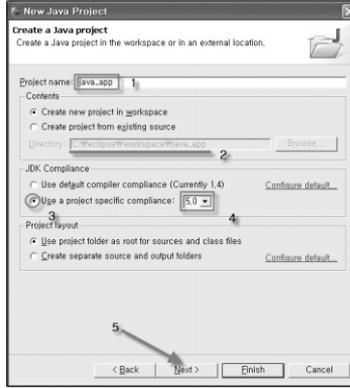
[그림 A-26] 이클립스 환경설정 (1)

- 2 'Java Project'를 체크하고 <Next> 버튼을 클릭한다.



[그림 A-27] 이클립스 환경설정 (2)

- 2 'Project name'에 'java\_app'라고 정하고, 'JDK Compliance'를 5.0으로 선택한 후에 <Next> 버튼을 클릭한다.



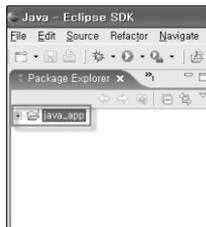
[그림 A-28] 이클립스 환경설정 (3)

- 4 그리고 <Finish> 버튼을 클릭한다.



[그림 A-29] 이클립스 환경설정 (4)

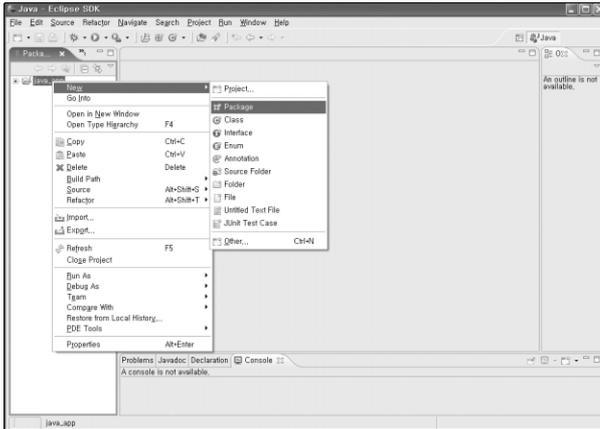
- 5 프로젝트가 올바르게 생성되었다면 다음과 같은 화면이 나올 것이다.



[그림 A-30] 이클립스 환경설정 (5)

## 실습하기 패키지 만들기

- 1 java\_app에서 마우스 오른쪽 버튼을 클릭한 후 [New] → [Package]를 지정한다.



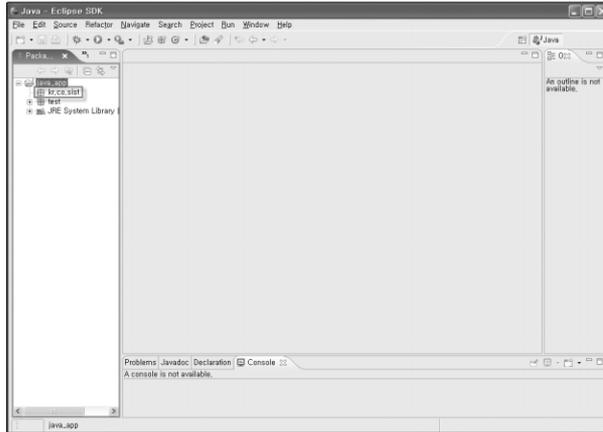
[그림 A-31] 패키지 만들기 (1)

- 2 Name에서 패키지명을 입력한 후에 <Finish> 버튼을 클릭한다.



[그림 A-32] 패키지 만들기 (2)

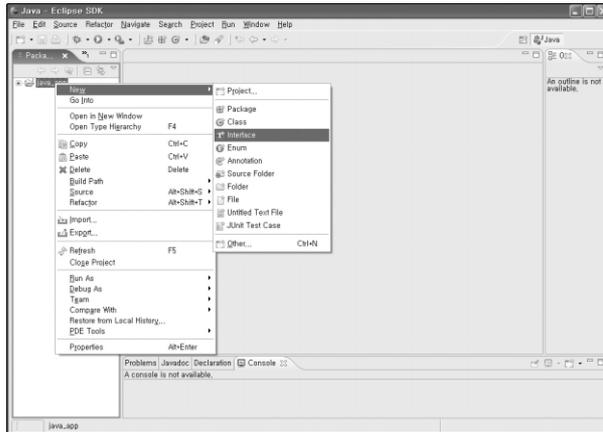
3 Package Explorer에 패키지가 만들어진다.



[그림 A-33] 패키지 만들기 (3)

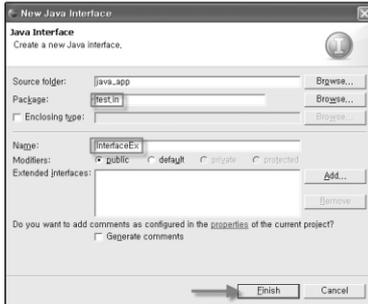
실습하기 인터페이스 만들기

1 java\_app에서 마우스 오른쪽 버튼을 클릭한 후 [New] → [Interface]를 지정한다.



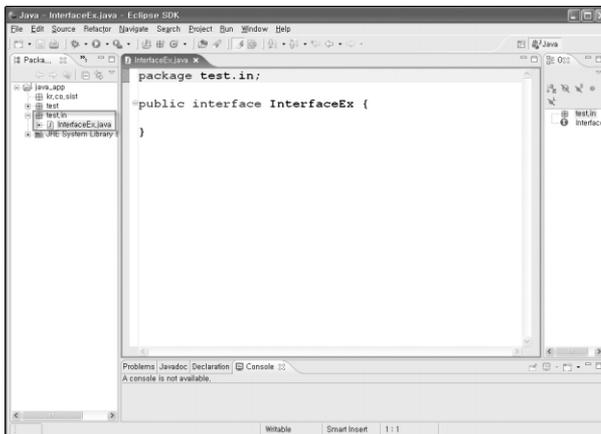
[그림 A-34] 인터페이스 만들기 (1)

- 2 Package명과 Interface명을 입력한 후에 <Finish> 버튼을 클릭한다.



[그림 A-35] 인터페이스 만들기 (2)

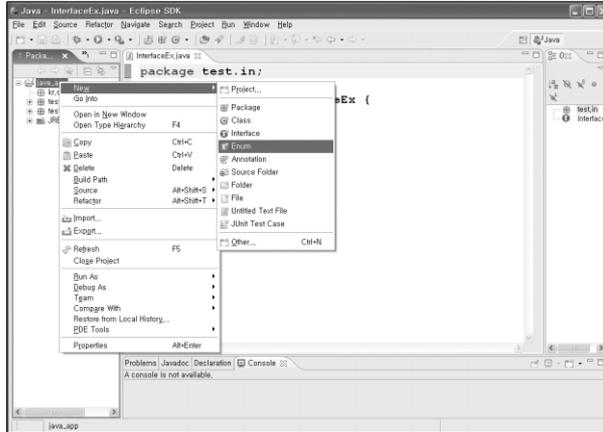
- 3 Package Explorer에 패키지 안에 인터페이스가 만들어진다.



[그림 A-36] 인터페이스 만들기 (3)

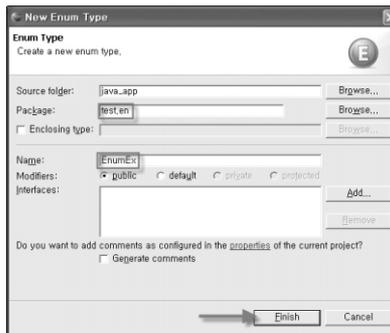
## 실습하기 Enum 만들기

- 1 java\_app에서 마우스 오른쪽 버튼을 클릭한 후 [New] → [Enum]을 지정한다.



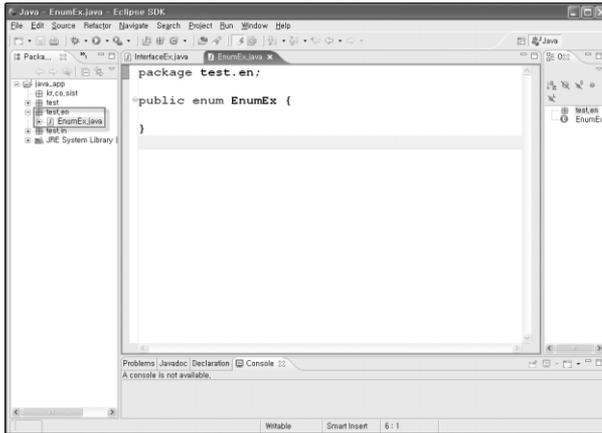
[그림 A-37] Enum 만들기 (1)

- 2 Package명과 Enum명을 입력한 후에 <Finish> 버튼을 클릭한다.



[그림 A-38] Enum 만들기 (2)

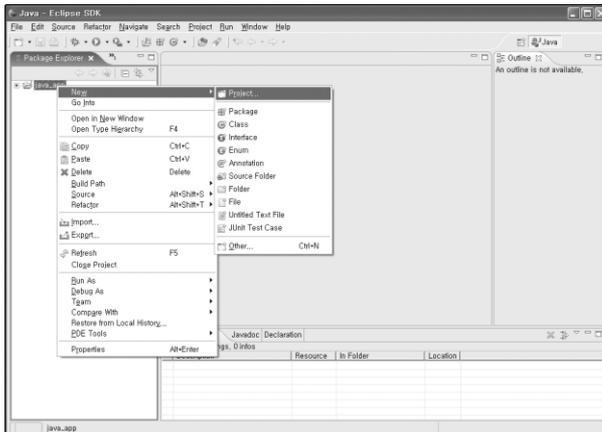
- 3 Package Explorer에 패키지 안에 Enum이 만들어진다.



[그림 A-39] Enum 만들기 (3)

## 실습하기 클래스 파일 생성 및 실행하기

- 1 프로젝트에서 마우스 오른쪽 버튼을 클릭한 후 [New] → [Class]를 지정한다.



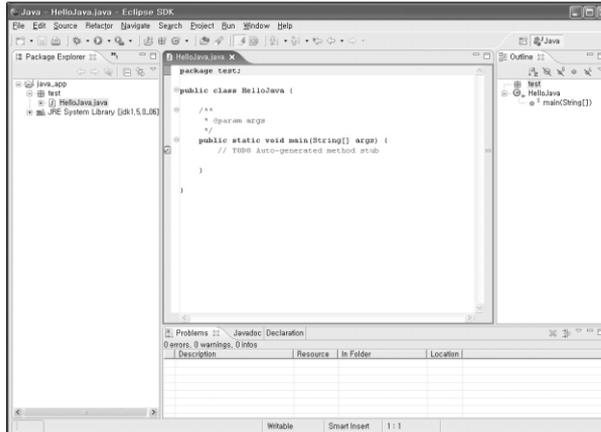
[그림 A-40] 클래스 파일 만들기 (1)

- 2 Package명(1번)과, Class명(2번)을 작성한 후에 3번 항목을 체크하면 main() 메서드가 자동으로 만들어진다. 여기서 2번을 작성할 때 파일명이 아니라 클래스명으로 작성해야 한다는 점에 유의하자.



[그림 A-41] 클래스 파일 만들기 (2)

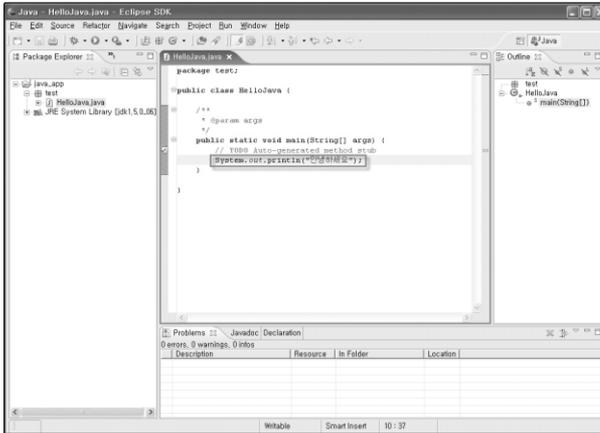
- 3 자바 파일을 생성한 화면은 다음과 같다.



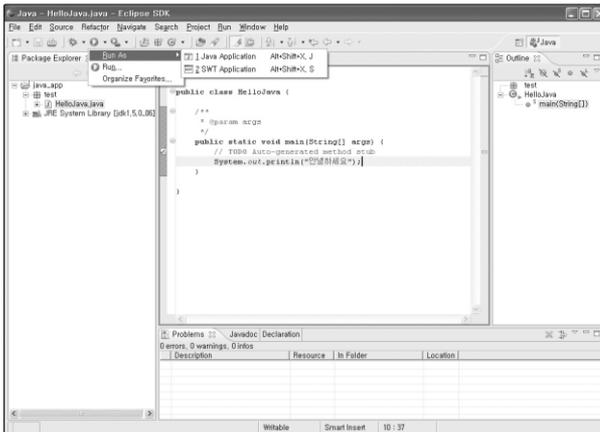
[그림 A-42] 클래스 파일 만들기 (3)

**4** 여기에 코드를 다음과 같이 작성해보자.

- `system.out.println("안녕하세요");`

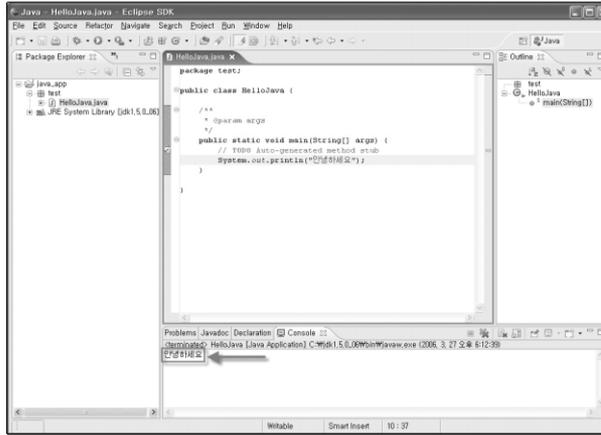


[그림 A-43] 클래스 파일 만들기 (4)

**5** 그리고 다음과 같이 프로그램을 실행한다.

[그림 A-44] 클래스 파일 만들기 (5)

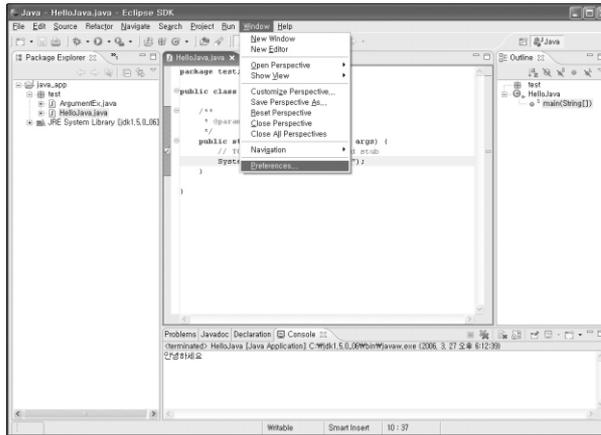
6 실행 결과 화면은 다음과 같다.



[그림 A-45] 클래스 파일 만들기 (6)

실습하기 자바 폰트와 크기 설정하기

1 [Window] → [Preferences] 메뉴를 선택한다.



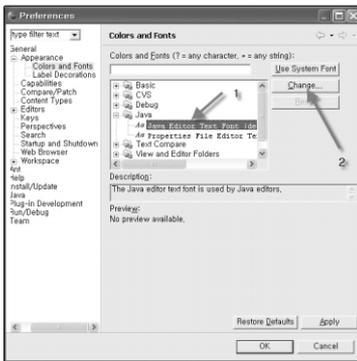
[그림 A-46] 폰트 및 크기 설정하기 (1)

- 2 'General' → 'Appearance' → 'Colors and Fonts'를 선택하면 오른쪽에 종류별로 설정할 수 있는 메뉴 리스트가 나온다.



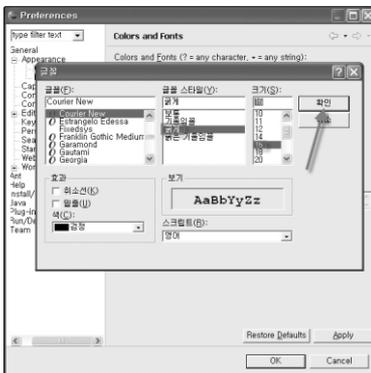
[그림 A-47] 폰트 및 크기 설정하기 (2)

- 3 오른쪽 메뉴 중 'Java'를 더블 클릭한 후에 1번을 선택하고 2번을 눌러서 폰트와 글씨 크기를 변경한다.



[그림 A-48] 폰트 및 크기 설정하기 (3)

- 4 아래와 같이 설정한 후에 <확인> 버튼을 클릭한다.



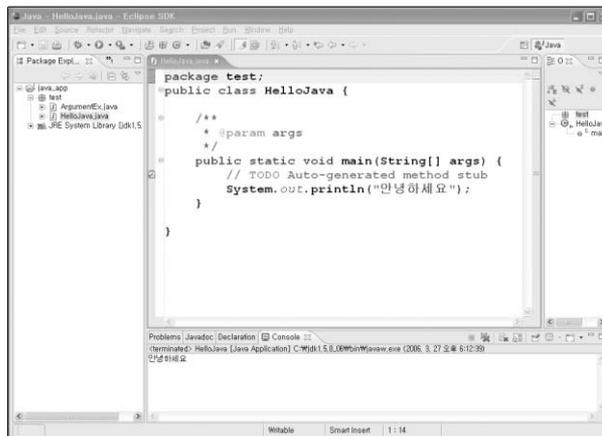
[그림 A-49] 폰트 및 크기 설정하기 (4)

5 <OK> 버튼을 클릭한다.



[그림 A-50] 폰트 및 크기 설정하기 (5)

6 그러면 글씨 크기와 굵기가 바뀌어 있음을 확인할 수 있다.

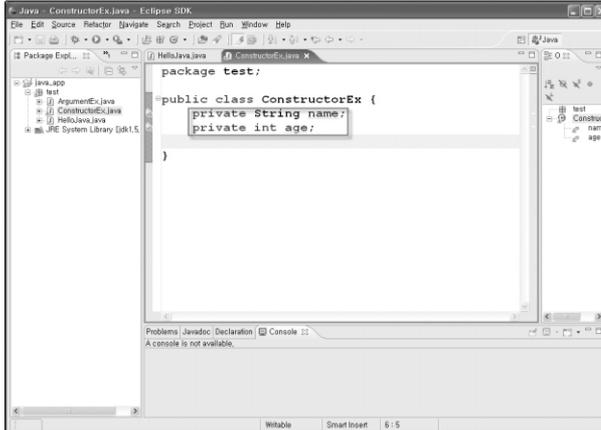


[그림 A-51] 폰트 및 크기 설정하기 (6)

## 실습하기

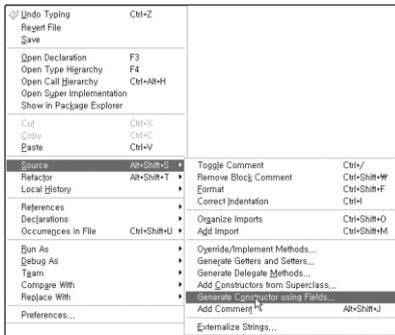
## 멤버변수를 이용한 생성자 만들기

- 1 아래와 같이 ConstructorEx.java 파일을 생성한다.



[그림 A-52] 멤버변수를 이용한 생성자 만들기 (1)

- 2 편집 창에서 마우스 오른쪽 버튼을 클릭하여 [Source] → [Generate Constructor using Fields]를 선택한다.



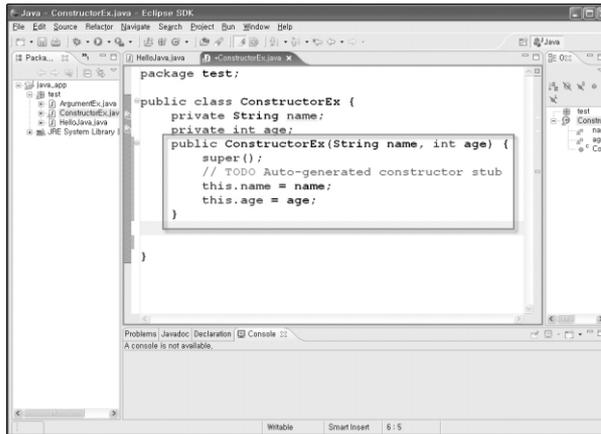
[그림 A-53] 멤버변수를 이용한 생성자 만들기 (2)

- 3 name, age 체크박스를 체크하고, <OK> 버튼을 누른다. 이 때 한 개만 체크하면 매개변수를 한 개를 갖는 생성자가 만들어진다.



[그림 A-54] 멤버변수를 이용한 생성자 만들기 (3)

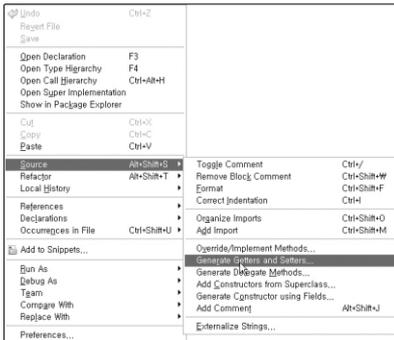
- 4 결과 화면은 다음과 같다.



[그림 A-55] 멤버변수를 이용한 생성자 만들기 (4)

## 실습하기 Setter, Getter 메서드 생성하기

- 1 ConstructorEx.java 파일의 편집 창에서 마우스 오른쪽 버튼을 클릭하여 [Source] → [Generate Setters and Getters]를 선택한다.



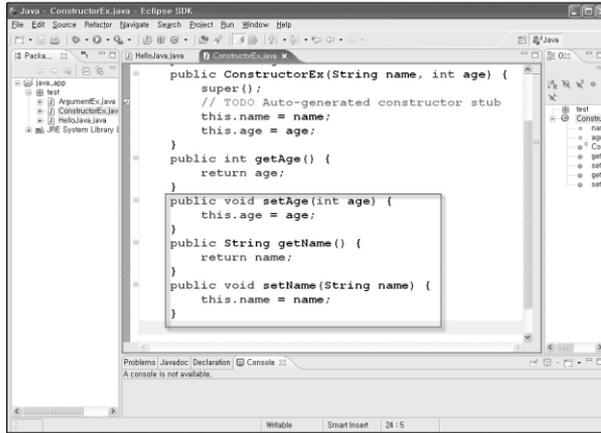
[그림 A-56] Setter, Getter 메서드 생성하기 (1)

- 2 name, age 체크박스를 체크하고, <OK> 버튼을 누른다. 이 때 한 개만 체크하면 한 개의 Setter와 Getter 메서드가 생성된다.



[그림 A-57] Setter, Getter 메서드 생성하기 (2)

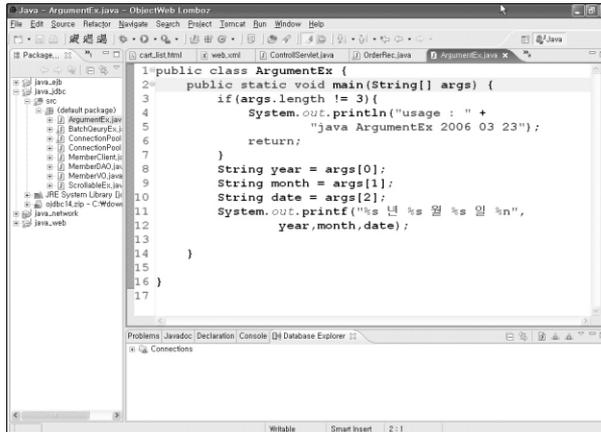
3 결과 화면은 다음과 같다.



[그림 A-58] Setter, Getter 메서드 생성하기 (3)

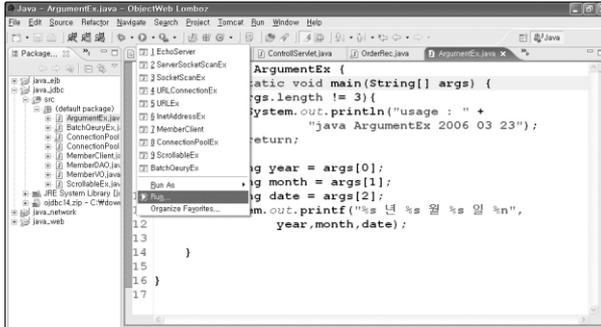
실습하기 실행시 파라미터 받기

1 파일을 아래와 같이 작성한다.



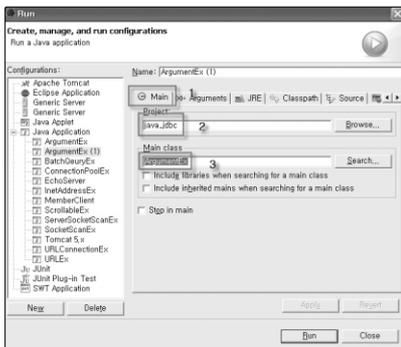
[그림 A-59] 실행시 파라미터 받기 (1)

- 2 <실행> 버튼을 클릭하여 drop 상태에서 [Run]을 클릭한다.



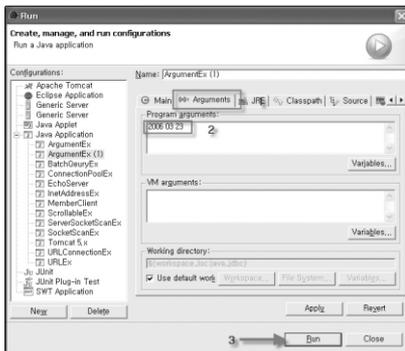
[그림 A-60] 실행시 파라미터 받기 (2)

- 3 창이 뜨면 메인 탭(1번)에서 현재 사용하고 있는 프로젝트(2번)를 찾고, 실행하려는 클래스를 찾거나 직접 기입한다.



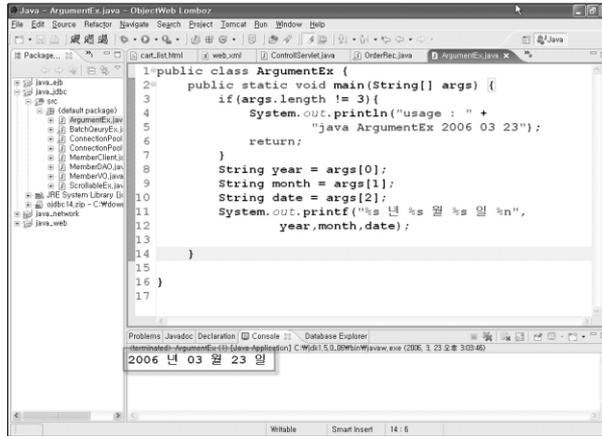
[그림 A-61] 실행시 파라미터 받기 (3)

- 4 Arguments → Program arguments → 2006 03 23라고 작성한 후에 <Run> 버튼(3번)을 클릭한다.



[그림 A-62] 실행시 파라미터 받기 (4)

5 성공 화면은 다음과 같다.



[그림 A-63] 실행시 파라미터 받기 (5)