

Java



애플릿에 대하여 애플릿의 작성 방법



애플릿 프로그래밍

01. 애플릿에 대하여

애플릿이란?

• 애플릿(applet)

- HTML 문서와 함께 웹 브라우저로 전송되어 HTML 문서의 일부로서 실행되는 프로그램
- 초창기에는 애니메이션 도구로 소개됨
- 현재는 웹 브라우저에서 복잡한 로직의 프로그램을 작성할 필요가 있을 때 사용됨
- 광범위한 지역에 지사가 분포된 대규모 기업의 인트라넷용으로도 사용됨



Applet

- ▶ Applet은 우리가 앞에서 보았던 AWT에서 Container의 역할을 하는 것으로 Web Browser에 다운되어 실행될 수 있는 Program을 의미한다.
- ▶ Applet이 나오면서 동적으로 물체가 움직이고, Game을 하는 등 특별히 Software를 Down받아 설치하지 않고 그냥 Web Browser상에서 Program이 실행될 수 있도록 해준 것.
- ▶ Applet은 Browser에 내장되어 있는 Java VM에 의해 동작이 된다.
- ▶ User는 별도의 Java VM을 설치하지 않아도 Applet이 실행될 수 있는데, Browser의 version이나 개발사에 따라 지원하는 version이 다를 수 있기 때문에 작성된 Applet Program이 원래 의도대로 동작하지 않을 수도 있다.
- ▶ 그래서 이러한 문제를 해결하기 위해서 Sun에서 해결책으로 나온 것이 바로 Java Plug-In이다.

Applet

● Applet의 실행

- ▶ Applet은 Application과 달리 `main()` 함수가 존재하지 않기 때문에 독자적으로는 실행이 불가능하다,
- ▶ Web Browser나 JDK가 설치되어 있는 `bin` folder에서 제공하는 `appletviewer`와 같은 Program내에서만 실행이 가능하다,
- ▶ Web Browser는 기본적으로 읽을 수 있는 문서형태가 HTML형태이므로, Applet code를 직접 Browser에 삽입하여 작성하는 것은 아니다,
- ▶ 즉, Applet code는 우리가 지금까지 작성했던 방식으로 `.java`형태로 작성해서 `compile` 시킨 후 HTML code에서 `compile`된 `CLASS` file을 불러 실행될 수 있도록 하는 방법을 사용한다,
- ▶ Browser에서 `CLASS` file을 불러오는 code는 `<applet>` tag를 사용,

Applet

● Applet관련 tag 및 method

➡ Applet관련 tag

- Web Browser에서 Applet code를 실행시키기 위해서 HTML에서 제공하는 tag가 바로 <applet>이다.
- 이 tag를 이용해서 HTML에서 Applet code를 지정해서 쉽게 Applet code를 사용할 수 있도록 해주는 기능을 한다.

❖ <applet>tag 형식

```
<applet code=" Applet class이름"  
        width=" Web Browser상의 Applet의 폭"  
        height=" Web Browser상의 Applet의 높이"  
        codebase=" Applet code가 있는 위치"  
>  
  <param name=" parameter name"  
        value=" parameter가 가지는 값"  
  >  
</applet>
```

Applet

● <applet>tag의 속성

- `code` : HTML에서 Applet으로 작성하여 compile된 CLASS file을 불러와서 실행할 수 있도록 지정하는 속성으로 CLASS file의 확장자는 생략가능하며 `package` 자체를 지정하여도 된다.
- `width, height` : Web Browser에서 Applet이 보여줄 수 있는 영역을 지정하는 속성이다. Web Browser의 크기변경에 따라 가변적으로 변경될 수 있다.
- `codebase` : Applet file이 있는 위치를 지정하는 속성으로 일반적으로 HTMLfile과 같은 위치에 있다면 생략이 가능하고 다른 위치에 있으면 반드시 경로를 설정해줘야 된다.
- `<param>tag` : 이 tag는 Applet이 실행될 때 HTML에서 Applet으로 초기값을 넘겨줄 수 있도록 해주는 기능으로 `name`은 Applet에서 값을 얻어갈 수 있도록 해주는 parameter 이름이고 `value`는 `name`에 의해서 실제적으로 넘어오는 parameter값을 지정하는 속성이다.

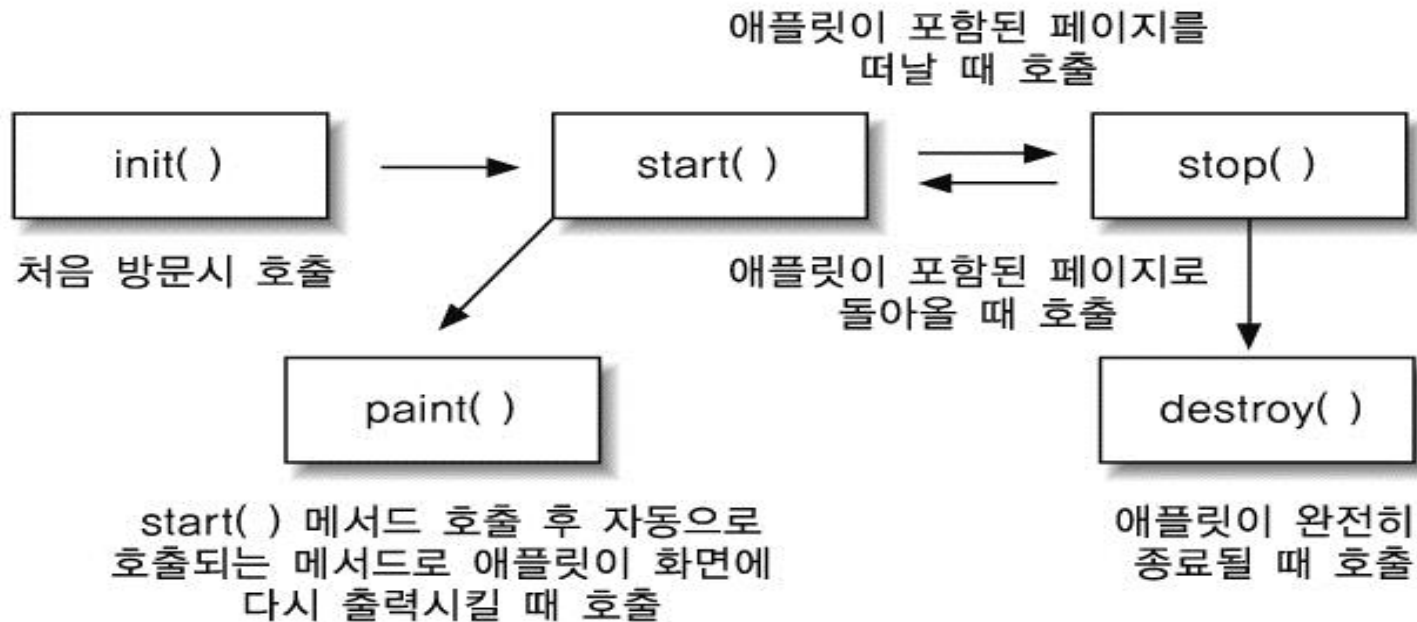
Applet

※ Applet 클래스의 주요 메서드

반환형	메서드	설명
void	init()	웹브라우저에서 처음으로 애플릿이 로드될 때 실행되는 메서드로 애플릿을 초기화한다.
	start()	init() 메서드 다음에 호출되거나 다른 페이지를 보고 있다가 다시 애플릿이 포함된 페이지로 되돌아 올 때 호출되는 메서드로 애플릿을 시작시킨다.
	stop()	애플릿이 중지되거나 애플릿이 포함되지 않은 페이지로 이동될 때 호출되는 메서드로 애플릿을 중지시킨다.
	destroy()	애플릿이 종료될 때 호출되는 메서드로, 주로 웹브라우저가 완전히 종료될 때 호출되는 메서드다.
	paint(Graphics g)	애플릿을 화면에 다시 출력시켜야 하는 경우에 호출되는 메서드다.
URL	getCodeBase()	애플릿 코드가 있는 URL을 얻어온다.
	getDocumentBase()	HTML 문서가 있는 URL을 얻어온다.
String	getParameter(String name)	HTML 문서에서 <param> 태그에 있는 name에 해당하는 값을 얻어온다.
Image	getImage(URL url)	url에 지정된 위치의 이미지를 얻어온다.
	getImage(URL url, String imgName)	url에 지정된 위치에 imgName을 가지는 이미지를 얻어온다.

Applet

● Applet Life Cycle



Applet

● Applet Security

- ▶ Java Applet은 Server로부터 Download되어 Web Browser를 통하여 실행되는 Program으로 Download된 Applet Program이 System에 접근하여 System의 정보를 읽거나 file을 가지고 갈 수도 있다면 문제가 아주 심각할 것이다.
- ▶ Applet은 이러한 행위를 하지 못하게 보안이 되어있고 이런 보안 문제가 발생하면 예외를 던지고 실행을 멈추게 한다.

❖ Applet Security Policy

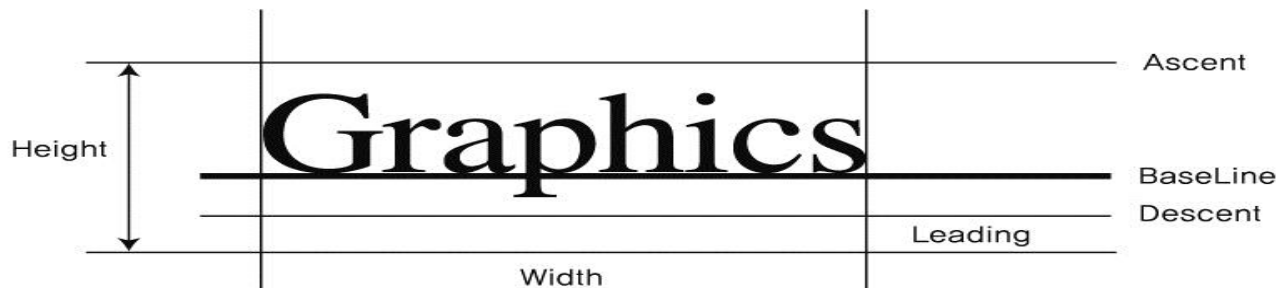
1. Applet이 Download된 System의 file을 읽거나 쓸 수 없다.
2. Applet이 Download된 System의 Program을 실행하거나 공유 library를 호출 할 수 없다.
3. Applet이 Download된 System의 정보를 알아낼 수 없다.
4. Binary code를 불러내 사용할 수 없다.
5. Applet을 제공한 Server 이외의 다른 System과 통신을 할 수 없다.

Applet

● FontMetrics class

- ➡ User가 정확한 위치에 문자를 출력시킬 수 있도록 FontMetrics class를 제공.

❖ FontMetrics의 문자열에 대한 정보

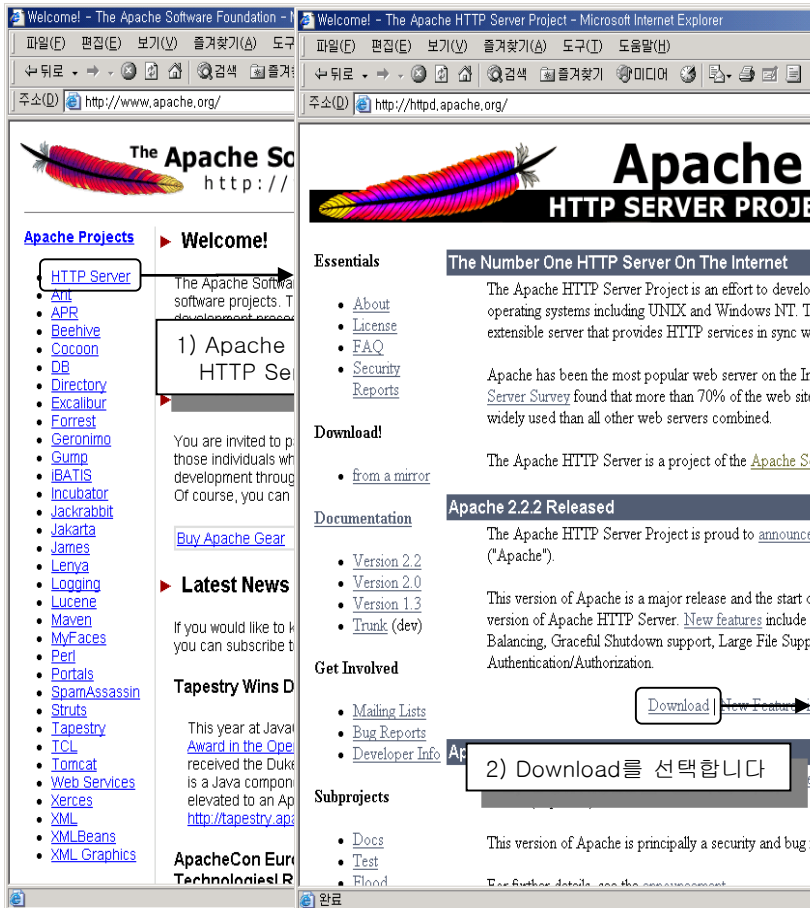


- Width : 문자열의 폭
- BaseLine : Font의 기준선(이 기준선에 의해 글자가 출력된다.)
- Descent : 기준선보다 아래에 있는 공간
- Ascent : 기준선보다 위에 있는 공간
- Leading : 현문자의 decent과 다음 문자라인의 Ascent 사이의 공간으로 아래글자와의 경계에 사용되는 빈 공간
- Height : $Ascent + Descent + Leading$

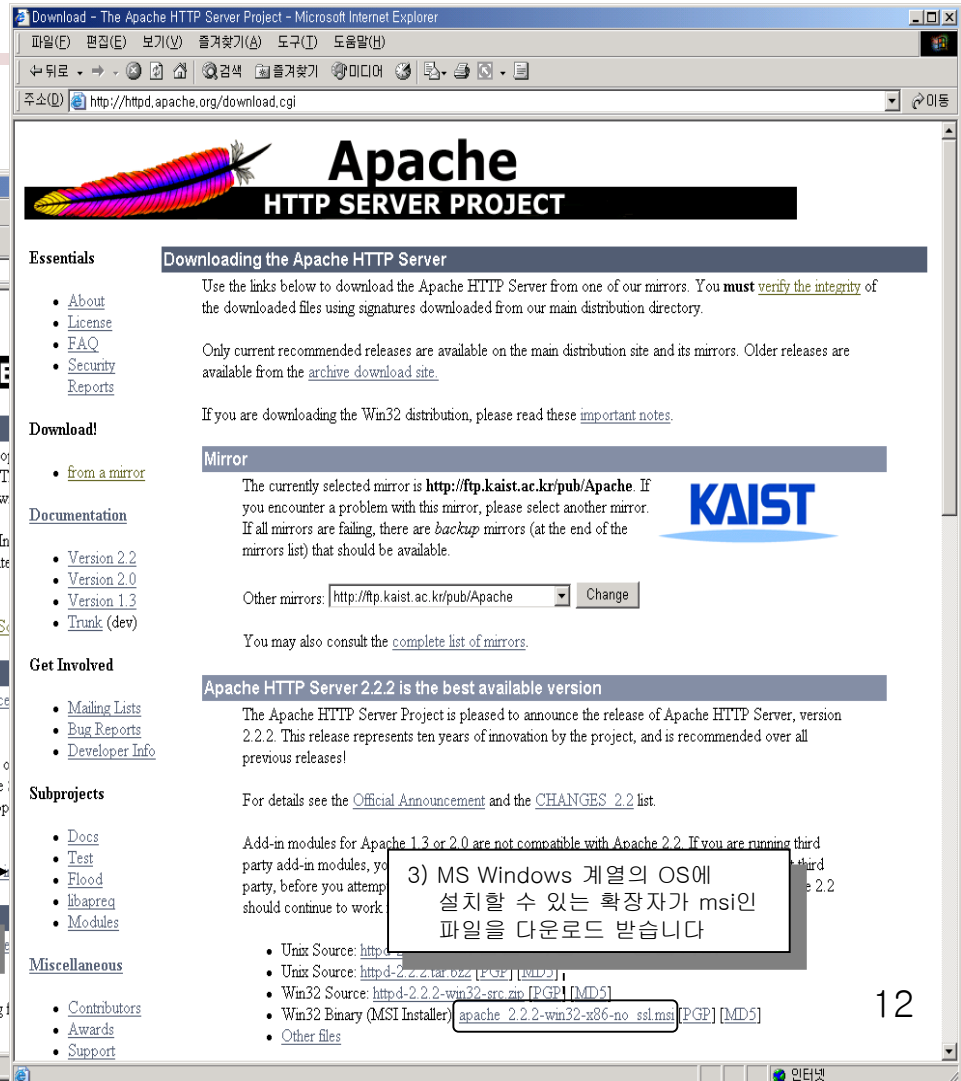
애플릿 프로그래밍

01. 애플릿에 대하여 아파치 웹 서버의 설치

•• 아파치 웹 서버를 다운로드 받을



The screenshot shows the Apache Software Foundation website. On the left, there is a sidebar with a list of 'Apache Projects' including Ant, APR, Beehive, Cocoon, DB, Directory, Excelsior, Forrest, Geronimo, Gump, iBATIS, Incubator, Jackrabbit, Jakarta, James, Lenya, Logging, Lucene, Maven, MyFaces, Perl, Portals, SpamAssassin, Struts, Tapestry, TCL, Tomcat, Web Services, Xerces, XML, XMLBeans, and XMLGraphics. The main content area has a 'Welcome!' message and a 'Latest News' section. A red box highlights the '1) Apache HTTP Server' link in the sidebar. Another red box highlights the 'Download' link in the 'Get Involved' section.



The screenshot shows the Apache HTTP Server Project download page. The page has a header with the Apache logo and the text 'Apache HTTP SERVER PROJECT'. Below the header, there are several sections: 'Essentials' with links to About, License, FAQ, Security, and Reports; 'Download!' with a link to 'from a mirror'; 'Documentation' with links to Version 2.2, Version 2.0, Version 1.3, and Trunk (dev); 'Get Involved' with links to Mailing Lists, Bug Reports, and Developer Info; 'Subprojects' with links to Docs, Test, Flood, libapreq, and Modules; and 'Miscellaneous' with links to Contributors, Awards, and Support. A red box highlights the '2) Download를 선택합니다' (Select Download) text. Another red box highlights the '3) MS Windows 계열의 OS에 설치할 수 있는 확장자가 msi인 파일을 다운로드 받습니다' (Download files with msi extension that can be installed on MS Windows system) text. The page also features a 'Mirror' section with a dropdown menu for selecting a mirror and a 'Change' button.

애플릿 프로그래밍

01. 애플릿에 대하여

아파치 웹 서버의 설치

- 1) 설치 프로그램의 첫번째 창입니다.

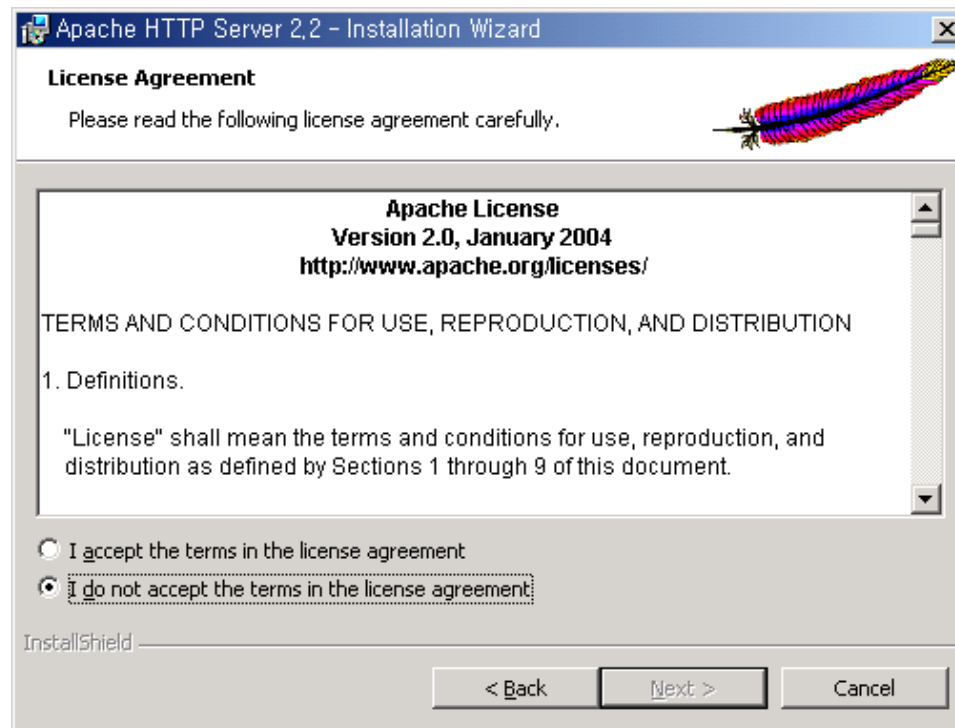


애플릿 프로그래밍

01. 애플릿에 대하여

아파치 웹 서버의 설치

- 2) License Agreement를 확인하는 창입니다.

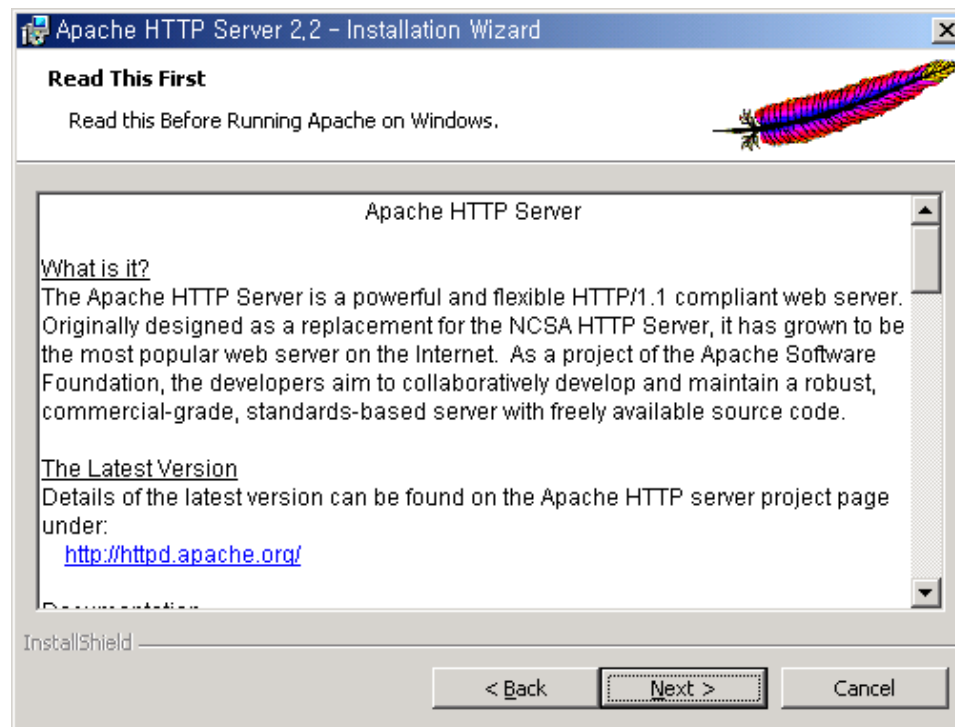


애플릿 프로그래밍

01. 애플릿에 대하여

아파치 웹 서버의 설치

- 3) 아파치 웹 서버에 대한 설명을 보여주는 창입니다.

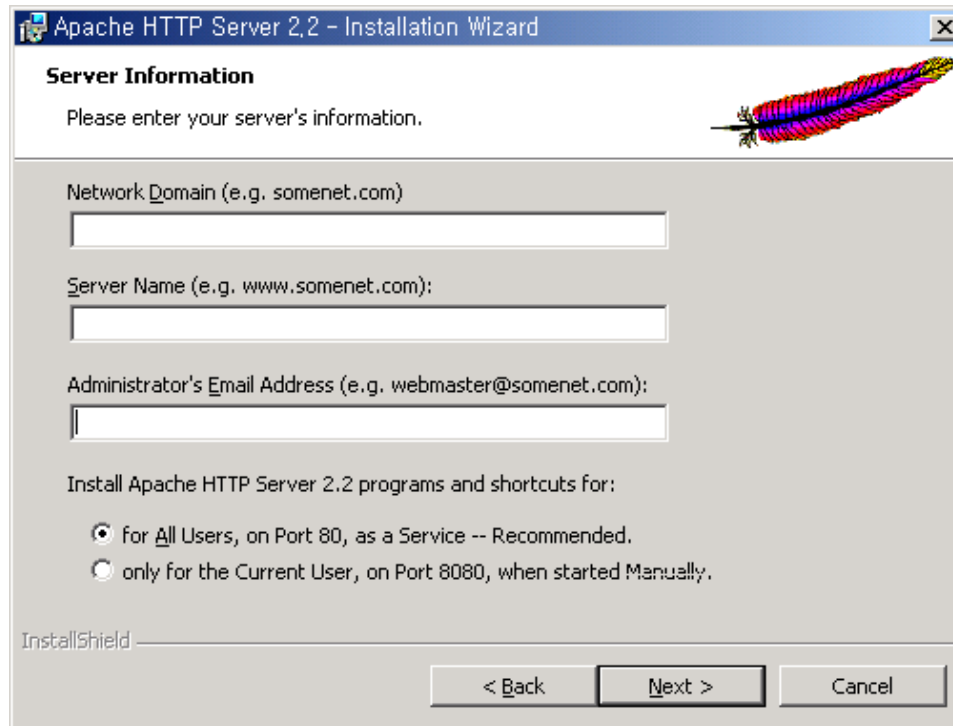


애플릿 프로그래밍

01. 애플릿에 대하여

아파치 웹 서버의 설치

- 4) 아파치 웹 서버에 대한 정보를 입력하는 창입니다.



Apache HTTP Server 2.2 - Installation Wizard

Server Information

Please enter your server's information.

Network Domain (e.g. somenet.com)

Server Name (e.g. www.somenet.com):

Administrator's Email Address (e.g. webmaster@somenet.com):

Install Apache HTTP Server 2.2 programs and shortcuts for:

☒ for All Users, on Port 80, as a Service -- Recommended.

☐ only for the Current User, on Port 8080, when started Manually.

InstallShield

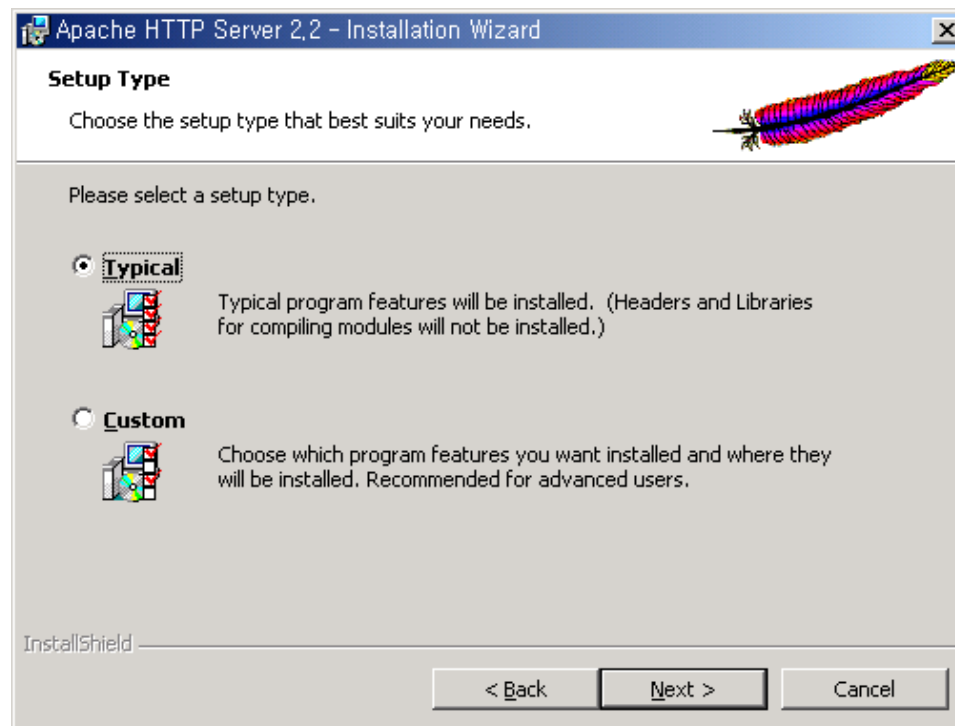
< Back Next > Cancel

애플릿 프로그래밍

01. 애플릿에 대하여

아파치 웹 서버의 설치

- 5) 설치 타입을 선택하는 창입니다.

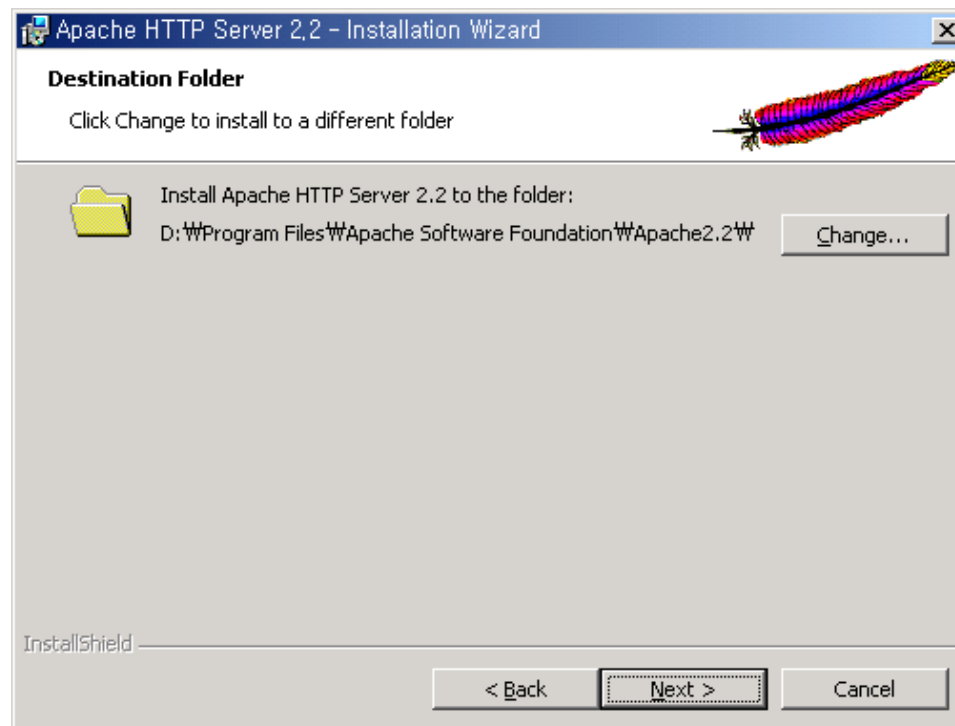


애플릿 프로그래밍

01. 애플릿에 대하여

아파치 웹 서버의 설치

- 6) 설치 디렉토리의 경로명을 확인하는 창입니다.

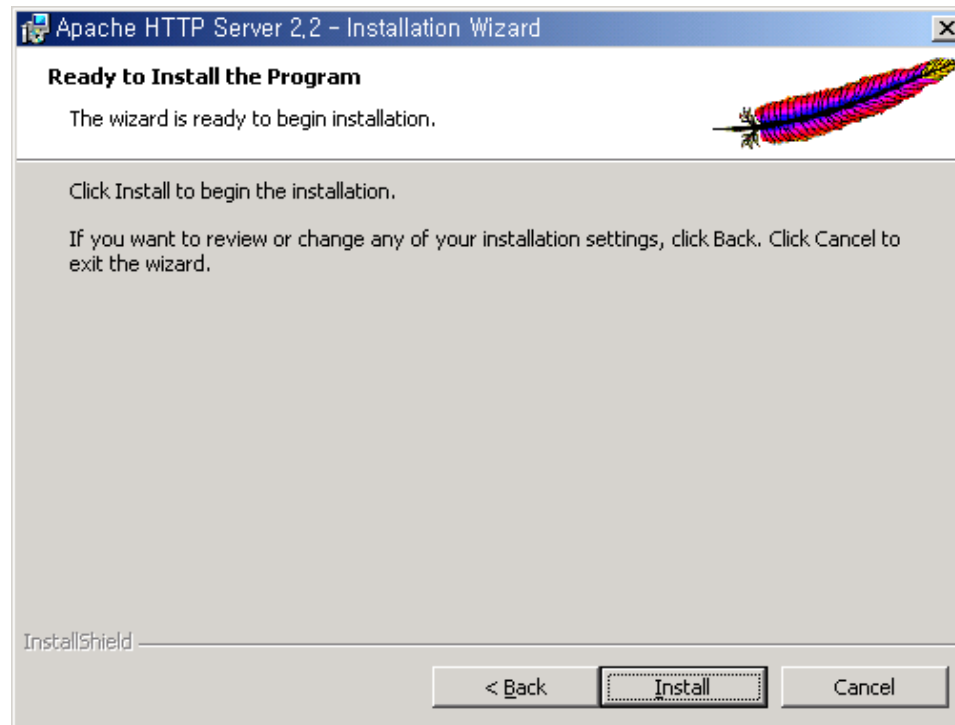


애플릿 프로그래밍

01. 애플릿에 대하여

아파치 웹 서버의 설치

- 7) 설치의 시작을 확인 받는 창입니다.

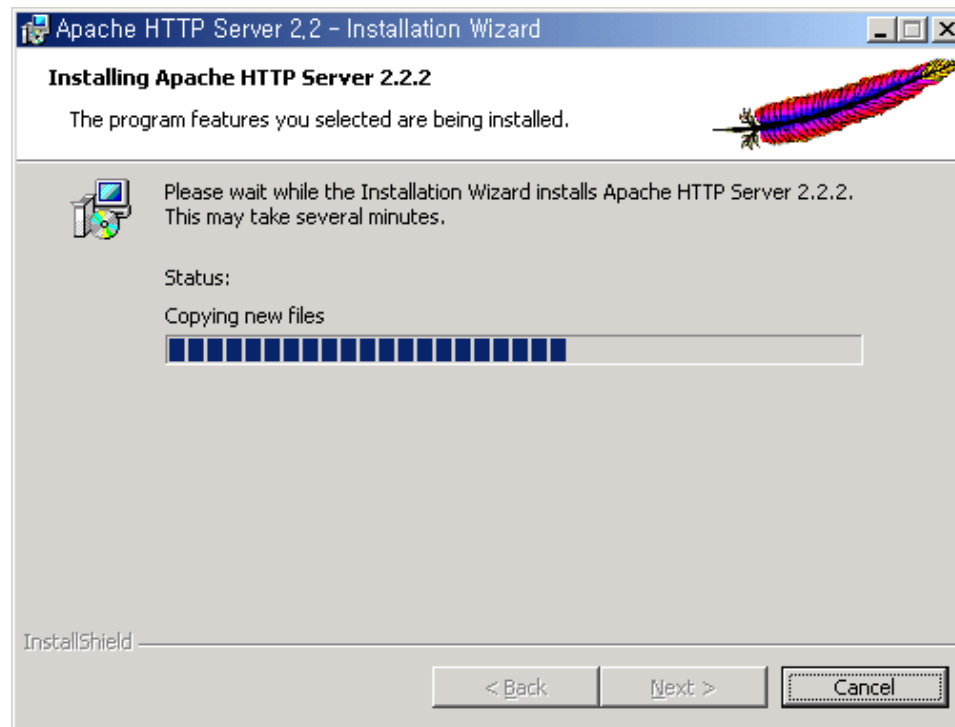


애플릿 프로그래밍

01. 애플릿에 대하여

아파치 웹 서버의 설치

- 8) 설치의 진행 과정을 보여주는 창입니다.

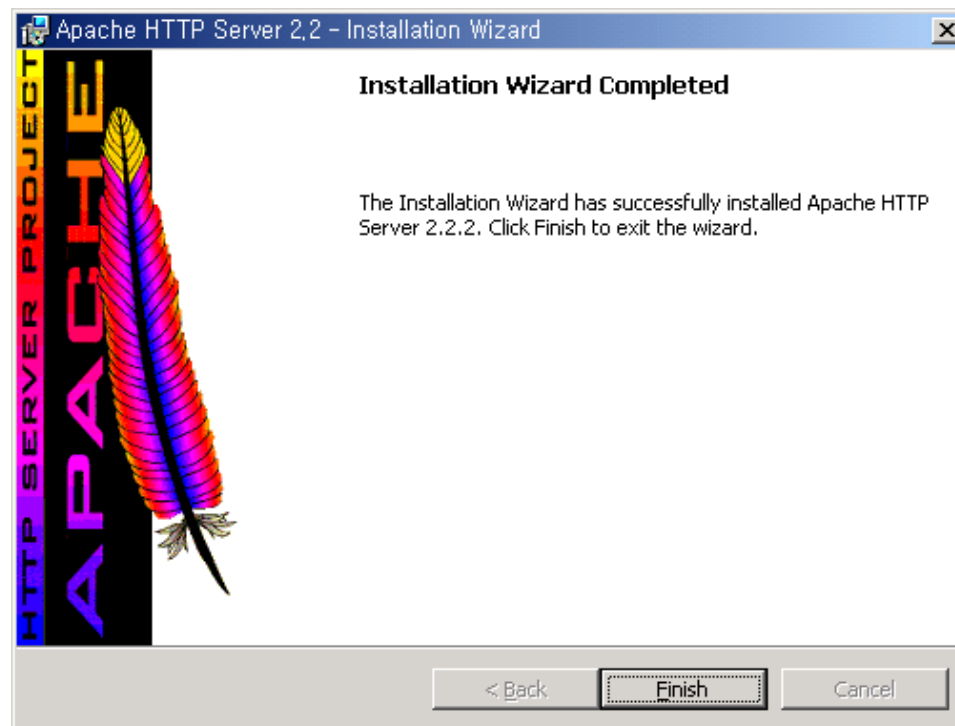


애플릿 프로그래밍

01. 애플릿에 대하여

아파치 웹 서버의 설치

- 9) 설치 프로그램의 마지막 창입니다.

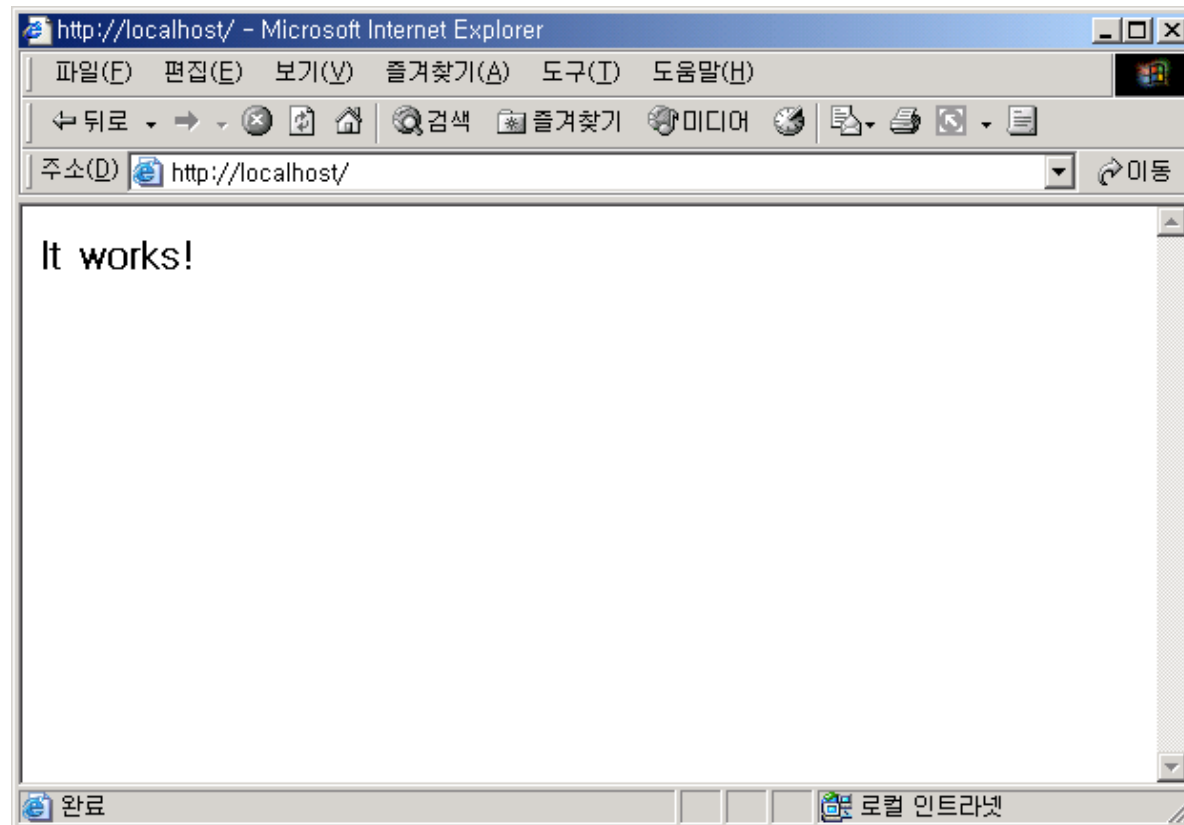


애플릿 프로그래밍

01. 애플릿에 대하여

아파치 웹 서버 시작하기

- 아파치 웹 서버의 디폴트 홈페이지

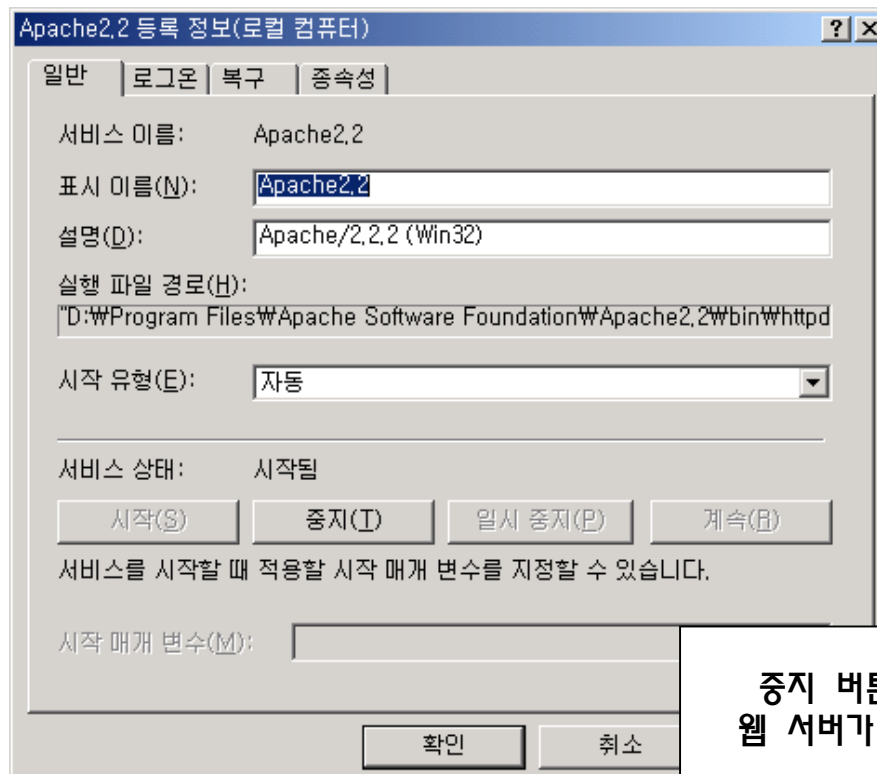


애플릿 프로그래밍

01. 애플릿에 대하여

아파치 웹 서버 종료하기

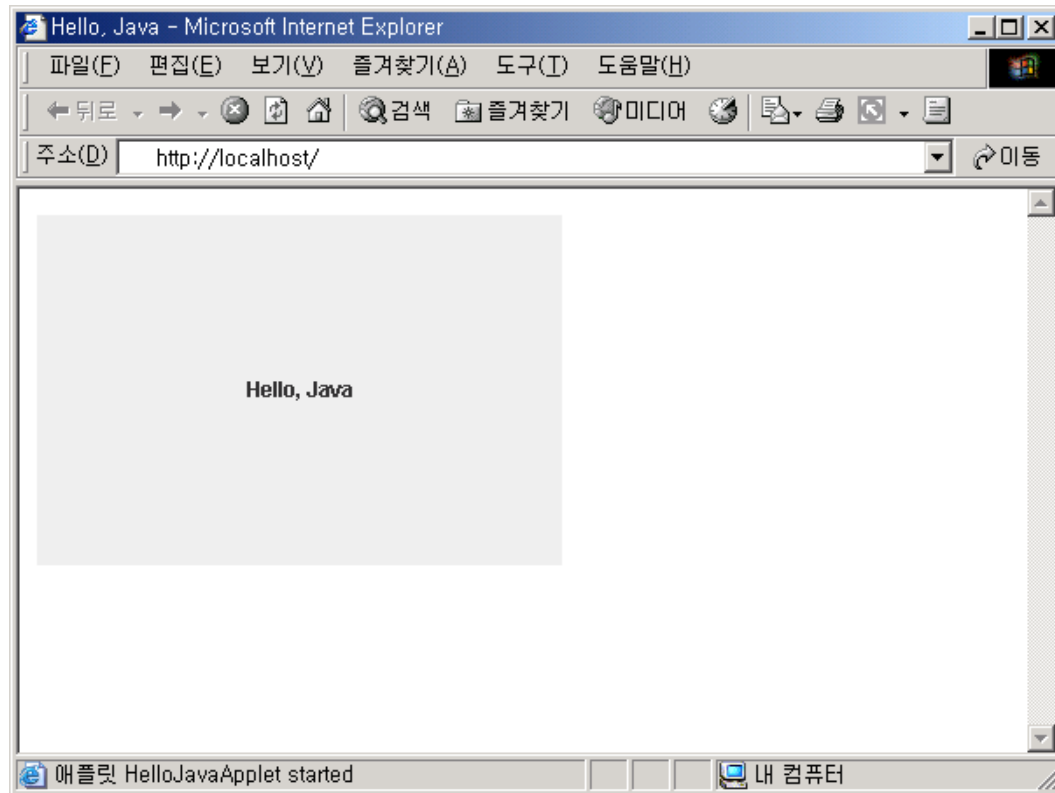
• 시작 -> 설정 -> 제어판 -> 관리도구 -> 서비스 메뉴를 선택했을 때 나타나는 창



중지 버튼을 누르면
웹 서버가 중지됩니다.

애플릿 프로그래밍

- 02. 애플릿의 작성 방법
- 애플릿으로 작동하는 Hello, Java 프로그램
 - 지금부터 작성할 예제의 화면 설계



애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿으로 작동하는 Hello, Java 프로그램

- 애플릿 클래스의 작성 방법

- 1) javax.swing.JApplet 클래스를 상속받는 public 클래스로 선언합니다.

```
public class HelloJavaApplet extends javax.swing.JApplet {  
    ...  
}
```

public 클래스로 선언해야 합니다

javax.swing.JApplet의 서브클래스로 선언해야 합니다

애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿으로 작동하는 Hello, Java 프로그램

• 애플릿 클래스의 작성 방법

- 2) 애플릿이 생성되고 나서 바로 해야 할 일은 init 메소드 안에서 해야 합니다.

```
public class HelloJavaApplet extends javax.swing.JApplet {  
    public void init() {  
  
    }  
}
```

..... 애플릿이 생성되고 나서 해야 할 일은 이 메소드 안에 써야 합니다

애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿으로 작동하는 Hello Java 프로그램

- [예제 22-1] 애플릿으로 작동하는

```
명령 프롬프트
E:\work\chap22\22-2>javac HelloJavaApplet.java
E:\work\chap22\22-2>dir
E 드라이브의 볼륨: 로컬 디스크
볼륨 일련 번호: 305B-17D4

E:\work\chap22\22-2 디렉터리

2006-06-23  01:12a    <DIR>          .
2006-06-23  01:12a    <DIR>          ..
2006-07-14  11:56a             407 HelloJavaApplet.class
2006-07-14  11:03a             146 HelloJavaApplet.java
                        2개 파일             553 바이트
                        2 디렉터리        505,298,944 바이트 남음

E:\work\chap22\22-2>
```

1) 애플릿 클래스를 컴파일하면
클래스 파일이 생성됩니다

2) 이 파일이 애플릿
사용될 수 있습니다

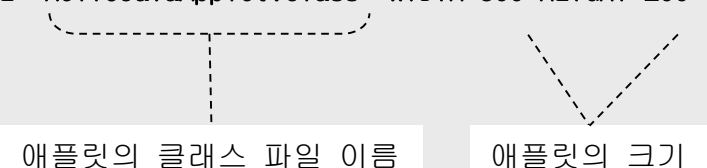
```
1  import java.awt.*;
2  import javax.swing.*;
3  public class HelloJavaApplet extends JApplet {
4      public void init() {
5          Container contentPane = getContentPane();
6          contentPane.add(new JLabel("Hello, Java", SwingConstants.CENTER));
7      }
8  }
```

02. 애플릿의 작성 방법

애플릿으로 작동하는 Hello, Java 프로그램

- [예제 22-2] HelloJavaApplet을 포함하는 HTML 문서

```
1  <HTML>
2  <HEAD><TITLE>Hello, Java</TITLE></HEAD>
3  <BODY>
4      <APPLET CODE="HelloJavaApplet.class" WIDTH=300 HEIGHT=200>
5      </APPLET>
6  </BODY>
7  </HTML>
```



애플릿의 클래스 파일 이름

애플릿의 크기

애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿으로 작동하는 Hello, Java 프로그램

- [예제 22-2] HelloJavaApplet을 포함하는 HTML 문서 - 실행 결과 (1)

1) 이 HTML 문서를 웹 브라우저로 여십시오.

2) 그러면 HTML 문서 안에 애플릿이 나타납니다.

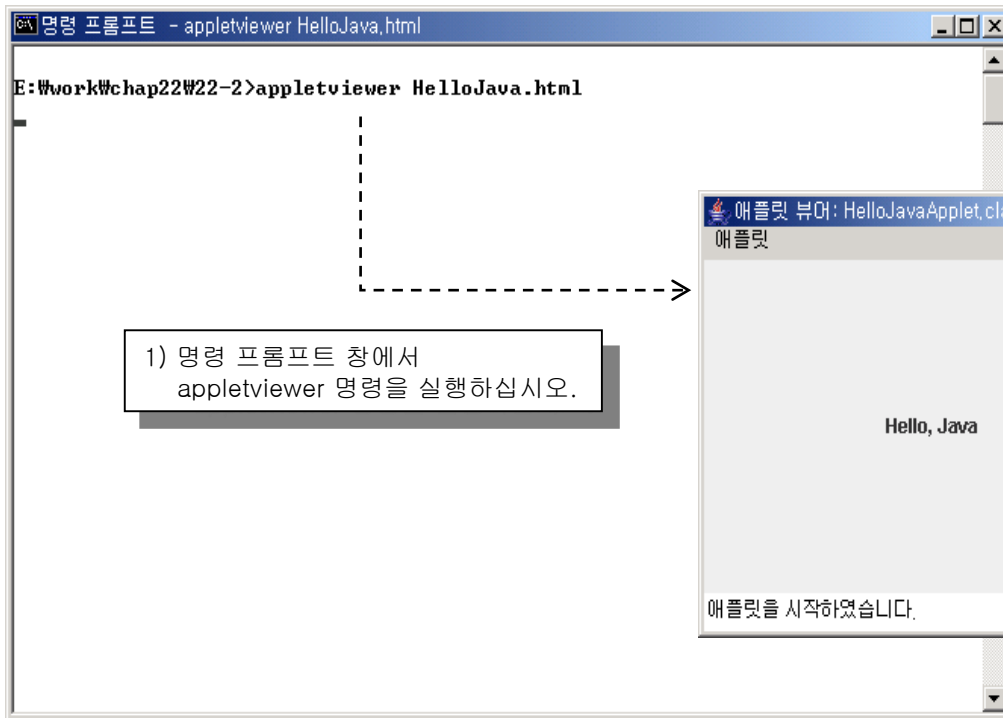
애플릿 HelloJavaApplet started

애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿으로 작동하는 Hello, Java 프로그램

- [예제 22-2] HelloJavaApplet을 포함하는 HTML 문서 - 실행 결과 (2)



1) 명령 프롬프트 창에서
appletviewer 명령을 실행하십시오.

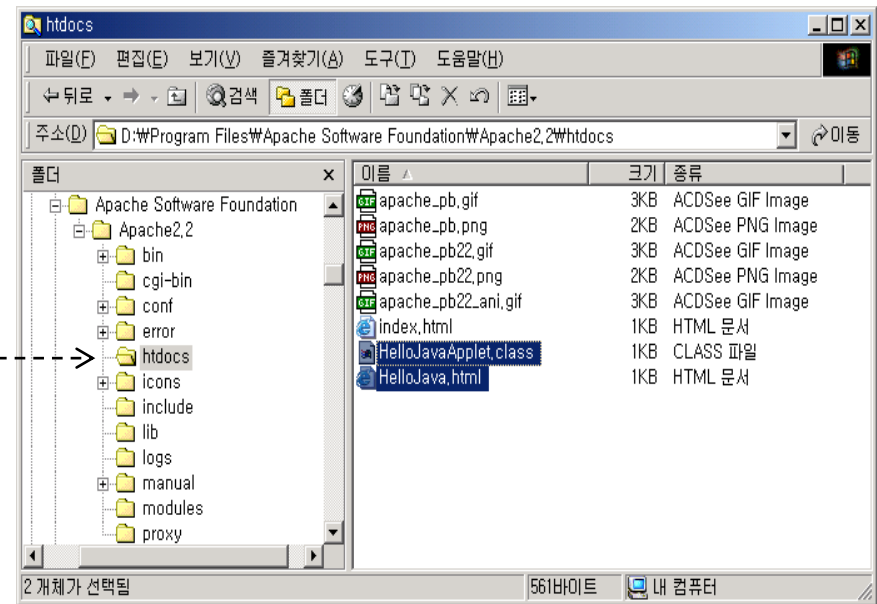
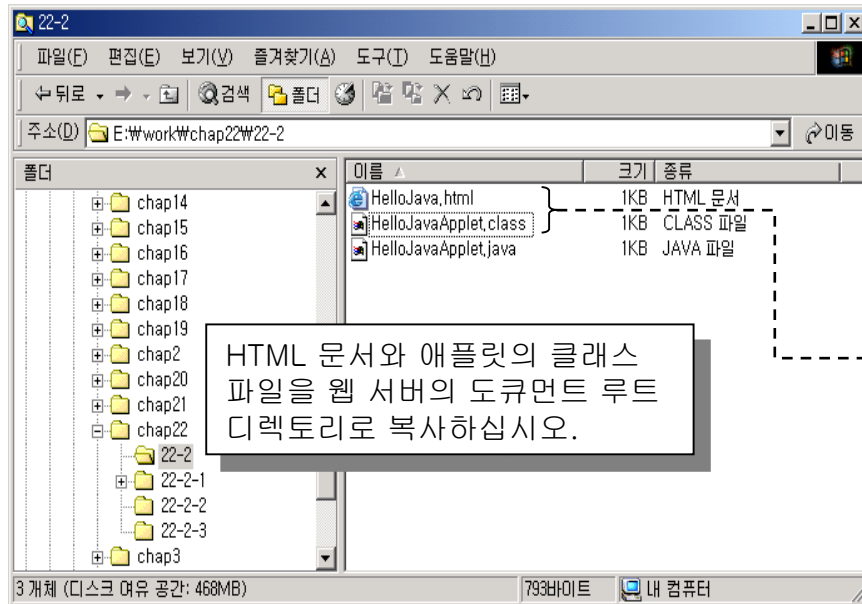
2) 그러면 애플릿 뷰어 창이 뜨고,
그 안에 애플릿이 나타납니다.

애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿으로 작동하는 Hello, Java 프로그램

- [예제 22-2] HelloJavaApplet을 포함하는 HTML 문서 – 웹 서버로의 설치

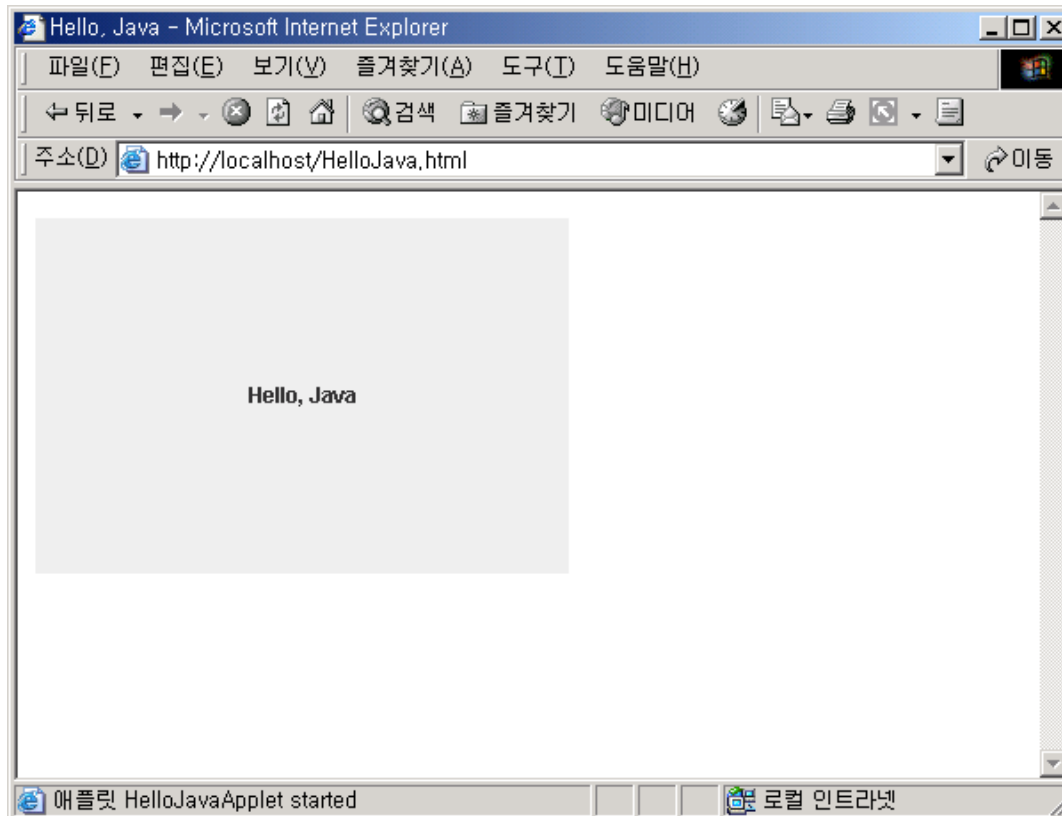


애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿으로 작동하는 Hello, Java 프로그램

- [예제 22-2] HelloJavaApplet을 포함하는 HTML 문서 - 실행 결과 (3)



애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿으로 작동하는 윈도우 프로그램

- [예제 22-3] 여러가지 컴포넌트가 있는 애플릿

애플릿 클래스

```
1 import java.awt.*;
2 import javax.swing.*;
3 public class ContactInfoApplet extends JApplet {
4     public void init() {
5         Container contentPane = getContentPane();
6         JPanel panel = new JPanel();
7         panel.setLayout(new GridLayout(3, 2));
8         JTextField text1 = new JTextField();
9         JTextField text2 = new JTextField();
10        JTextField text3 = new JTextField();
11        panel.add(new JLabel("이름"));
12        panel.add(text1);
13        panel.add(new JLabel("주소"));
14        panel.add(text2);
15        panel.add(new JLabel("전화번호"));
16        panel.add(text3);
17        contentPane.add(panel, BorderLayout.CENTER);
18        JButton button = new JButton("입력");
19        contentPane.add(button, BorderLayout.SOUTH);
20        button.addActionListener(
21            new InsertActionListener(text1, text2, text3));
22    }
```

컴포넌트를
만들어서
배치합니다

입력 버튼 리스너를 등록합니다

입력 버튼을 처리하는 리스너 클래스

```
1 import java.awt.event.*;
2 import javax.swing.*;
3 class InsertActionListener implements ActionListener {
4     JTextField text1, text2, text3;
5     InsertActionListener(JTextField text1, JTextField text2,
6         JTextField text3) {
7         this.text1 = text1;
8         this.text2 = text2;
9         this.text3 = text3;
10    }
11    public void actionPerformed(ActionEvent e) {
12        String name = text1.getText();
13        String address = text2.getText();
14        String phoneno = text3.getText();
15        System.out.println("이름: " + name);
16        System.out.println("주소: " + address);
17        System.out.println("전화번호 " + phoneno);
18    }
```

애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿으로 작동하는 윈도우 프로그램

- [예제 22-3] 여러가지 컴포넌트가 있는 애플릿 - 설치 방법

1) 도큐먼트 루트 디렉토리 아래에 이런 디렉토리를 만드십시오

2) [예제 22-3]을 컴파일한 결과물이 이 디렉토리에 복사하십시오

3) 텍스트 에디터로 이런 HTML 문서를 만들어서 같은 디렉토리에 저장하십시오

```
<HTML>
<HEAD><TITLE>연락처 입력 프로그램</TITLE></HEAD>
<BODY>
  <H1>연락처 입력 프로그램</H1>
  <APPLET CODE="ContactInfoApplet.class" WIDTH=250 HEIGHT=100>
</APPLET>
</BODY>
</HTML>
```

애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿으로 작동하는 윈도우 프로그램

- [예제 22-3] 여러가지 컴포넌트가 있는 애플릿 - 실행 결과

The screenshot shows a Microsoft Internet Explorer window titled "연락처 입력 프로그램 - Microsoft Internet Explorer". The address bar shows "http://localhost/customer/ContactInfo.html". The page content includes a form with the following fields:

이름	강수영
주소	서울시 동작구 CCC동
전화번호	02-814-0000

Below the form is a button labeled "입력". A dashed box highlights the button and a text box that says: "2) 입력 버튼을 눌렀을 때 호출되는 System.out.println 메서드는 자바 콘솔 창으로 메시지를 출력합니다".

To the right of the browser window is a "Java 콘솔" window. It displays the following text:

```
Java Plug-in 1.5.0_07
JRE 버전 사용 1.5.0_07 Java HotSpot(TM) Client VM
사용자 홈 디렉토리 = D:\Documents and Settings\User

c: 콘솔 창 지우기
f: 완성 대기열에서 객체 완성
g: 가비지 컬렉션
h: 이 도움말 메시지 표시
l: 클래스로더 목록 덤프
m: 메모리 사용 인쇄
o: 로깅 트리거
p: 프록시 구성 재로드
q: 콘솔 숨기기
r: 정책 구성 재로드
s: 시스템 및 배포 등록 정보 덤프
t: 스레드 목록 덤프
v: 스레드 스택 덤프
x: 클래스로더 캐시 지우기
0-5: 추적 수준을 <n>(으)로 설정

이름: 강수영
주소: 서울시 동작구 CCC동
전화번호 02-814-0000
```

At the bottom of the Java console window are buttons for "지우기", "복사", and "닫기".

02. 애플릿의 작성 방법

애플릿으로 작동하는 그래픽 프로그램

- <PARAM> 태그에 대하여

1) <APPLET>과 </APPLET> 태그 사이에는 <PARAM> 태그를 쓸 수 있습니다.

```
<APPLET CODE="BarGraphApplet.class" WIDTH=500 HEIGHT=300>  
  <PARAM NAME="KOREAN" VALUE=100>  
  <PARAM NAME="ENGLISH" VALUE=85>  
  <PARAM NAME="MATH" VALUE=99>  
</APPLET>
```

} 애플릿 클래스에 전달되는 파라미터

애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿으로 작동하는 그래픽 프로그램

- <PARAM> 태그에 대하여
 - 2) <PARAM> 태그의 값은 다음과 같은 방법으로 프로그램에서 가져올 수 있습니다.

```
public class BarGraphApplet extends JApplet {  
    public void init() {  
        String str1 = getParameter("KOREAN");  
        String str2 = getParameter("ENGLISH");  
        String str3 = getParameter("MATH");  
        ...  
    }  
}
```

} HTML 문서에 있는 애플릿 파라미터를 가져오는 getParameter 메소드

애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿으로 작동하는 그래픽 프로그램

- [예제 22-4] 그래프가 그려진 애플릿

애플릿 클래스

```
1 import java.awt.*;
2 import javax.swing.*;
3 public class BarGraphApplet extends JApplet {
4     public void init() {
5         int korean = Integer.parseInt(getParameter("KOREAN"));
6         int english = Integer.parseInt(getParameter("ENGLISH"));
7         int math = Integer.parseInt(getParameter("MATH"));
8         Container contentPane = getContentPane();
9         contentPane.add(new DrawingPanel(korean, english, math));
10    }
11 }
```

그림을 그리는 패널을 생성해서
애플릿에 추가합니다.

HTML 문서에 있는 PARAM
태그의 값을 가져옵니다.

그래프를 그리는 패널 클래스

```
1 import java.awt.*;
2 import javax.swing.*;
3 class DrawingPanel extends JPanel {
4     int korean, english, math;
5     DrawingPanel(int korean, int english, int math) { // 생성자
6         this.korean = korean;
7         this.english = english;
8         this.math = math;
9     }
10    public void paint(Graphics g) {
11        g.clearRect(0, 0, getWidth(), getHeight()); -----
12        g.drawLine(50, 250, 350, 250);
13        for (int cnt = 1; cnt < 11; cnt++) {
14            g.drawString(cnt * 10 + "", 25, 255 - 20 * cnt);
15            g.drawLine(50, 250 - 20 * cnt, 350, 250 - 20 * cnt);
16        }
17        g.drawLine(50, 20, 50, 250);
18        g.drawString("국어", 100, 270);
19        g.drawString("영어", 200, 270);
20        g.drawString("수학", 300, 270);
21        g.setColor(Color.RED);
22        if (korean > 0)
23            g.fillRect(110, 250 - korean * 2, 10, korean * 2);
24        if (english > 0)
25            g.fillRect(210, 250 - english * 2, 10, english * 2);
26        if (math > 0)
27            g.fillRect(310, 250 - math * 2, 10, math * 2);
28    }
29 }
```

패널 전체를 흰색으로
클리어합니다

좌표를
그립니다

막대 그래프를
그립니다

애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿으로 작동하는 그래픽 프로그램

- [예제 22-4] 그래프가 그려진 애플릿 - 설치 방법

1) 도큐먼트 루트 디렉토리 아래에 이런 디렉토리를 만드십시오

2) [예제 22-4]를 컴파일한 결과물을 이 디렉토리에 복사하십시오

3) 텍스트 에디터로 이런 HTML 문서를 만들어서 같은 디렉토리에 저장하십시오

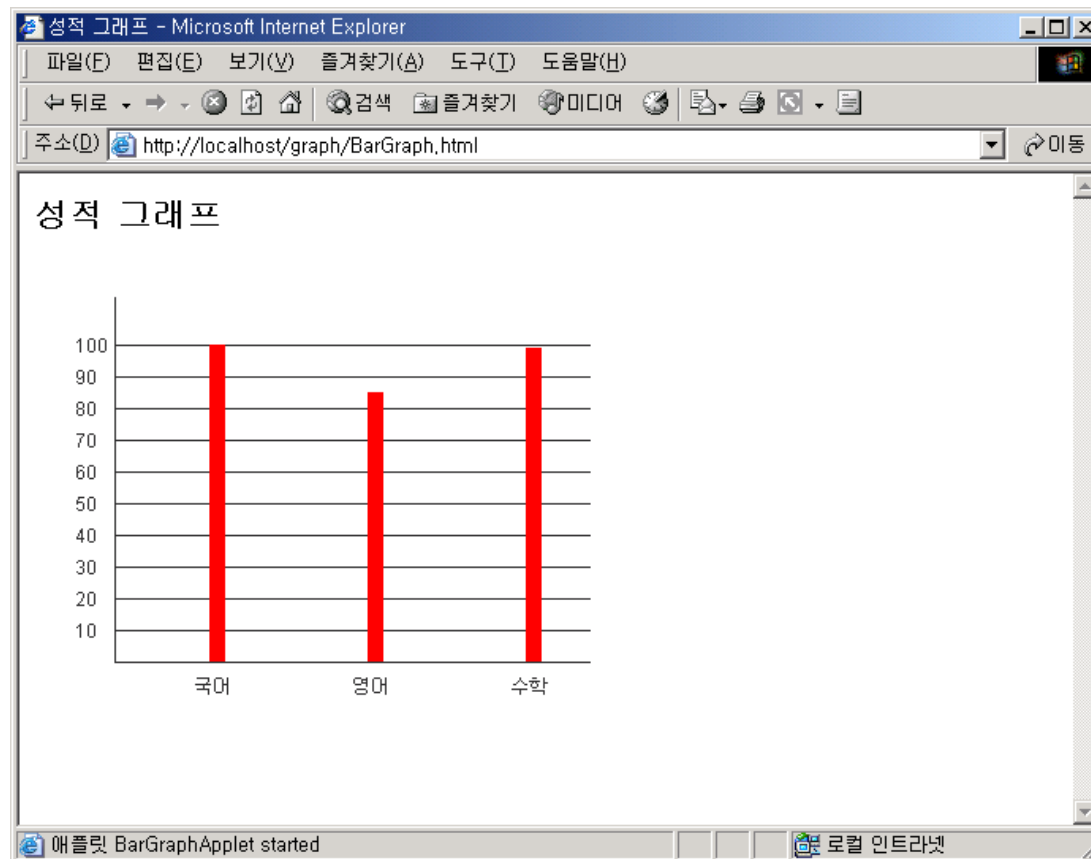
```
<HTML>
<HEAD><TITLE>성적 그래프</TITLE></HEAD>
<BODY>
<H1>성적 그래프</H1>
  <APPLET CODE="BarGraphApplet.class" WIDTH=500 HEIGHT=300>
    <PARAM NAME="KOREAN" VALUE=100>
    <PARAM NAME="ENGLISH" VALUE=85>
    <PARAM NAME="MATH" VALUE=99>
  </APPLET>
</BODY>
</HTML>
```

애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿으로 작동하는 그래픽 프로그램

- [예제 22-4] 그래프가 그려진 애플릿 - 실행 결과

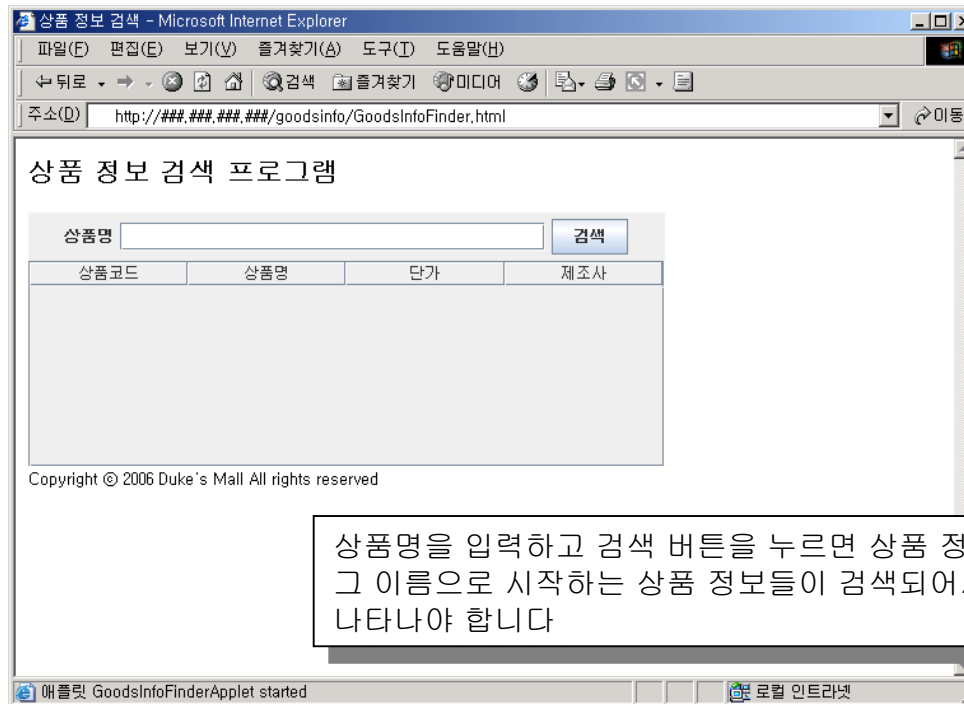


애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿과 데이터베이스

- 지금부터 작성할 애플릿의 화면 설계



애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿과 데이터베이스

• [예제 22-5] 데이터베이스를 사용하는 애플릿

애플릿 클래스

```
1 import java.awt.*;
2 import javax.swing.*;
3 import javax.swing.table.*;
4 public class GoodsInfoFinder {
5     public void init() {
6         Container contentPane = getContentPane();
7         JPanel panel = new JPanel();
8         JTextField text = new JTextField(20);
9         JButton button = new JButton("검색");
10        panel.add(new JLabel("상품명:"));
11        panel.add(text);
12        panel.add(button);
13        contentPane.add(panel);
14        String colNames[] = {"code", "name", "price", "maker"};
15        DefaultTableModel model = new DefaultTableModel(colNames, 0);
16        JTable table = new JTable(model);
17        contentPane.add(new JScrollPane(table));
18        button.addActionListener(new SearchActionListener(table, text));
19    }
20 }
```

검색 버튼 리스너 클래스

```
1 import java.awt.event.*;
2 import javax.swing.*;
3 import javax.swing.table.*;
4 import java.sql.*;
5 class SearchActionListener implements ActionListener {
6     JTable table;
7     JTextField text;
8     SearchActionListener(JTable table, JTextField text) { // 생성자
9         this.table = table;
10        this.text = text;
11    }
12    public void actionPerformed(ActionEvent e) {
13        String name = text.getText();
14        DefaultTableModel model = (DefaultTableModel) table.getModel();
15        model.setRowCount(0);
16        Connection conn = null;
17        Statement stmt = null;
18        try {
19            Class.forName("com.mysql.jdbc.Driver");
20            conn = DriverManager.getConnection(
21                "jdbc:mysql://###.###.###.###:3306/malldb", "root", "1234");
22            stmt = conn.createStatement();
23            ResultSet rs = stmt.executeQuery(
24                "select * from goodsinfo where "
25                + "name like '" + toLatin1(name) + "%'");
26            while (rs.next()) {
27                String arr[] = new String[4];
28                arr[0] = toUnicode(rs.getString("code"));
29                arr[1] = toUnicode(rs.getString("name"));
30                arr[2] = toUnicode(rs.getString("price"));
31                arr[3] = toUnicode(rs.getString("maker"));
32                model.addRow(arr);
33            }
34        } catch (ClassNotFoundException cnfe) {
35            showErrorMessage("해당 클래스를 찾을 수 없습니다." +
36                cnfe.getMessage());
37        }
38    }
39 }
```

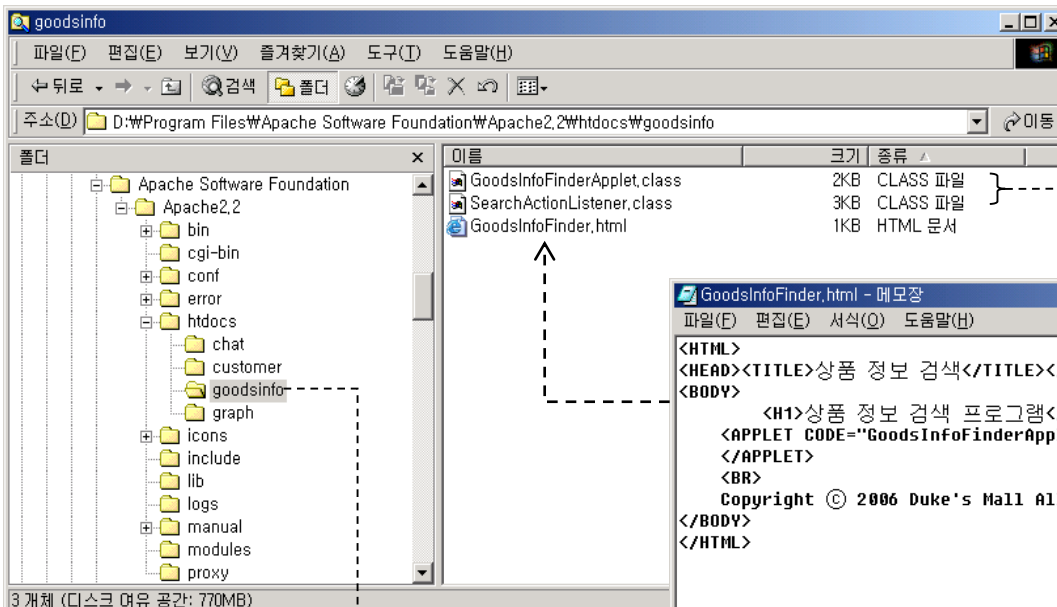
```
34 }
35 catch (SQLException se) {
36     showErrorMessage("데이터베이스 에러입니다.");
37 }
38 catch (java.io.UnsupportedEncodingException uee) {
39     showErrorMessage("지원되지 않는 문자셋입니다.");
40 }
41 finally {
42     try {
43         stmt.close();
44     }
45     catch (Exception ignored) {
46     }
47     try {
48         conn.close();
49     }
50     catch (Exception ignored) {
51     }
52 }
53 }
54 private String toUnicode(String str)
55     throws java.io.UnsupportedEncodingException {
56     return new String(str.getBytes("ISO-8859-1"));
57 }
58 private String toLatin1(String str)
59     throws java.io.UnsupportedEncodingException {
60     return new String(str.getBytes(), "ISO-8859-1");
61 }
62 private void showErrorMessage(String str) {
63     JOptionPane.showMessageDialog(text, str, "에러 메시지",
64         JOptionPane.ERROR_MESSAGE);
65 }
66 }
```

애플릿 프로그래밍

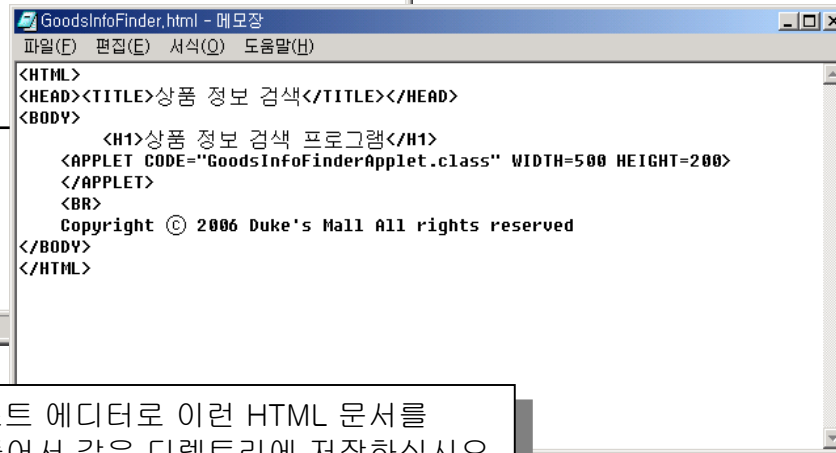
02. 애플릿의 작성 방법

애플릿과 데이터베이스

- [예제 22-5] 데이터베이스를 사용하는 애플릿 - 설치 방법



2) [예제 22-5]를 컴파일한 결과물을 이 디렉토리에 복사하십시오



1) 도큐먼트 루트 디렉토리 아래에 이런 디렉토리를 만드십시오

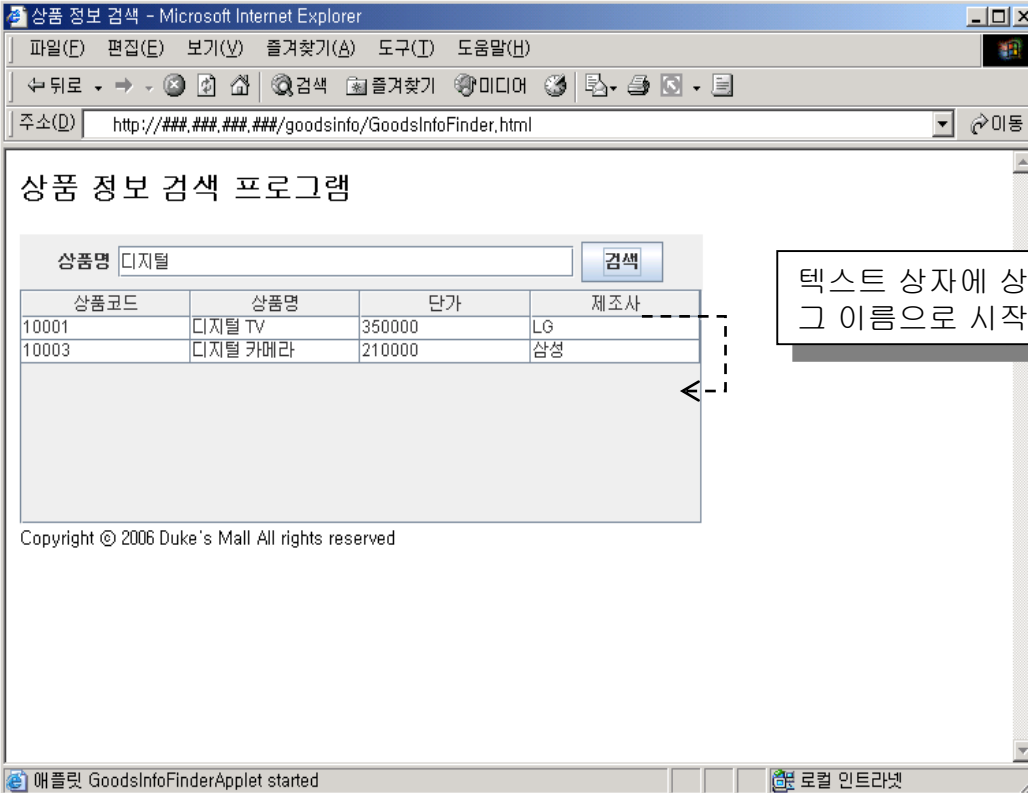
3) 텍스트 에디터로 이런 HTML 문서를 만들어서 같은 디렉토리에 저장하십시오

애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿과 데이터베이스

- [예제 22-5] 데이터베이스를 사용하는 애플릿 - 실행 결과



상품 정보 검색 프로그램

상품명

상품코드	상품명	단가	제조사
10001	디지털 TV	350000	LG
10003	디지털 카메라	210000	삼성

Copyright © 2006 Duke's Mall All rights reserved

애플릿 GoodsInfoFinderApplet started

로컬 인트라넷

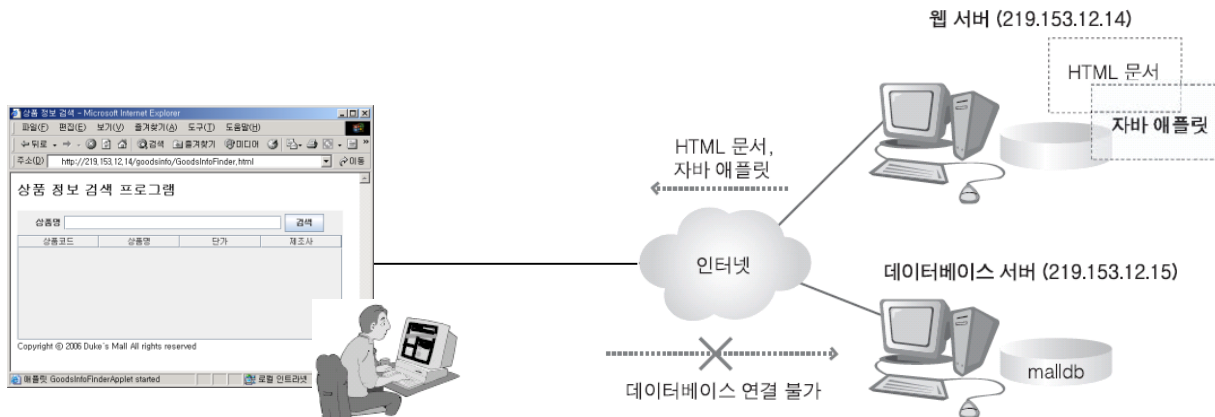
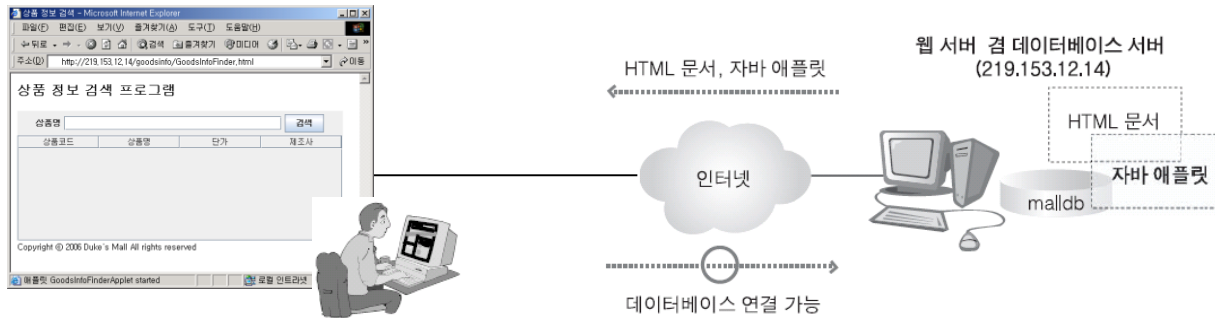
텍스트 상자에 상품명을 입력하고 검색 버튼을 누르면
그 이름으로 시작하는 모든 상품 정보가 검색됩니다

애플릿 프로그래밍

02. 애플릿의 작성 방법

애플릿과 데이터베이스

- 데이터베이스로의 연결이 가능한 경우와 불가능한 경우



02. 애플릿의 작성 방법

애플릿의 제약

• 애플릿으로 할 수 없는 일

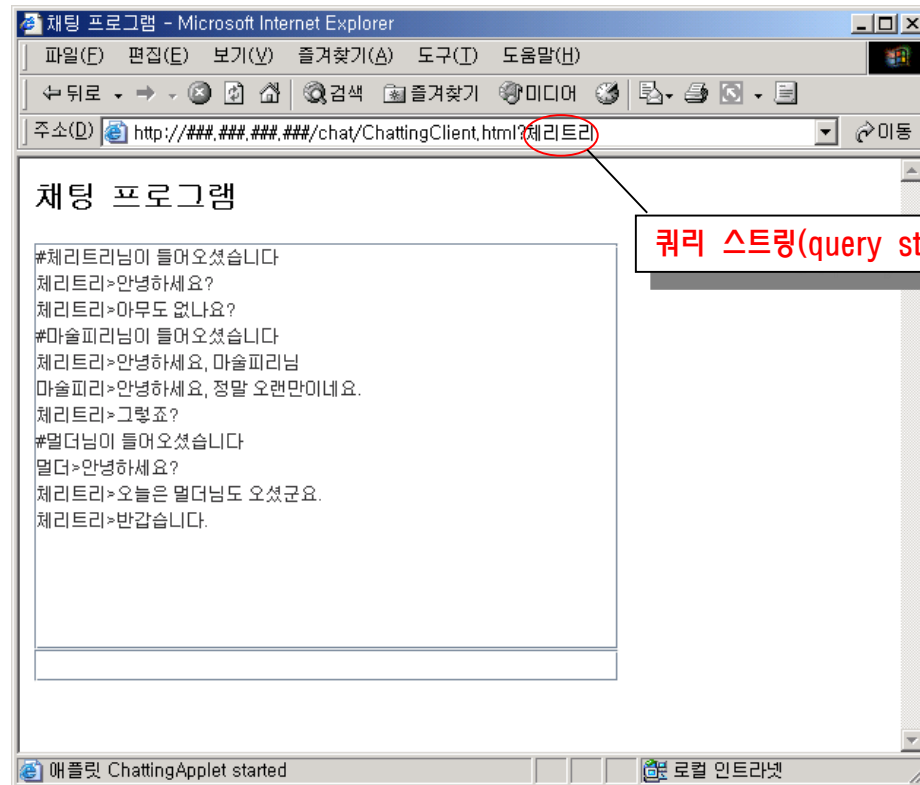
- 웹 서버와 다른 IP 주소의 컴퓨터에 있는 데이터베이스 서버에 연결할 수 없습니다.
- 웹 서버와 다른 IP 주소의 컴퓨터에 있는 서버 프로그램에 연결할 수 없습니다.
- 애플릿이 실행되고 있는 로컬 컴퓨터에 파일을 읽거나 쓸 수 없습니다.
- 애플릿이 실행되고 있는 로컬 컴퓨터에서 새로운 프로그램의 실행을 시작할 수 없습니다.
- 애플릿이 실행되고 있는 로컬 컴퓨터에서 자바가 아닌 다른 언어로 작성된 라이브러리 의 코드를 실행 할 수 없습니다.

애플릿 프로그래밍

02. 애플릿의 작성 방법

채팅 클라이언트 애플릿

• 채팅 클라이언트 애플릿의 화면 설계



애플릿 프로그래밍

02. 애플릿의 작성 방법

채팅 클라이언트 애플릿

- 애플릿 클래스 안에서 쿼리 스트링을 읽어오는 방법

```
public class ChattingApplet extends JApplet {  
    public void init() {  
        ...  
        URL url = getDocumentBase();  
        String name = url.getQuery();  
        ...  
    }  
}
```

애플릿이 포함된 HTML 문서의 URL을 가져오는 메소드입니다.

URL에서 쿼리 스트림만 가져오는 메소드입니다.

02. 애플릿의 작성 방법

채팅 클라이언트 애플릿

- 텍스트 상자에서 Enter 키를 눌렀을 때의 이벤트 리스너 등록 방법

```
text.addActionListener(new EnterActionListener());
```

JTextField 객체

Enter 키를 눌렀을 때 호출되는 action listener

02. 애플릿의 작성 방법

채팅 클라이언트 애플릿

- 채팅 서버로부터 오는 메시지를 수신하는 스레드의 시작

```
public class ChattingApplet extends JApplet {  
    public void init() {  
        ...  
        Thread thread = new ReceiverThread();  
        thread.start();  
        ...  
    }  
}
```

} init 메소드 안에서 메시지 수신 스레드를 생성해서 시작합니다

02. 애플릿의 작성 방법

채팅 클라이언트 애플릿

- 애플릿이 소멸되기 직전에 호출되는 `destroy` 메소드

```
public class HelloJavaApplet extends javax.swing.JApplet {  
    public void destroy() {
```

..... 애플릿이 소멸되기 전에 해야 할 일은
이 메소드 안에 써야 합니다

```
    }  
}
```

애플릿 프로그래밍

02. 애플릿의 작성 방법

채팅 클라이언트 애플릿

• [예제 22-6] 채팅 클라이언트 애플릿

애플릿 클래스

```
1 import java.awt.*;
2 import java.awt.event.*;
3 import javax.swing.*;
4 import java.net.*;
5 import java.io.*;
6 public class ChattingApplet extends JApplet {
7     Socket socket;
8     public void init() {
9         try {
10             socket = new Socket(InetAddress.getByName("127.0.0.1"), 8080);
11         } catch (IOException ioe) {
12             JOptionPane.showMessageDialog(this, "서버에 연결할 수 없습니다.", "오류",
13                 JOptionPane.ERROR_MESSAGE);
14             return;
15         }
16         Container contentPane = getContentPane();
17         JTextArea text1 = new JTextArea(10, 30);
18         JTextField text2 = new JTextField(20);
19         text1.setEditable(false);
20         contentPane.add(new JScrollPane(text1));
21         contentPane.add(text2, BorderLayout.EAST);
22         URL url = getDocumentBase();
23         String name = url.getQuery();
24         ActionListener listener = new EnterListener(text2, socket, name);
25         text2.addActionListener(listener);
26         Thread thread = new ReceiverThread(text1, socket, name);
27         thread.start();
28     }
29     public void destroy() {
30         try {
31             socket.close();
32         } catch (IOException ioe) {
33             JOptionPane.showMessageDialog(this, "서버에 연결할 수 없습니다.", "오류",
34                 JOptionPane.ERROR_MESSAGE);
35         }
36     }
37 }
```

메시지를 수신하는 스레드 클래스

```
1 import java.io.*;
2 import java.net.*;
3 import javax.swing.*;
4 class ReceiverThread extends Thread {
5     JTextArea text;
6     Socket socket;
7     ReceiverThread(JTextArea text, Socket socket) {
8         this.text = text;
9         this.socket = socket;
10    }
11    public void run() {
12        try {
13            BufferedReader reader = new BufferedReader(
14                new InputStreamReader(socket.getInputStream()));
15            while (true) {
16                final String str = reader.readLine();
17                SwingUtilities.invokeLater(
18                    new Runnable() {
19                        public void run() {
20                            text.append(str + "\n");
21                        }
22                    }
23                );
24            }
25        } catch (IOException ioe) {
26            JOptionPane.showMessageDialog(text, ioe.getMessage(), "에러 메시지",
27                JOptionPane.ERROR_MESSAGE);
28            System.out.println("수신 종료");
29        }
30    }
31 }
```

검색 버튼 리스너 클래스

```
1 import java.awt.event.*;
2 import javax.swing.*;
```

ActionListener {

```
    JTextArea text, Socket socket, String name) {
```

```
        socket.getOutputStream());
```

```
        JOptionPane.showMessageDialog(text, ioe.getMessage(), "에러 메시지",
            JOptionPane.ERROR_MESSAGE);
```

```
    public void actionPerformed(ActionEvent e) {
```

```
    }
```

```
        JOptionPane.showMessageDialog(text, ioe.getMessage(), "에러 메시지",
            JOptionPane.ERROR_MESSAGE);
```

```
        text.setText("");
```

```
    }
```

```
    }
```

```
    }
```

애플릿 프로그래밍

02. 애플릿의 작성 방법

채팅 클라이언트 애플릿

- [예제 22-6] 채팅 클라이언트 애플릿 – 설치 방법

The screenshot shows a web browser window titled 'chat' with the address bar displaying 'D:\Program Files\Apache Software Foundation\Apache2.2\htdocs\chat'. The left sidebar shows a file tree with folders like 'bin', 'conf', 'htdocs', and 'manual'. The main content area lists files: 'ChattingApplet.class' (2KB, CLASS 파일), 'EnterActionListener.class' (2KB, CLASS 파일), 'ReceiverThread\$.class' (1KB, CLASS 파일), 'ReceiverThread.class' (2KB, CLASS 파일), and 'ChattingClient.html' (1KB, HTML 문서). A callout box points to the 'ChattingClient.html' file with the text: '2) [예제 22-6]를 컴파일한 결과물을 이 디렉토리에 복사하십시오'. Below this, a callout box points to the 'htdocs' folder in the file tree with the text: '1) 도큐먼트 루트 디렉토리 아래에 이런 디렉토리를 만드십시오'. At the bottom, a callout box points to the 'ChattingClient.html' file in the file list with the text: '3) 텍스트 에디터로 이런 HTML 문서를 만들어서 같은 디렉토리에 저장하십시오'. The 'ChattingClient.html' file content is shown in a separate window titled 'ChattingClient.html - 메모장' with the following code:

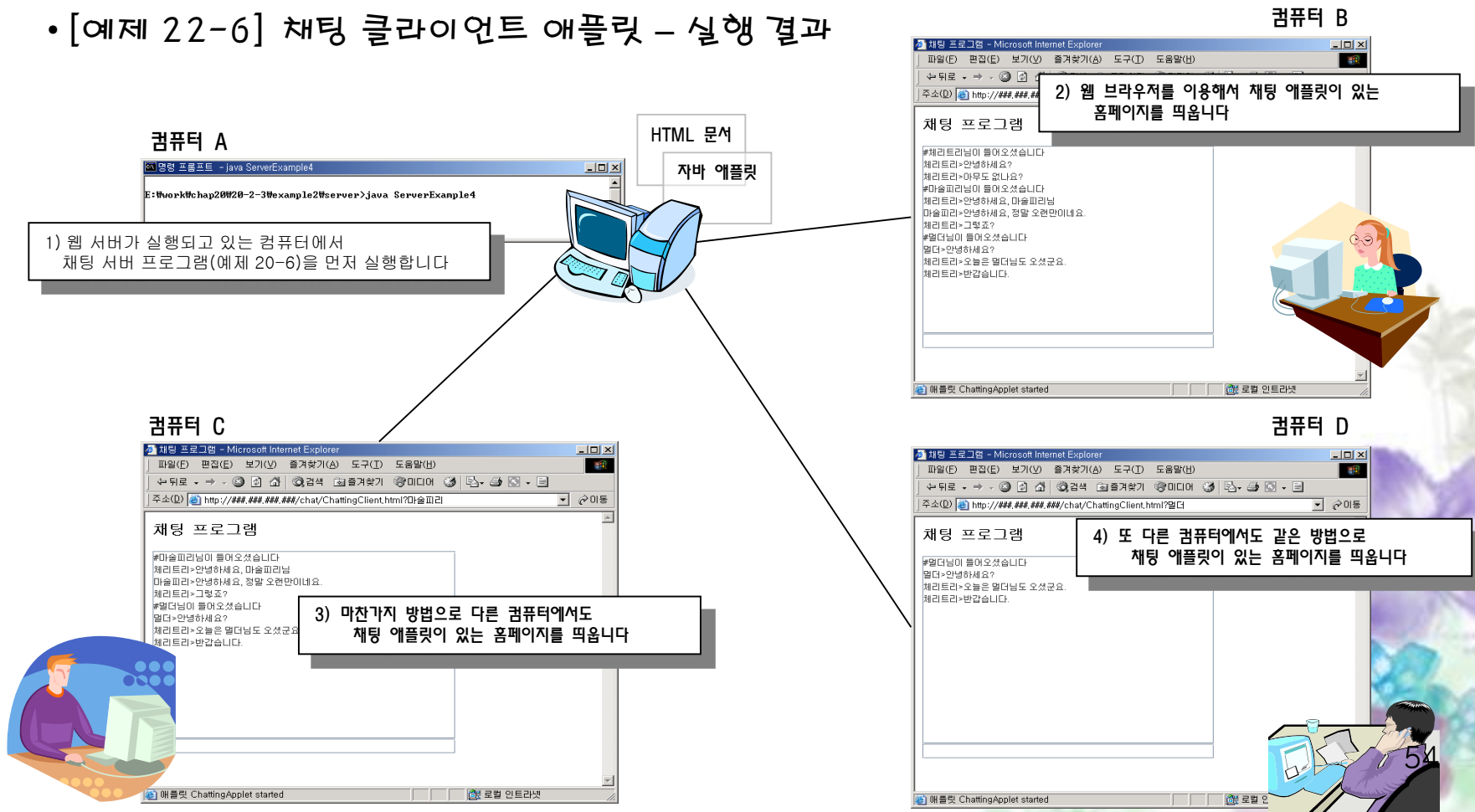
```
<HTML>
<HEAD><TITLE>채팅 프로그램</TITLE></HEAD>
<BODY>
  <H1>채팅 프로그램</H1>
  <APPLET CODE="ChattingApplet.class" WIDTH=400 HEIGHT=300>
</APPLET>
</BODY>
</HTML>
```

애플릿 프로그래밍

02. 애플릿의 작성 방법

채팅 클라이언트 애플릿

- [예제 22-6] 채팅 클라이언트 애플릿 - 실행 결과



정기

이

한

시

다

^^!