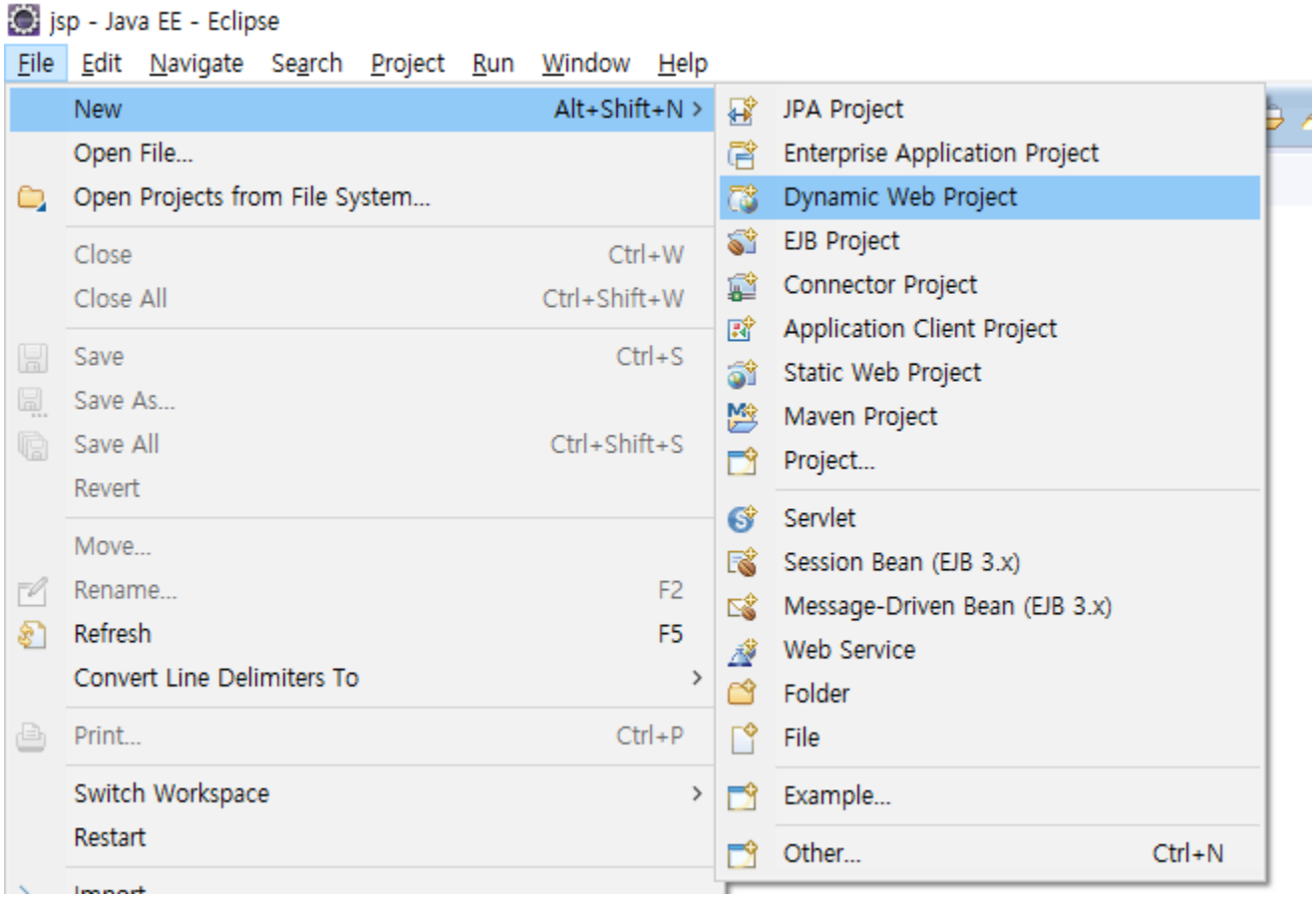
서블릿의 기초

❖ 프로젝트 생성

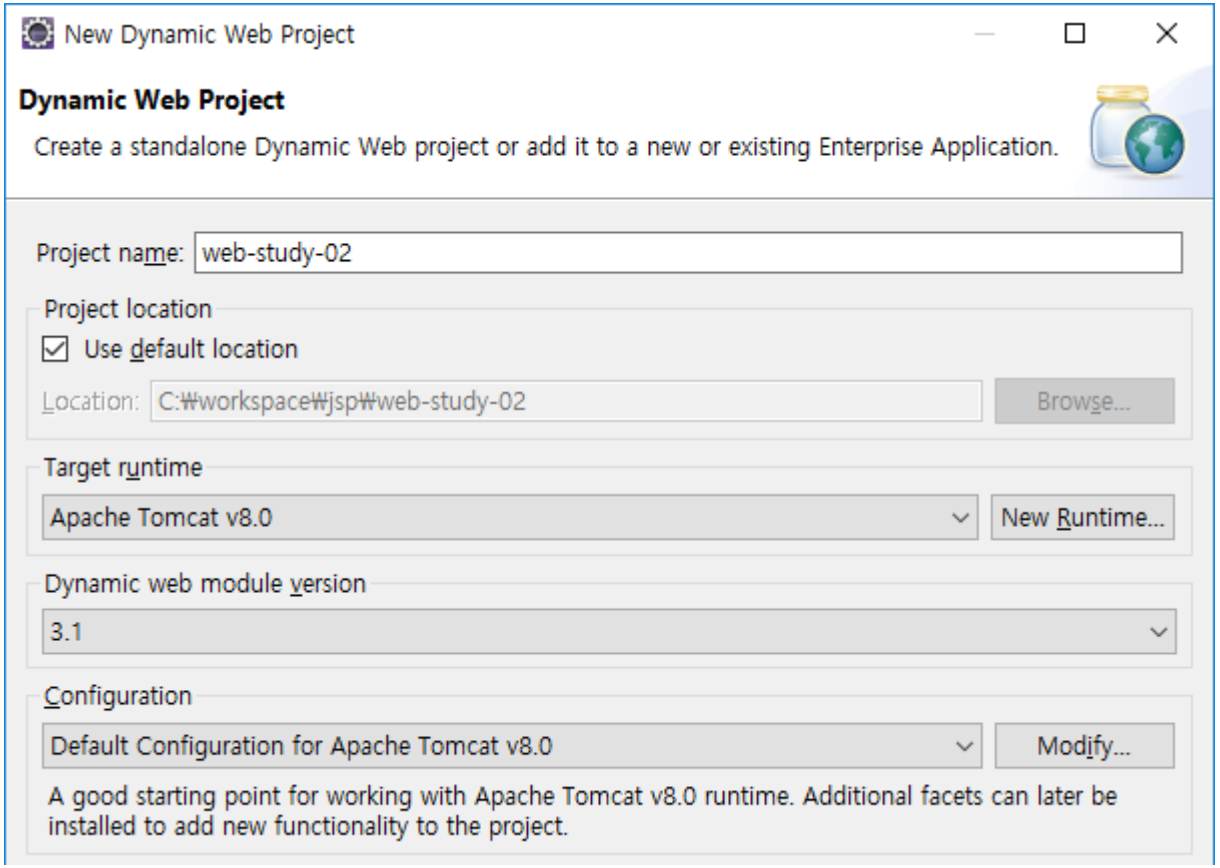
- File → New → Dynamic Web Project



서블릿 프로그램

❖ 프로젝트 생성

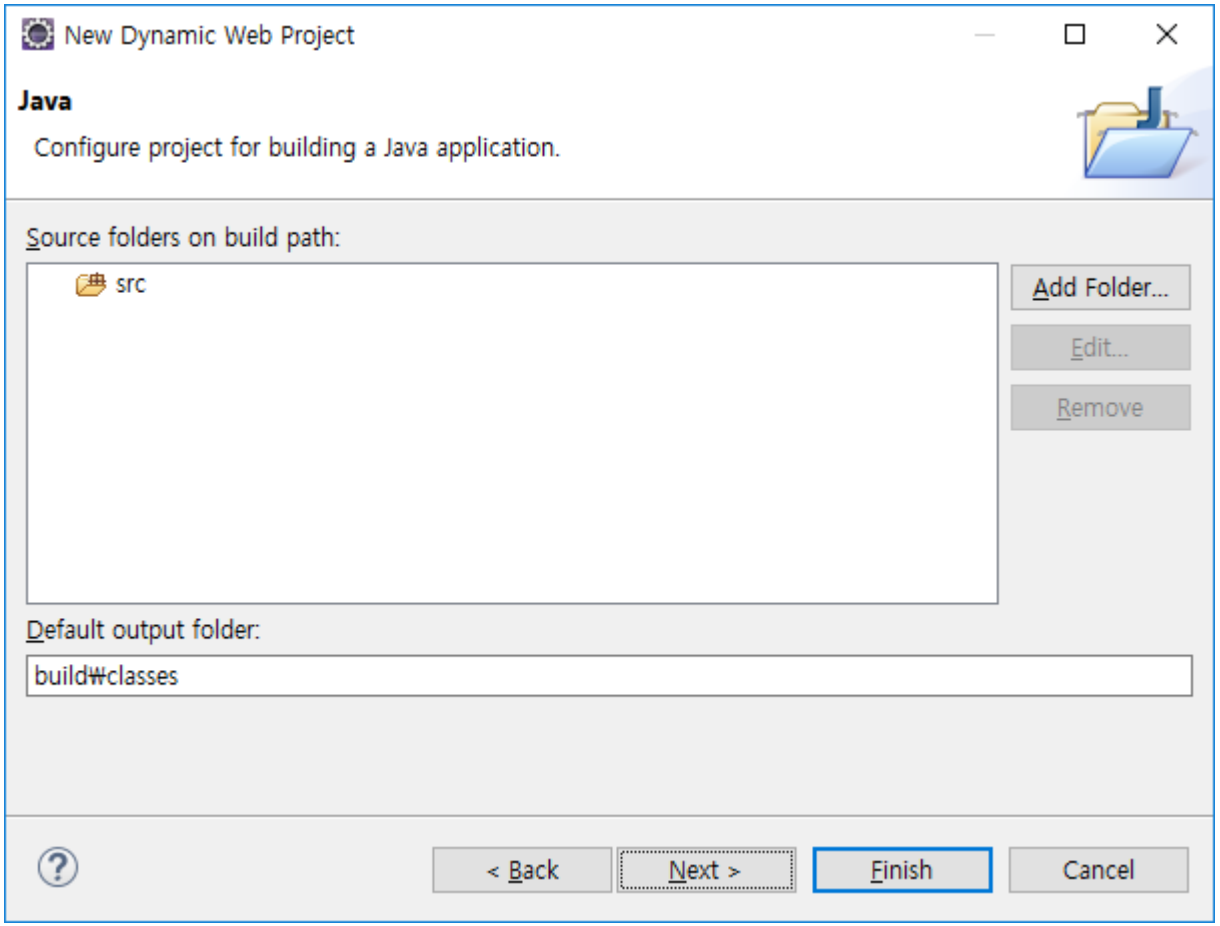
- ㅇ 프로젝트명 : web-study-02



서블릿 프로그램

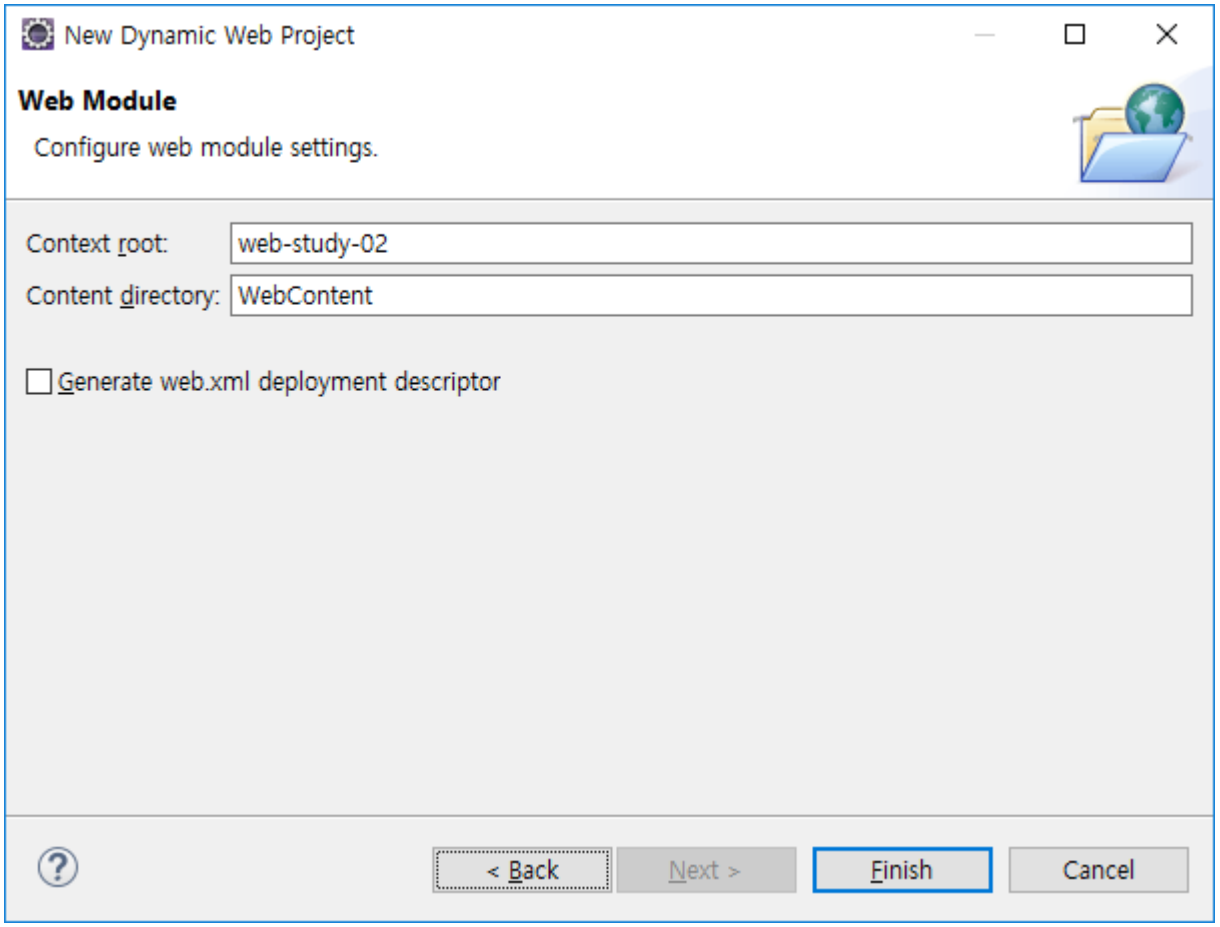
❖ 프로젝트 생성

- ㅇ 기본 폴더 구조 설정
  - ▪ 디폴트 유지



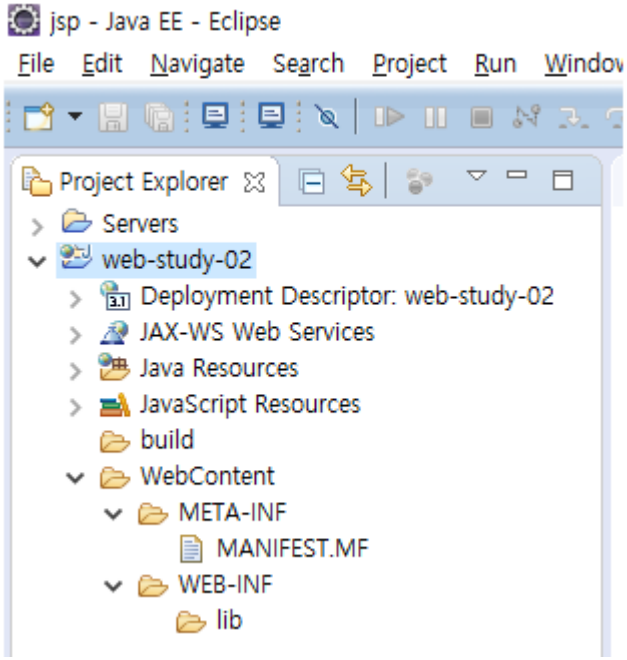
❖ 프로젝트 생성

- ㅇ 컨텍스트 패스 및 web.xml 파일 생성 여부 설정



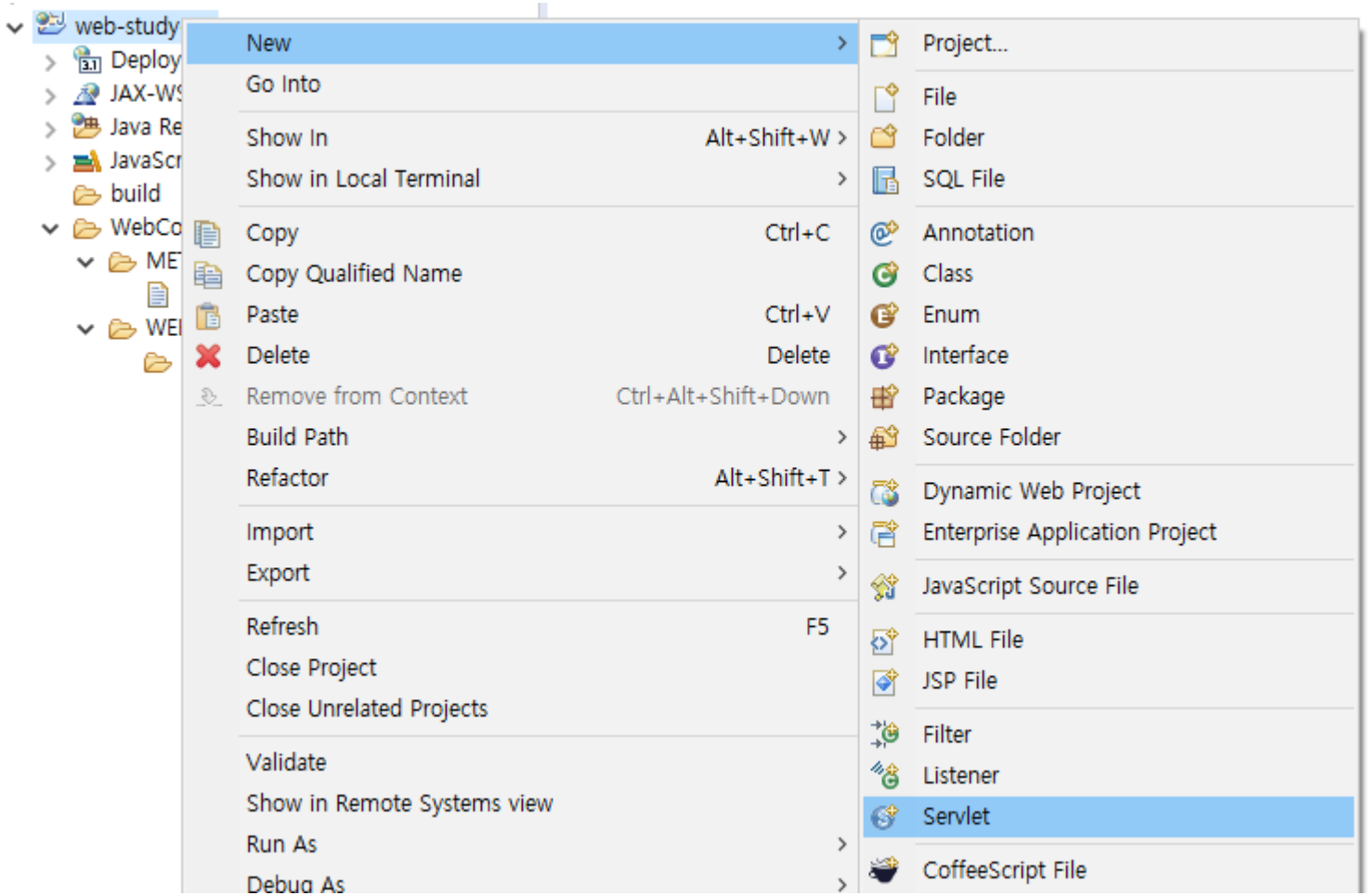
서블릿 프로그램

❖ 프로젝트 기본 구조



❖ 서블릿 생성

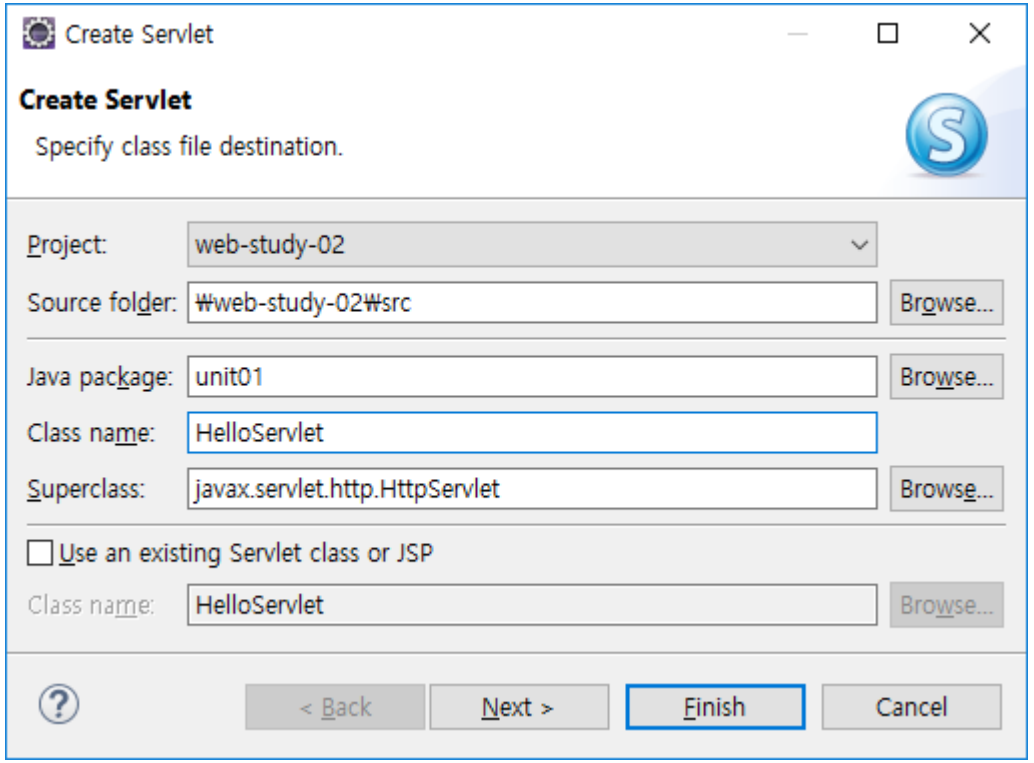
- ○ 프로젝트 선택 → New → Servlet



서블릿 프로그램

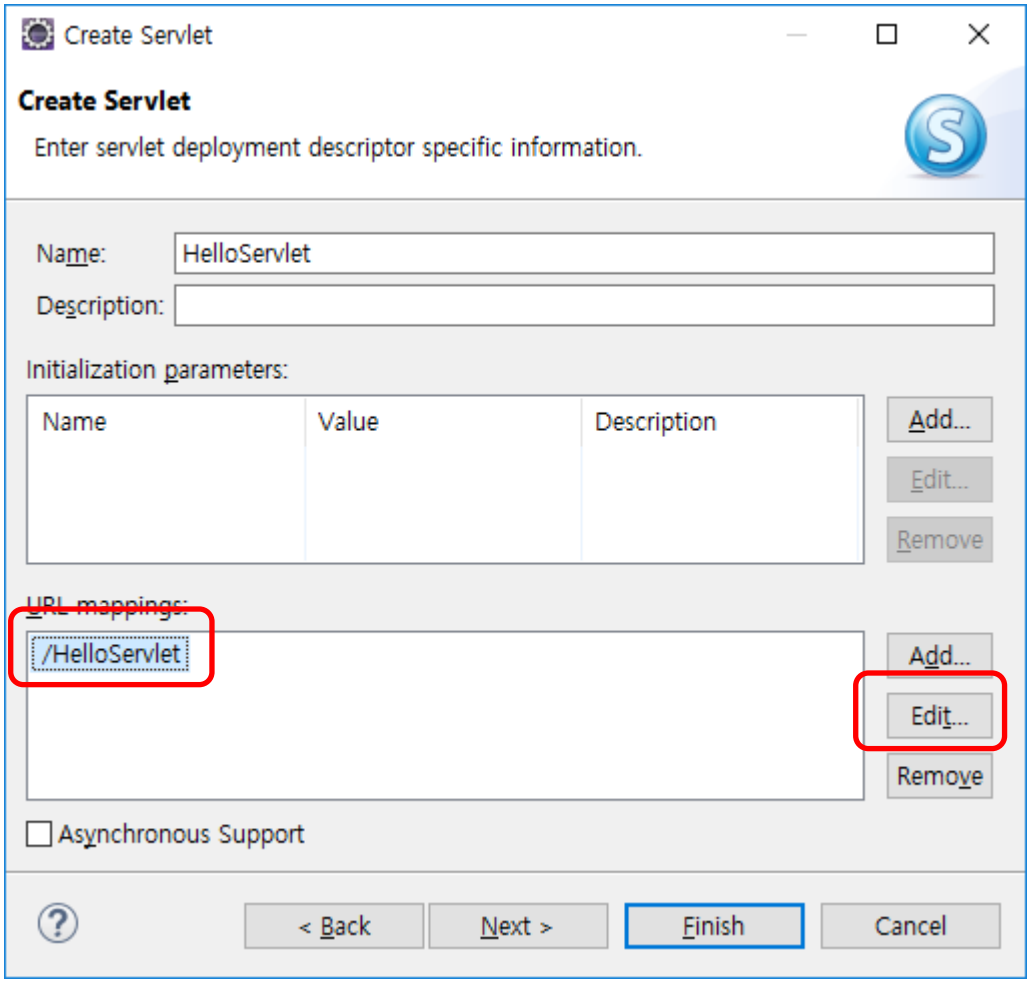
❖ 서블릿 생성

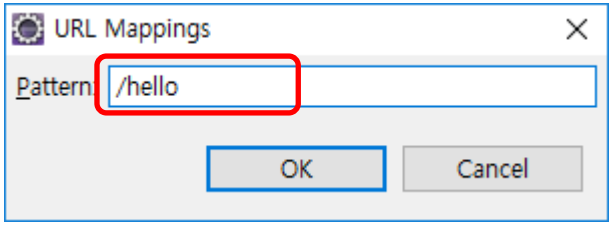
- Java package : unit01
- Class name : HelloServlet



❖ 서블릿 생성

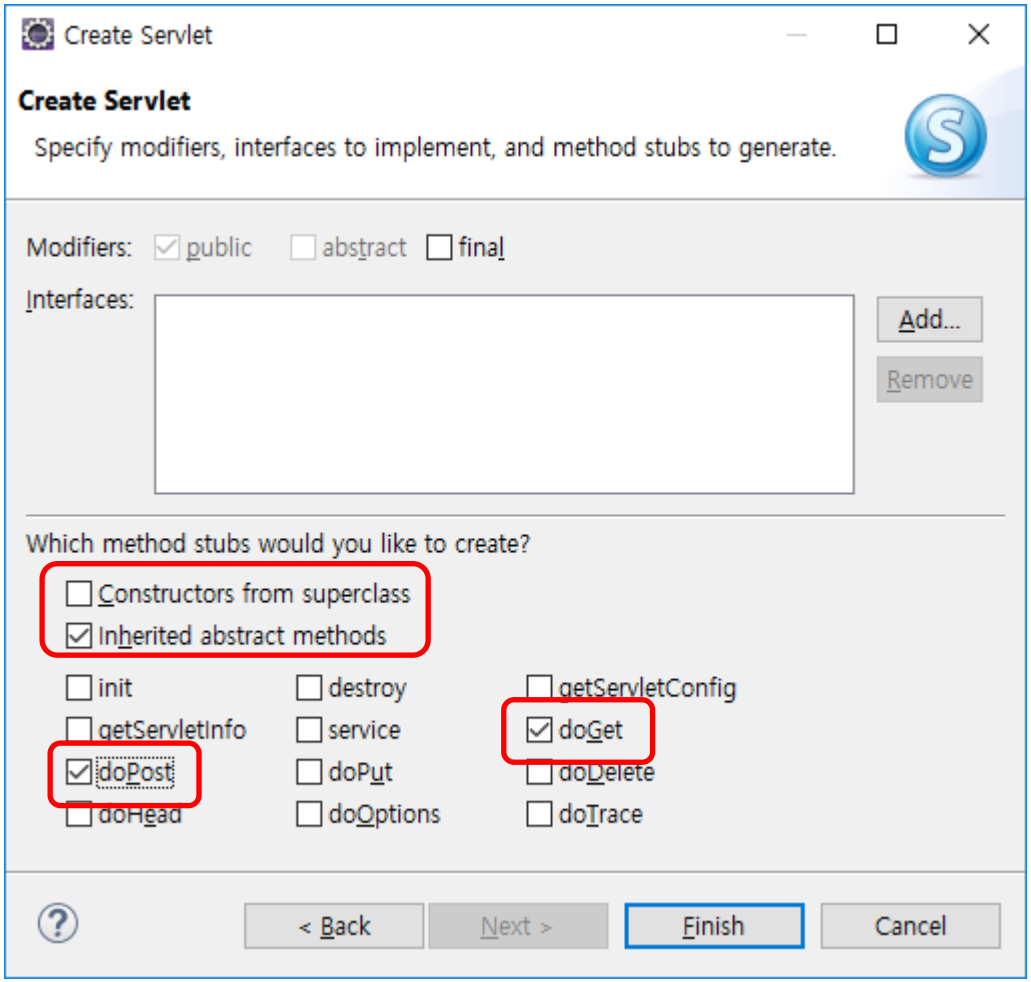
- URL 매핑명 변경 : /HelloServlet → /hello





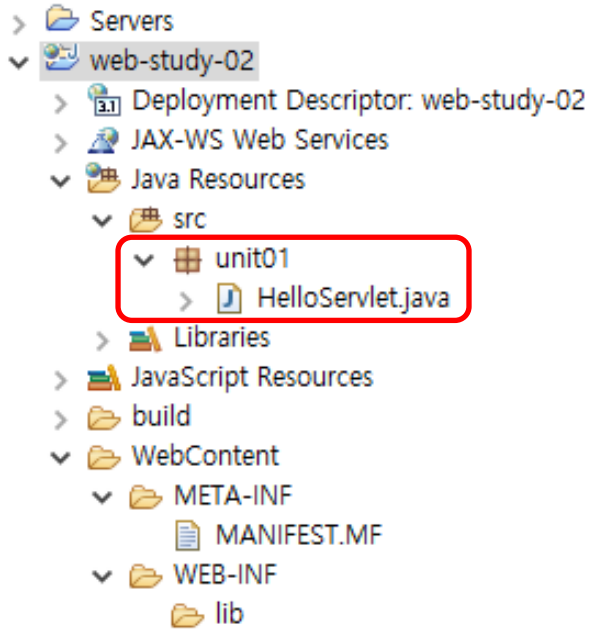
❖ 서블릿 생성

- ㅇ 요청 처리 메서드 선택



서블릿 프로그램

❖ 서블릿 생성



서블릿 프로그램

❖ 디폴트 서블릿 코드

```
package unit01;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

@WebServlet("/hello")
public class HelloServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;
```

❖ 디폴트 서블릿 코드

```
    protected void doGet(HttpServletRequest request,
                         HttpServletResponse response)
                  throws ServletException, IOException {

        // TODO Auto-generated method stub
        response.getWriter()
                .append("Served at: ")
                .append(request.getContextPath());
    }

    protected void doPost(HttpServletRequest request,
                         HttpServletResponse response)
                  throws ServletException, IOException {

        // TODO Auto-generated method stub
        doGet(request, response);
    }

}
```

❖ 서블릿 코드 수정

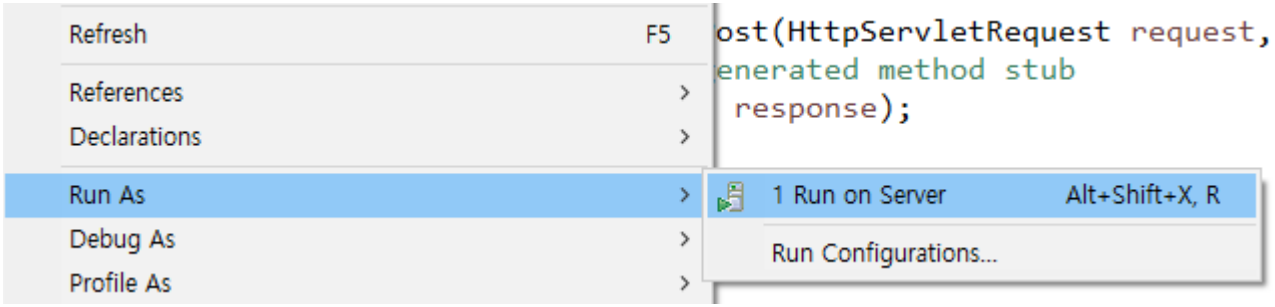
```
   :
   protected void doGet(HttpServletRequest request,
                        HttpServletResponse response)
                  throws ServletException, IOException {

      response.setContentType("text/html");
      PrintWriter out = response.getWriter();
      out.print("<html><body><h1>");
      out.print("Hello Servlet");
      out.print("</h1></body></html>");
      out.close();
   }
   :
}
```

서블릿 프로그램

❖ 실행

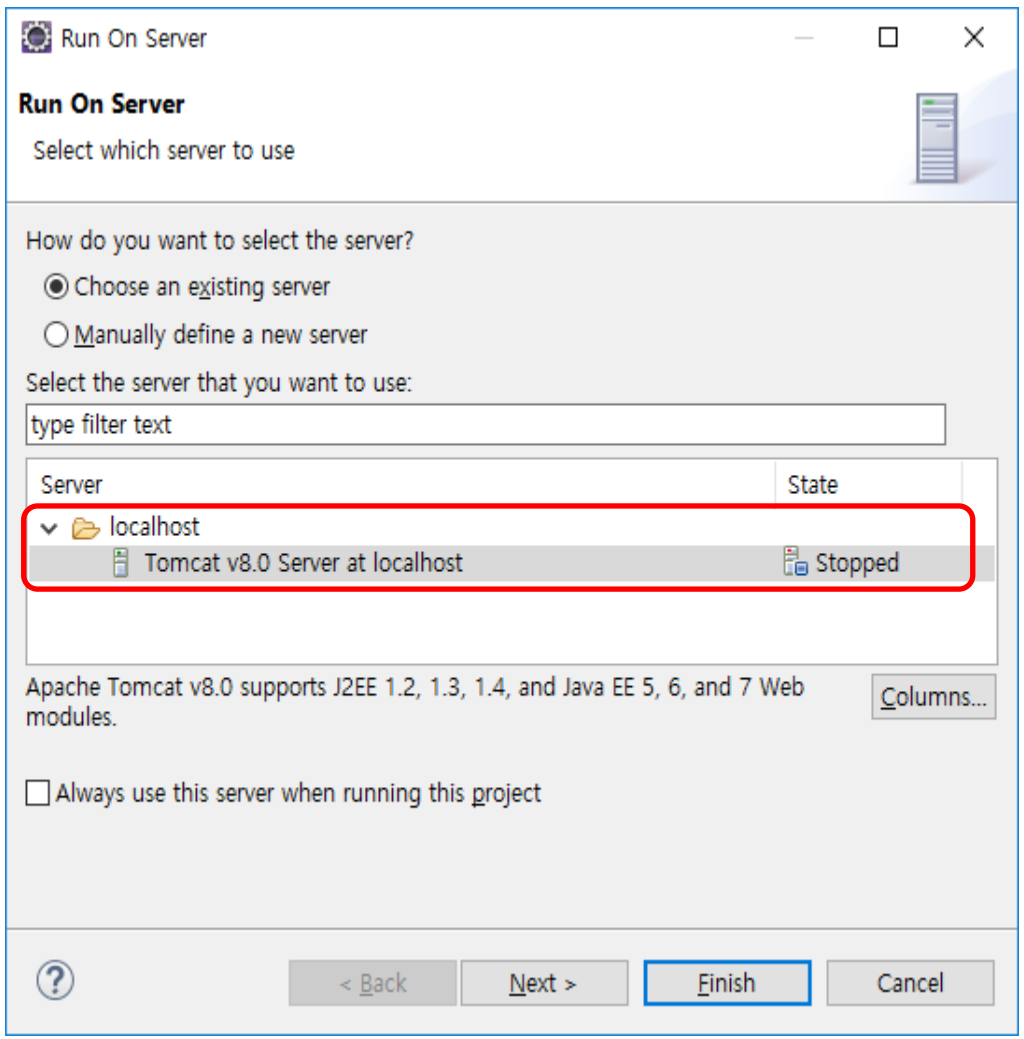
- 서블릿 클래스 선택 → Run As → Run On Server (단축키 : Ctrl+F11)

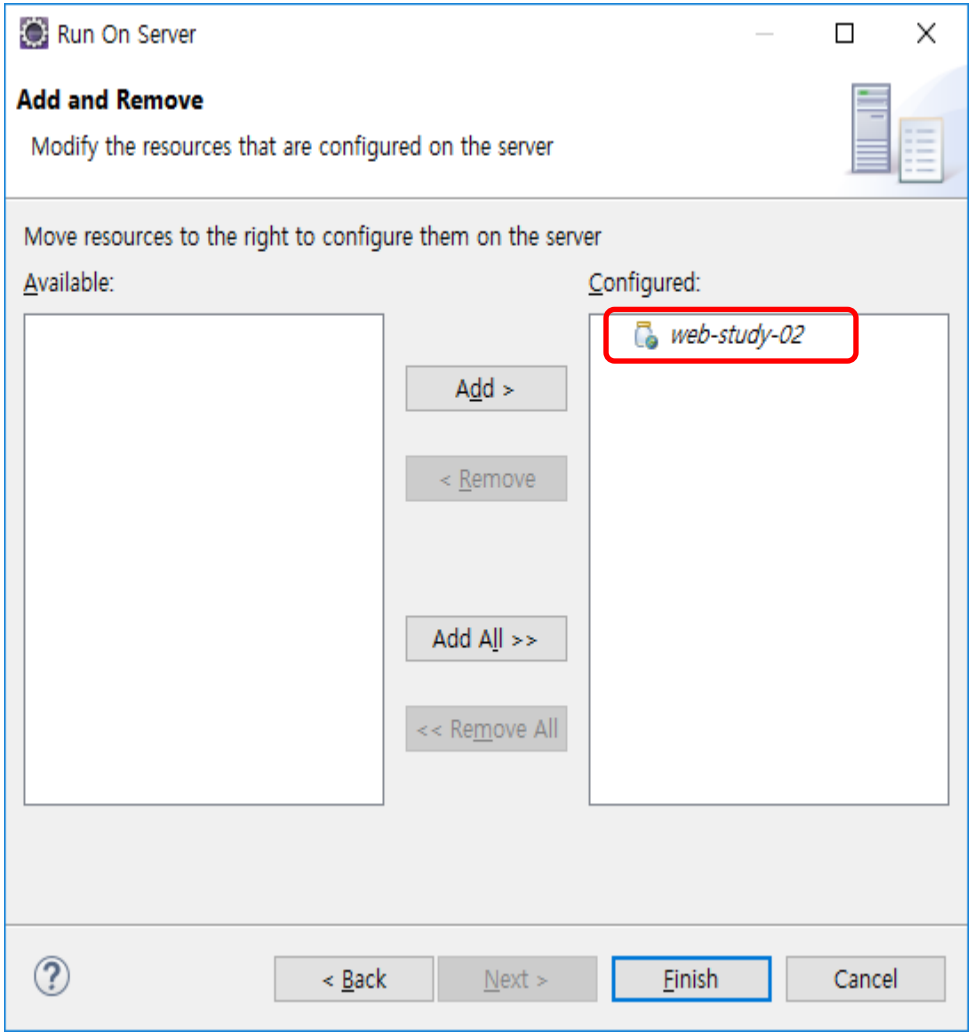


서블릿 프로그램

❖ 실행

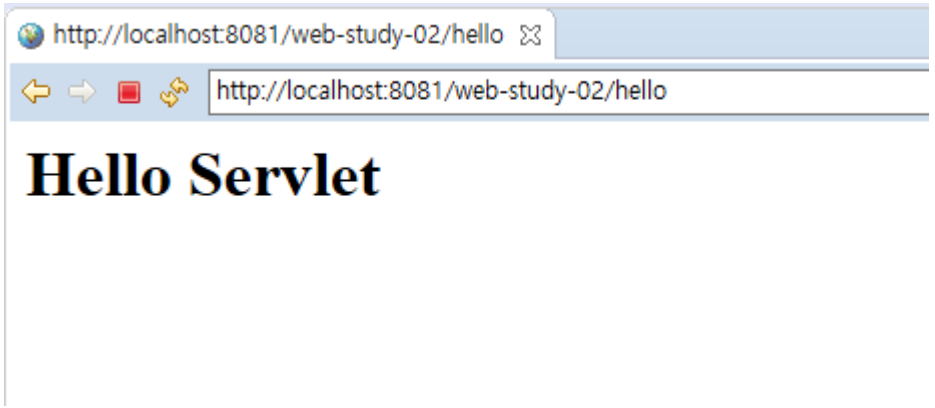
○ 서버 선택





서블릿 프로그램

❖ 실행



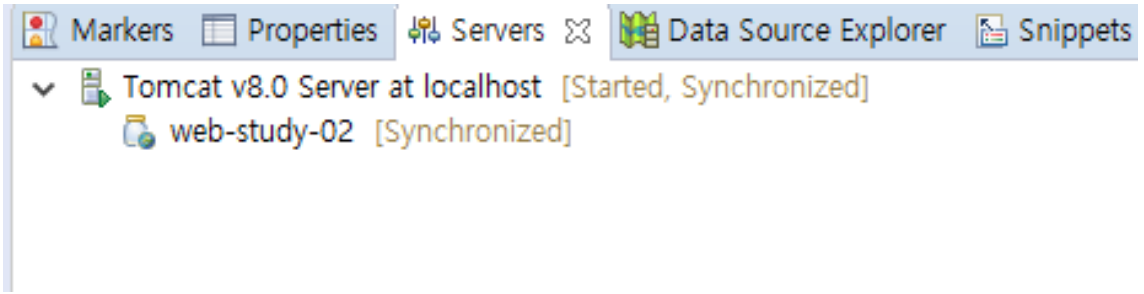
- url
  - http://localhost:포트번호/web-study-02/hello

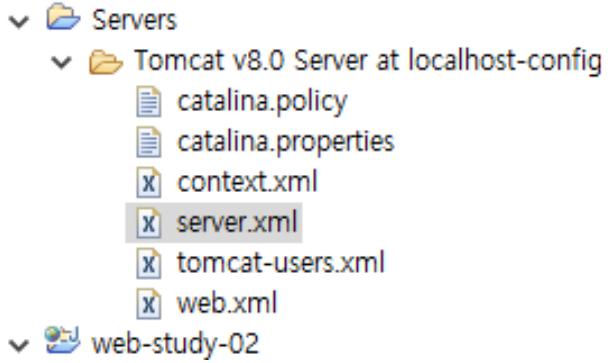
컨텍스트 패스 **서블릿 요청 URL 패턴**

- 컨텍스트 패스(Context Path)
  - 한 웹 서버에서 웹 애플리케이션을 구분

서블릿 프로그램

❖ 서버 환경 변화

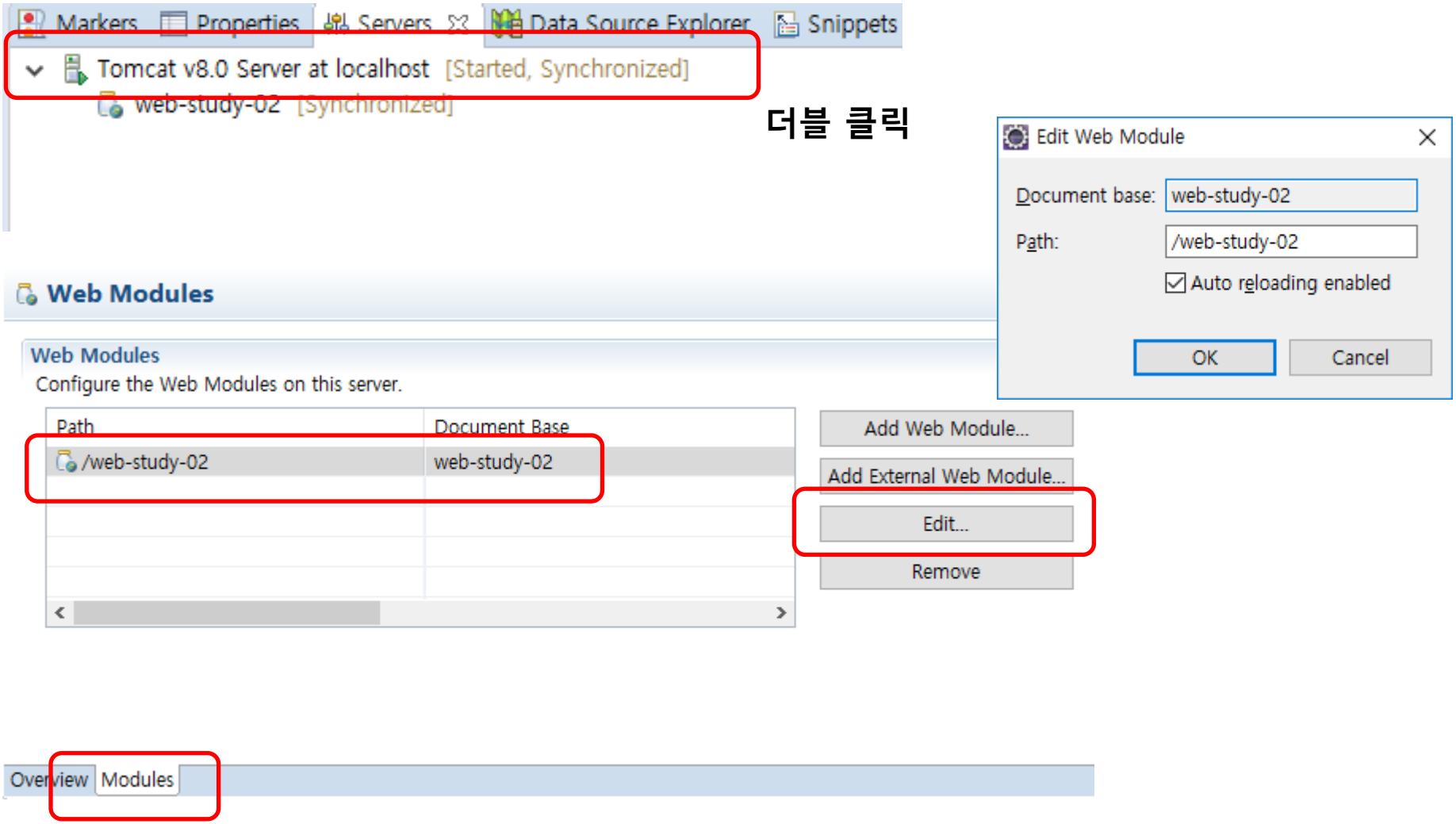





```
<?xml version="1.0" encoding="UTF-8"?>
    :

       <Context docBase="web-study-02"
           path="/web-study-02"
           reloadable="true"
    source="org.eclipse.jst.jee.server:web-study-02"/>
        </Host>
      </Engine>
    </Service>
</Server>
```

❖ 컨텍스트 패스 변경


Markers
Properties
Servers
Data Source Explorer
Snippets
Tomcat v8.0 Server at localhost [Started, Synchronized]
web-study-02 [Synchronized]
더블 클릭
Edit Web Module
Document base: web-study-02
Path: /web-study-02
Auto reloading enabled
OK
Cancel
Web Modules
Web Modules
Configure the Web Modules on this server.
Path
Document Base
/web-study-02
web-study-02
Add Web Module...
Add External Web Module...
Edit...
Remove
Overview
Modules

❖ @WebServlet 어노테이션

- URL 매핑 패턴 지정

```
package unit01;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

@WebServlet("/hello")
public class HelloServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;
```

서블릿 프로그램

❖ 서블릿 클래스 구조

❖ 요청 처리 메서드

- doGet
  - HTTP의 GET 요청 처리시 호출
- doPost
  - HTTP의 POST 요청 처리시 호출
- 매개변수
  - HttpServletRequest request : 요청 정보 처리
  - HttpServletResponse response : 응답 처리

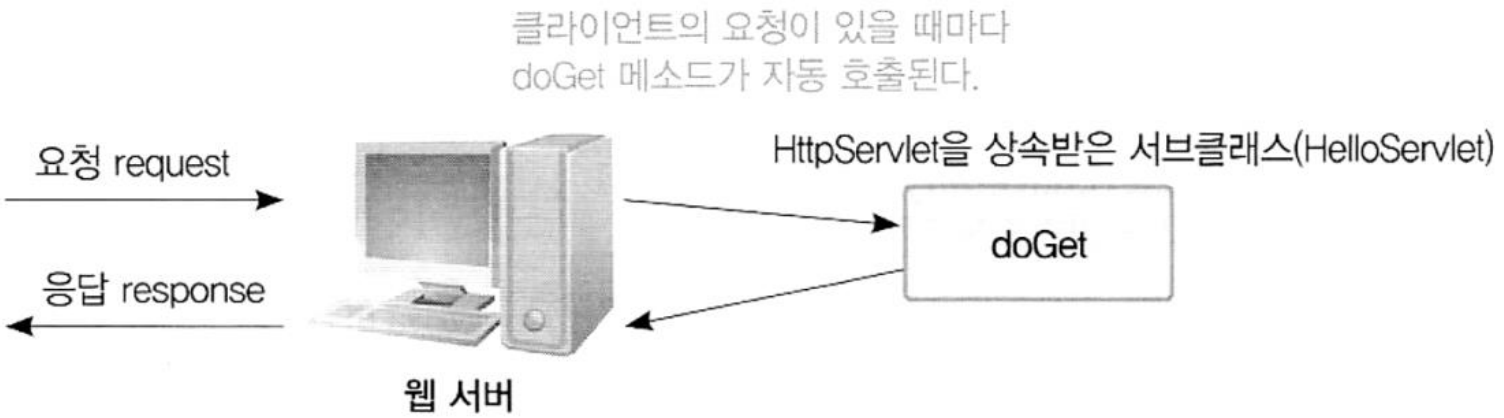
```
protected void doGet(HttpServletRequest request,
                     HttpServletResponse response)
                throws ServletException, IOException {
}

protected void doPost(HttpServletRequest request,
                     HttpServletResponse response)
                throws ServletException, IOException {

}
```

서블릿 프로그램

❖ 서블릿의 HTTP 요청 처리 과정



❖ 응답 내보내기

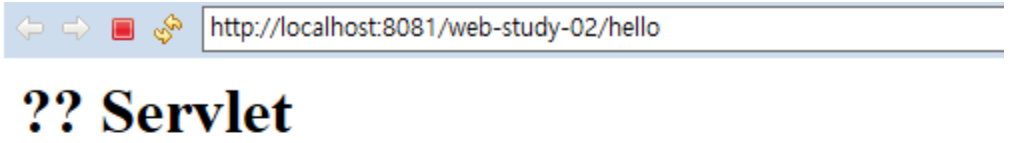
- ㅇ 응답 내용 형태 지정(MIME 타입 지정)
  - response.setContextType() 메서드로 지정

```
    :
    protected void doGet(HttpServletRequest request,
                         HttpServletResponse response)
                    throws ServletException, IOException {

        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.print("<html><body><h1>");
        out.print("안녕 Servlet");
        out.print("</h1></body></html>");
        out.close();
    }
    :
}
```

❖ 응답 내보내기

- ○ 응답 내용 형태 지정(MIME 타입 지정)
  - ▪ 한글 응답시 문자셋 문제 발생
    - 응답 문자셋은 UTF-8이지만 브라우저는 이를 모름
    - 디폴트 EUC-KR로 처리
  - ▪ 컨텐트 타입에 사용 문자셋을 지정




```
    :
    protected void doGet(HttpServletRequest request,
                         HttpServletResponse response)
                    throws ServletException, IOException {

        response.setContentType("text/html;charset=UTF-8");
        :
    }
    :
}
```

❖ 응답 내보내기

- 출력 스트림을 얻어 출력

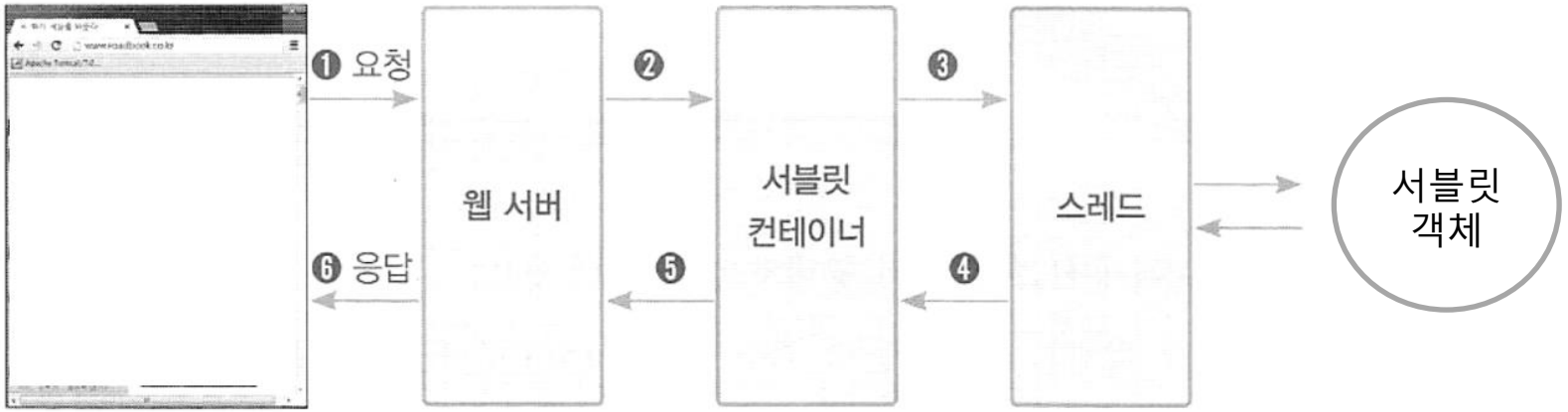
```
PrintWriter out = response.getWriter();
```

- 출력하기
  - response.setContentType("text/html;charset=UTF-8")으로 출력 형태를 html로 지정했으므로 html 코드를 작성하여 출력

```
out.print("<html><body><h1>");
out.print("안녕 Servlet");
out.print("</h1></body></html>");
out.close();
```

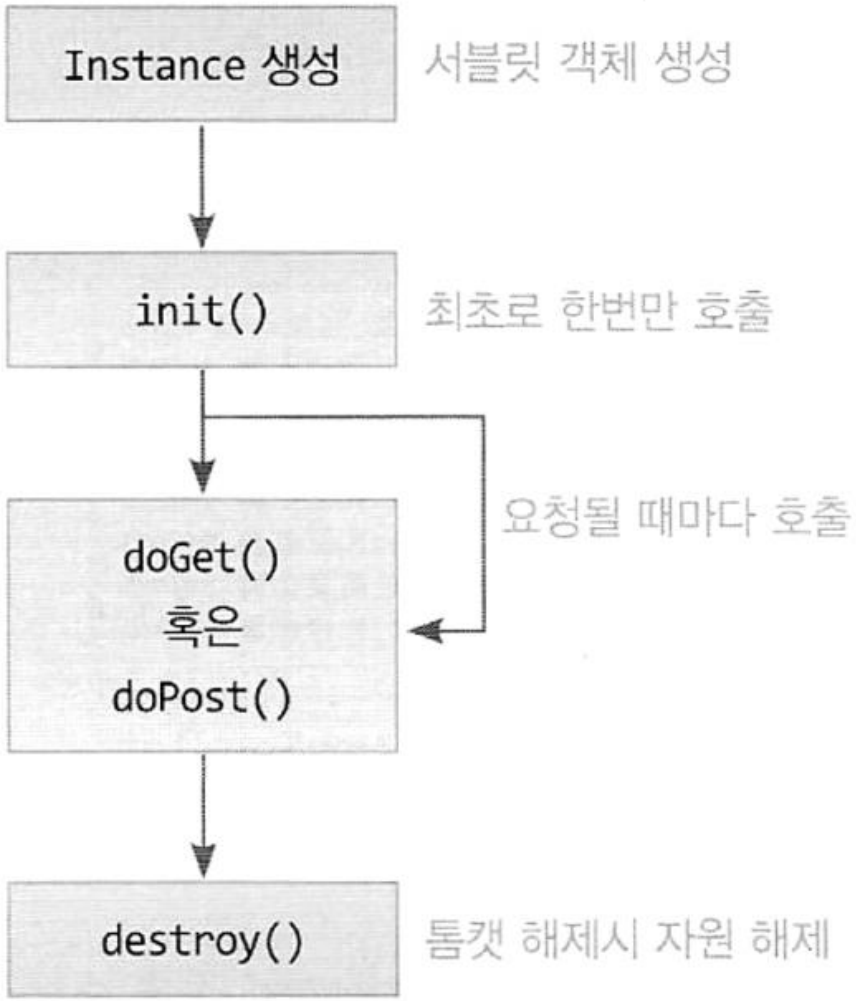
서블릿 프로그램

❖ 서블릿 작동 절차



서블릿 프로그램

❖ 서블릿 라이프 사이클



❖ 서블릿 라이프 사이클

- 서블릿 클래스 추가
  - 클래스명 : LifeCycle
  - URL 패턴 : /LifeCycle
  - 메서드 선택
    - init, destroy, doGet, doPost

```
@WebServlet("/LifeCycle")
public class LifeCycle extends HttpServlet {
    private static final long serialVersionUID = 1L;

    int initCount =1;
    int doGetCount = 1;
    int destroyCount =1;

    public LifeCycle() {
        super();
    }
```

서블릿 프로그램

```
    public void init(ServletConfig config) throws ServletException {
        System.out.println("init 메서드는 첫 요청만 호출됨 : " + initCount++);
    }

    public void destroy() {
        System.out.println("destroy 메서드는 톰캣이 종료될 때만 호출됨 : " +
                           destroyCount++);
    }

    protected void doGet(HttpServletRequest request,
                         HttpServletResponse response)
                  throws ServletException, IOException {

        System.out.println("doGet 메서드가 요청될 때마다 호출됨 : " +
                           doGetCount++);
    }
:
}
```