

Servlet & JSP

Part4-HTTP프로토콜

백 성 애

1. HTTP란?

- ▶ -HTTP는 웹서버와 클라이언트(브라우저)간에 통신하기 위한 프로토콜(규약)이다.
- ▶ - HTTP는 TCP/IP를 기반으로 한 지점에서 다른 지점으로 요청과 응답을 전송한다.
- ▶ - **HTTP구조는 요청/응답의 끊임 없는 주고 받음**이라고 할 수 있다.
- ▶ 웹브라우저와 웹서버간의 통신은 웹브라우저의 요청으로 부터 시작되는데, 브라우저는 HTTP프로토콜에 맞게 요청을 웹 서버에 전송하고,
- ▶ 웹서버는 이에 따라 응답을 HTTP프로토콜에 맞게 브라우저에 전송한다.

2. 요청 메시지 형태-요청라인

▶ 1) 요청라인

- ▶ <요청방식><요청자원><HTTP버전 정보><crLf>
- ▶ -웹브라우저는 웹서버가 사용하는 포트로 연결을 시도하고
- ▶ 요청이 이루어진 다음에 요청 메시지의 요청 라인을 통해
- ▶ 요청방식, 요청자원, HTTP버전 정보를 웹서버에 전달한다.
- ▶ ex) GET /index.jsp HTTP/1.1

2. 요청 메시지 형태-헤더부분

▶ 2) 헤더부분

- ▶ <헤더이름:헤더값><crLf>
- ▶ <헤더이름:헤더값><crLf>
- ▶
- ▶
- ▶ - 요청라인 다음에 웹클라이언트는 헤더정보, 구성정보, 받아들일 수 있는 문서 포맷 등 정보를 옵션으로 웹서버에 전달한다.
- ▶ 모든 헤더 정보는 한 줄에 하나씩만 기술되며, 헤더이름과 값의 쌍으로 구성된다.

2. 요청 메시지 형태-헤더부분

- ▶ 헤더 정보 끝에는 공백라인 <cr>lf>을 추가하여 헤더 끝임을 알려준다.
- ▶ ex) User-Agent:Mozilla/2.03Gold(WinNT:1)
- ▶ Accept: text/html, image/gif....

2. 요청 메시지 형태- 엔티티 바디

- ▶ - 요청과 헤더 정보를 보낸 다음에 웹클라이언트는 추가적인 정보를 웹서버에 보낼 수 있다.
- ▶ - 요청방식이 POST인 경우 데이터가 여기에 포함되어 웹서버에 전달된다.
- ▶ - GET방식인 경우에는 바디 부분이 빈상태로 전달된다.

3. 응답 메시지 형태

```
C:\> C:\WINDOWS\System32\cmd.exe

HTTP/1.1 200 OK
Content-type: text/html
Content-Length: 6680
Date: Tue, 17 Jan 2006 11:58:27 GMT
ETag: "6680-1000723040000"
Server: Apache Tomcat/4.0 (HTTP/1.1 Connector)
Last-Modified: Mon, 17 Sep 2001 10:37:20 GMT

<!doctype html public "-//w3c//dtd html 4.0 tra
/TR/REC-html40/strict.dtd">
                                <html>
                                    <head>
content="text/html; charset=iso-8859-1">
                                    <tit
```

3. 응답 메시지 형태-응답라인

- ▶ - 웹클라이언트로부터 요청받은 웹서버는 요청에 맞는 작업을 수행하고, 응답을 크게 세 부분으로 구성하여 전달한다.
- ▶
- ▶ 1) 응답라인
- ▶ **<HTTP버전><상태코드><설명><crLf>**
- ▶
- ▶ -http버전은 웹서버가 사용하는 http버전을 의미
- ▶ - 상태코드는 클 요청에 따른 서버 결과를 세자리 숫자로 표현
 - ▶ ex) HTTP/1.1 200 OK "

3. 응답 메시지 형태-응답라인

- ▶ 1xx : 정보 -요청이 받아들여졌고 처리중임.
- ▶ 2xx : 성공 - 요청이 성공적으로 처리됨
- ▶ 3xx : 경로변경 - 요청을 완전히 처리하기 위해
추가적인 액션이 수행되어야 함
- ▶ 4xx : 클라이언트 에러 -요청 문법 오류 또는
요청이 처리될 수 없음
- ▶ 5xx : 서버 에러
 - ▶ - 클의 정당한 요청을 서버가 처리하지 못함

3. 응답 메시지 형태- 헤더부분

- ▶ 2) 헤더부분

- ▶ <헤더이름:헤더값><crLf>

- ▶

- ▶ <헤더이름:헤더값><crLf>

- ▶

- ▶ -응답라인 다음에 서버는 자신과 요청된 자원에 대한 정보를 헤더 정보를 통해 전달한다.

3. 응답 메시지 형태- 엔티티바디부분

▶ 3) 엔티티바디



- ▶ - 클 요청이 정상적으로 수행됐다면 요청 자료가 이 부분에 포함되어 전송된다.
- ▶ [자료는 파일내용일 수도 있고 CGI, 서블릿 등의 결과물일 수도 있다.]

4. 텔넷으로 웹서버와 HTTP통신

- ▶ **[실습]**
- ▶ 도스창에서
- ▶ `c:\Windows\System32>telnet localhost 9090`
- ▶ `GET /index.jsp HTTP/1.0`
- ▶ 엔터 두번치기
- ▶ `GET /MyWeb/HelloServlet HTTP/1.0`
- ▶ 엔터 두번치기
- ▶ `GET /servlet/GetPostServlet?name=Tom&addr=LA
HTTP/1.0`
- ▶ 엔터 두번치기
- ▶ -----

4. 텔넷으로 웹서버와 HTTP통신

- ▶ 텔넷을 통해 http프로토콜 동작방식을 알아보자.
- ▶ 아래 순서 대로 실행해보자.
- ▶ 1. 톰캣을 startup한다.
- ▶ 2. 도스창을 열어 텔넷으로 웹서버에 접속 시도
- ▶ 3. 텔넷에 연결되면 명령창에 다음과 같이 입력한후 엔터를 두번 친다.
GET /index.html HTTP/1.0
- ▶
- ▶ ...이 때 주의 텔넷을 이용해 테스트할때 입력한 내용이
텔넷창에 보이지 않는다..그래도 묵묵히 위와같이 입력한 뒤
엔터를 두번 쳐주면 응답이 올 것이다. (대소문자도 주의하자)
- ▶
- ▶ 위 예제는 서버에 index.html 파일을 get방식으로 요청하는 예제
로, 서버에 index.html파일이 존재해야 아래와 같은 응답을
받을 수 있다.
- ▶
- ▶ 4. 드디어 서버에서 응답이 왔다. 아래처럼...

4. 텔넷으로 웹서버와 HTTP통신



A screenshot of a Windows command prompt window titled "C:\WINDOWS\System32\cmd.exe". The window displays the output of a Telnet session to a web server. The first line, "HTTP/1.1 200 OK", is highlighted with a red box. Subsequent lines, including "Content-type: text/html", "Content-Length: 6680", "Date: Tue, 17 Jan 2006 11:58:27 GMT", "ETag: '6680-1000723040000'", "Server: Apache Tomcat/4.0 (HTTP/1.1 Connector)", and "Last-Modified: Mon, 17 Sep 2001 10:37:20 GMT", are enclosed in a yellow box. Below these headers, the beginning of an HTML document is shown, including the DOCTYPE declaration and the opening tags for <html>, <head>, and <title>.

```
C:\WINDOWS\System32\cmd.exe

HTTP/1.1 200 OK
Content-type: text/html
Content-Length: 6680
Date: Tue, 17 Jan 2006 11:58:27 GMT
ETag: "6680-1000723040000"
Server: Apache Tomcat/4.0 (HTTP/1.1 Connector)
Last-Modified: Mon, 17 Sep 2001 10:37:20 GMT

<!doctype html public "-//w3c//dtd html 4.0 tr
/TR/REC-html40/strict.dtd">
    <html>
        <head>
            content="text/html; charset=iso-8859-1">
                <ti
```

참고 도서**

웹 개발자를 위한 서블릿 JSP -프리렉