스프링의 정석

13강. 서블릿과 JSP(1) ~ 16강.서블릿과 JSP(4)

```java
@Controller
public class TwoDice {
    @RequestMapping("/rollDice")
    public void main(HttpServletResponse response) throws IOException {
        int idx1 = (int)(Math.random()*6)+1;
        int idx2 = (int)(Math.random()*6)+1;

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```

[Controller]

```java
@WebServlet("/rollDice2")
public class TwoDiceServlet extends HttpServlet {
    int getRandomInt(int range) { return new Random().nextInt(range)+1; }
    public void service(HttpServletRequest request, HttpServletResponse response) throws IOException {
        int idx1 = getRandomInt(6);
        int idx2 = getRandomInt(6);

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```

[Servlet]

```java
@Controller
public class TwoDice {
    @RequestMapping("/rollDice")
    public void main(HttpServletResponse response) throws IOException {
        int idx1 = (int)(Math.random()*6)+1;
        int idx2 = (int)(Math.random()*6)+1;

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```

[Controller]

```java
@WebServlet("/rollDice2")
public class TwoDiceServlet extends HttpServlet {
    int getRandomInt(int range) { return new Random().nextInt(range)+1; }
    public void service(HttpServletRequest request, HttpServletResponse response) throws IOException {
        int idx1 = getRandomInt(6);
        int idx2 = getRandomInt(6);

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```

[Servlet]

```java
@Controller
public class TwoDice {
    @RequestMapping("/rollDice")
    public void main(HttpServletResponse response) throws IOException {
        int idx1 = (int)(Math.random()*6)+1;
        int idx2 = (int)(Math.random()*6)+1;

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```

[Controller]

```java
@WebServlet("/rollDice2")
public class TwoDiceServlet extends HttpServlet {
    int getRandomInt(int range) { return new Random().nextInt(range)+1; }
    public void service(HttpServletRequest request, HttpServletResponse response) throws IOException {
        int idx1 = getRandomInt(6);
        int idx2 = getRandomInt(6);

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```

[Servlet]

```
@Controller
public class TwoDice {
    @RequestMapping("/rollDice")
    public void main(HttpServletResponse response) throws IOException {
        int idx1 = (int)(Math.random()*6)+1;
        int idx2 = (int)(Math.random()*6)+1;

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```
[Controller]

```
@WebServlet("/rollDice2")
public class TwoDiceServlet extends HttpServlet {
    int getRandomInt(int range) { return new Random().nextInt(range)+1; }
    public void service(HttpServletRequest request, HttpServletResponse response) throws IOException {
        int idx1 = getRandomInt(6);
        int idx2 = getRandomInt(6);

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```
[Servlet]

```
@Controller
public class TwoDice {
    @RequestMapping("/rollDice")
    public void main(HttpServletResponse response) throws IOException {
        int idx1 = (int)(Math.random()*6)+1;
        int idx2 = (int)(Math.random()*6)+1;

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```
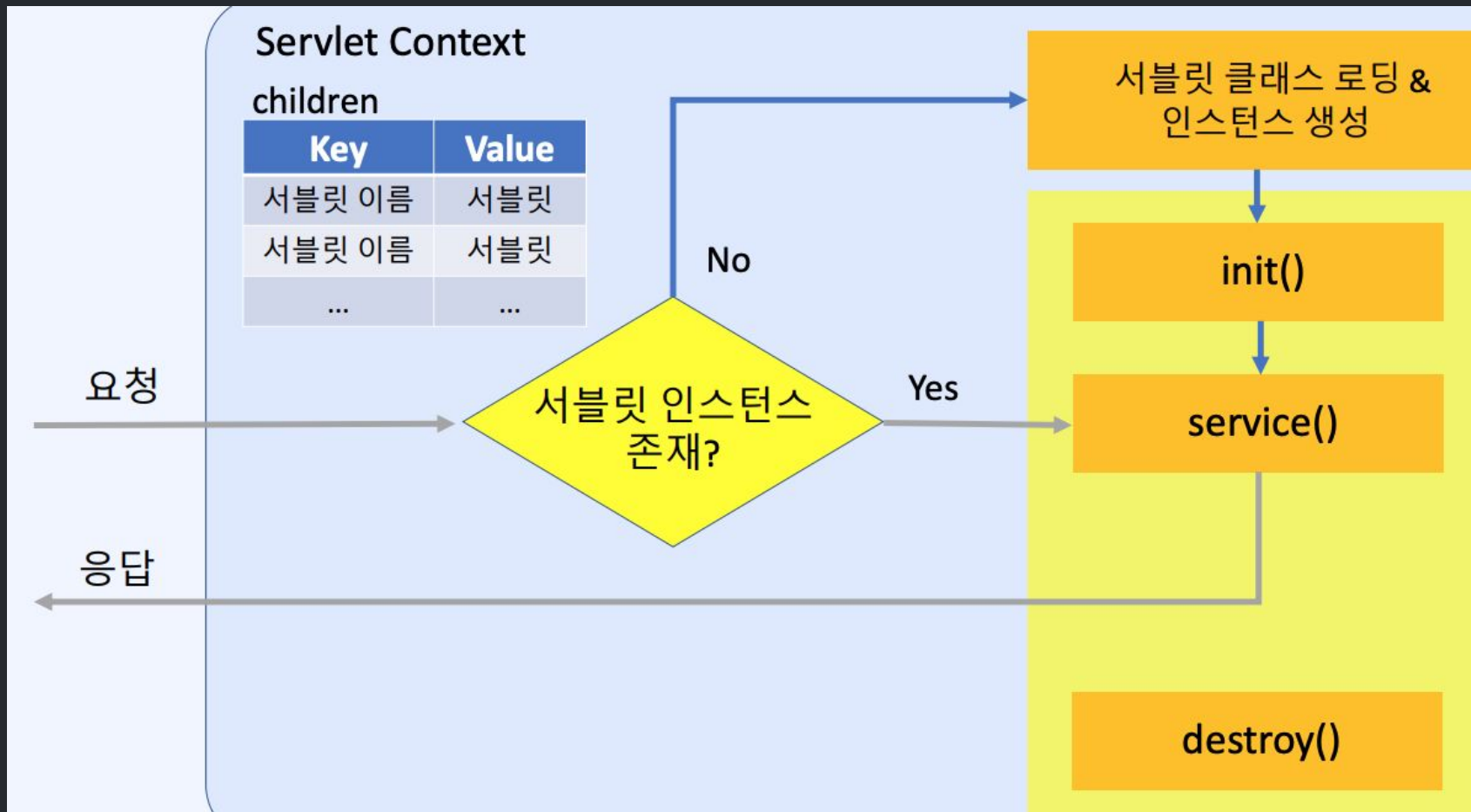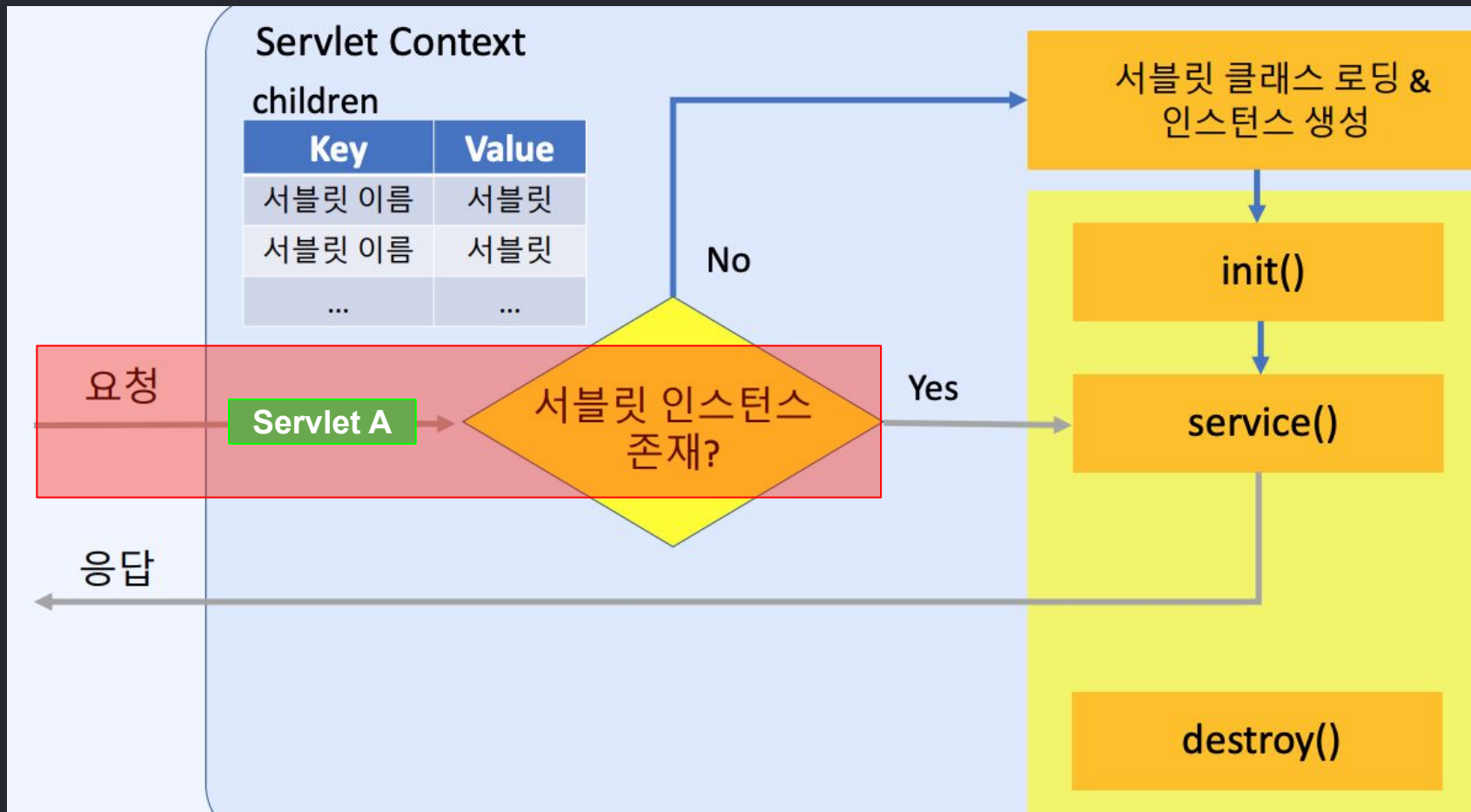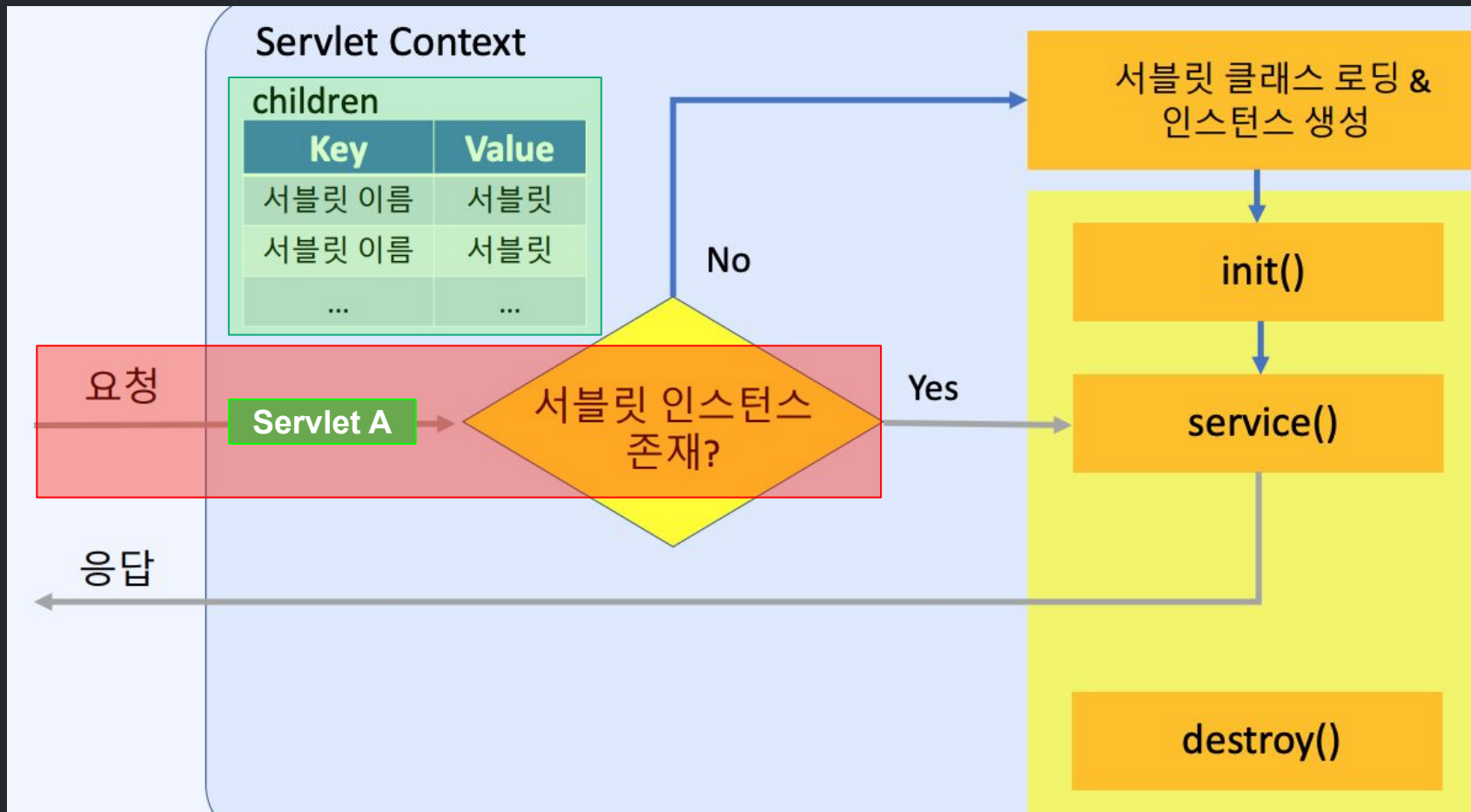
[Controller]

```
@WebServlet("/rollDice2")
public class TwoDiceServlet extends HttpServlet {
    int getRandomInt(int range) { return new Random().nextInt(range)+1; }
    public void service(HttpServletRequest request, HttpServletResponse response) throws IOException {
        int idx1 = getRandomInt(6);
        int idx2 = getRandomInt(6);

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```
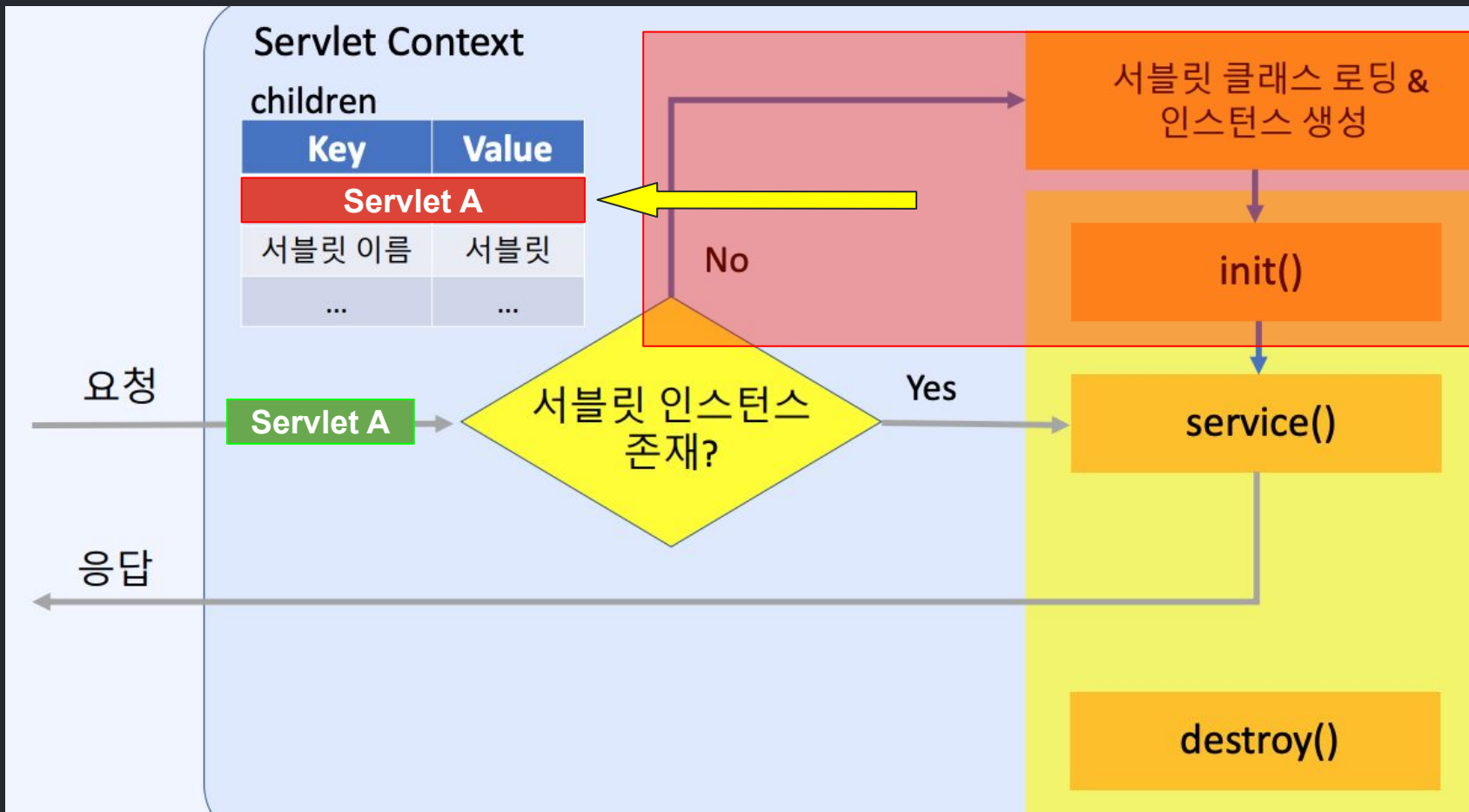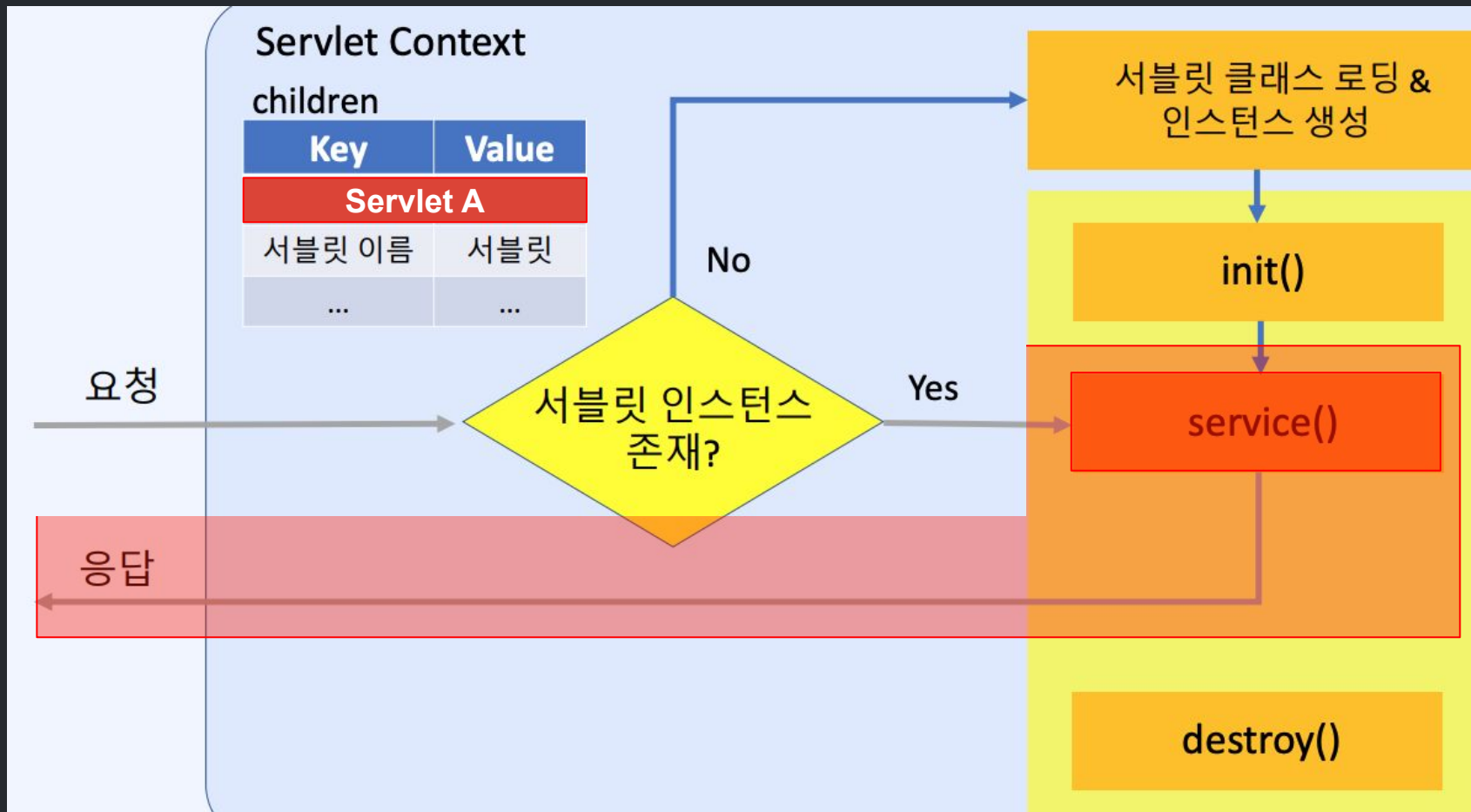
[Servlet]

Servlet Context

children

Key	Value
서블릿 이름	서블릿
서블릿 이름	서블릿
...	...

요청

Servlet A

서블릿 인스턴스 존재?

No

Yes

응답

서블릿 클래스 로딩 & 인스턴스 생성

init()

service()

destroy()

Servlet Context

children

Key	Value
~~Servlet A~~	
서블릿 이름	서블릿
...	...

요청

응답

서블릿 인스턴스 존재?

No

Yes

서블릿 클래스 로딩 & 인스턴스 생성

init()

service()

destroy()

```jsp
<%@ page contentType="text/html;charset=utf-8"%>
<%@ page import="java.util.Random" %>

<%-- <%! 클래스 영역 %> --%>
<%!
    int getRandomInt(int range){
        return new Random().nextInt(range)+1;
    }
%>


<%-- <%  메서드 영역 - service()의 내부 %> --%>
<%
    int idx1 = getRandomInt(6);
    int idx2 = getRandomInt(6);
%>
<html>
<head>
    <title>twoDice.jsp</title>
</head>
<body>
    <img src='resources/img/dice<%=idx1%>.jpg'>
    <img src='resources/img/dice<%=idx2%>.jpg'>
</body>
</html>
```

```java
@WebServlet("/rollDice2")
public class TwoDiceServlet extends HttpServlet {
    int getRandomInt(int range) {
        return new Random().nextInt(range)+1;
    }

    public void service(HttpServletRequest request, H
        int idx1 = getRandomInt(6);
        int idx2 = getRandomInt(6);

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```
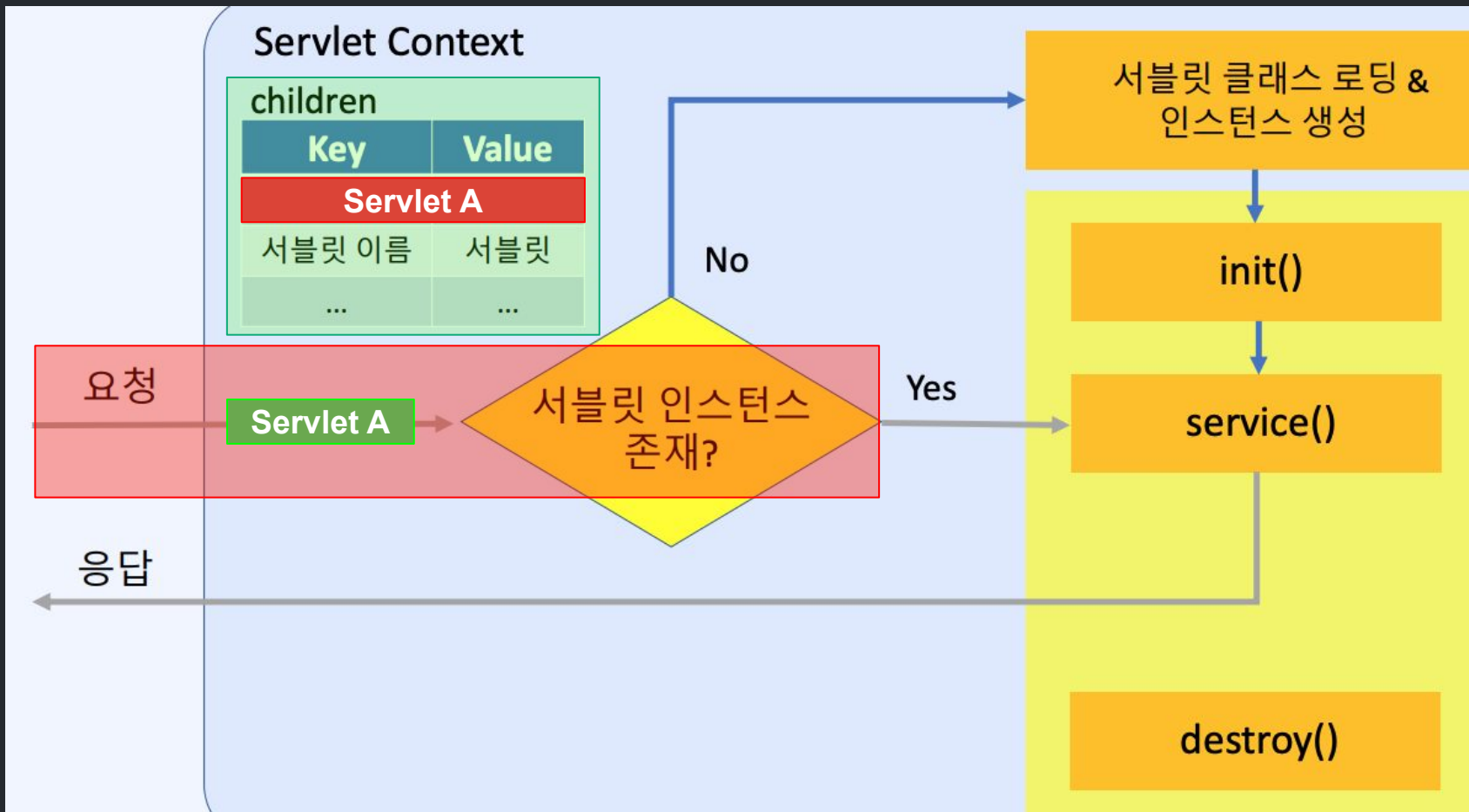
<% %>

```
<%@ page contentType="text/html;charset=utf-8"%>
<%@ page import="java.util.Random" %>

<%-- <%! 클래스 영역 %> --%>
<%!
    int getRandomInt(int range){
        return new Random().nextInt(range)+1;
    }
%>


<%-- <% 메서드 영역 - service()의 내부 %> --%>
<%
    int idx1 = getRandomInt(6);
    int idx2 = getRandomInt(6);
%>
<html>
<head>
    <title>twoDice.jsp</title>
</head>
<body>
    <img src='resources/img/dice<%=idx1%>.jpg'>
    <img src='resources/img/dice<%=idx2%>.jpg'>
</body>
</html>
```
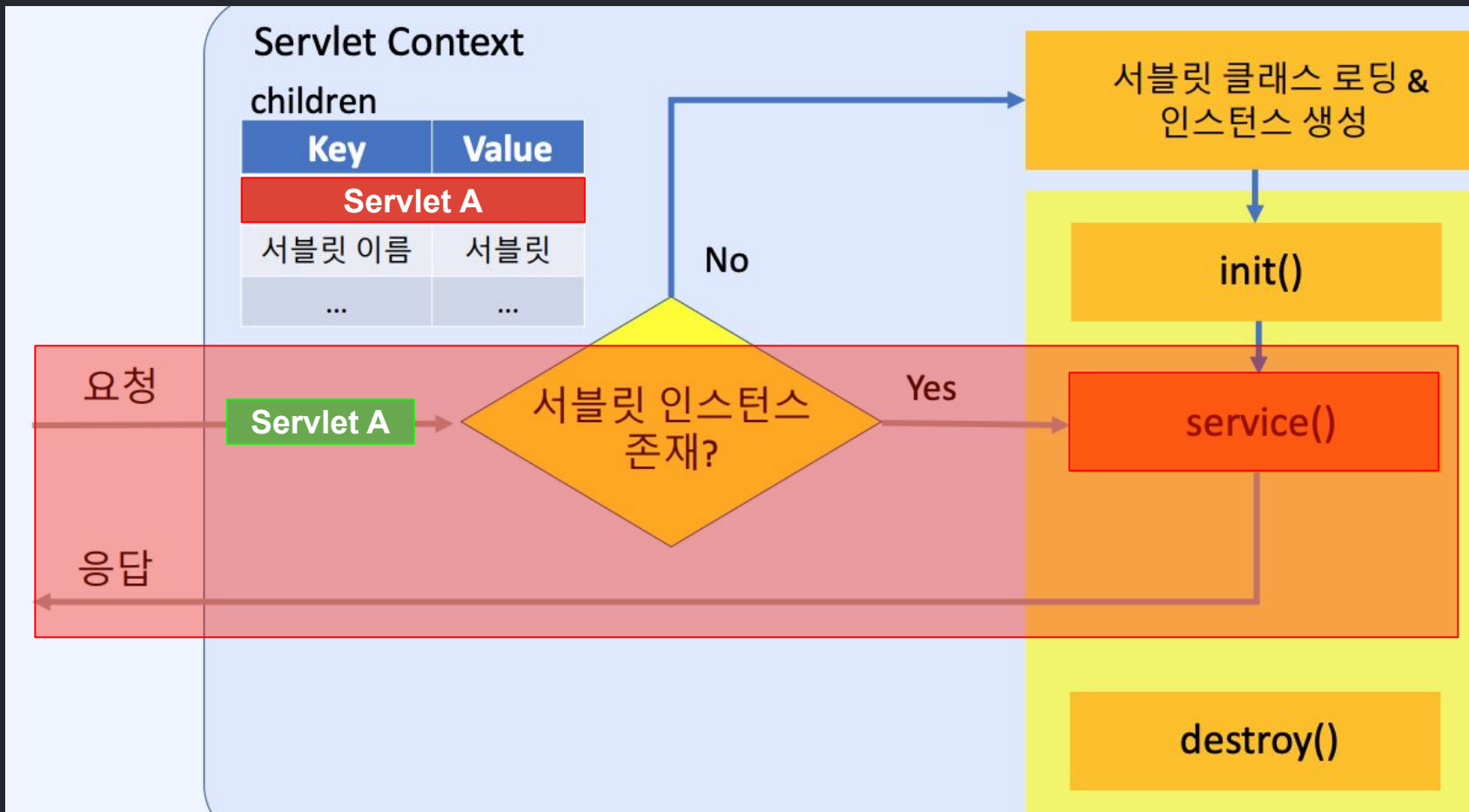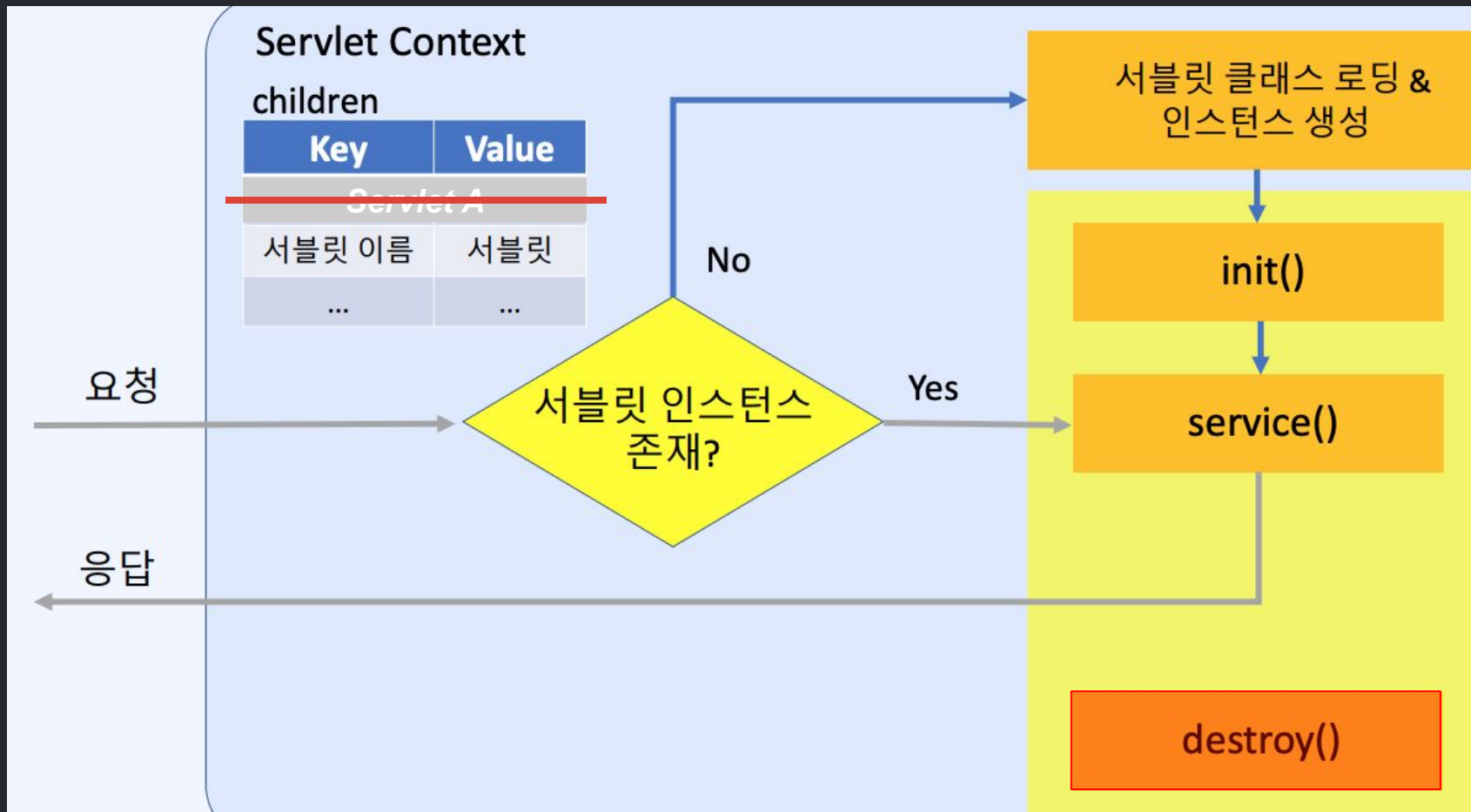
```
@WebServlet("/rollDice2")
public class TwoDiceServlet extends HttpServlet {
    int getRandomInt(int range) {
        return new Random().nextInt(range)+1;
    }

    public void service(HttpServletRequest request, H
        int idx1 = getRandomInt(6);
        int idx2 = getRandomInt(6);

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```

<%@ page %>

```jsp
<%@ page contentType="text/html;charset=utf-8"%>
<%@ page import="java.util.Random" %>

<%-- <%! 클래스 영역 %> --%>
<%!
    int getRandomInt(int range){
        return new Random().nextInt(range)+1;
    }
%>


<%-- <% 메서드 영역 - service()의 내부 %> --%>
<%
    int idx1 = getRandomInt(6);
    int idx2 = getRandomInt(6);
%>
<html>
<head>
    <title>twoDice.jsp</title>
</head>
<body>
    <img src='resources/img/dice<%=idx1%>.jpg'>
    <img src='resources/img/dice<%=idx2%>.jpg'>
</body>
</html>
```

```java
@WebServlet("/rollDice2")
public class TwoDiceServlet extends HttpServlet {
    int getRandomInt(int range) {
        return new Random().nextInt(range)+1;
    }

    public void service(HttpServletRequest request, H
        int idx1 = getRandomInt(6);
        int idx2 = getRandomInt(6);

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```

<%@ page %>

```jsp
<%@ page contentType="text/html;charset=utf-8"%>
<%@ page import="java.util.Random" %>

<%-- <%! 클래스 영역 %> --%>
<%!
    int getRandomInt(int range){
        return new Random().nextInt(range)+1;
    }
%>


<%-- <%  메서드 영역 – service()의 내부 %> --%>
<%
    int idx1 = getRandomInt(6);
    int idx2 = getRandomInt(6);
%>
<html>
<head>
    <title>twoDice.jsp</title>
</head>
<body>
    <img src='resources/img/dice<%=idx1%>.jpg'>
    <img src='resources/img/dice<%=idx2%>.jpg'>
</body>
</html>
```

```java
@WebServlet("/rollDice2")
public class TwoDiceServlet extends HttpServlet {
    int getRandomInt(int range) {
        return new Random().nextInt(range)+1;
    }

    public void service(HttpServletRequest request, H
        int idx1 = getRandomInt(6);
        int idx2 = getRandomInt(6);

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```

<%@ page %>

```jsp
<%@ page contentType="text/html;charset=utf-8"%>
<%@ page import="java.util.Random" %>

<%-- <%! 클래스 영역 %> --%>
<%!
    int getRandomInt(int range){
        return new Random().nextInt(range)+1;
    }
%>


<%-- <% 메서드 영역 - service()의 내부 %> --%>
<%
    int idx1 = getRandomInt(6);
    int idx2 = getRandomInt(6);
%>
<html>
<head>
    <title>twoDice.jsp</title>
</head>
<body>
    <img src='resources/img/dice<%=idx1%>.jpg'>
    <img src='resources/img/dice<%=idx2%>.jpg'>
</body>
</html>
```

```java
import java.util.Random;

@WebServlet("/rollDice2")
public class TwoDiceServlet extends HttpServlet {
    int getRandomInt(int range) {
        return new Random().nextInt(range)+1;
    }

    public void service(HttpServletRequest request, H
        int idx1 = getRandomInt(6);
        int idx2 = getRandomInt(6);

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```

```jsp
<%@ page contentType="text/html;charset=utf-8"%>
<%@ page import="java.util.Random" %>

<%-- <%! 클래스 영역 %> --%>
<%!
    int getRandomInt(int range){
        return new Random().nextInt(range)+1;
    }
%>
```

<%! %>

```jsp
<%-- <% 메서드 영역 - service()의 내부 %> --%>
<%
    int idx1 = getRandomInt(6);
    int idx2 = getRandomInt(6);
%>
<html>
<head>
    <title>twoDice.jsp</title>
</head>
<body>
    <img src='resources/img/dice<%=idx1%>.jpg'>
    <img src='resources/img/dice<%=idx2%>.jpg'>
</body>
</html>
```

```java
@WebServlet("/rollDice2")
public class TwoDiceServlet extends HttpServlet {
    int getRandomInt(int range) {
        return new Random().nextInt(range)+1;
    }

    public void service(HttpServletRequest request, H
        int idx1 = getRandomInt(6);
        int idx2 = getRandomInt(6);

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```

```jsp
<%@ page contentType="text/html;charset=utf-8"%>
<%@ page import="java.util.Random" %>

<%-- <%! 클래스 영역 %> --%>
<%!
    int getRandomInt(int range){
        return new Random().nextInt(range)+1;
    }
%>

<%-- <% 메서드 영역 – service()의 내부 %> --%>
<%
    int idx1 = getRandomInt(6);
    int idx2 = getRandomInt(6);
%>                                        <% %>
<html>
<head>
    <title>twoDice.jsp</title>
</head>
<body>
    <img src='resources/img/dice<%=idx1%>.jpg'>
    <img src='resources/img/dice<%=idx2%>.jpg'>
</body>
</html>
```

```java
@WebServlet("/rollDice2")
public class TwoDiceServlet extends HttpServlet {
    int getRandomInt(int range) {
        return new Random().nextInt(range)+1;
    }

    public void service(HttpServletRequest request, H
        int idx1 = getRandomInt(6);
        int idx2 = getRandomInt(6);

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```

```jsp
<%@ page contentType="text/html;charset=utf-8"%>
<%@ page import="java.util.Random" %>


<%-- <%! 클래스 영역 %> --%>
<%!
    int getRandomInt(int range){
        return new Random().nextInt(range)+1;
    }
%>


<%-- <%  메서드 영역 - service()의 내부 %> --%>
<%
    int idx1 = getRandomInt(6);
    int idx2 = getRandomInt(6);
%>
<html>
<head>
    <title>twoDice.jsp</title>
</head>
<body>
    <img src='resources/img/dice<%=idx1%>.jpg'>
    <img src='resources/img/dice<%=idx2%>.jpg'>
</body>
</html>
```

```java
@WebServlet("/rollDice2")
public class TwoDiceServlet extends HttpServlet {
    int getRandomInt(int range) {
        return new Random().nextInt(range)+1;
    }

    public void service(HttpServletRequest request, H
        int idx1 = getRandomInt(6);
        int idx2 = getRandomInt(6);

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```

```jsp
<%@ page contentType="text/html;charset=utf-8"%>
<%@ page import="java.util.Random" %>

<%-- <%! 클래스 영역 %> --%>
<%!
    int getRandomInt(int range){
        return new Random().nextInt(range)+1;
    }
%>


<%-- <% 메서드 영역 - service()의 내부 %> --%>
<%
    int idx1 = getRandomInt(6);
    int idx2 = getRandomInt(6);
%>
<html>
<head>
    <title>twoDice.jsp</title>
</head>
<body>
    <img src='resources/img/dice<%=idx1%>.jpg'>
    <img src='resources/img/dice<%=idx2%>.jpg'>
</body>
</html>
```

<%= %>

```java
@WebServlet("/rollDice2")
public class TwoDiceServlet extends HttpServlet {
    int getRandomInt(int range) {
        return new Random().nextInt(range)+1;
    }

    public void service(HttpServletRequest request, H
        int idx1 = getRandomInt(6);
        int idx2 = getRandomInt(6);

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```

```jsp
<%@ page contentType="text/html;charset=utf-8"%>
<%@ page import="java.util.Random" %>

<%-- <%! 클래스 영역 %> --%>
<%!
    int getRandomInt(int range){
        return new Random().nextInt(range)+1;
    }
%>

<%-- <% 메서드 영역 - service()의 내부 %> --%>
<%
    int idx1 = getRandomInt(6);    2
    int idx2 = getRandomInt(6);    3
%>
<html>
<head>
    <title>twoDice.jsp</title>
</head>
<body>
                              <%= %>
    <img src='resources/img/dice<%=idx1%>.jpg'>   dice2.jpg
    <img src='resources/img/dice<%=idx2%>.jpg'>   dice3.jpg
</body>
</html>
```

```java
@WebServlet("/rollDice2")
public class TwoDiceServlet extends HttpServlet {
    int getRandomInt(int range) {
        return new Random().nextInt(range)+1;
    }

    public void service(HttpServletRequest request, H
        int idx1 = getRandomInt(6);
        int idx2 = getRandomInt(6);

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```

<%-- 주석 --%>

```jsp
<%@ page contentType="text/html;charset=utf-8"%>
<%@ page import="java.util.Random" %>

<%-- <%! 클래스 영역 %> --%>
<%!
    int getRandomInt(int range){
        return new Random().nextInt(range)+1;
    }
%>

<%-- <% 메서드 영역 - service()의 내부 %> --%>
<%
    int idx1 = getRandomInt(6);
    int idx2 = getRandomInt(6);
%>
<html>
<head>
    <title>twoDice.jsp</title>
</head>
<body>
    <img src='resources/img/dice<%=idx1%>.jpg'>
    <img src='resources/img/dice<%=idx2%>.jpg'>
</body>
</html>
```

```java
@WebServlet("/rollDice2")
public class TwoDiceServlet extends HttpServlet {
    int getRandomInt(int range) {
        return new Random().nextInt(range)+1;
    }

    public void service(HttpServletRequest request, H
        int idx1 = getRandomInt(6);
        int idx2 = getRandomInt(6);

        response.setContentType("text/html");
        response.setCharacterEncoding("utf-8");
        PrintWriter out = response.getWriter();
        out.println("<html>");
        ...
    }
}
```

```jsp
<%@ page contentType="text/html;charset=utf-8"%>
<%@ page import="java.util.Calendar" %>
<%
    String year  = request.getParameter("year");
    String month = request.getParameter("month");
    String day   = request.getParameter("day");

    int yyyy = Integer.parseInt(year);
    int mm   = Integer.parseInt(month);
    int dd   = Integer.parseInt(day);

    // 2. 처리
    Calendar cal = Calendar.getInstance();
    cal.set(yyyy, mm - 1, dd);

    int dayOfWeek = cal.get(Calendar.DAY_OF_WEEK);
    char yoil = " 일월화수목금토".charAt(dayOfWeek);

    // 3. 출력
%>
<html>
<head>
    <title>yoilTeller.jsp</title>
</head>
<body>
    <h1><%=yyyy%>년 <%=mm%>월 <%=dd%>일은 <%=yoil%>요일입니다.</h1>
</body>
</html>
```

```jsp
<%@ page contentType="text/html;charset=utf-8"%>
<%@ page import="java.util.Calendar" %>
<%
    String year  = request.getParameter("year");
    String month = request.getParameter("month");
    String day   = request.getParameter("day");

    int yyyy = Integer.parseInt(year);
    int mm   = Integer.parseInt(month);
    int dd   = Integer.parseInt(day);

    // 2. 처리
    Calendar cal = Calendar.getInstance();
    cal.set(yyyy, mm - 1, dd);

    int dayOfWeek = cal.get(Calendar.DAY_OF_WEEK);
    char yoil = " 일월화수목금토".charAt(dayOfWeek);

    // 3. 출력
%>
<html>
<head>
    <title>yoilTeller.jsp</title>
</head>
<body>
    <h1><%=yyyy%>년 <%=mm%>월 <%=dd%>일은 <%=yoil%>요일입니다.</h1>
</body>
</html>
```

```jsp
<%@ page contentType="text/html;charset=utf-8"%>
<%@ page import="java.util.Calendar" %>
<%
    String year  = request.getParameter("year");
    String month = request.getParameter("month");
    String day   = request.getParameter("day");

            기본객체는 객체 생성 없이 사용할 수 있다

    int yyyy = Integer.parseInt(year);
    int mm   = Integer.parseInt(month);
    int dd   = Integer.parseInt(day);

    // 2. 처리
    Calendar cal = Calendar.getInstance();
    cal.set(yyyy, mm - 1, dd);

    int dayOfWeek = cal.get(Calendar.DAY_OF_WEEK);
    char yoil = " 일월화수목금토".charAt(dayOfWeek);

    // 3. 출력
%>
<html>
<head>
    <title>yoilTeller.jsp</title>
</head>
<body>
    <h1><%=yyyy%>년 <%=mm%>월 <%=dd%>일은 <%=yoil%>요일입니다.</h1>
</body>
</html>
```

```jsp
<%@ page contentType="text/html;charset=utf-8"%>
<%@ page import="java.util.Calendar" %>
<%

    String year  = request.getParameter("year");
    String month = request.getParameter("month");
    String day   = request.getParameter("day");

    int yyyy = Inte
    int mm   = Inte
    int dd   = Inte

    // 2. 처리
    Calendar cal =
    cal.set(yyyy, m

    int dayOfWeek = // ...
    char yoil = " 일

    // 3. 출력
%>
<html>
<head>
    <title>yoilTel
</head>
<body>
    <h1><%=yyyy%>년 <%=mm%>월 <%=dd%>일은 <%=yoil%>요일입니다.</h1>
</body>
</html>
```

기본객체는 객체 생성 없이 사용할 수 있다

```java
public final class yoilTeller_jsp extends org.apache.jasper.runtime.HttpJspBase
        implements org.apache.jasper.runtime.JspSourceDependent,
                   org.apache.jasper.runtime.JspSourceImports {
    // ...
```

service() 의 지역 변수(lv) = 기본 객체

```java
    public void _jspService(final javax.servlet.http.HttpServletRequest request,
                            final javax.servlet.http.HttpServletResponse response)
        throws java.io.IOException, javax.servlet.ServletException {

        final javax.servlet.jsp.PageContext pageContext;
        javax.servlet.http.HttpSession session = null;
        final javax.servlet.ServletContext application;
        final javax.servlet.ServletConfig config;
        javax.servlet.jsp.JspWriter out = null;
        final java.lang.Object page = this;
        //...
```

기본 객체	유효 범위	설명
pageContext	1개 JSP페이지	JSP페이지의 시작부터 끝까지. 해당 JSP 내부에서만 접근가능. 페이지당 1개
request	1+개 JSP페이지	요청의 시작부터 응답까지. 다른 JSP로 전달 가능. 요청마다 1개
session	n개 JSP페이지	session의 시작부터 종료까지(로그인 ~ 로그아웃). 클라이언트마다 1개
application	context 전체	Web Application의 시작부터 종료까지. context내부 어디서나 접근 가능 모든 클라이언트가 공유. context마다 1개

속성 관련 메서드	설명
void setAttribute(String name, Object value)	지정된 값(value)을 지정된 속성 이름(name)으로 저장
Object getAttribute(String name)	지정된 이름(name)으로 저장된 속성의 값을 반환
void removeAttribute(String name)	지정된 이름(name)의 속성을 삭제
Enumeration getAttributeNames()	기본 객체에 저장된 모든 속성의 이름을 반환

@WebServlet으로 URL매핑할 때 사용

```
@WebServlet(urlPatterns = {"/hello", "/hello/*"}, loadOnStartup = 1)
@WebServlet("/hello")
```

요청이 들어오기 전에 미리 init하기 위한 설정

우선순위	종류	URL Pattern	매칭 URL
1	exact mapping	/login/hello.do	localhost/ch2/login/hello.do
2	path mapping	/login/*	localhost/ch2/login/ localhost/ch2/login/hello localhost/ch2/login/hello.do localhost/ch2/login/test/
3	extenstion mapping	*.do	localhost/ch2/hi.do localhost/ch2/login/hello.do
4	default mapping	/	localhost/ch2/ localhost/ch2/hello.do localhost/ch2/login/ localhost/ch2/login/hello localhost/ch2/login/hello.do

```
exact mapping: 정확히 일치하는 URL
path mapping : 경로/와일드카드 URL
extension mapping: 확장자 맵핑
default mapping : 기본 맵핑(모든 주소)
```

ServletContext

children(서블릿)	
Key(String, 서블릿 이름)	**Value(StandardWrapper)**
com.fastcampus.ch2.HelloServlet	0x100
jsp	0x200
default	0x300
...	...

0x100 → HelloServlet

0x200 → JspServlet

0x300 → DefaultServlet

servlet Mappings	
Key(String)	**Value(String, 서블릿 이름)**
/hello	com.fastcampus.ch2.HelloServlet
*.jsp	jsp
/	default
...	...

request →

ServletContext

children(서블릿)	
Key(String, 서블릿 이름)	Value(StandardWrapper)
com.fastcampus.ch2.HelloServlet	0x100
jsp	0x200
default	0x300

0x100 → HelloServlet

0x200 → JspServlet

0x300 → DefaultServlet

2. 처리할 서블릿을 children에서 찾아 요청을 전달한다

servlet Mappings	
Key(String)	Value(String, 서블릿 이름)
/hello	com.fastcampus.ch2.HelloServlet
*.jsp	jsp
/	default
...	...

1. 요청이 들어오면 servlet Mappings에서 매핑을 확인한다

request →

ServletContext
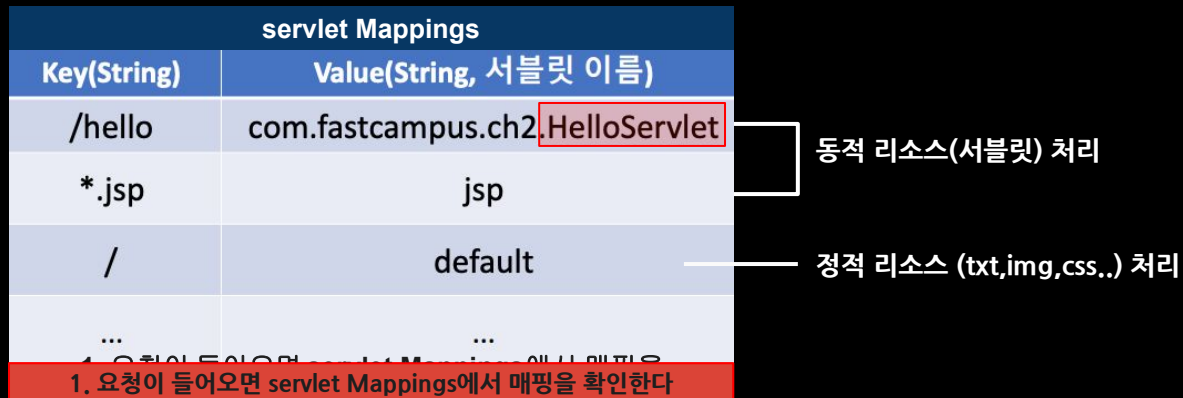
children(서블릿)	
Key(String, 서블릿 이름)	Value(StandardWrapper)
com.fastcampus.ch2.HelloServlet	0x100
jsp	0x200
default	0x300

0x100 → HelloServlet

0x200 → JspServlet

0x300 → DefaultServlet

2. 처리할 서블릿을 children에서 찾아 요청을 전달한다

servlet Mappings	
Key(String)	Value(String, 서블릿 이름)
/hello	com.fastcampus.ch2.HelloServlet
*.jsp	jsp
/	default
...	...

동적 리소스(서블릿) 처리

정적 리소스 (txt,img,css..) 처리

request

1. 요청이 들어오면 servlet Mappings에서 매핑을 확인한다

ServletContext

children(서블릿)	
Key(String, 서블릿 이름)	**Value(StandardWrapper)**
com.fastcampus.ch2.HelloServlet	0x100
jsp	0x200
default	0x300

0x100 → **HelloServlet**

0x200 → **JspServlet**

0x300 → **DefaultServlet**

2. 처리할 서블릿을 children에서 찾아 요청을 전달한다

스프링에서는
Default Servlet 대신
Dispatcher Servlet 사용

servlet Mappings	
Key(String)	**Value(String, 서블릿 이름)**
/hello	com.fastcampus.ch2.HelloServlet
*.jsp	jsp
/	default
...	...

동적 리소스(서블릿) 처리

정적 리소스 (txt,img,css..) 처리

request

1. 요청이 들어오면 servlet Mappings에서 매핑을 확인한다

```
<% java code %>
<%=person.getCar().getColor()%>
```

→

```
${ value }
${ person.car.color }
```

<% java code %> ➡ $ { value }

<%=person.getCar().getColor()%> ${ person.car.color }

- 객체 안에 getter가 정의되어 있어야, EL을 사용할 수 있다
- EL를 사용할 때는 지역변수를 사용할 수 없다
- 지역 변수를 사용하기 위해서 저장소에 저장해야한다 (pageContext)

```java
Person person = new Person();
request.setAttribute("person", person);
request.setAttribute("name", "남궁성");
request.setAttribute("list", new java.util.ArrayList());
```

```jsp
<%@ page contentType="text/html;charset=utf-8"%>
<%@ taglib prefix="c"   uri="http://java.sun.com/jsp/jstl/core"%>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<html>
<head>
    <title>JSTL</title>
</head>
<body>
<c:set var="to"   value="10"/>
<c:set var="arr"  value="10,20,30,40,50,60,70"/>
<c:forEach var="i" begin="1" end="${to}">
    ${i}
</c:forEach>
<br>
<c:if test="${not empty arr}">                                    if문
    <c:forEach var="elem" items="${arr}" varStatus="status">      for문
        ${status.count}. arr[${status.index}]=${elem}<BR>
    </c:forEach>
</c:if>
<c:if test="${param.msg != null}">
    msg=${param.msg}
    msg=<c:out value="${param.msg}"/>
</c:if>
<br>
<c:if test="${param.msg == null}">메시지가 없습니다.<br></c:if>
<c:set var="age" value="${param.age}"/>
<c:choose>
    <c:when test="${age >= 19}">성인입니다.</c:when>
    <c:when test="${0 <= age && age < 19}">성인이 아닙니다.</c:when>
    <c:otherwise>값이 유효하지 않습니다.</c:otherwise>
</c:choose>
<br>
<c:set var="now" value="<%=new java.util.Date() %>"/>
Server time is <fmt:formatDate value="${now}" type="both" pattern="yyyy/MM/dd HH:mm:ss"/>
</body>
</html>
```

```jsp
<%@ page contentType="text/html;charset=utf-8"%>
<%@ taglib prefix="c"   uri="http://java.sun.com/jsp/jstl/core"%>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<html>
<head>
    <title>JSTL</title>
</head>
<body>
<c:set var="to"   value="10"/>
<c:set var="arr"  value="10,20,30,40,50,60,70"/>
<c:forEach var="i" begin="1" end="${to}">
    ${i}
</c:forEach>
<br>
<c:if test="${not empty arr}">
    <c:forEach var="elem" items="${arr}" varStatus="status">
        ${status.count}. arr[${status.index}]=${elem}<BR>
    </c:forEach>
</c:if>
<c:if test="${param.msg != null}">
    msg=${param.msg}
    msg=<c:out value="${param.msg}"/>
</c:if>
<br>
<c:if test="${param.msg == null}">메시지가 없습니다.<br></c:if>
<c:set var="age" value="${param.age}"/>
<c:choose>
    <c:when test="${age >= 19}">성인입니다.</c:when>
    <c:when test="${0 <= age && age < 19}">성인이 아닙니다.</c:when>
    <c:otherwise>값이 유효하지 않습니다.</c:otherwise>
</c:choose>
<br>
<c:set var="now" value="<%=new java.util.Date() %>"/>
Server time is <fmt:formatDate value="${now}" type="both" pattern="yyyy/MM/dd HH:mm:ss"/>
</body>
</html>
```

taglib : 태그 라이브러리 = 태그에 대해 정의되어있다
```jsp
<%@ taglib prefix="c"   uri="http://java.sun.com/jsp/jstl/core"%>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
```

if문

for문

```jsp
<%@ page contentType="text/html;charset=utf-8"%>
<%@ taglib prefix="c"   uri="http://java.sun.com/jsp/jstl/core"%>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<html>
<head>
    <title>JSTL</title>
</head>
<body>
<c:set var="to"    value="10"/>
<c:set var="arr"   value="10,20,30,40,50,60,70"/>
<c:forEach var="i" begin="1" end="${to}">
    ${i}
</c:forEach>
<br>
<c:if test="${not empty arr}">
    <c:forEach var="elem" items="${arr}" varStatus="status">
        ${status.count}. arr[${status.index}]=${elem}<BR>
    </c:forEach>
</c:if>
<c:if test="${param.msg != null}">
    msg=${param.msg}
    msg=<c:out value="${param.msg}"/>
</c:if>
<br>
<c:if test="${param.msg == null}">메시지가 없습니다.<br></c:if>
<c:set var="age" value="${param.age}"/>
<c:choose>
    <c:when test="${age >= 19}">성인입니다.</c:when>
    <c:when test="${0 <= age && age < 19}">성인이 아닙니다.</c:when>
    <c:otherwise>값이 유효하지 않습니다.</c:otherwise>
</c:choose>
<br>
<c:set var="now" value="<%=new java.util.Date() %>"/>
Server time is <fmt:formatDate value="${now}" type="both" pattern="yyyy/MM/dd HH:mm:ss"/>
</body>
</html>
```

if문

for문

---

taglib : 태그 라이브러리 = 태그에 대해 정의되어있다

```jsp
<%@ taglib prefix="c"   uri="http://java.sun.com/jsp/jstl/core"%>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
```

JSP

```jsp
<%
    if(msg !=null) {
%>
        msg=${param.msg}
<%
    } else {
%>
        메시지가 없습니다.
<%
    }
%>
```

```
<c:set var="to" value="10"/>
<c:set var="arr" value="10,20,30,40,50"/>  <%-- 배열 형태 --%>
```

- setAttribute()와 같이 저장소에 데이터를 저장하는 역할 (scope 생략)

```
<c:forEach var="i" begin="1" end="${to}">
    ${i}
</c:forEach>
```

- <c:forEach> : 반복문

```
<c:if test="${not empty arr}">
    <c:forEach var="elem" items="${arr}" varStatus="status">
        ${status.count}.arr[${status.index}]=${elem}<br>
    </c:forEach>
</c:if>
```

- <c:if> : 조건문(if)

```
<c:out value="${param.msg}"/>
```

- <c:out> : value값에 태그가 포함되어 있는 경우, 태그를 해석하지 않고 텍스트로 받아들인다

```
<fmt:formatDate value="${now}" type="both" pattern="yyyy/MM/dd">
```

Filter

- 공통적인 요청 전처리와 응답 후처리에 사용한다 (예) 로깅, 인코딩 등

- 여러개의 필터가 있는 경우 처리 순서는 아래와 같다

<u>필터A, 필터B 가 있는 경우</u>

- 필터A 전처리 / 다음 필터 호출

- 필터B 전처리 / 서블릿 호출

- 서블릿 처리

- 필터B 후처리

- 필터A 후처리

Filter A	Filter B	Servlet
1.전처리 2.다음 필터 호출 3.후처리	1.전처리 2.서블릿호출 3.후처리	2.처리