

MVC 아키텍처

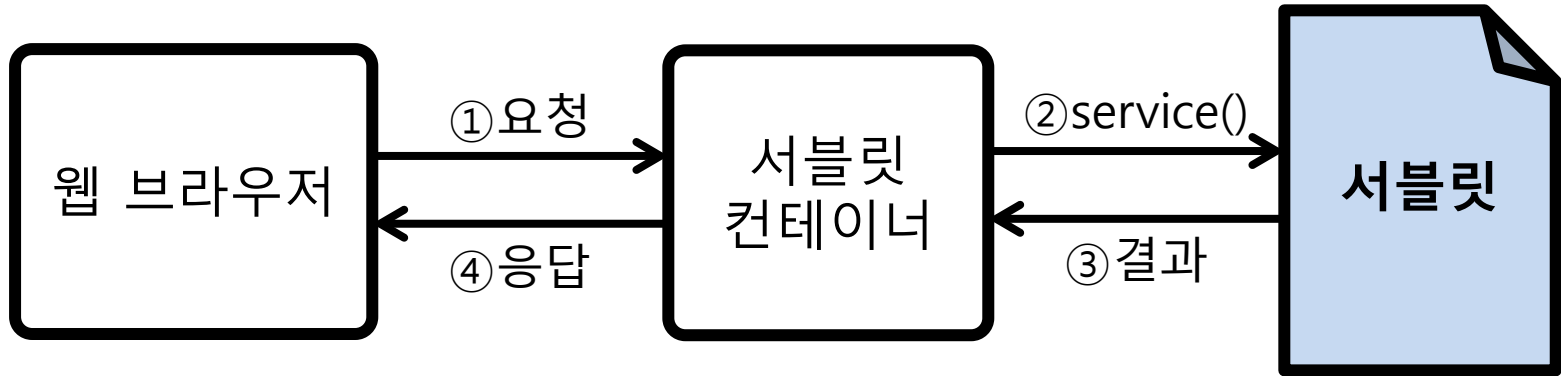
5.1 MVC 이해하기

5.1 MVC 이해하기

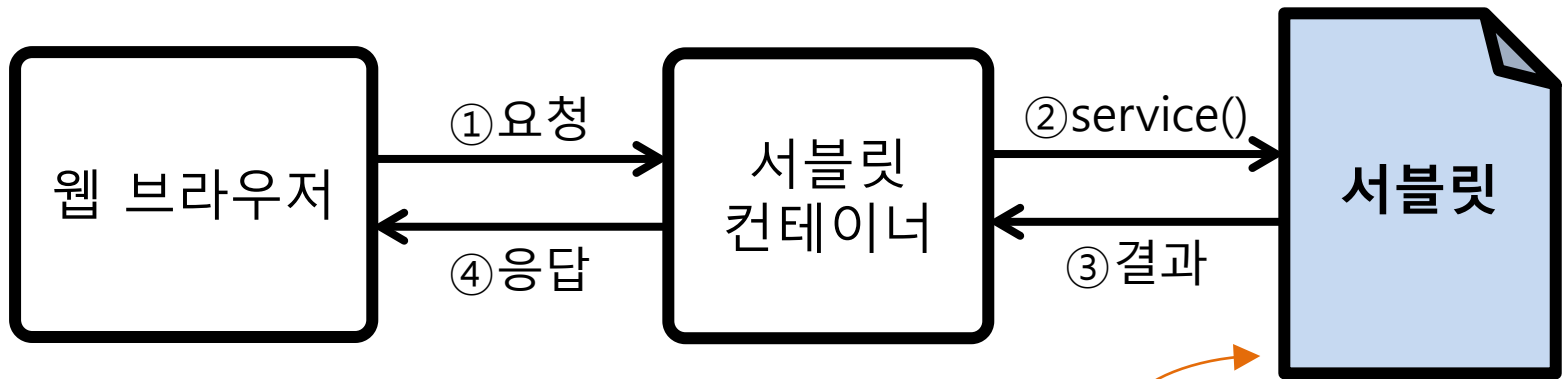
클라이언트 요청
처리

올인원 방식

5.1 MVC 이해하기

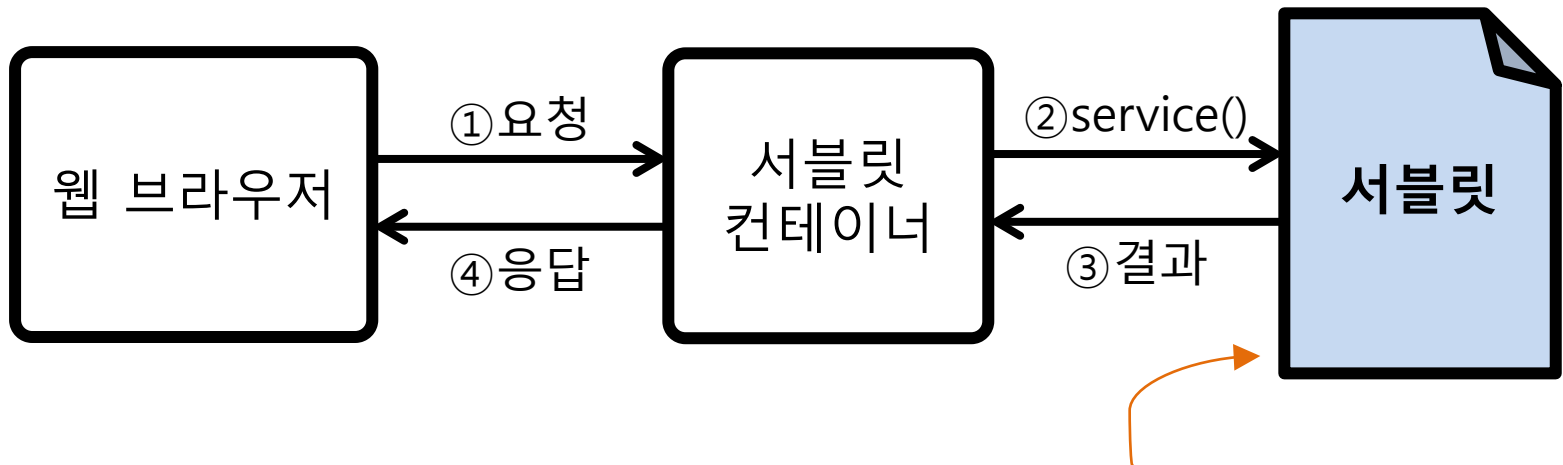


5.1 MVC 이해하기



- ✓ 요청 파라미터 가공
- ✓ 비즈니스 로직 및 데이터 지속성 처리
- ✓ 결과 데이터 가공 및 결과 화면 생성

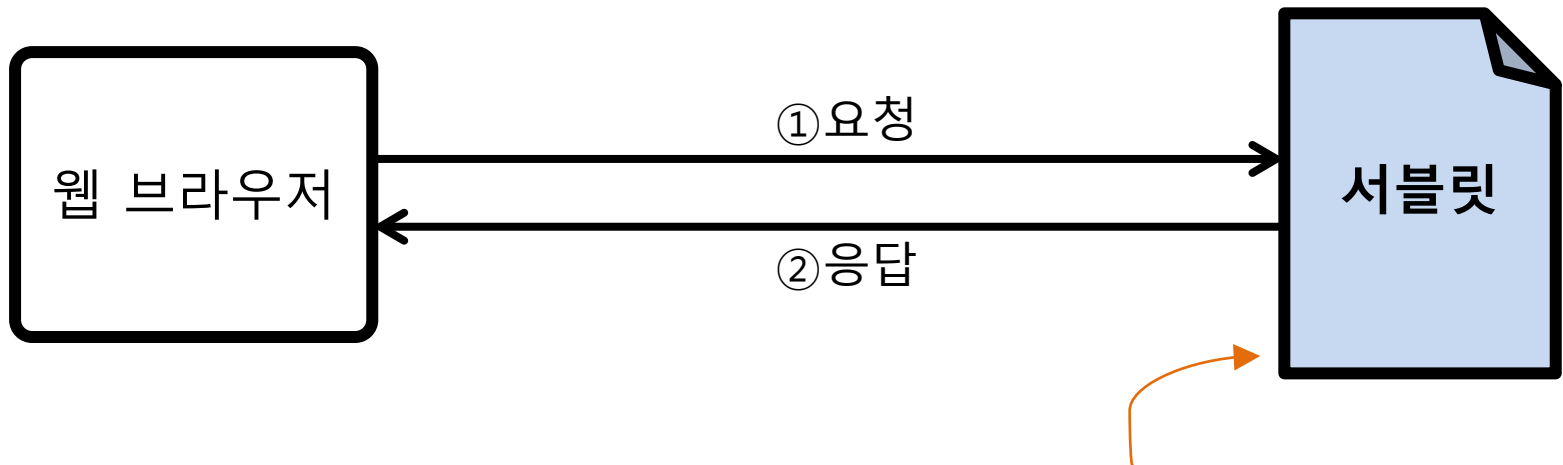
5.1 MVC 이해하기



“All-in-one”

- ✓ 요청 파라미터 가공
- ✓ 비즈니스 로직 및 데이터 지속성 처리
- ✓ 결과 데이터 가공 및 결과 화면 생성

5.1 MVC 이해하기



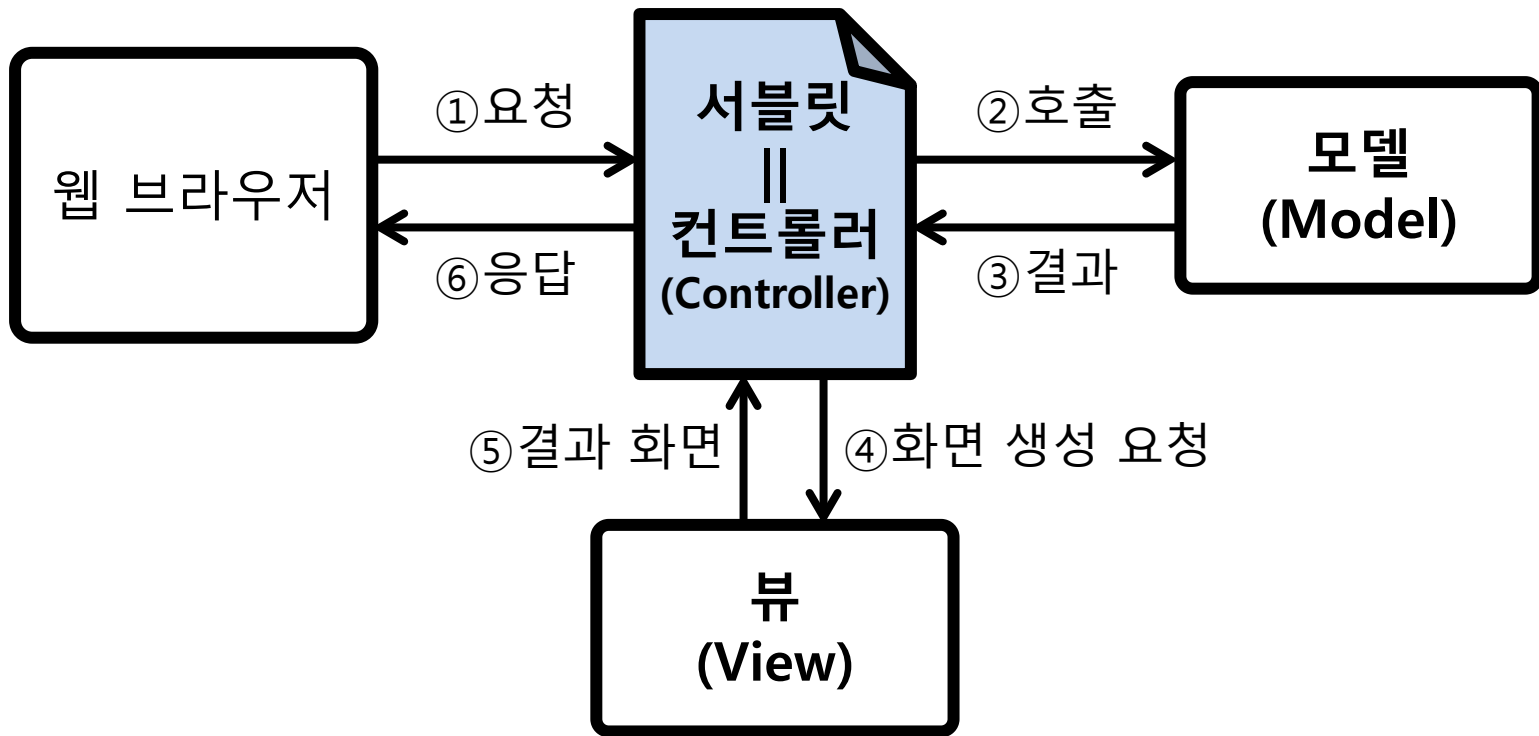
“All-in-one”

- ✓ 요청 파라미터 가공
- ✓ 비즈니스 로직 및 데이터 지속성 처리
- ✓ 결과 데이터 가공 및 결과 화면 생성

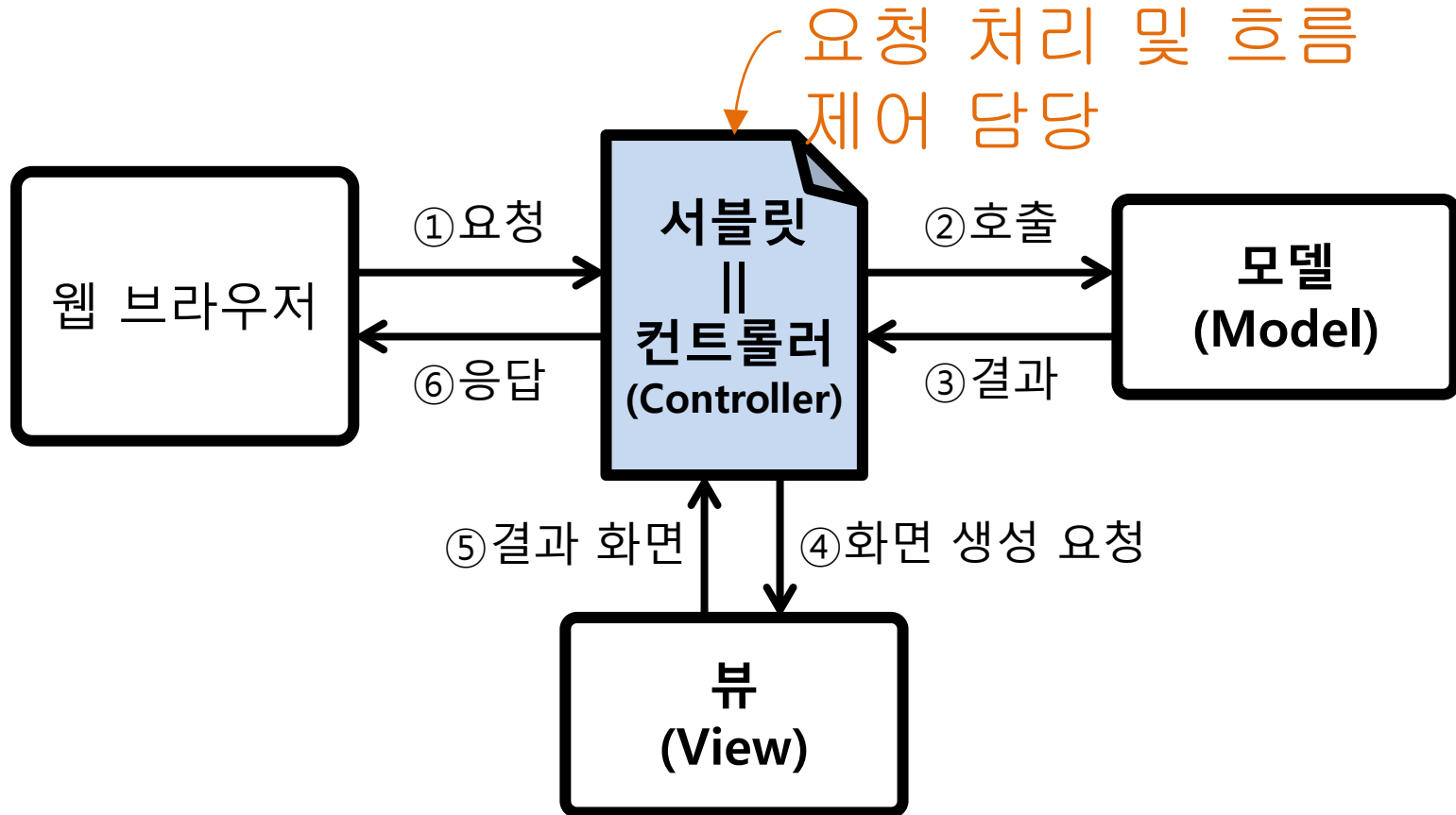
5.1 MVC 이해하기

MVC 아키텍처

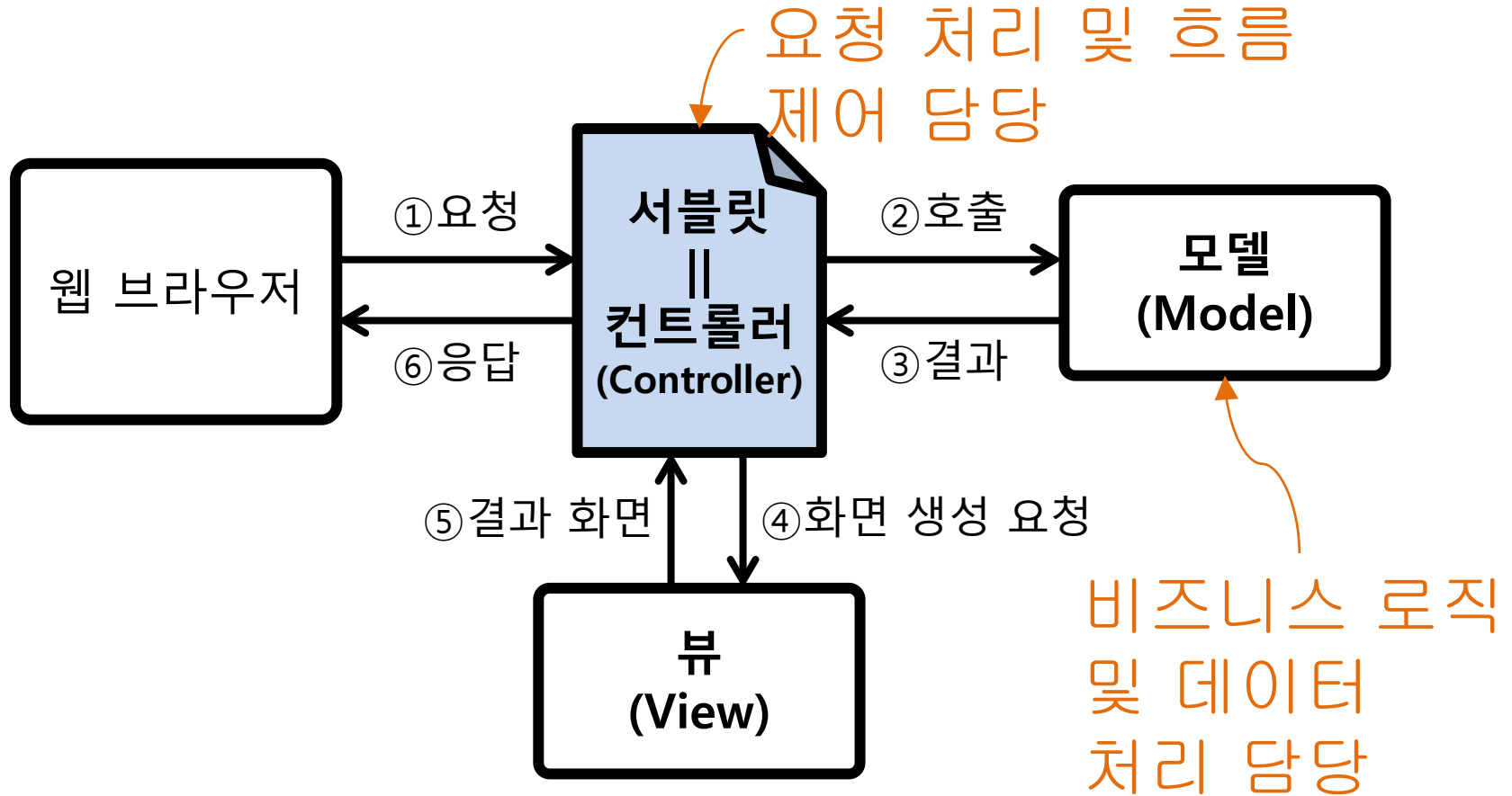
5.1 MVC 이해하기



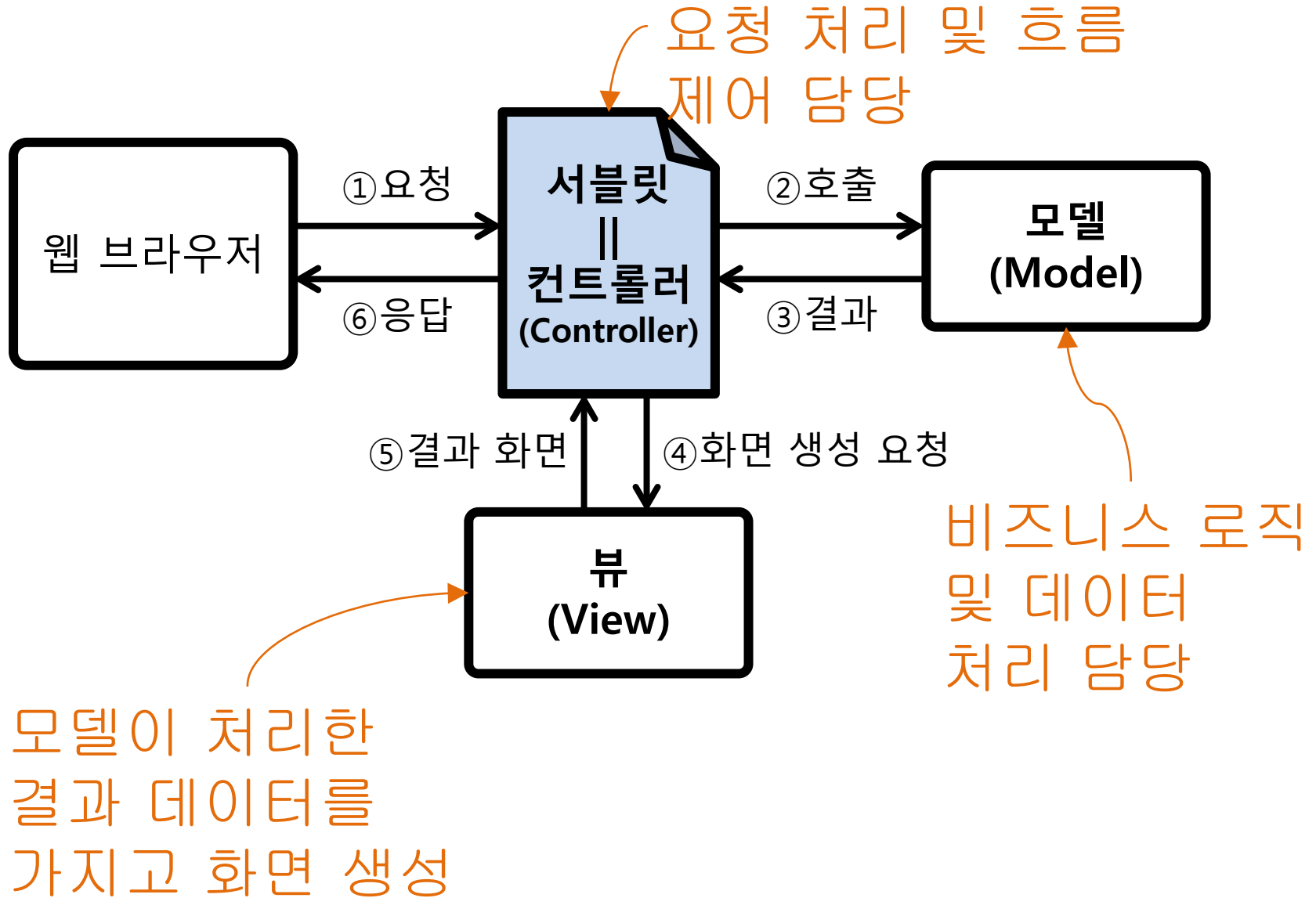
5.1 MVC 이해하기



5.1 MVC 이해하기



5.1 MVC 이해하기



5.1 MVC 이해하기

MVC 아키텍처의 등장 배경

5.1 MVC 이해하기


세계화와 인터넷의 등장

5.1 MVC 이해하기

세계화와
인터넷의
등장

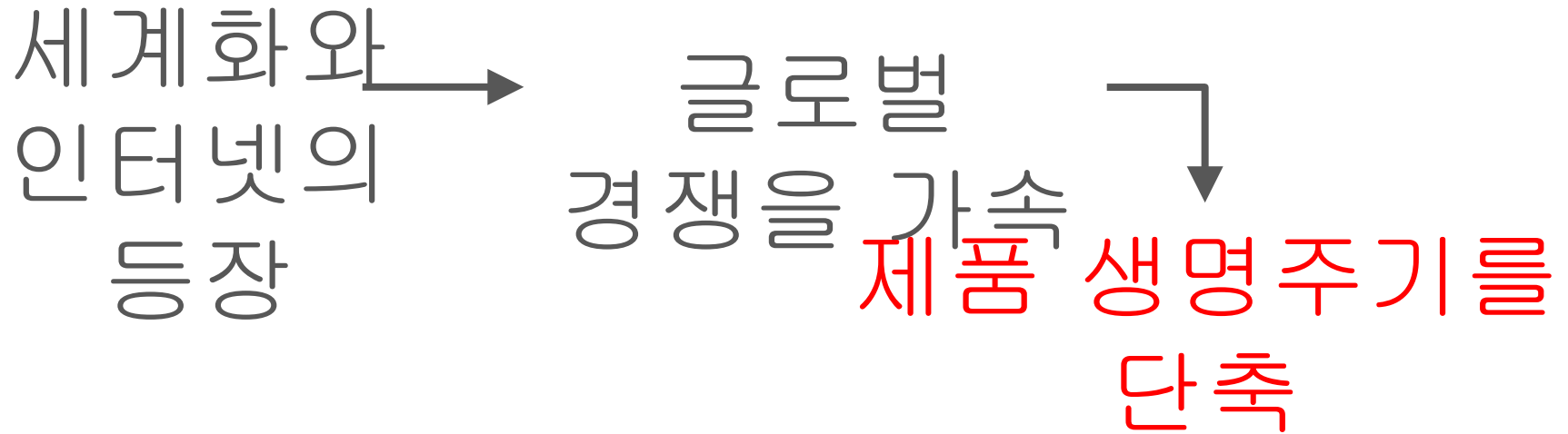
5.1 MVC 이해하기

세계화와
인터넷의
등장

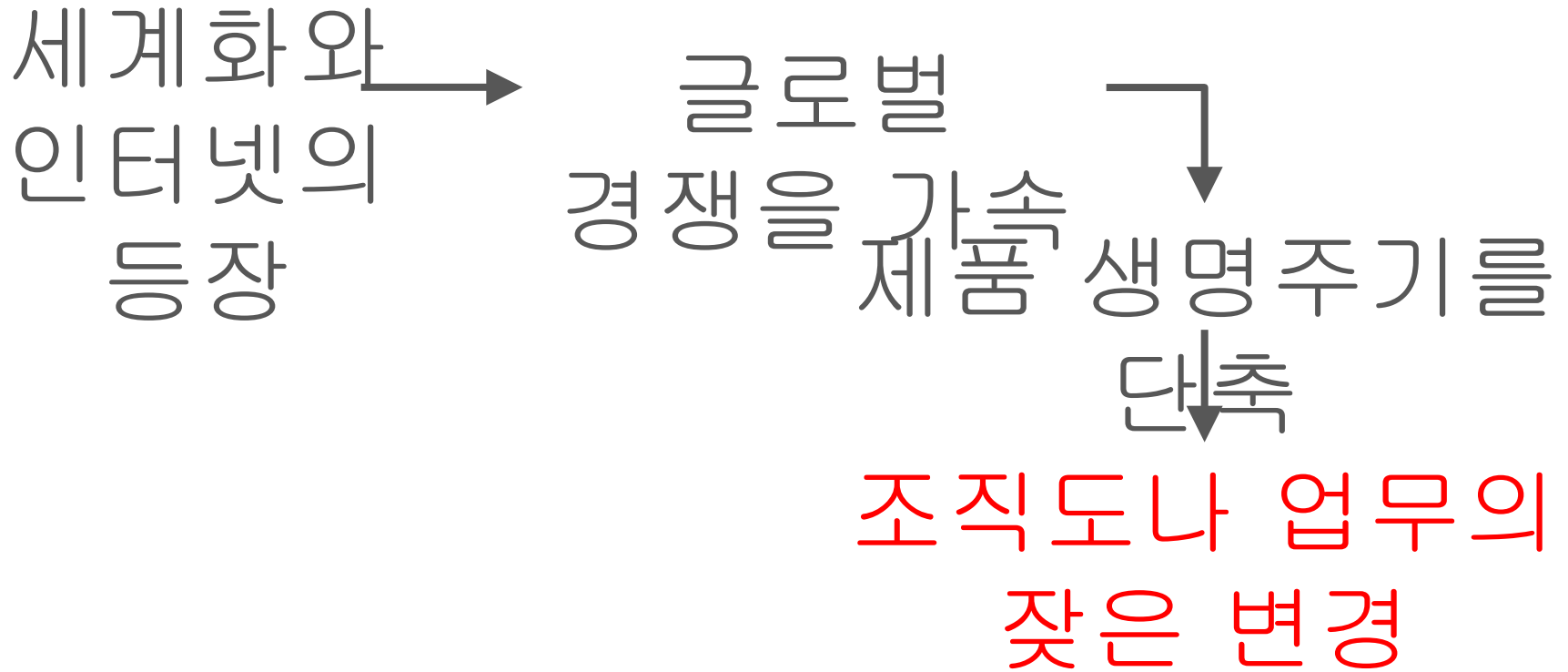


글로벌
경쟁을 가속

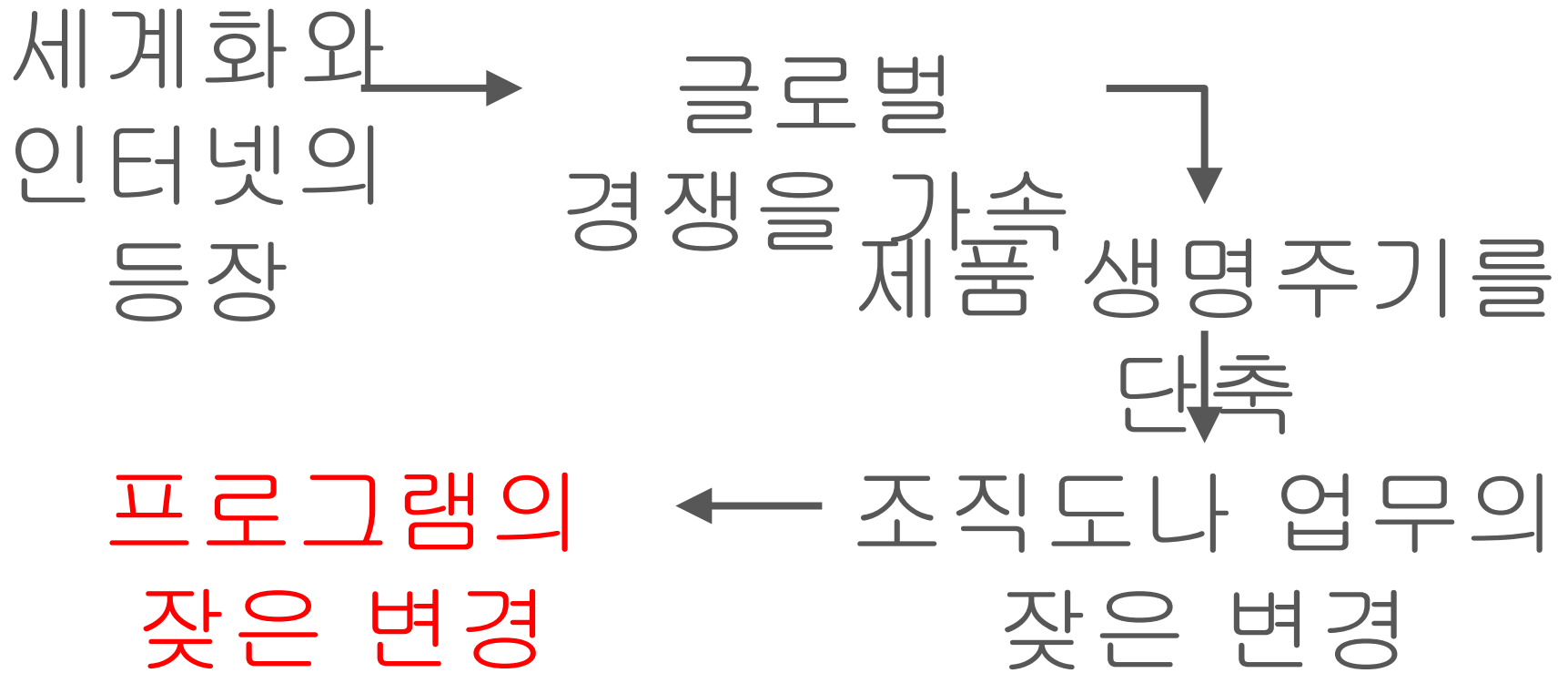
5.1 MVC 이해하기



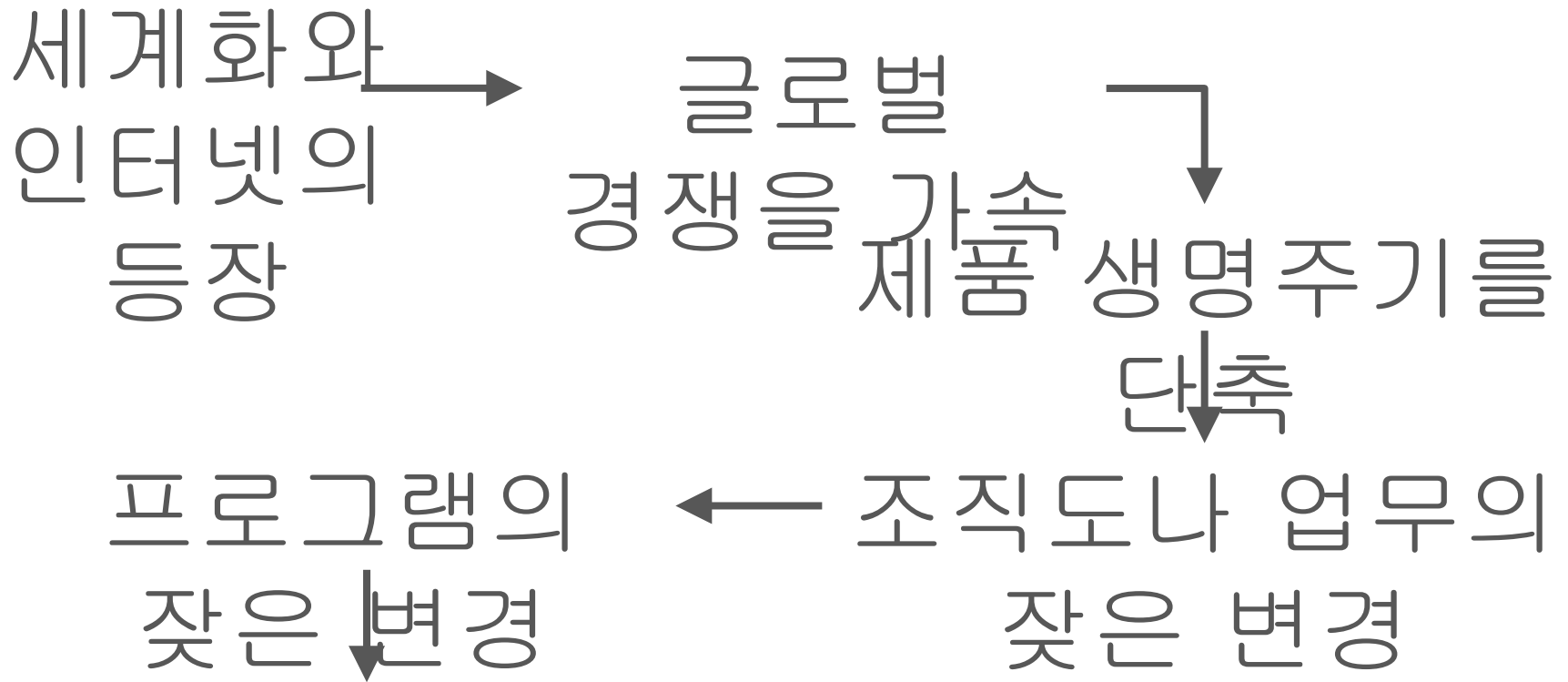
5.1 MVC 이해하기



5.1 MVC 이해하기

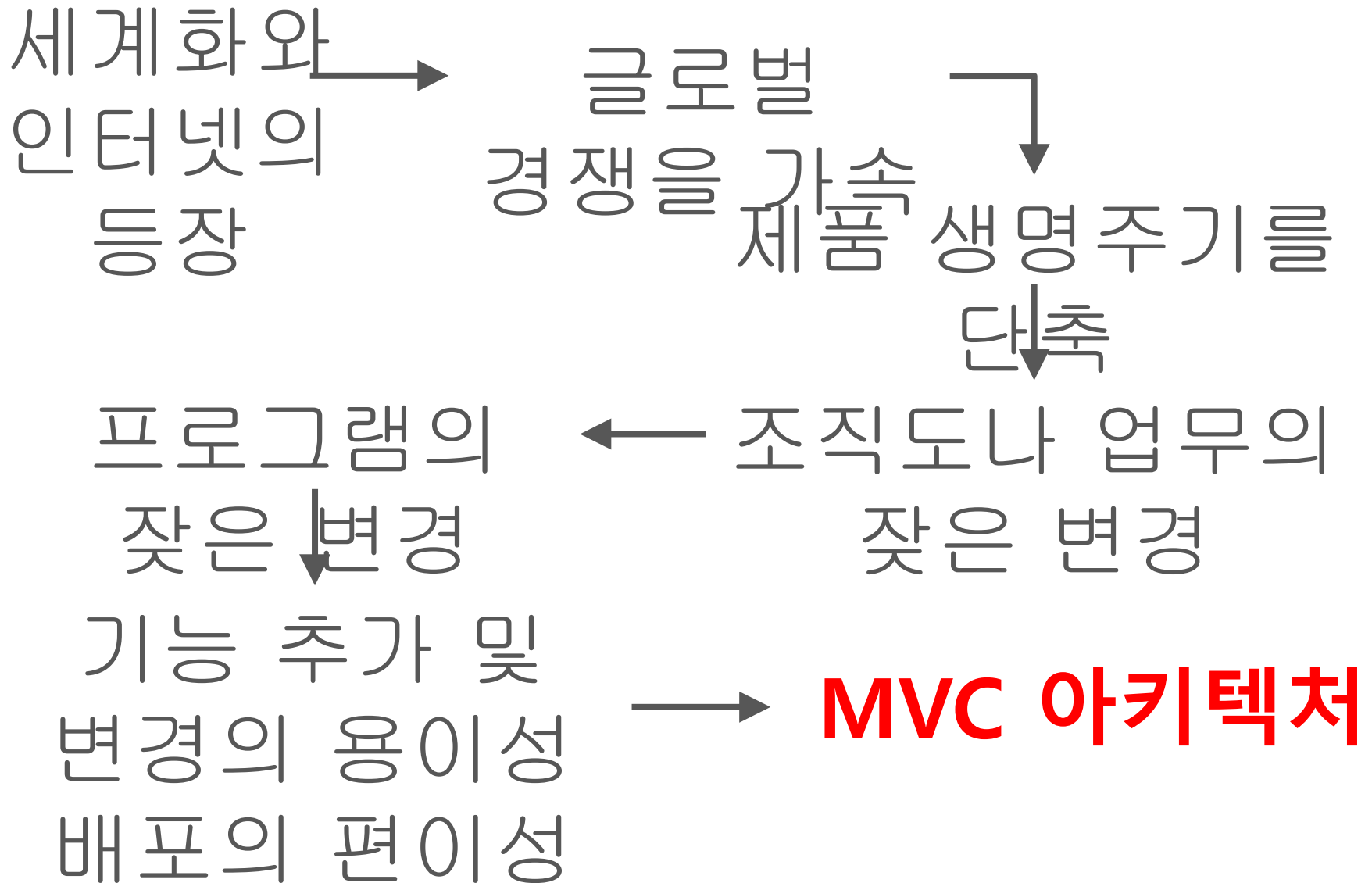


5.1 MVC 이해하기




기능 추가 및
변경의 용이성
배포의 편이성

5.1 MVC 이해하기



MVC 아키텍처의 특징



유지 보수가 쉽도록,
중복 코드의 작성을
최소화하고
기존 코드의 재사용을
높임

MVC 아키텍처의 특징



유지 보수가 쉽도록,
중복 코드의 작성을
최소화하고
기존 코드의 재사용을
높임



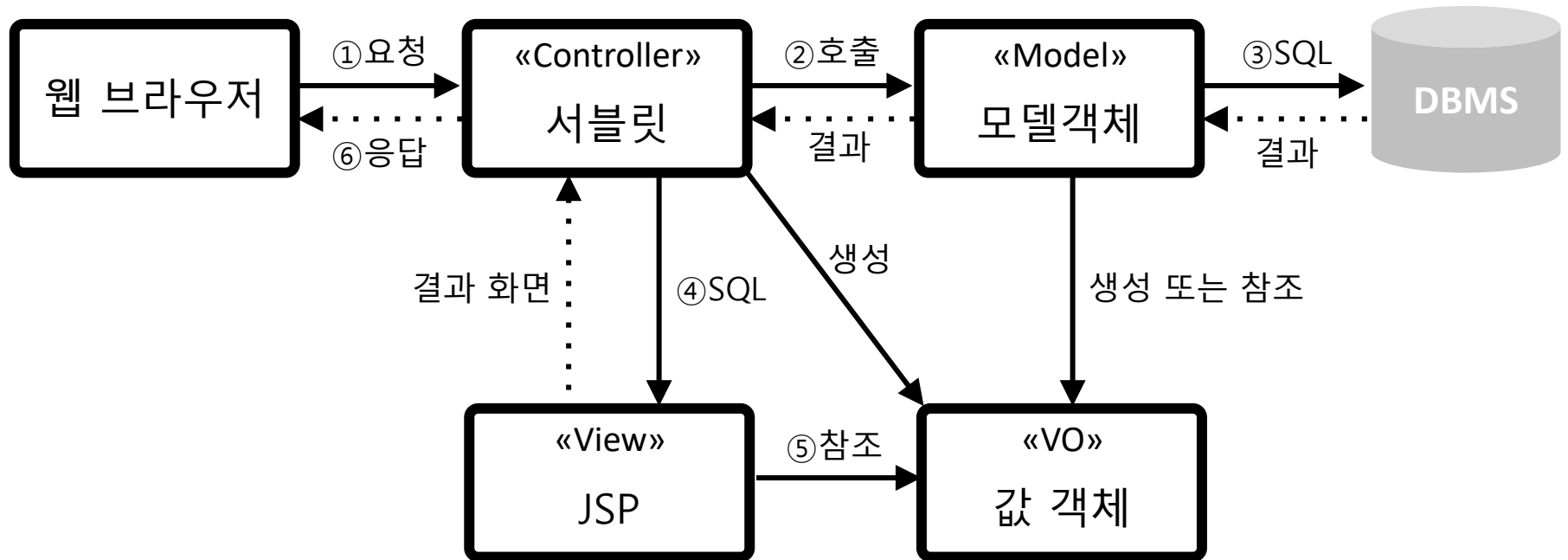
- ✓ 역할을 세분화
- ✓ 의존성 최소화

5.1 MVC 이해하기

MVC의 실행 흐름

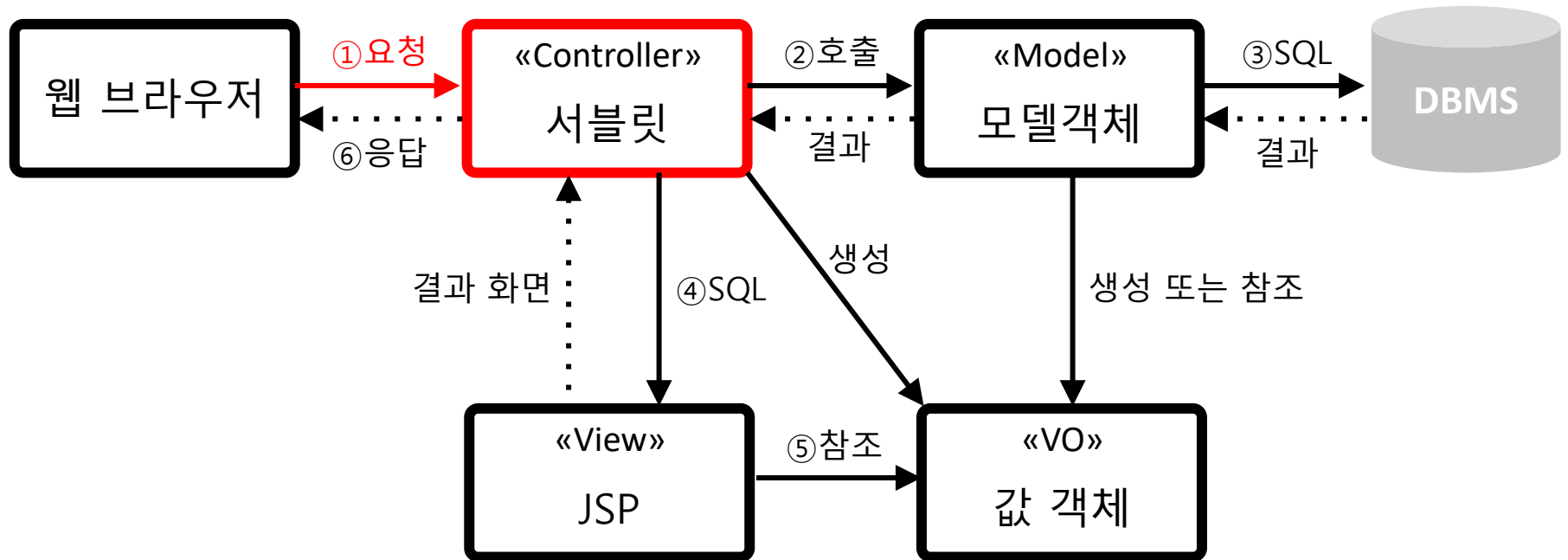
5.1 MVC 이해하기

데이터 가져오기



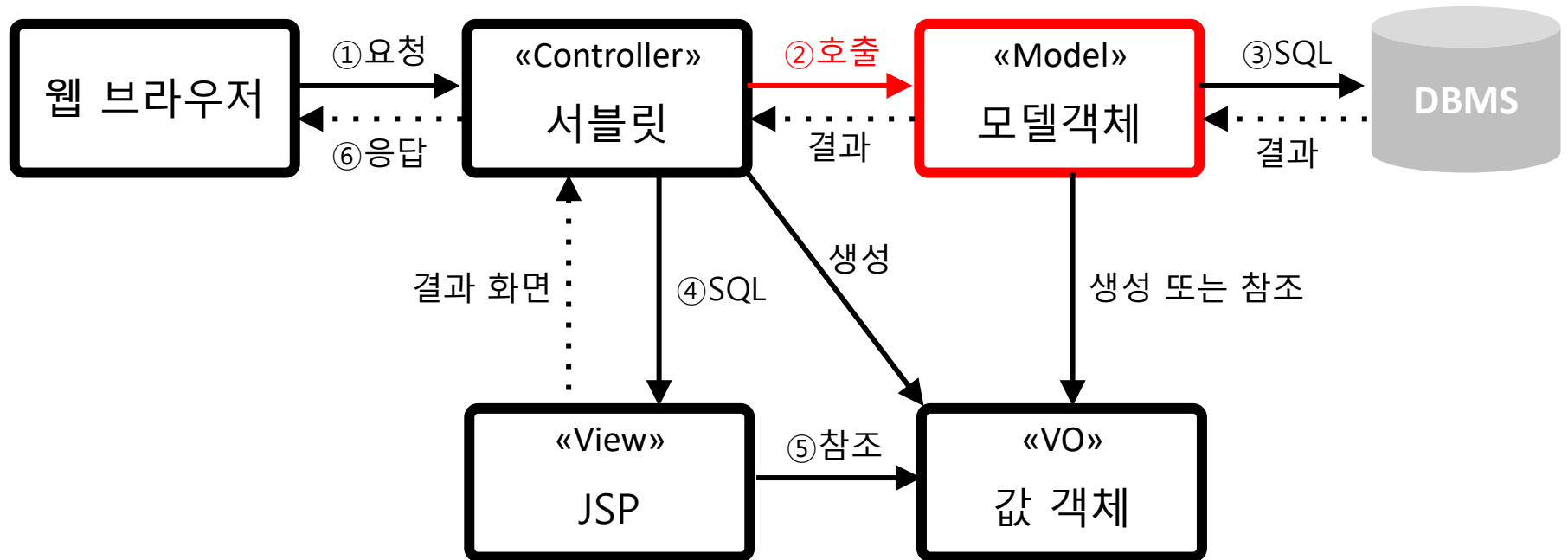
5.1 MVC 이해하기

데이터 가져오기



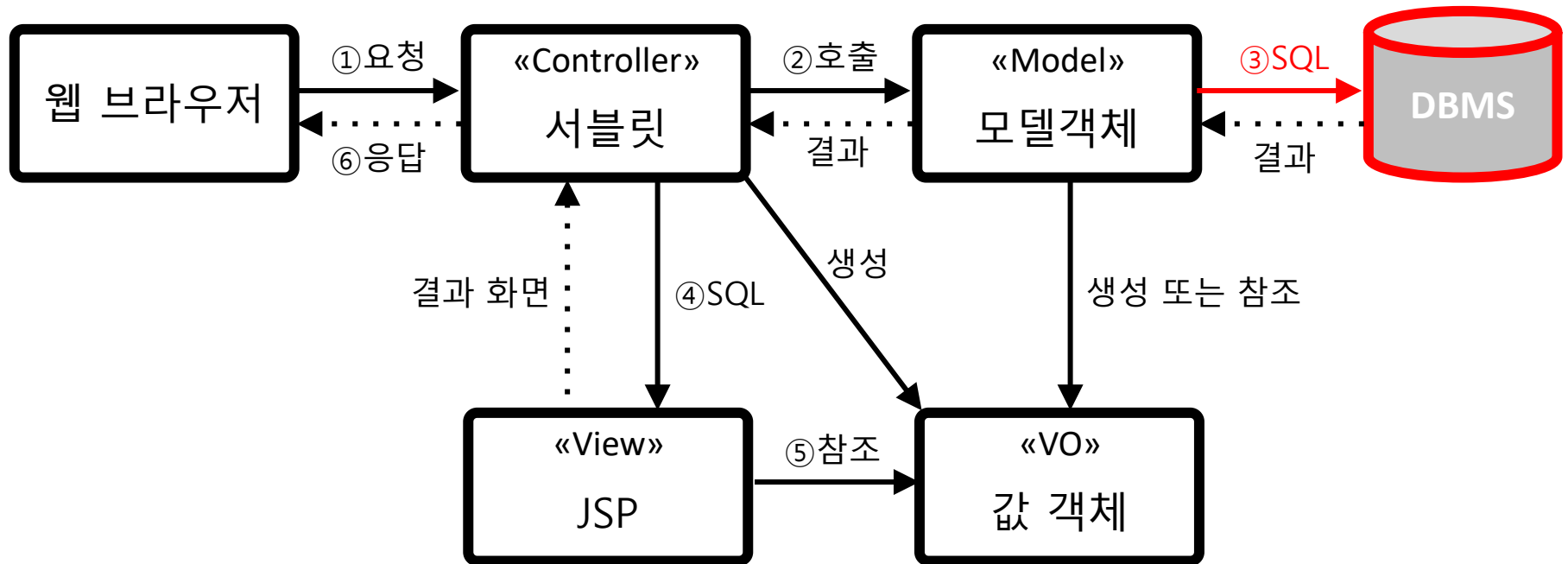
5.1 MVC 이해하기

데이터 가져오기



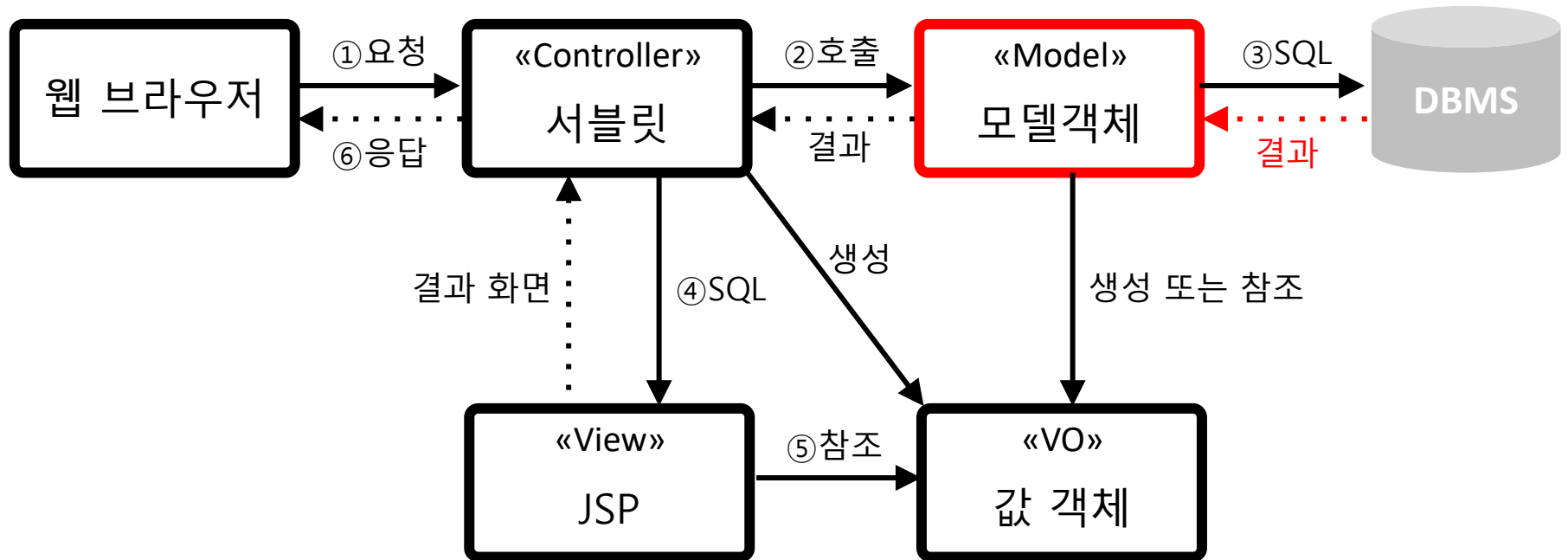
5.1 MVC 이해하기

데이터 가져오기



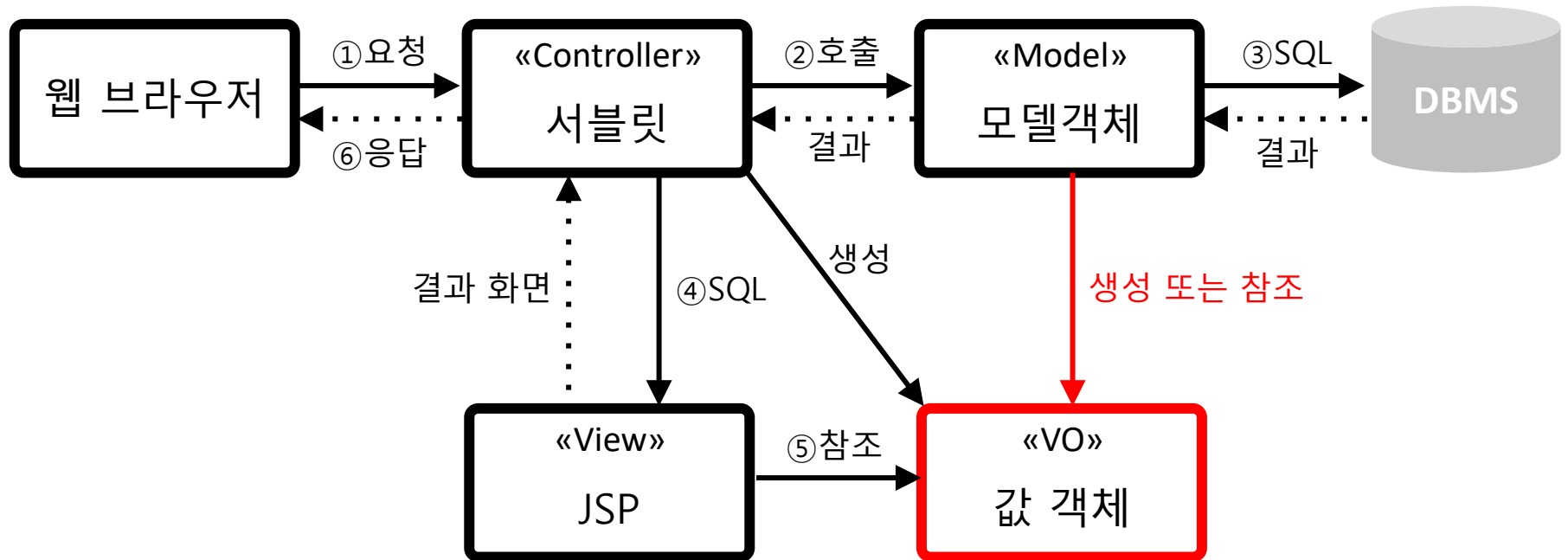
5.1 MVC 이해하기

데이터 가져오기



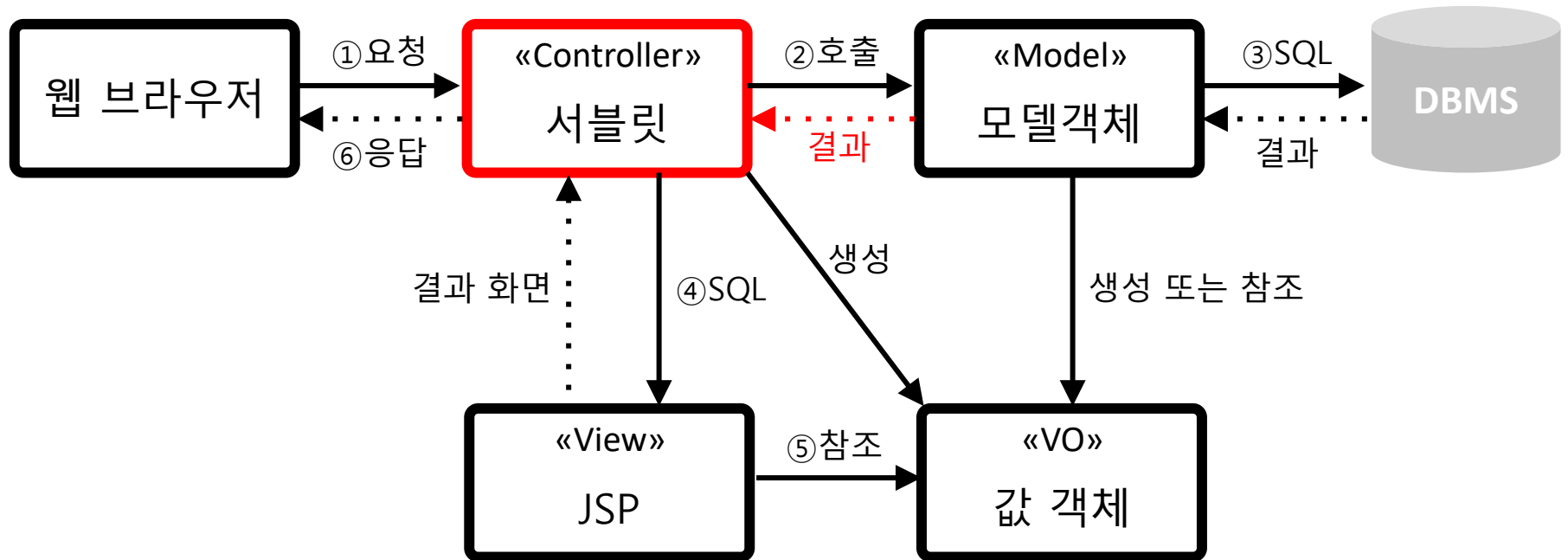
5.1 MVC 이해하기

데이터 가져오기



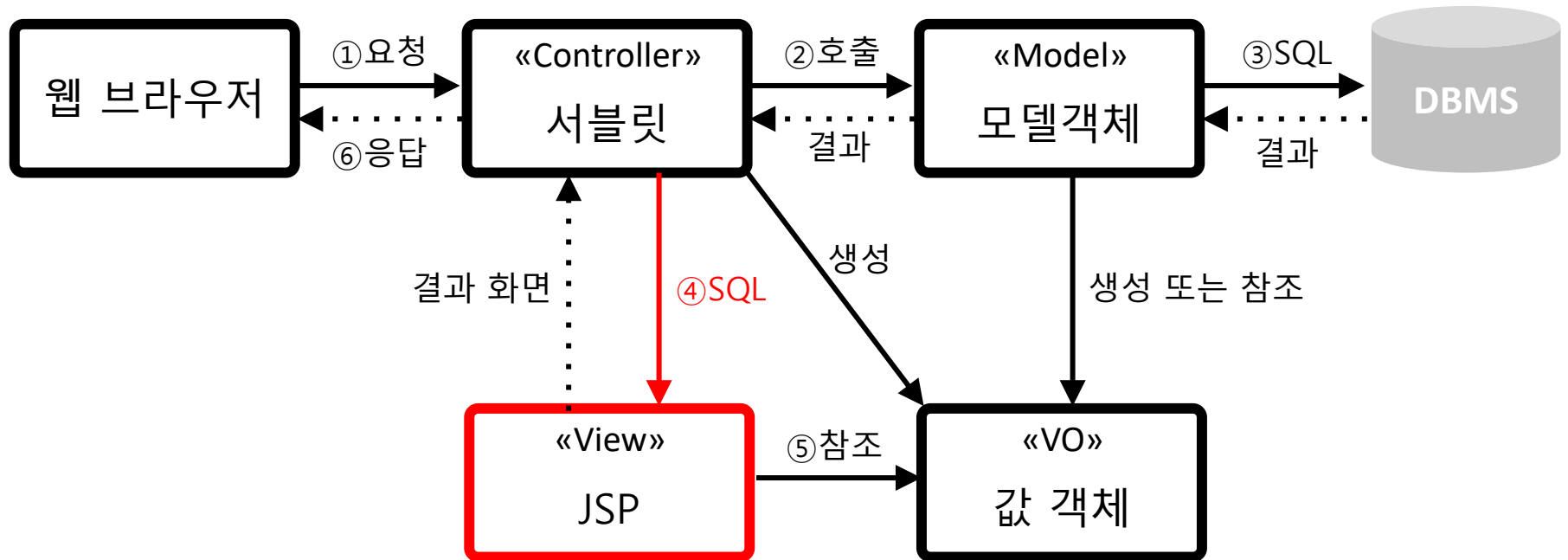
5.1 MVC 이해하기

데이터 가져오기



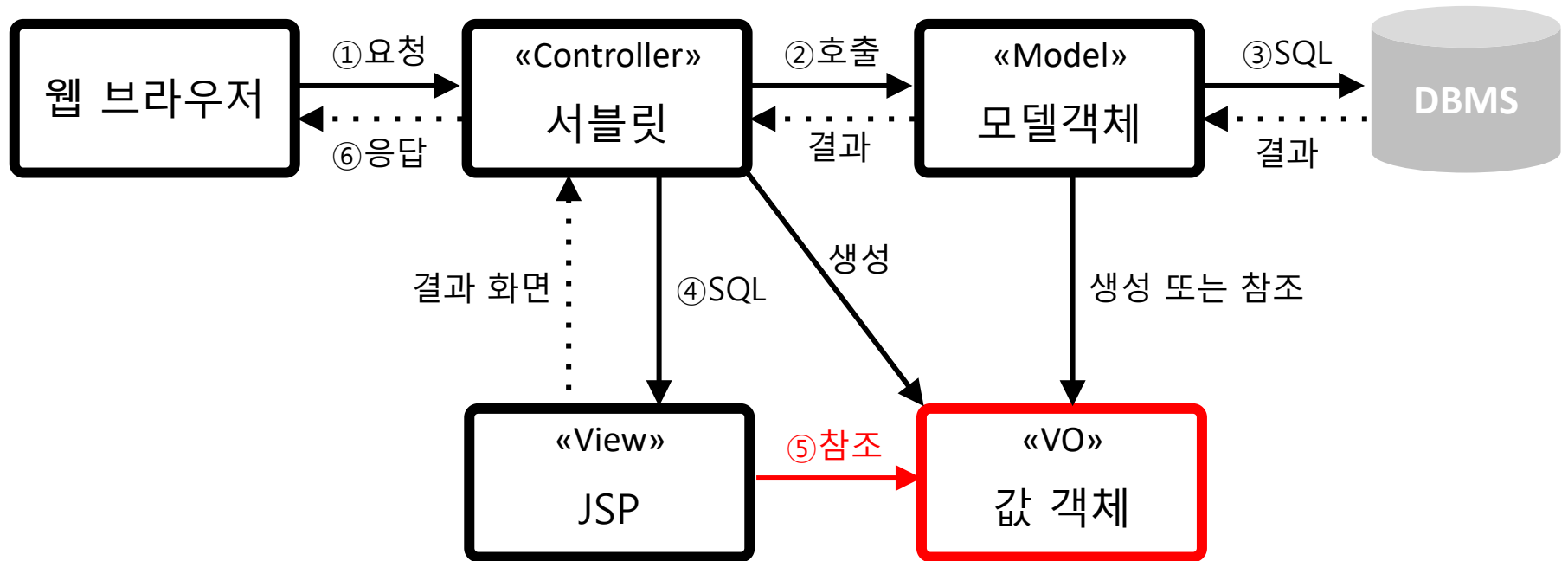
5.1 MVC 이해하기

데이터 가져오기



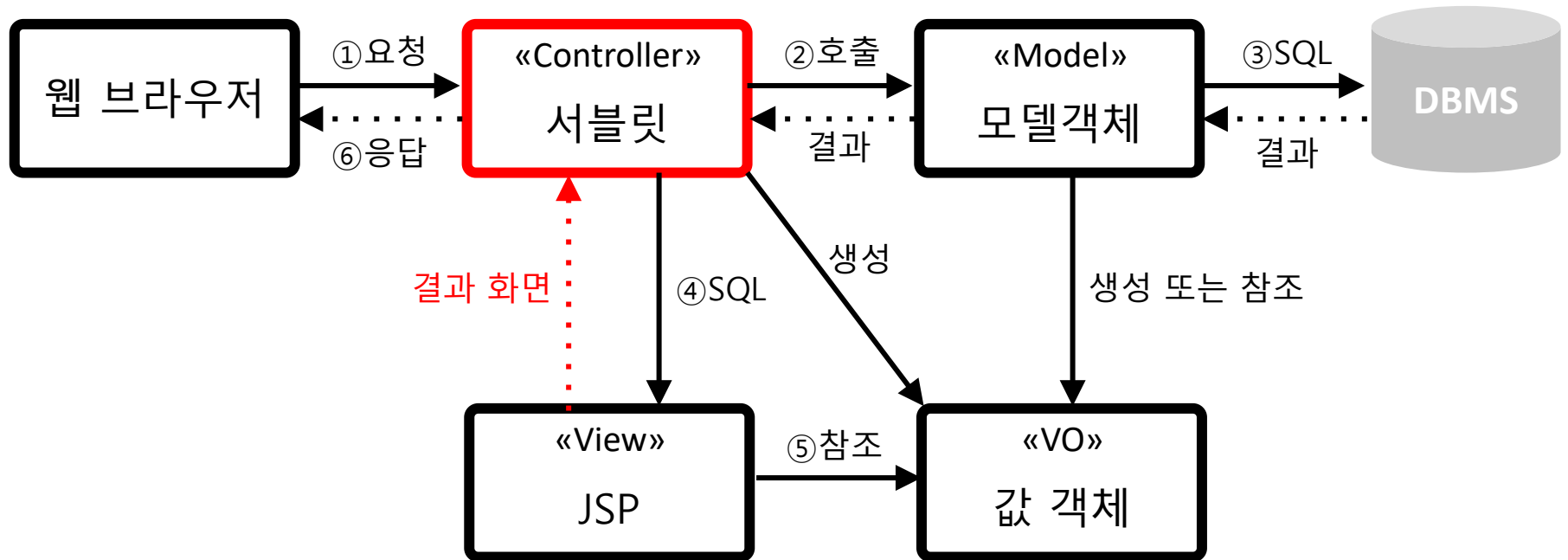
5.1 MVC 이해하기

데이터 가져오기



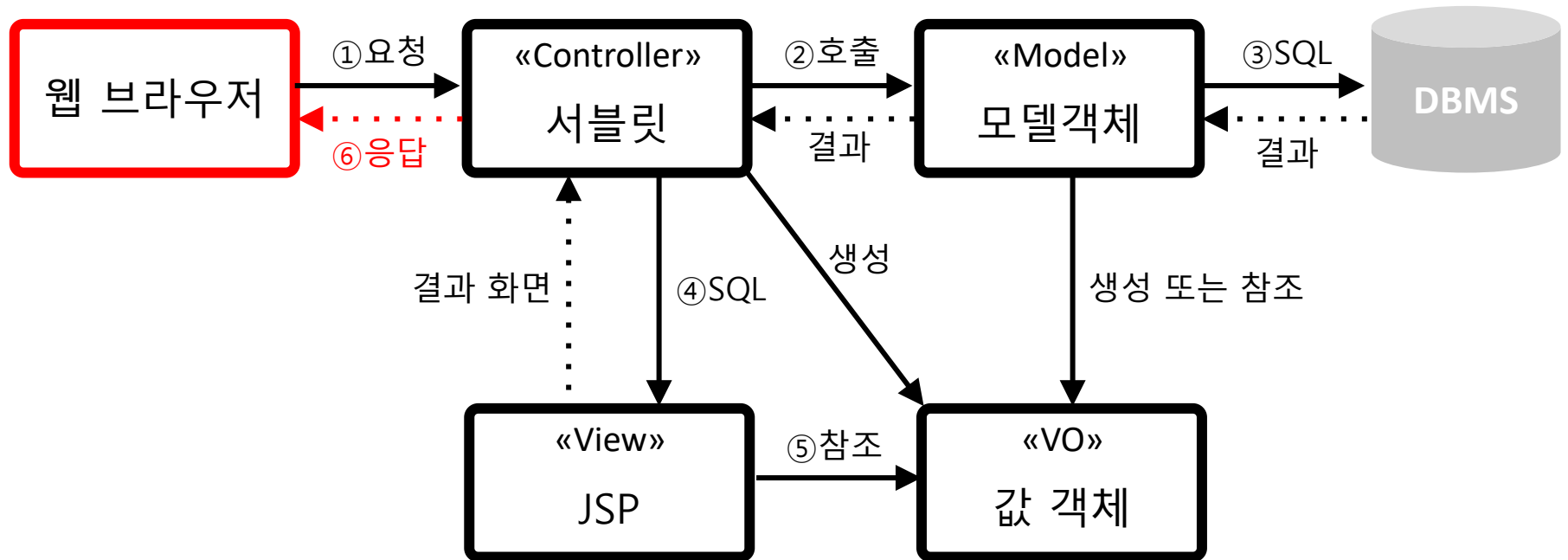
5.1 MVC 이해하기

데이터 가져오기



5.1 MVC 이해하기

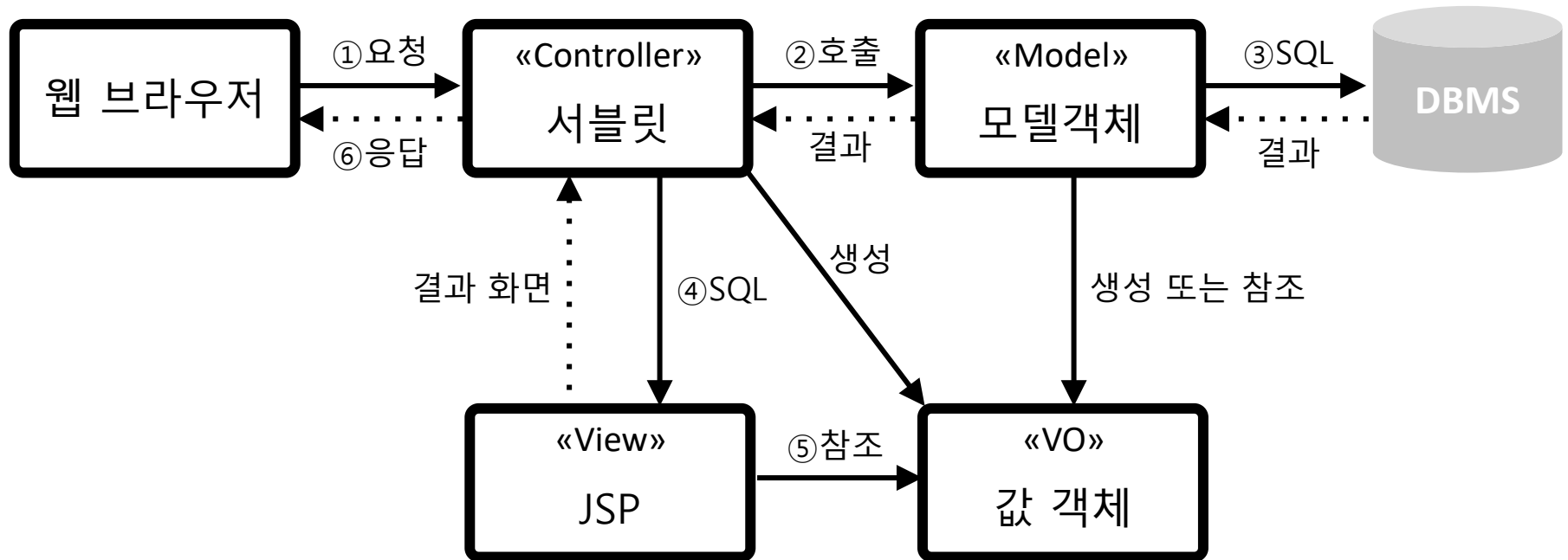
데이터 가져오기



5.2 뷰 컴포넌트와 JSP

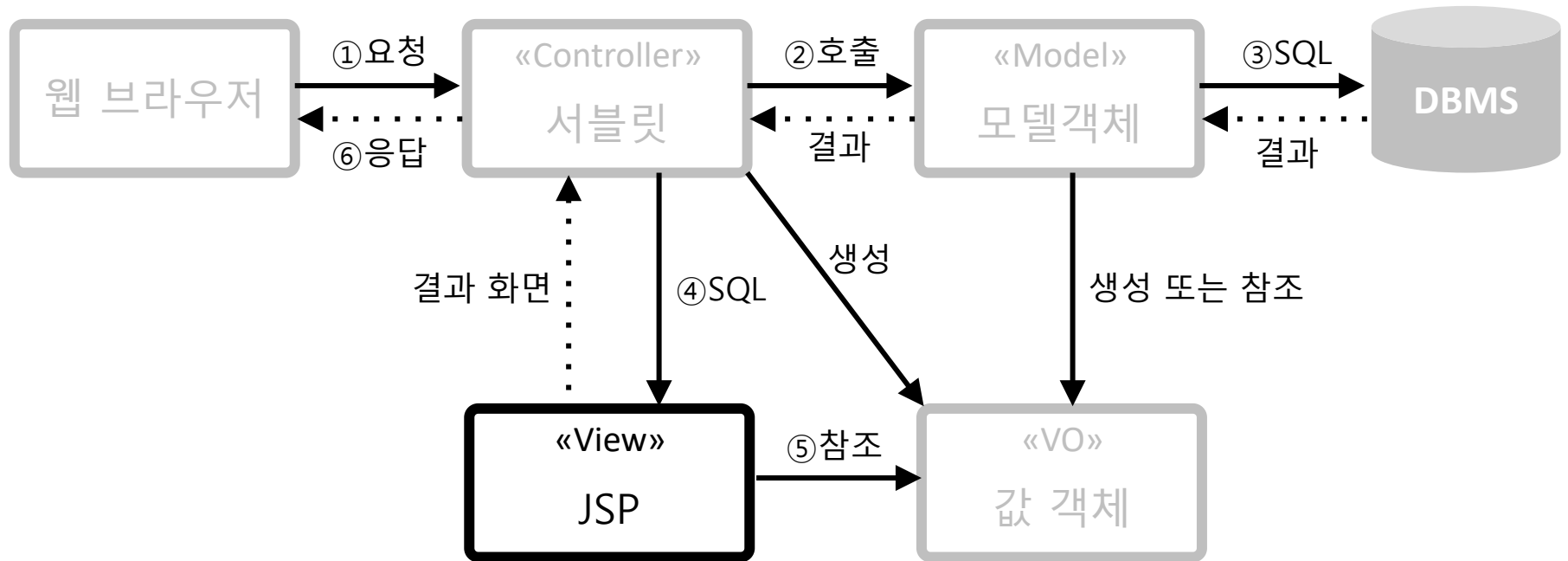
5.2 뷰 컴포넌트와 JSP

MVC 아키텍처



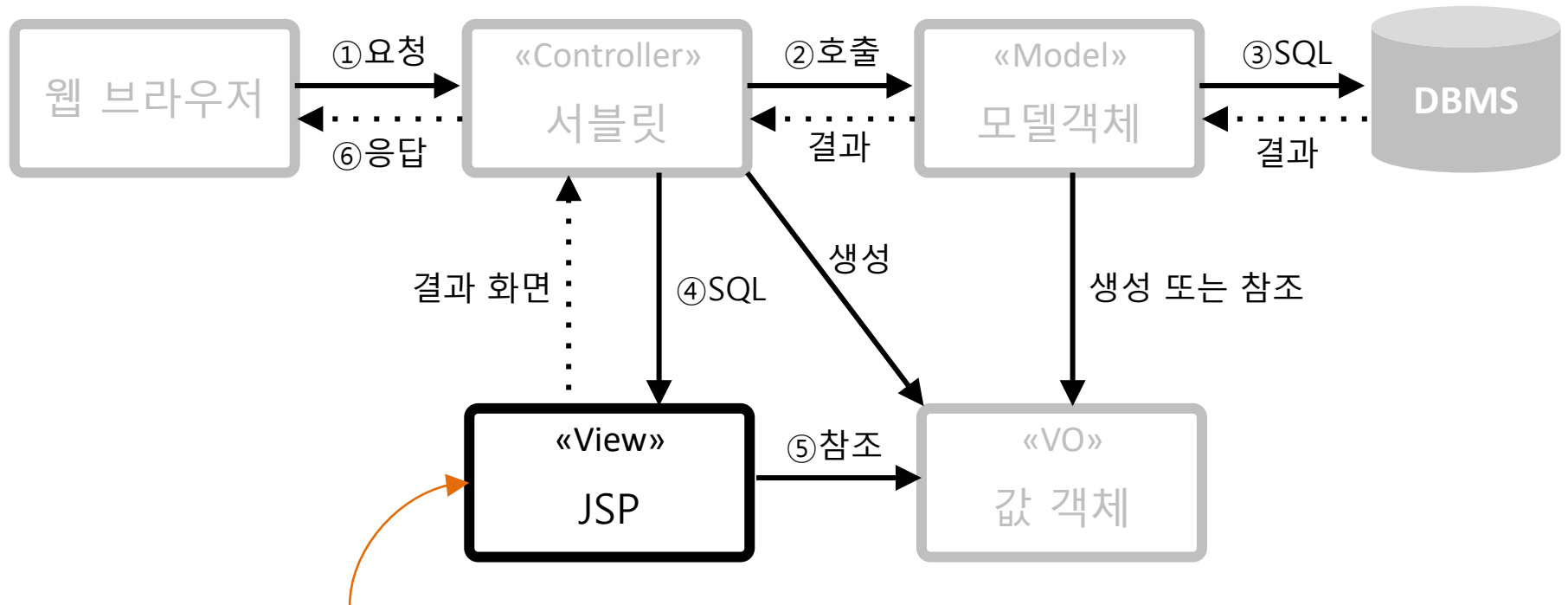
5.2 뷰 컴포넌트와 JSP

MVC 아키텍처



5.2 뷰 컴포넌트와 JSP

MVC 아키텍처



뷰 컴포넌트를 만들 때
JSP를 사용하는 이유?

5.2 뷰 컴포넌트와 JSP

“화면 만들기가
쉽다!”

5.2 뷰 컴포넌트와 JSP

JSP 사용 전

5.2 뷰 컴포넌트와 JSP

```
public class MemberAddServlet extends HttpServlet {  
    ...  
    protected void doGet(...) ... {  
        response.setContentType("text/html; charset=UTF-8");  
        PrintWriter out = response.getWriter();  
        out.println("<html><head><title>회원 등록</title></head>");  
        out.println("<body><h1>회원 등록</h1>");  
        out.println("<form action='add' method='post'>");  
        out.println("이름: <input type='text' name='name'><br>");  
        out.println("이메일: <input type='text' name='email'><br>");  
        out.println("암호: <input type='password' name='password'><br>");  
        out.println("<input type='submit' value='추가'>");  
        out.println("<input type='reset' value='취소'>");  
        out.println("</form>");  
        out.println("</body></html>");  
    }  
    ...  
}
```



출력문을 사용하여
일일이 HTML을
출력해야 한다.

5.2 뷰 컴포넌트와 JSP

JSP 사용 후

5.2 뷰 컴포넌트와 JSP

```
<%@ page language="java" contentType="text/html; charset=UTF-8" ...>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" ...>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>회원 등록</title>
</head>
<body>
<jsp:include page="/Header.jsp"/>
<h1>회원 등록</h1>
<form action='add' method='post'>
이름: <input type='text' name='name'><br>
이메일: <input type='text' name='email'><br>
암호: <input type='password' name='password'><br>
<input type='submit' value='추가'>
<input type='reset' value='취소'>
</form>
<jsp:include page="/Tail.jsp"/>
</body>
</html>
```

출력할 내용은 그냥
작성한다.



5.2 뷰 컴포넌트와 JSP

JSP 구동원리

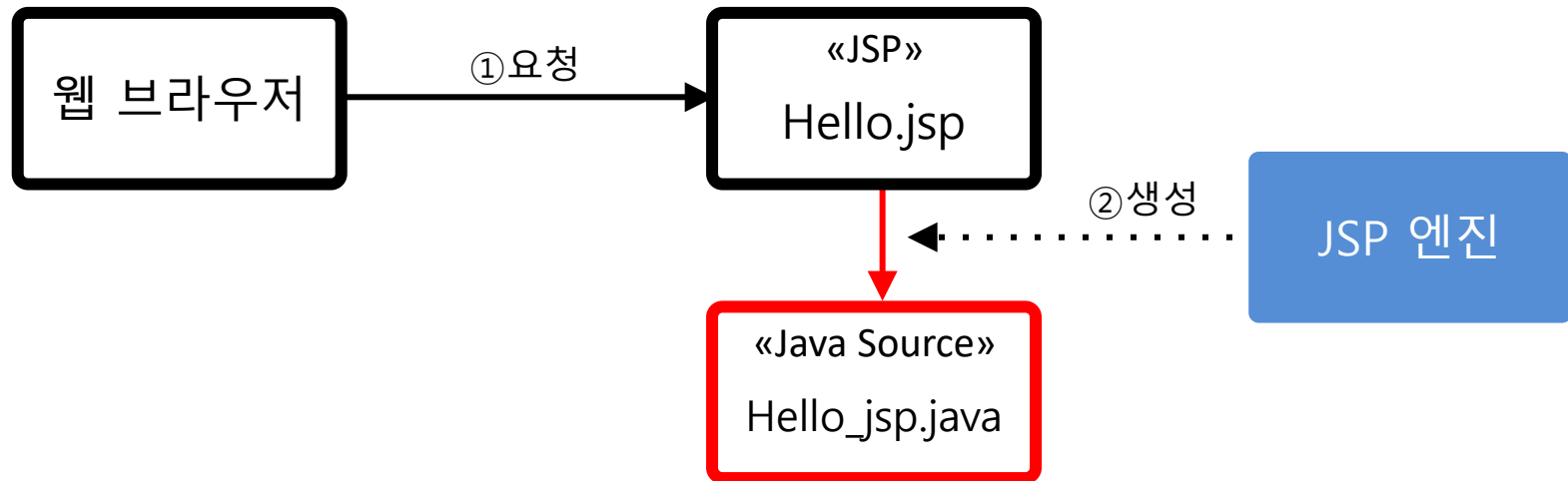
5.2 뷰 컴포넌트와 JSP

JSP 구동원리

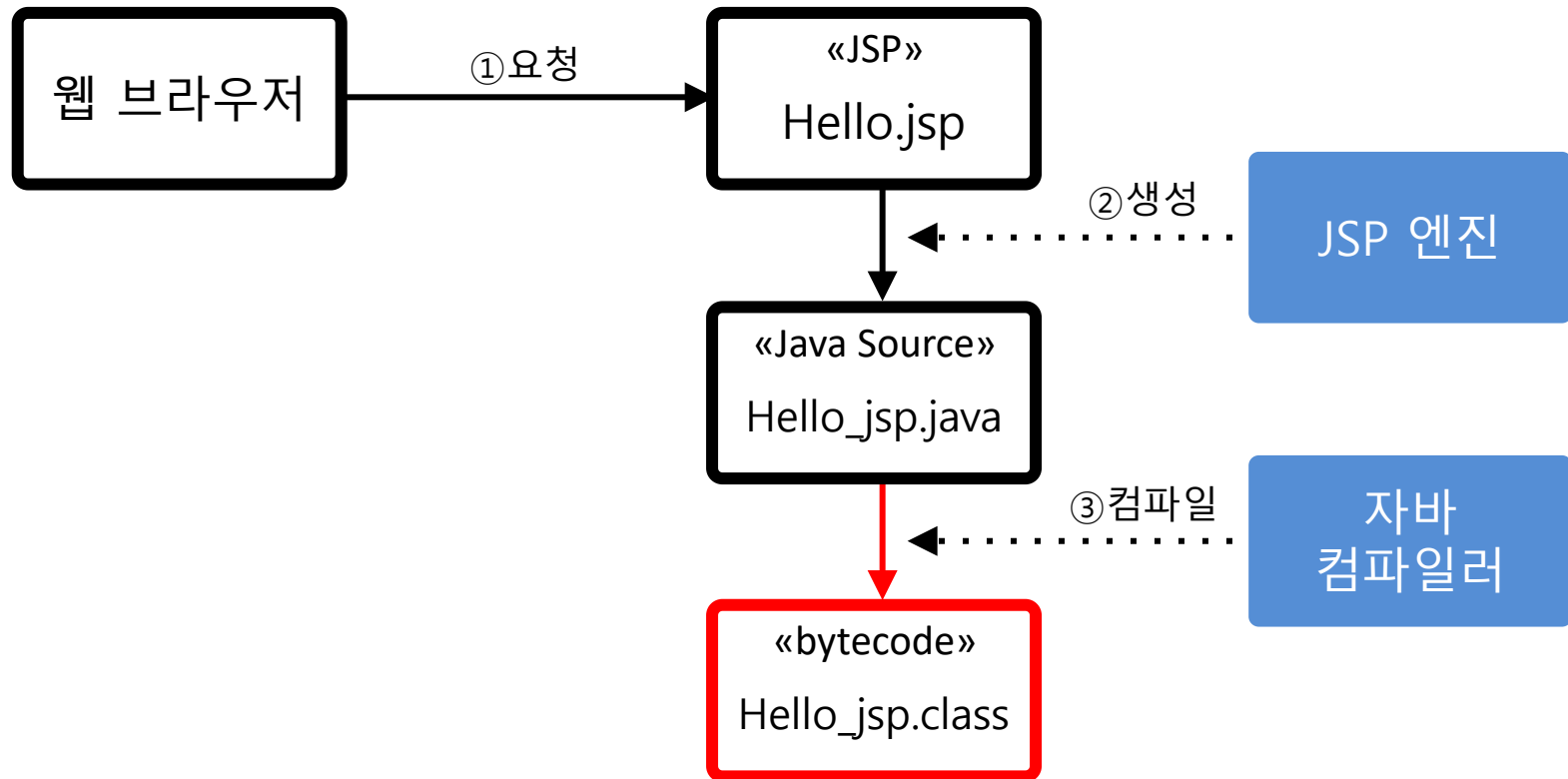


5.2 뷰 컴포넌트와 JSP

JSP 구동원리

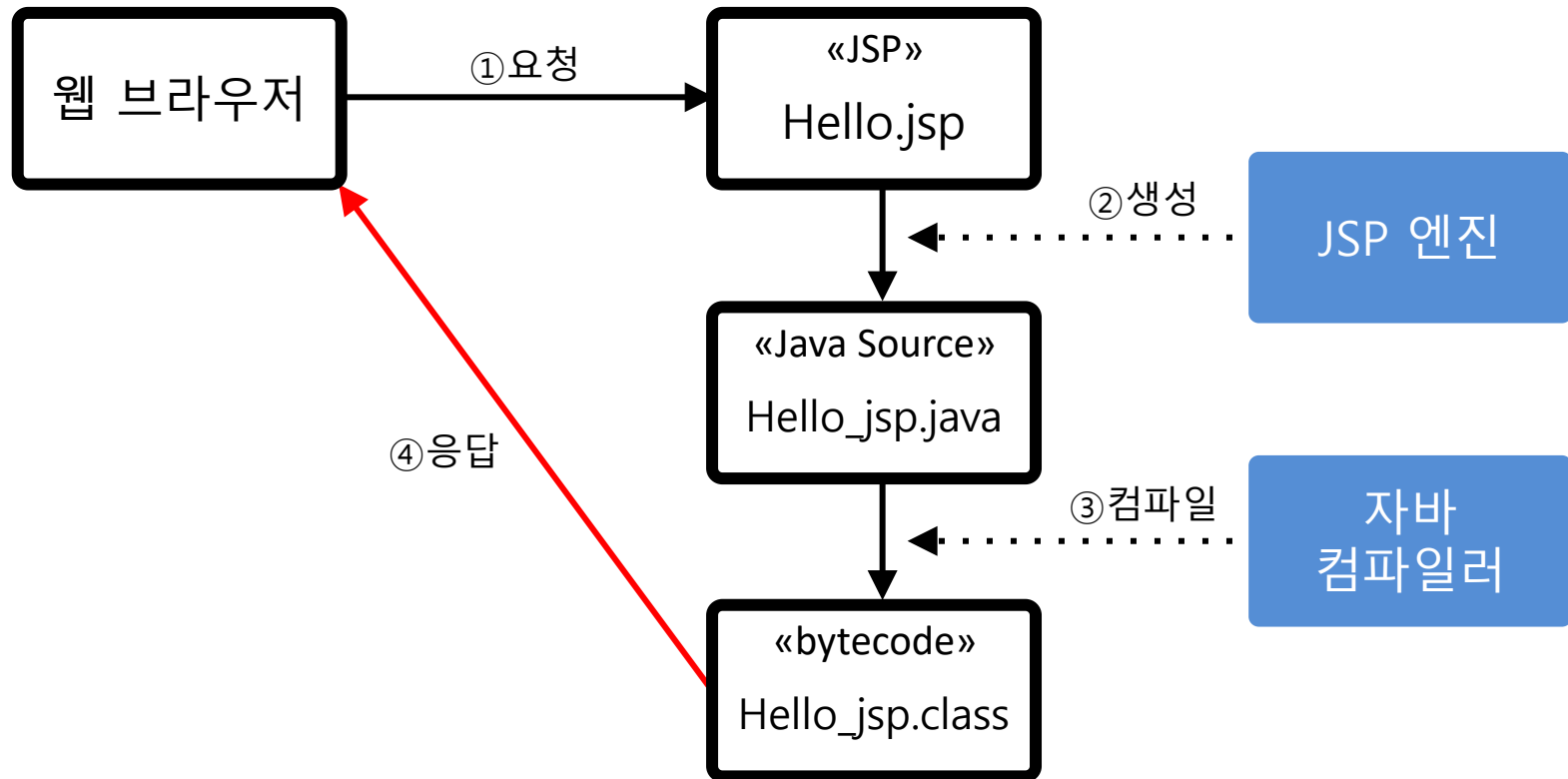


JSP 구동원리



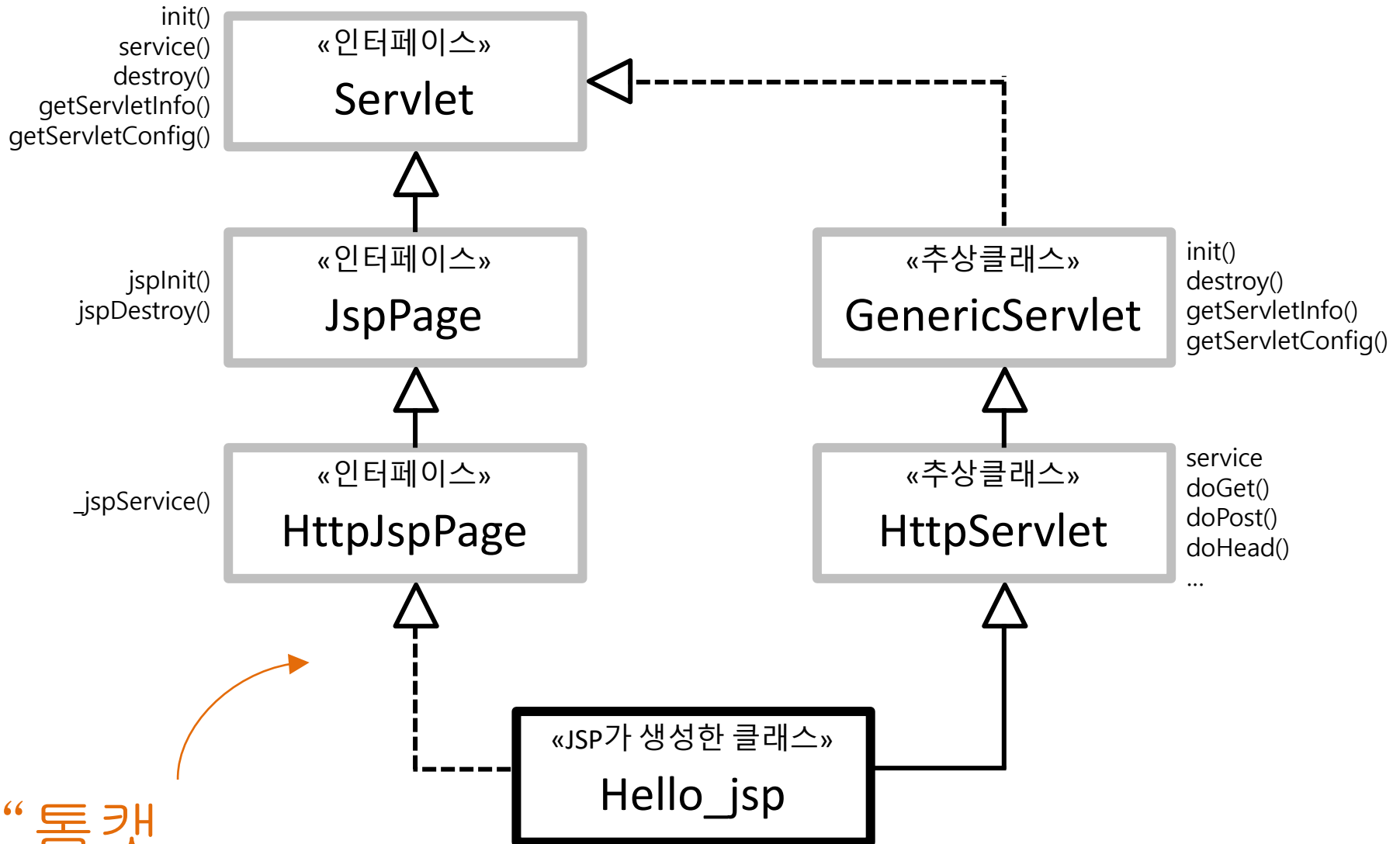
5.2 뷰 컴포넌트와 JSP

JSP 구동원리



JSP 클래스의 상속 관계

5.2 뷰 컴포넌트와 JSP



“톰캣
서버의 경우”

5.2 뷰 컴포넌트와 JSP

«JSP가 생성한 클래스»

Hello_jsp

5.2 뷰 컴포넌트와 JSP



톰캣 실행을 위한 임시 작업 디렉토리

5.2 뷰 컴포넌트와 JSP



톰캣 실행을 위한 임시 작업 디렉토리



예) c:\javaide\workspace\metadata\plugins\
→ org.eclipse.wst.server.core\tmp0\work
→ Catalina\localhost\...\jsp\
→ Hello_jsp.java
→ Hello_jsp.class

5.3 JSP의 주요 구성요소

5.3 JSP의 주요 구성 요소

템플릿 데이터

5.3 JSP의 주요 구성 요소

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>계산기</title>
</head>
<body>
<h2>JSP 계산기</h2>
```



템플릿 데이터
→ 자바 출력문을
생성

5.3 JSP의 주요 구성 요소

```
out.write("<!DOCTYPE html PUBLIC \"-//W3C//DTD HTML 4.01 Transitional//EN\" \n");  
out.write("\t\"http://www.w3.org/TR/html4/loose.dtd\">\n");  
out.write("<html>\n");  
out.write("<head>\n");  
out.write("<meta http-equiv=\"Content-Type\" content=\"text/html; charset=UTF-  
8\">\n");  
out.write("<title>계산기</title>\n");  
out.write("</head>\n");  
out.write("<body>\n");  
out.write("<h2>JSP 계산기</h2>\n");
```



JSP 엔진이 생성한 서블릿
코드의 일부

5.3 JSP의 주요 구성 요소

<h2>JSP 계산기</h2>

5.3 JSP의 주요 구성 요소

<h2>JSP 계산기</h2>

템플릿
데이터



5.3 JSP의 주요 구성 요소

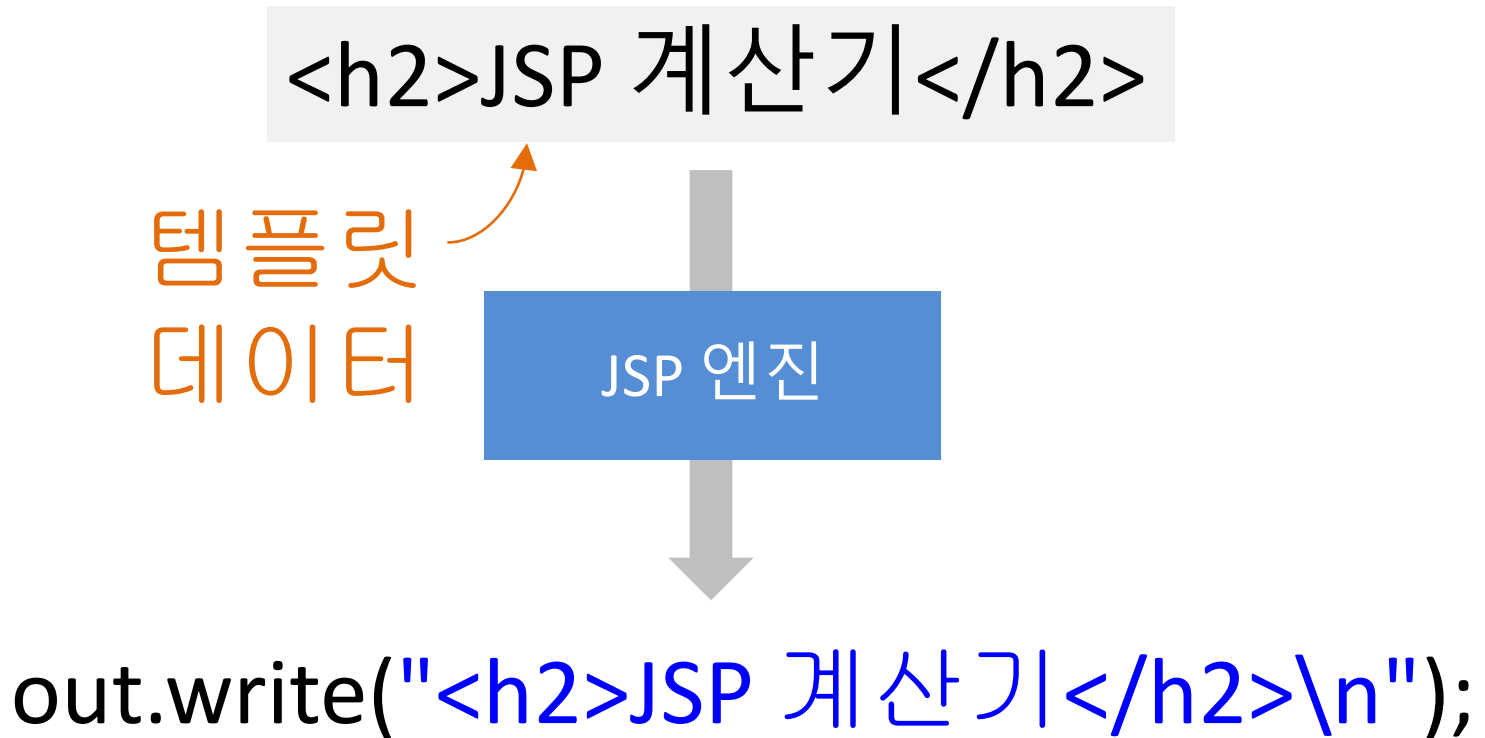
<h2>JSP 계산기</h2>

템플릿
데이터

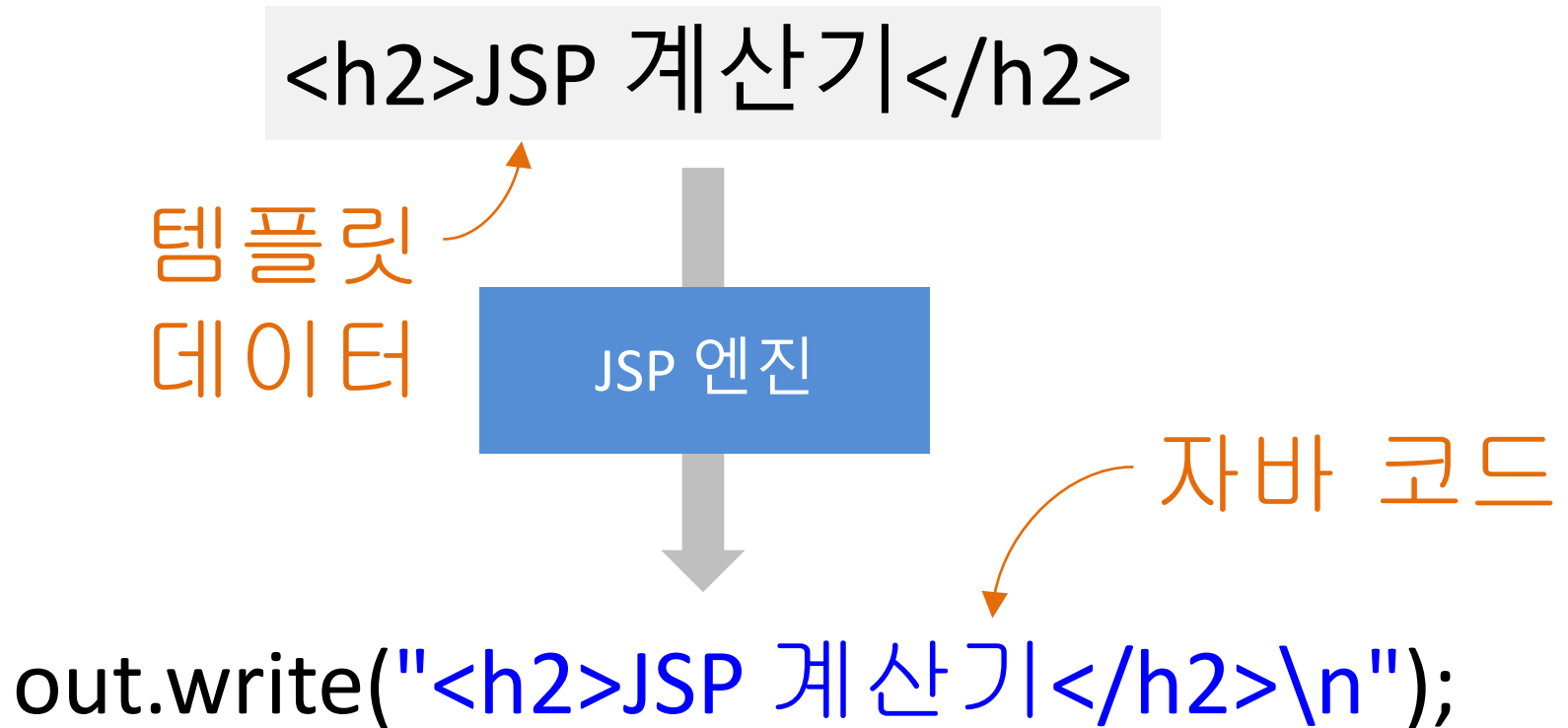


JSP 엔진

5.3 JSP의 주요 구성 요소



5.3 JSP의 주요 구성 요소



5.3 JSP의 주요 구성 요소

JSP 지시자

5.3 JSP의 주요 구성 요소

<%@ page

language="java"

contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8" %>

JSP 지시자



5.3 JSP의 주요 구성 요소

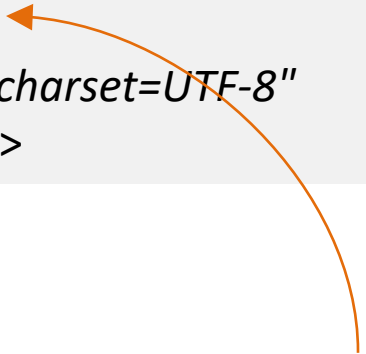
```
<%@ page
```

```
  language="java"
```

```
  contentType="text/html; charset=UTF-8"
```

```
  pageEncoding="UTF-8" %>
```

JSP 페이지에서
사용할
프로그래밍 언어
지정



5.3 JSP의 주요 구성 요소

```
<%@ page  
    language="java"  
    contentType="text/html; charset=UTF-8"  
    pageEncoding="UTF-8" %>
```

자바 코드
생성



```
public void _jspService(...) throws java.io.IOException, javax.servlet.ServletException {  
    final javax.servlet.jsp.PageContext pageContext;  
    javax.servlet.http.HttpSession session = null;  
    ...  
    try {  
        response.setContentType("text/html; charset=UTF-8");  
        ...  
    }  
}
```

5.3 JSP의 주요 구성 요소

```
<%@ page  
    language="java"  
    contentType="text/html; charset=UTF-8"  
    pageEncoding="UTF-8" %>
```

출력할 데이터의
문자집합을 지정하는
자바 코드 생성



```
public void _jspService(...) throws java.io.IOException, javax.servlet.ServletException {  
    final javax.servlet.jsp.PageContext pageContext;  
    javax.servlet.http.HttpSession session = null;  
    ...  
    try {  
        response.setContentType("text/html; charset=UTF-8");  
        ...  
    }  
}
```

5.3 JSP의 주요 구성 요소

```
<%@ page import="java.util.*,java.io.*" %>
```

JSP 페이지에서 사용할
자바 패키지 импорт 코드
생성

```
import javax.servlet.*;  
import javax.servlet.http.*;  
import javax.servlet.jsp.*;
```

import java.util.*;
import java.io.*;

```
public final class Calculator_jsp extends org.apache.jasper.runtime.HttpJspBase  
    implements org.apache.jasper.runtime.JspSourceDependent {
```

5.3 JSP의 주요 구성 요소

스크립트릿

5.3 JSP의 주요 구성 요소

<%

String v1 = "";

String v2 = "";

String result = "";

String[] selected = {"", "", "", ""};

//값이 있을 때만 꺼낸다.

if (request.getParameter("v1") != null) {

 v1 = request.getParameter("v1");

 v2 = request.getParameter("v2");

 String op = request.getParameter("op");

 result = calculate(Integer.parseInt(v1), Integer.parseInt(v2), op);

 if ("+".equals(op)) {

 ...

 }

}

%>

5.3 JSP의 주요 구성 요소

<%

```
String v1 = "";  
String v2 = "";  
String result = "";  
String[] selected = {"", "", "", ""};
```

//값이 있을 때만 꺼낸다.

```
if (request.getParameter("v1") != null) {  
    v1 = request.getParameter("v1");  
    v2 = request.getParameter("v2");  
    String op = request.getParameter("op");
```

```
    result = calculate(Integer.parseInt(v1), Integer.parseInt(v2), op);
```

```
    if ("+".equals(op)) {
```

```
        ...
```

```
    }
```

```
}
```

%>

5.3 JSP의 주요 구성 요소

<%

```
String v1 = "";  
String v2 = "";  
String result = "";  
String[] selected = {"", "", "", ""};
```

```
//값이 있을 때만 꺼낸다.
```

```
if (request.getParameter("v1") != null) {  
    v1 = request.getParameter("v1");  
    v2 = request.getParameter("v2");  
    String op = request.getParameter("op");
```

```
    result = calculate(Integer.parseInt(v1), Integer.parseInt(v2), op);
```

```
    if ("+".equals(op)) {
```

```
        ...
```

```
    }
```

```
}
```

%>

스크립트릿 → 서블릿
클래스의 `_jspService()`
안으로 그대로 복사됨



5.3 JSP의 주요 구성 요소

```
public void _jspService(...) throws ... {
```

```
...
```

```
try {
```

```
...
```

```
    out.write('\n');
```

```
String v1 = "";
```

```
String v2 = "";
```

```
String result = "";
```

```
String[] selected = {"", "", "", ""};
```

```
//값이 있을 때만 꺼낸다.
```

```
if (request.getParameter("v1") != null) {
```

```
    v1 = request.getParameter("v1");
```

```
    v2 = request.getParameter("v2");
```

```
    String op = request.getParameter("op");
```

```
    result = calculate(
```

```
        Integer.parseInt(v1),
```

```
        Integer.parseInt(v2),
```

```
        op);
```

JSP 페이지에 있던
스크립트릿



5.3 JSP의 주요 구성 요소

JSP 선언문

5.3 JSP의 주요 구성 요소

<%!

```
private String calculate(int a, int b, String op) {  
    int r = 0;  
  
    if ("+".equals(op)) {  
        r = a + b;  
    } else if ("-".equals(op)) {  
        r = a - b;  
    } else if ("*".equals(op)) {  
        r = a * b;  
    } else if ("/".equals(op)) {  
        r = a / b;  
    }  
  
    return Integer.toString(r);  
}
```

%>

5.3 JSP의 주요 구성 요소

<%!

```
private String calculate(int a, int b, String op) {  
    int r = 0;  
  
    if ("+".equals(op)) {  
        r = a + b;  
    } else if ("-".equals(op)) {  
        r = a - b;  
    } else if ("*".equals(op)) {  
        r = a * b;  
    } else if ("/".equals(op)) {  
        r = a / b;  
    }  
  
    return Integer.toString(r);  
}
```

%>

클래스 선언부에
들어갈
변수 또는 메서드 선언



5.3 JSP의 주요 구성 요소

```
public final class Calculator_jsp extends  
    org.apache.jasper.runtime.HttpJspBase  
    implements org.apache.jasper.runtime.JspSourceDependent {
```

```
private String calculate(int a, int b, String op) {
```

```
    int r = 0;
```

```
    if ("+".equals(op)) {
```

```
        r = a + b;
```

```
    } else if ("-".equals(op)) {
```

```
        r = a - b;
```

```
    } else if ("*".equals(op)) {
```

```
        r = a * b;
```

```
    } else if ("/".equals(op)) {
```

```
        r = a / b;
```

```
    }
```

```
    return Integer.toString(r);
```

```
}
```

```
...
```

JSP 선언문을 그대로
가쳐옴



5.3 JSP의 주요 구성 요소

JSP 표현식

5.3 JSP의 주요 구성 요소

<h2>JSP 계산기</h2>

<form action="Calculator.jsp" method="get">

 <input type="text" name="v1" size="4" value="<%=v1%>">

 <select name="op">

 <option value="+" <%=selected[0]%>>+</option>

 <option value="-" <%=selected[1]%>>-</option>

 <option value="*" <%=selected[2]%>>*</option>

 <option value="/" <%=selected[3]%>>/</option>

 </select>

 <input type="text" name="v2" size="4" value="<%=v2%>">

 <input type="submit" value="=">

 <input type="text" size="8" value="<%=result%>">

</form>

5.3 JSP의 주요 구성 요소

<h2>JSP 계산기</h2>

<form action="Calculator.jsp" method="get">

<input type="text" name="v1" size="4" value="<%=v1%>">

<select name="op">

<option value="+" <%=selected[0]%>>+</option>

<option value="-" <%=selected[1]%>>-</option>

<option value="*" <%=selected[2]%>>*</option>

<option value="/" <%=selected[3]%>>/</option>

</select>

<input type="text" name="v2" size="4" value="<%=v2%>">

<input type="submit" value="=">

<input type="text" size="8" value="<%=result%>">

</form>

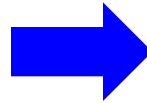
자바 출력문을

생성

out.print(v1);

5.3 JSP의 주요 구성 요소

```
or.jsp" method="get">
' name="v1" size="4" value="<%=v1%>">
">
e="+" <%=selected[0]%>>+</option>
e="-" <%=selected[1]%>>-</option>
e="*" <%=selected[2]%>>*</option>
e="/" <%=selected[3]%>>/</option>
' name="v2" size="4" value="<%=v2%>">
nit" value="=">
' size="8" value="<%=result%>"><br>
```

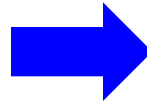


```
out.write("<h2>JSP 계 산 기</h2>\n");
out.write("<form action=\"Calculator.jsp\">");
out.write("\t<input type=\"text\" name=\"v1\">");
out.print(v1);
out.write("\n");
out.write("\t<select name=\"op\">\n");
out.write("\t\t<option value=\"+\">");
out.print(selected[0]);
out.write(">+</option>\n");
out.write("\t\t<option value=\"-\">");
out.print(selected[1]);
out.write(">-</option>\n");
out.write("\t\t<option value=\"*\">");
out.print(selected[2]);
out.write(">*</option>\n");
out.write("\t\t<option value=\"/\">");
out.print(selected[3]);
out.write(">/</option>\n");
out.write("\t</select> \n");
out.write("\t<input type=\"text\" name=\"v2\">");
out.print(v2);
```

5.3 JSP의 주요 구성 요소

```
or.jsp" method="get">
' name="v1" size="4" value="<%=v1%>">
">
e="+" <%=selected[0]%>>+</option>
e="-" <%=selected[1]%>>-</option>
e="*" <%=selected[2]%>>*</option>
e="/" <%=selected[3]%>>/</option>

' name="v2" size="4" value="<%=v2%>">
nit" value="=">
' size="8" value="<%=result%>"><br>
```

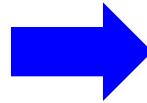


```
out.write("<h2>JSP 계 산 기</h2>\n");
out.write("<form action=\"Calculator.jsp\">");
out.write("\t<input type=\"text\" name=\"v1\">");
out.print(v1);
out.write("\n");
out.write("\t<select name=\"op\">\n");
out.write("\t\t<option value=\"+\">");
out.print(selected[0]);
out.write(">+</option>\n");
out.write("\t\t<option value=\"-\">");
out.print(selected[1]);
out.write(">-</option>\n");
out.write("\t\t<option value=\"*\">");
out.print(selected[2]);
out.write(">*</option>\n");
out.write("\t\t<option value=\"/\">");
out.print(selected[3]);
out.write(">/</option>\n");
out.write("\t</select> \n");
out.write("\t<input type=\"text\" name=\"v2\">");
out.print(v2);
```

5.3 JSP의 주요 구성 요소

```
or.jsp" method="get">
' name="v1" size="4" value="<%=v1%>">
">
e="+" <%=selected[0]%>>+</option>
e="-" <%=selected[1]%>>-</option>
e="*" <%=selected[2]%>>*</option>
e="/" <%=selected[3]%>>/</option>

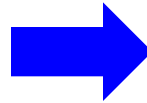
' name="v2" size="4" value="<%=v2%>">
nit" value="=">
' size="8" value="<%=result%>"><br>
```



```
out.write("<h2>JSP 계 산 기</h2>\n");
out.write("<form action=\"Calculator.jsp\">");
out.write("\t<input type=\"text\" name=\"v1\">");
out.print(v1);
out.write("\n");
out.write("\t<select name=\"op\">\n");
out.write("\t\t<option value=\"+\">+</option>");
out.print(selected[0]);
out.write(">+</option>\n");
out.write("\t\t<option value=\"-\">-</option>");
out.print(selected[1]);
out.write(">-</option>\n");
out.write("\t\t<option value=\"*\">*</option>");
out.print(selected[2]);
out.write(">*</option>\n");
out.write("\t\t<option value=\"/\">/</option>");
out.print(selected[3]);
out.write(">/</option>\n");
out.write("\t</select> \n");
out.write("\t<input type=\"text\" name=\"v2\">");
out.print(v2);
```

5.3 JSP의 주요 구성 요소

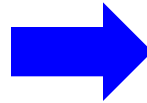
```
or.jsp" method="get">
' name="v1" size="4" value="<%=v1%>">
">
e="+" <%=selected[0]%>>+</option>
e="-" <%=selected[1]%>>-</option>
e="*" <%=selected[2]%>>*</option>
e="/" <%=selected[3]%>>/</option>
' name="v2" size="4" value="<%=v2%>">
nit" value="=">
' size="8" value="<%=result%>"><br>
```



```
out.write("<h2>JSP 계 산 기</h2>\n");
out.write("<form action=\"Calculator.jsp\">");
out.write("\t<input type=\"text\" name=\"v1\">");
out.print(v1);
out.write("\n");
out.write("\t<select name=\"op\">\n");
out.write("\t\t<option value=\"+\">+</option>");
out.print(selected[0]);
out.write(">+</option>\n");
out.write("\t\t<option value=\"-\">-</option>");
out.print(selected[1]);
out.write(">-</option>\n");
out.write("\t\t<option value=\"*\">*</option>");
out.print(selected[2]);
out.write(">*</option>\n");
out.write("\t\t<option value=\"/\">/</option>");
out.print(selected[3]);
out.write(">/</option>\n");
out.write("\t</select> \n");
out.write("\t<input type=\"text\" name=\"v2\">");
out.print(v2);
```

5.3 JSP의 주요 구성 요소

```
or.jsp" method="get">
' name="v1" size="4" value="<%=v1%>">
">
e="+" <%=selected[0]%>>+</option>
e="-" <%=selected[1]%>>-</option>
e="*" <%=selected[2]%>>*</option>
e="/" <%=selected[3]%>></option>
' name="v2" size="4" value="<%=v2%>">
nit" value="=">
' size="8" value="<%=result%>"><br>
```

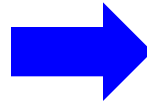


```
out.write("<h2>JSP 계 산 기</h2>\n");
out.write("<form action=\"Calculator.jsp\">");
out.write("\t<input type=\"text\" name=\"v1\">");
out.print(v1);
out.write("\n");
out.write("\t<select name=\"op\">\n");
out.write("\t\t<option value=\"+\">+</option>");
out.print(selected[0]);
out.write(">+</option>\n");
out.write("\t\t<option value=\"-\">-</option>");
out.print(selected[1]);
out.write(">-</option>\n");
out.write("\t\t<option value=\"*\">*</option>");
out.print(selected[2]);
out.write(">*</option>\n");
out.write("\t\t<option value=\"/\">/</option>");
out.print(selected[3]);
out.write("></option>\n");
out.write("\t</select> \n");
out.write("\t<input type=\"text\" name=\"v2\">");
out.print(v2);
```

5.3 JSP의 주요 구성 요소

```
or.jsp" method="get">
' name="v1" size="4" value="<%=v1%>">
">
e="+" <%=selected[0]%>>+</option>
e="-" <%=selected[1]%>>-</option>
e="*" <%=selected[2]%>>*</option>
e="/" <%=selected[3]%>>/</option>

' name="v2" size="4" value="<%=v2%>">
nit" value="=">
' size="8" value="<%=result%>"><br>
```



```
out.write("<h2>JSP 계 산 기</h2>\n");
out.write("<form action=\"Calculator.jsp\">");
out.write("\t<input type=\"text\" name=\"v1\">");
out.print(v1);
out.write("\n");
out.write("\t<select name=\"op\">\n");
out.write("\t\t<option value=\"+\">+</option>");
out.print(selected[0]);
out.write(">+</option>\n");
out.write("\t\t<option value=\"-\">-</option>");
out.print(selected[1]);
out.write(">-</option>\n");
out.write("\t\t<option value=\"*\">*</option>");
out.print(selected[2]);
out.write(">*</option>\n");
out.write("\t\t<option value=\"/\">/</option>");
out.print(selected[3]);
out.write(">/</option>\n");
out.write("\t</select> \n");
out.write("\t<input type=\"text\" name=\"v2\">");
out.print(v2);
```