

서블릿 프로그래밍

- 웹 애플리케이션과 CGI 규칙
- 실습

웹 애플리케이션과 CGI 규칙

CGI 규칙

웹 브라우저

웹 서버

CGI 규칙



CGI 규칙



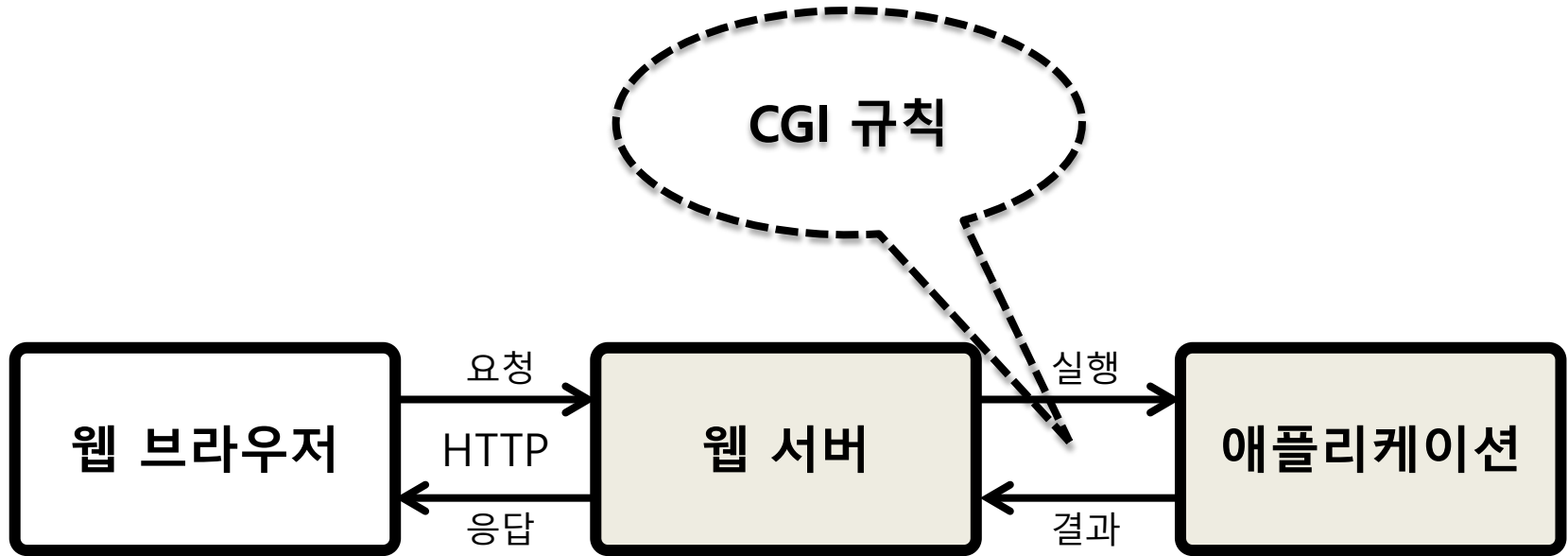
CGI 규칙



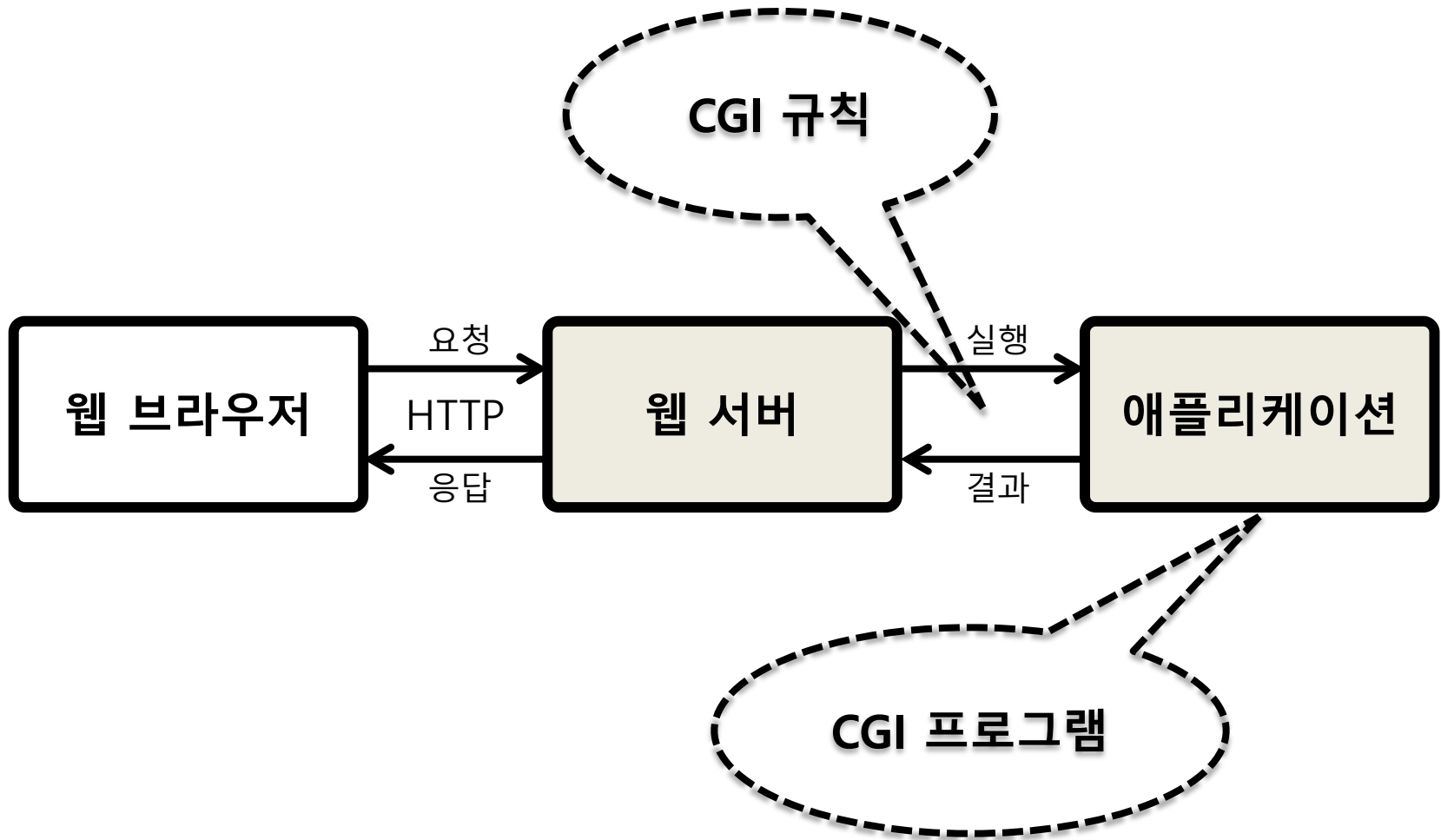
CGI 규칙



CGI 규칙



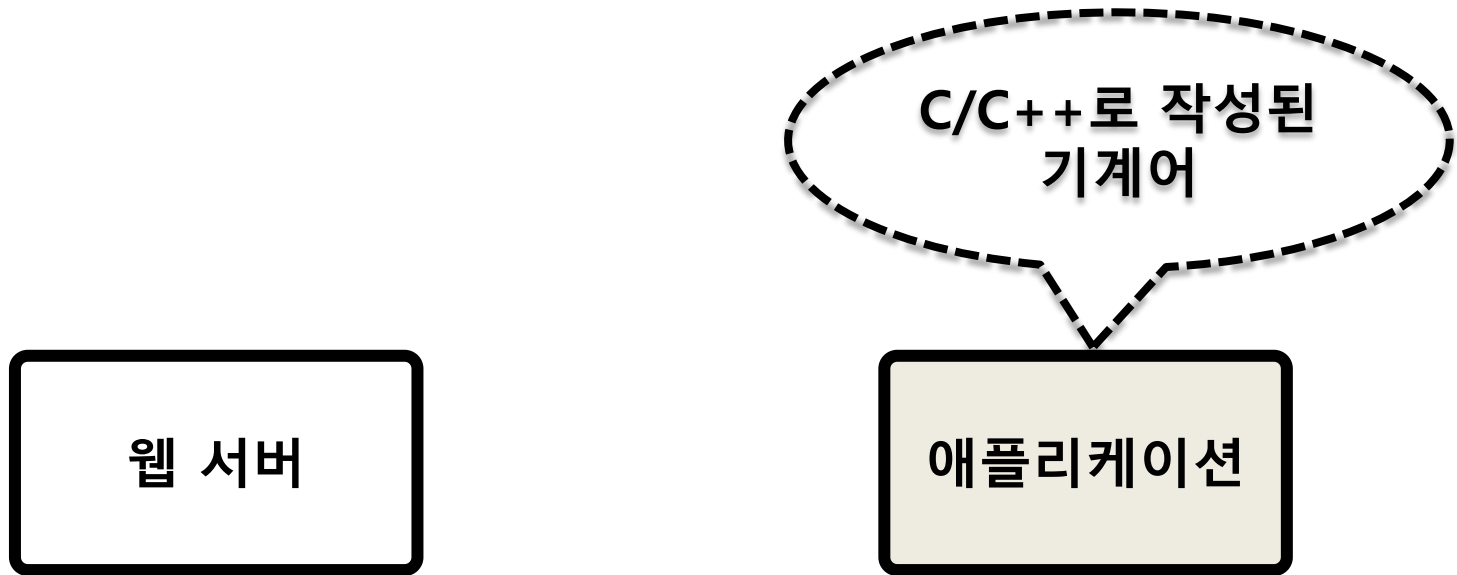
CGI 규칙



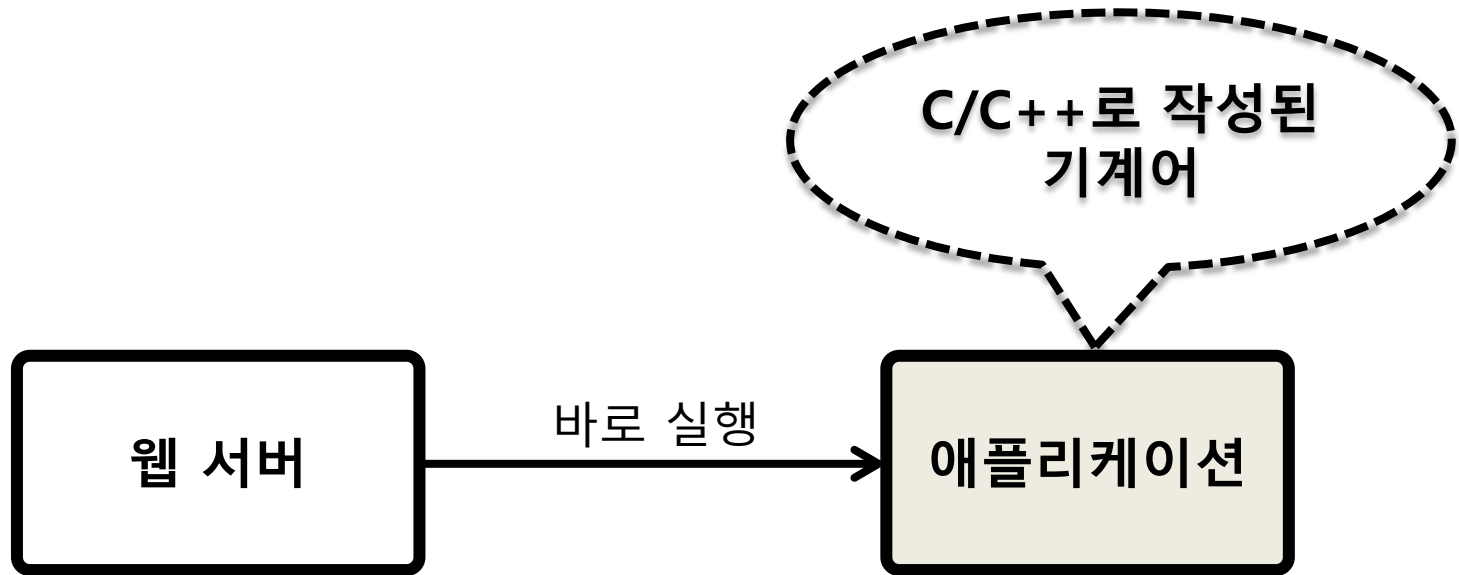
CGI 프로그램 유형

-컴파일 방식-

컴파일 방식



컴파일 방식



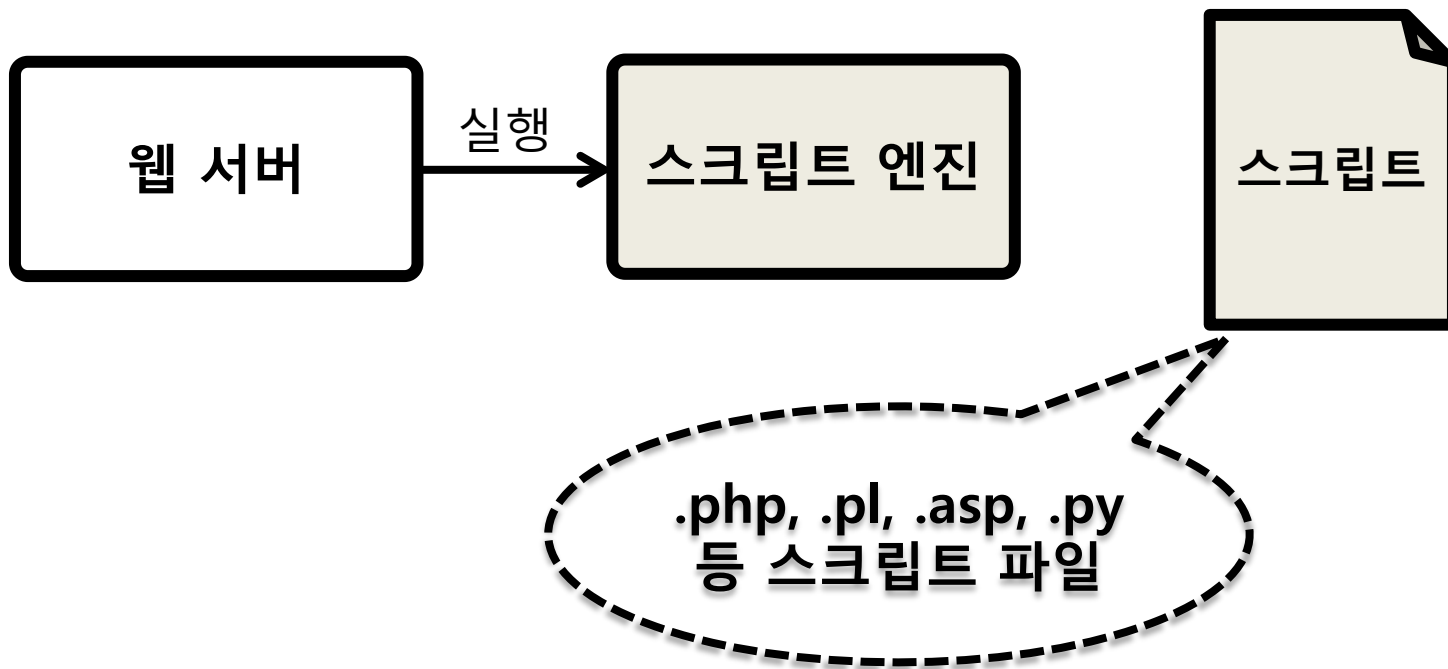
CGI 프로그램 유형

-인터프리터 방식-

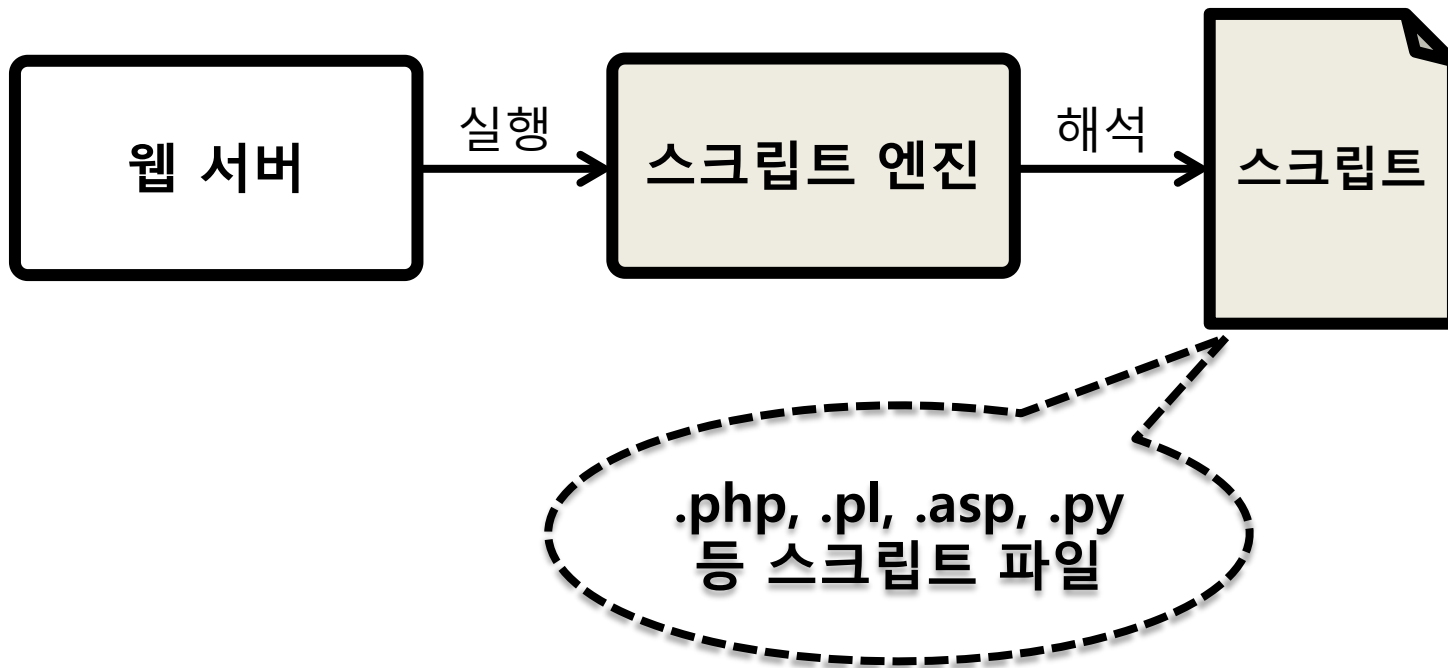
인터프리터 방식



인터프리터 방식



인터프리터 방식



자바 웹 애플리케이션 실행 방식

자바 웹 애플리케이션 실행



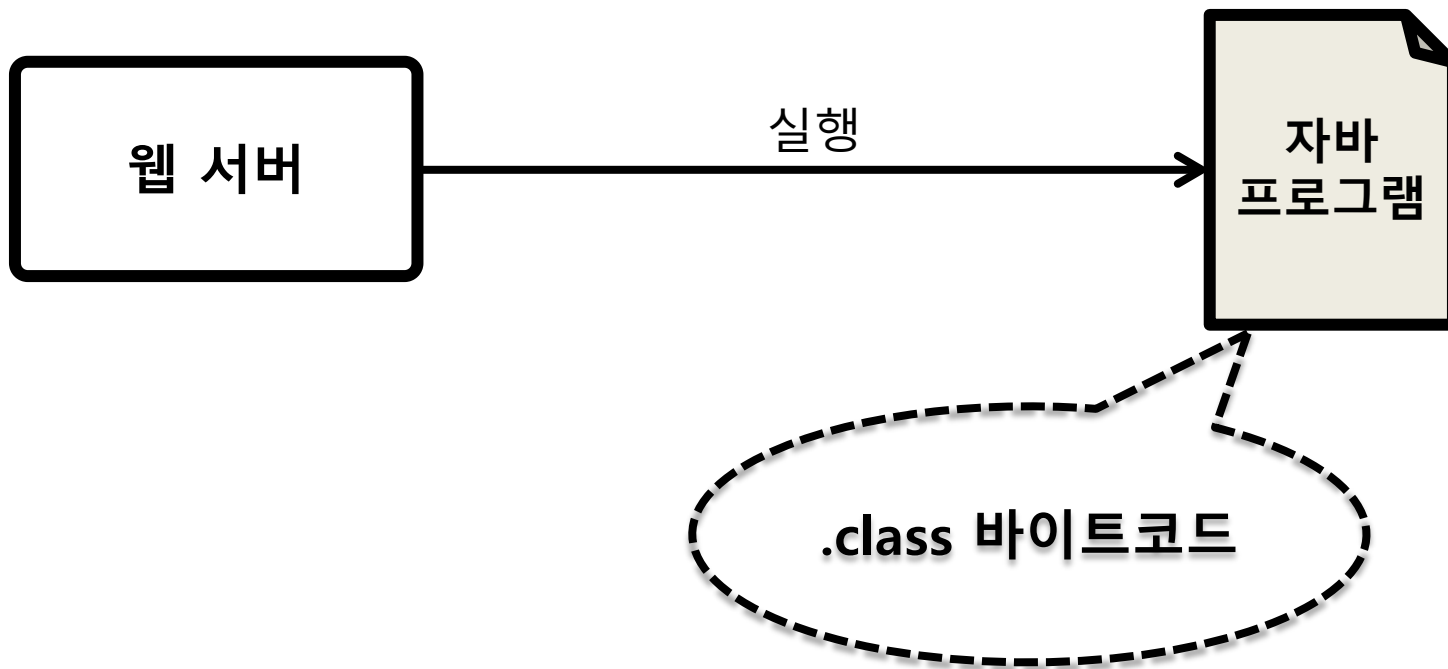
웹 서버

자바
프로그램

자바 웹 애플리케이션 실행

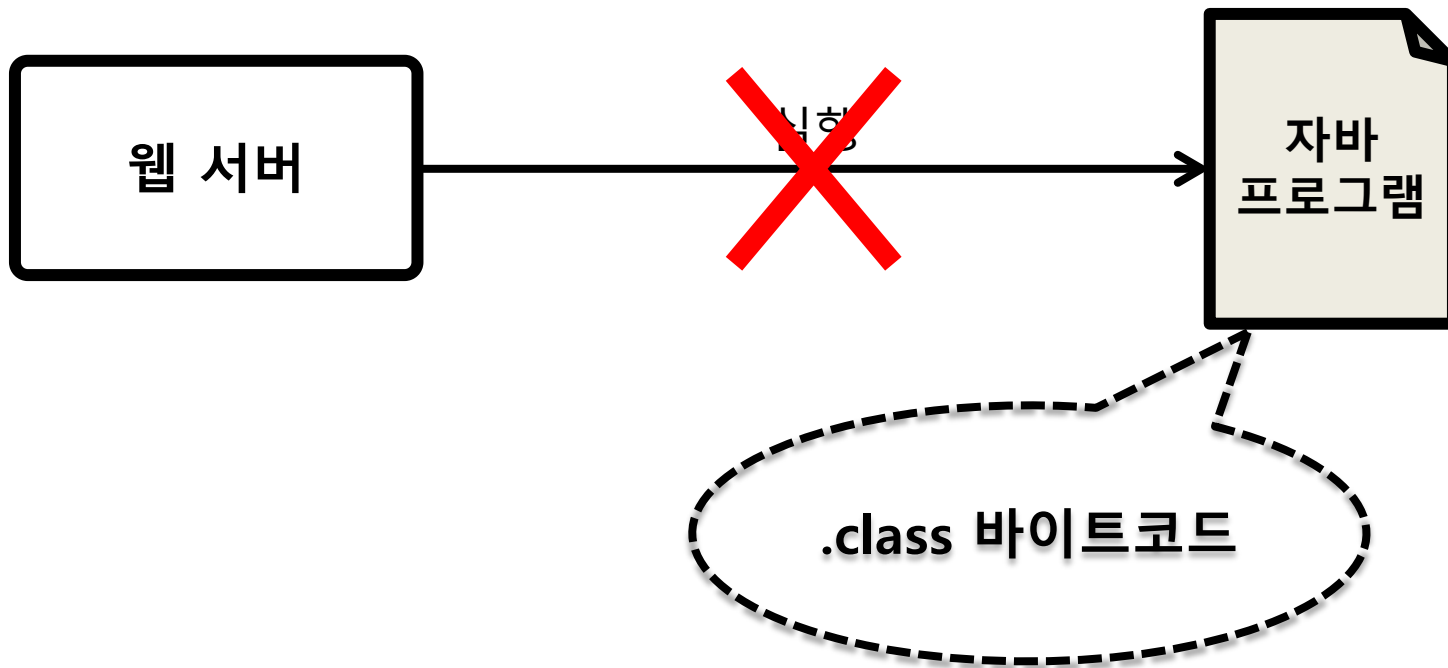


자바 웹 애플리케이션 실행

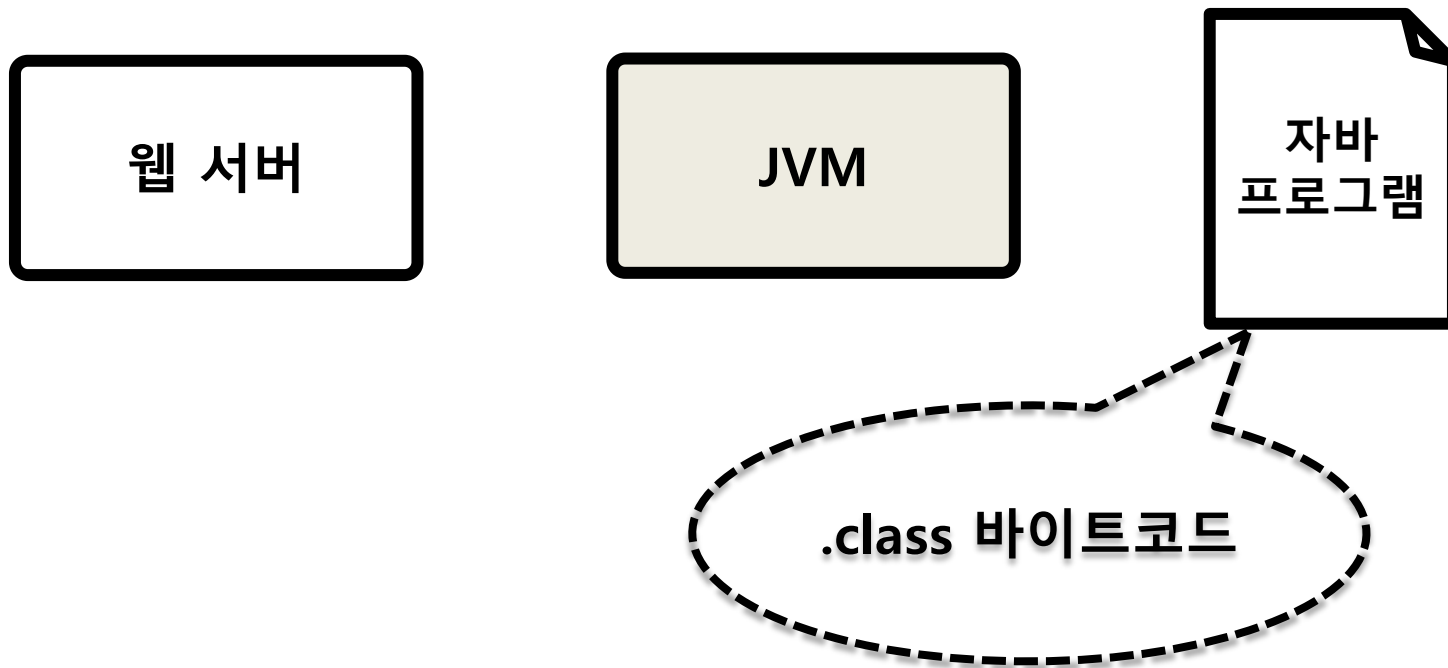


자바 웹 애플리케이션 실행

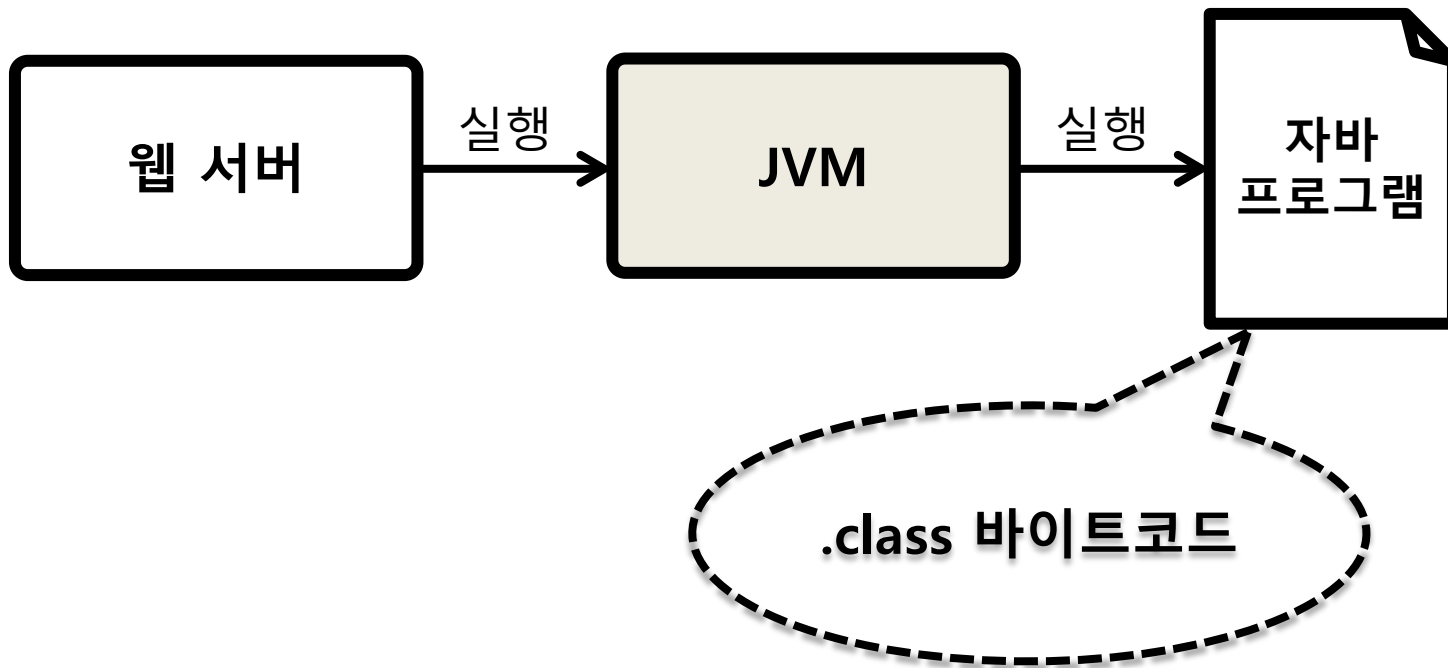
실행 불가!



자바 웹 애플리케이션 실행

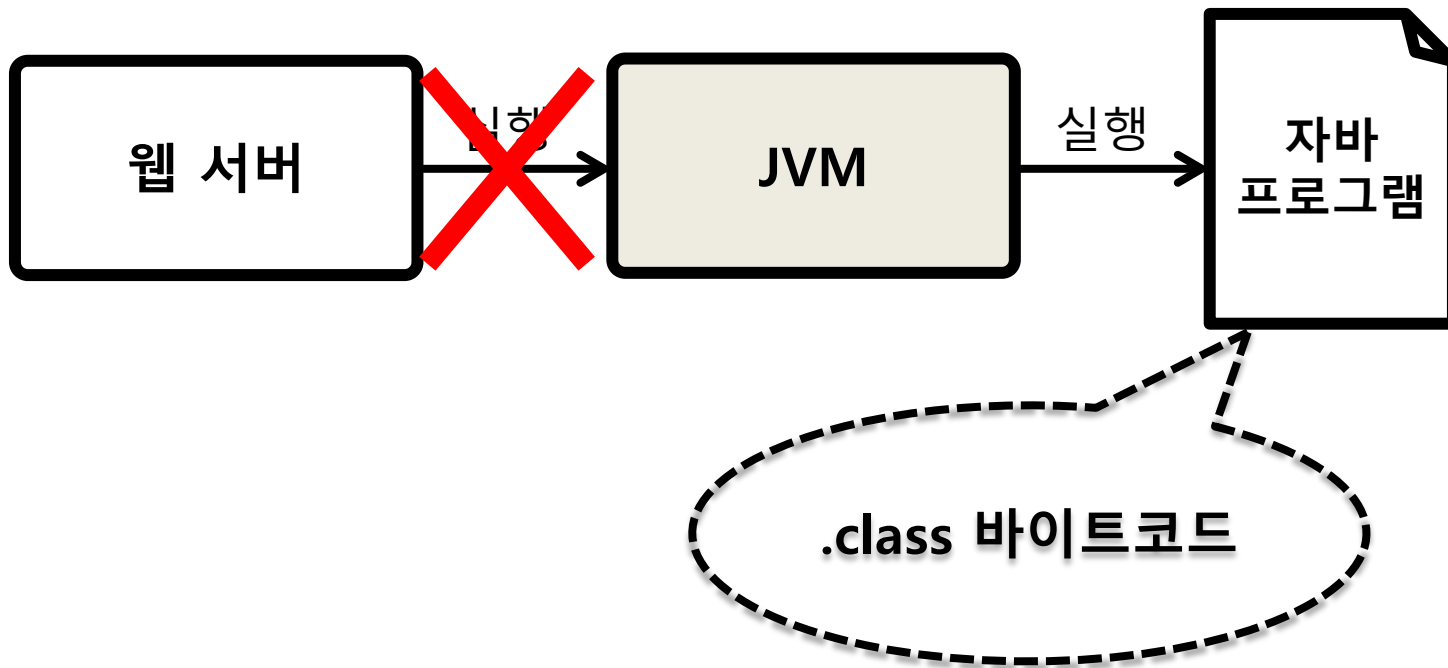


자바 웹 애플리케이션 실행

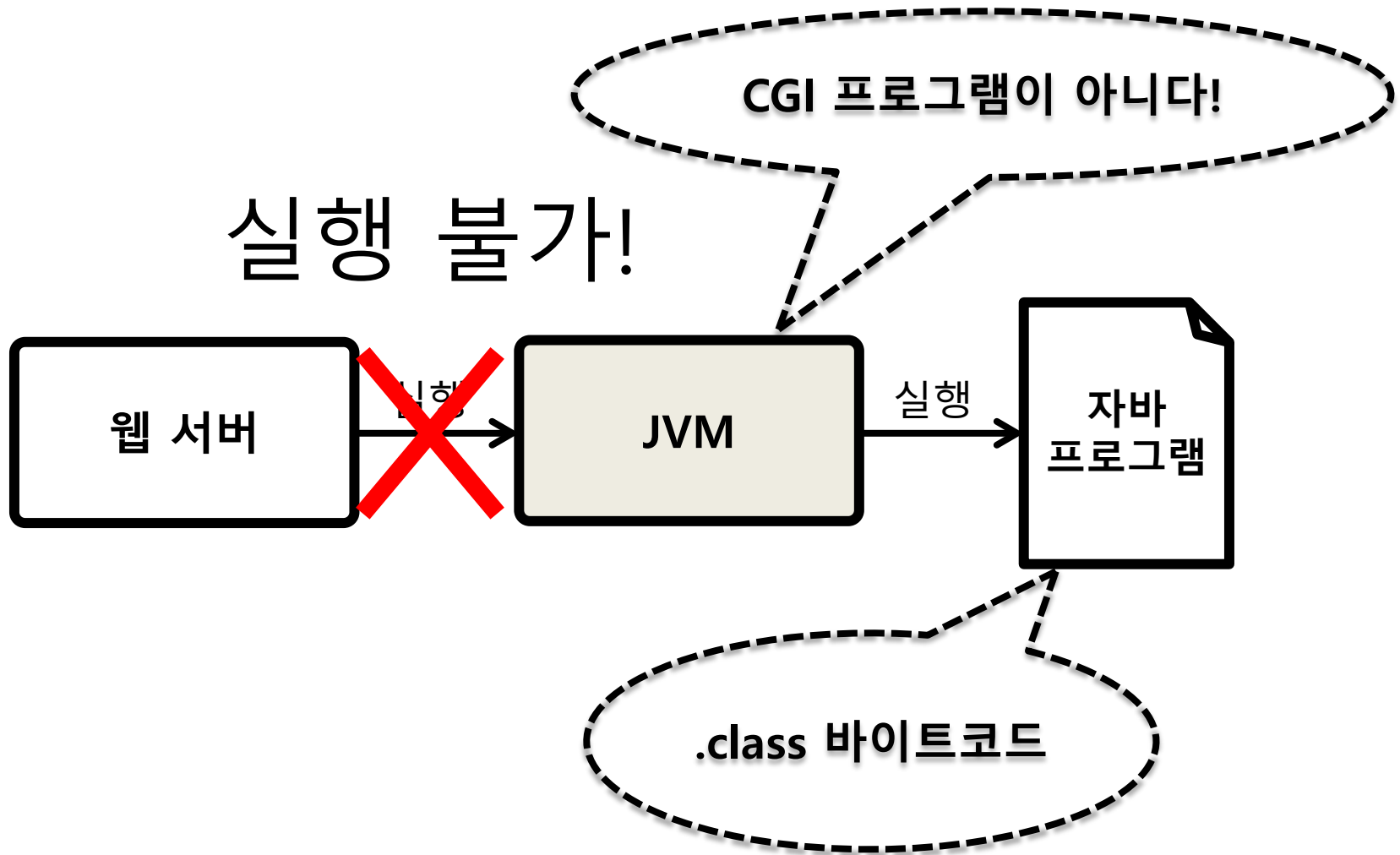


자바 웹 애플리케이션 실행

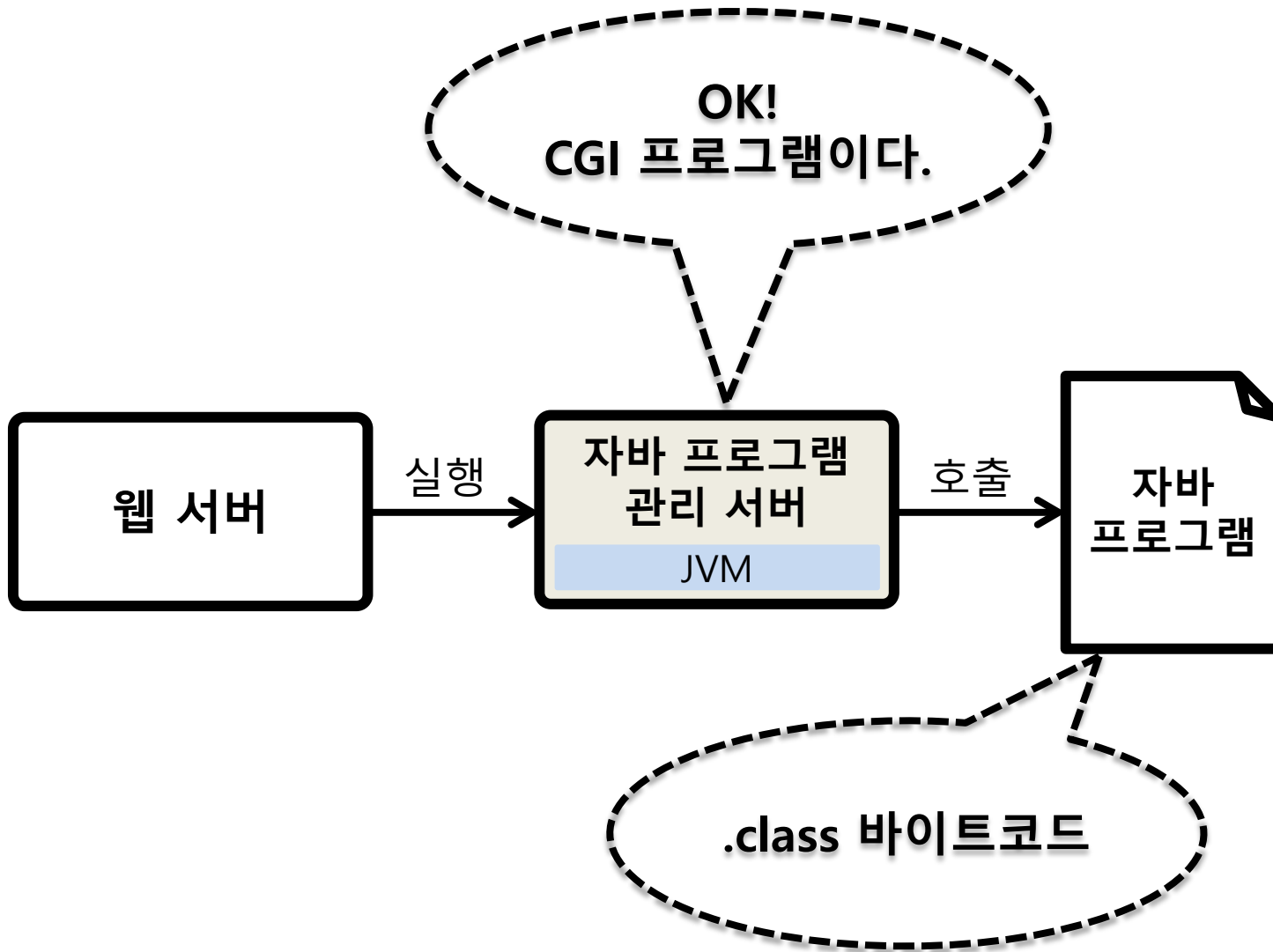
실행 불가!



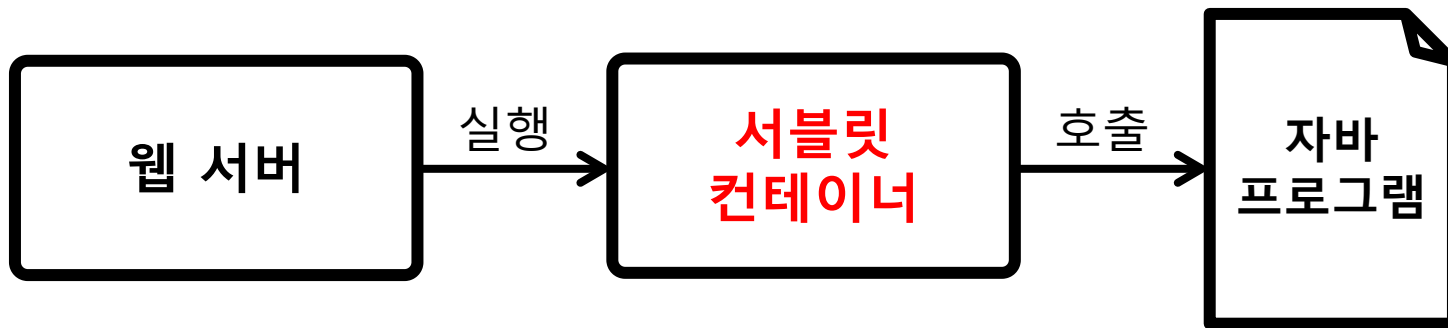
자바 웹 애플리케이션 실행



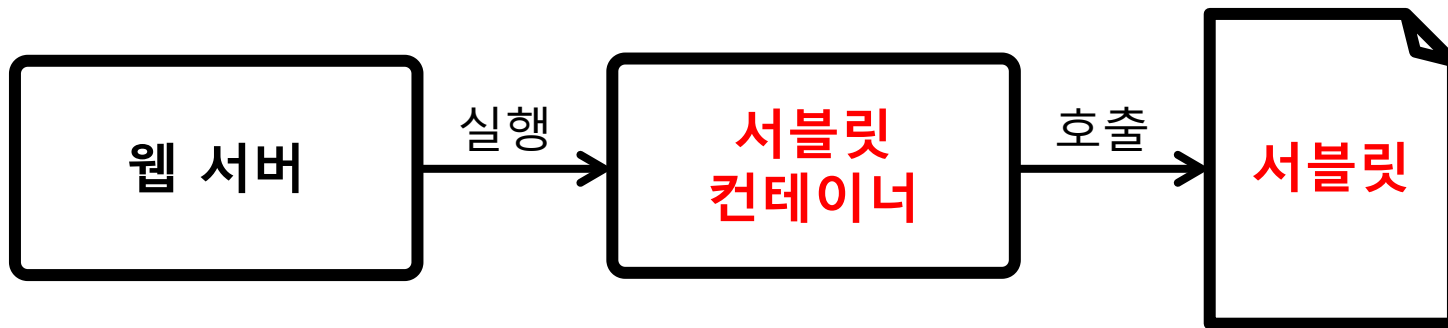
자바 웹 애플리케이션 실행



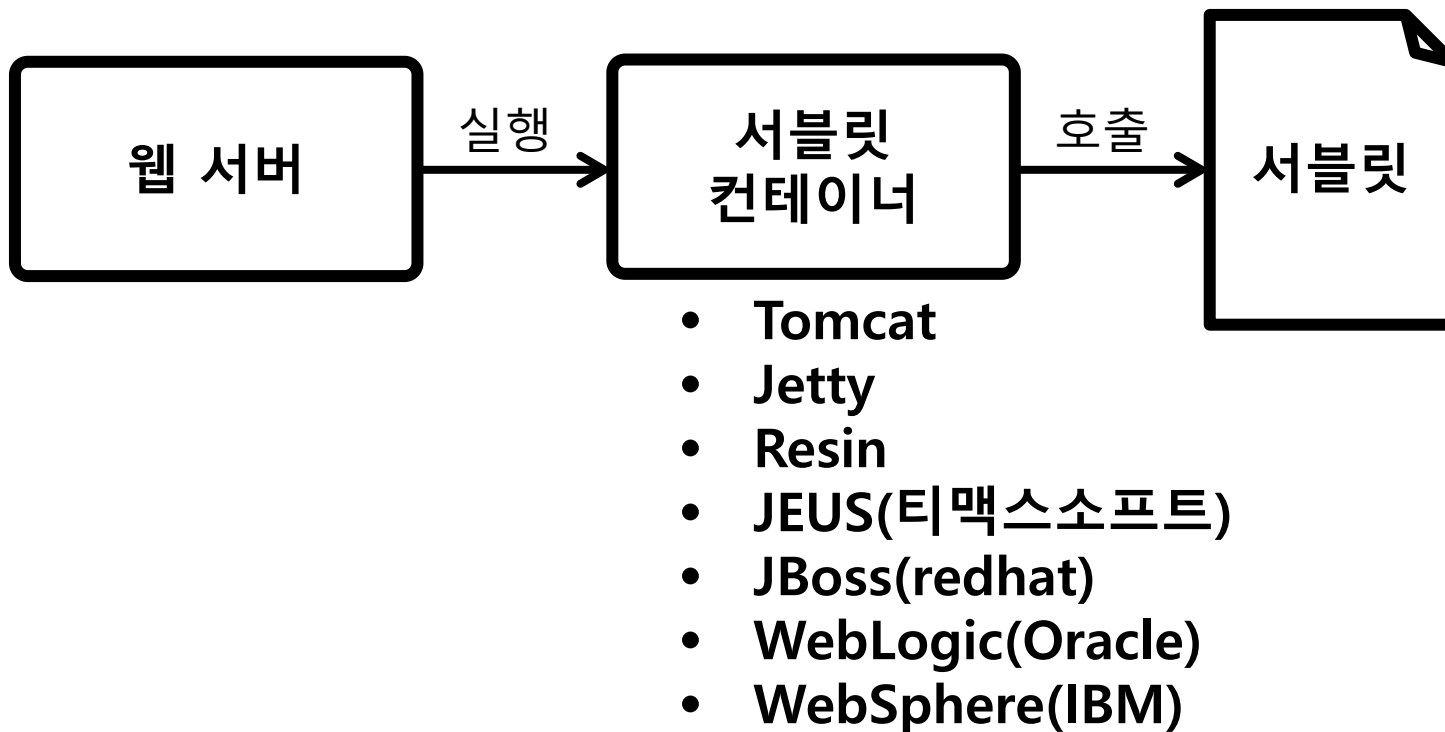
자바 웹 애플리케이션 실행



자바 웹 애플리케이션 실행



자바 웹 애플리케이션 실행

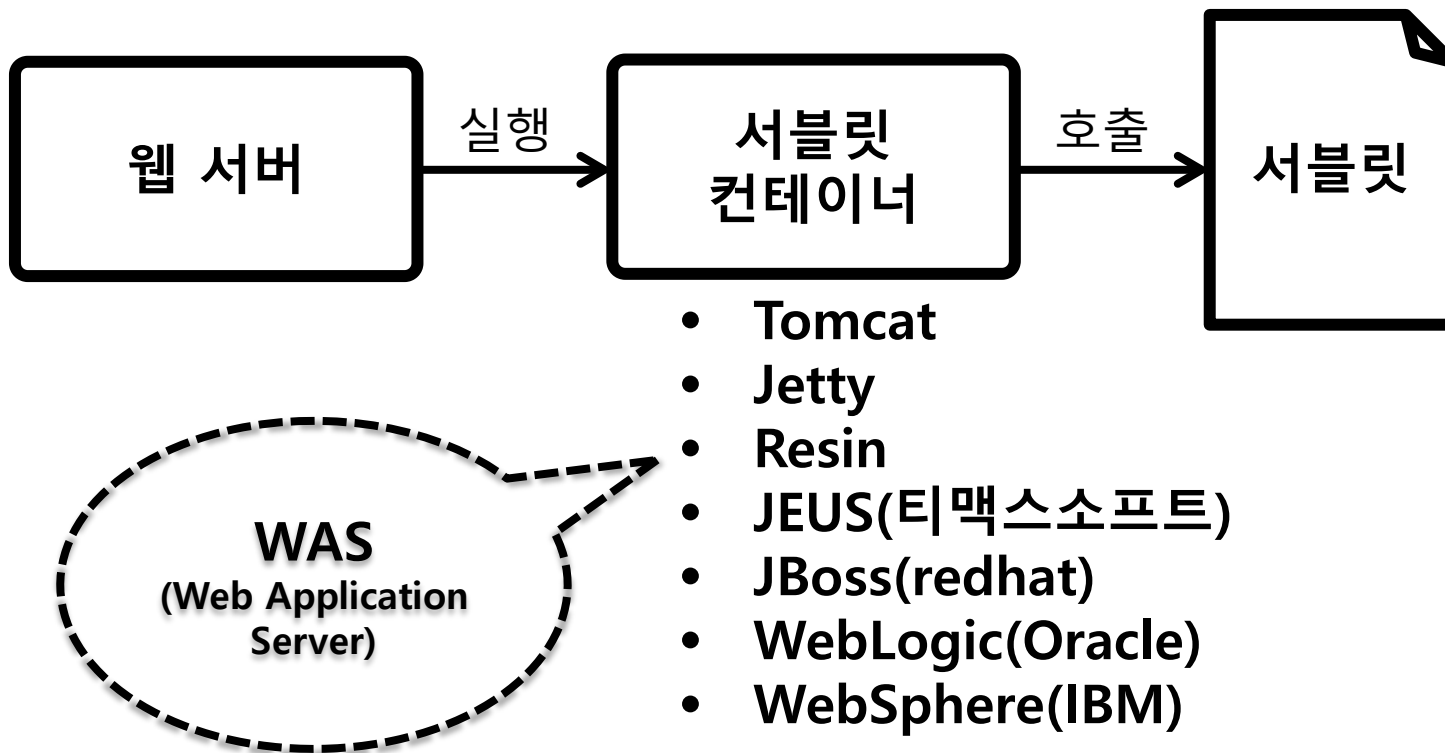


자바 웹 애플리케이션 실행



- **Tomcat**
- Jetty
- Resin
- JEUS(티맥스소프트)
- JBoss(redhat)
- WebLogic(Oracle)
- WebSphere(IBM)

자바 웹 애플리케이션 실행



3.6 GenericServlet의 사용

2.1 HTTP 프로토콜의 이해

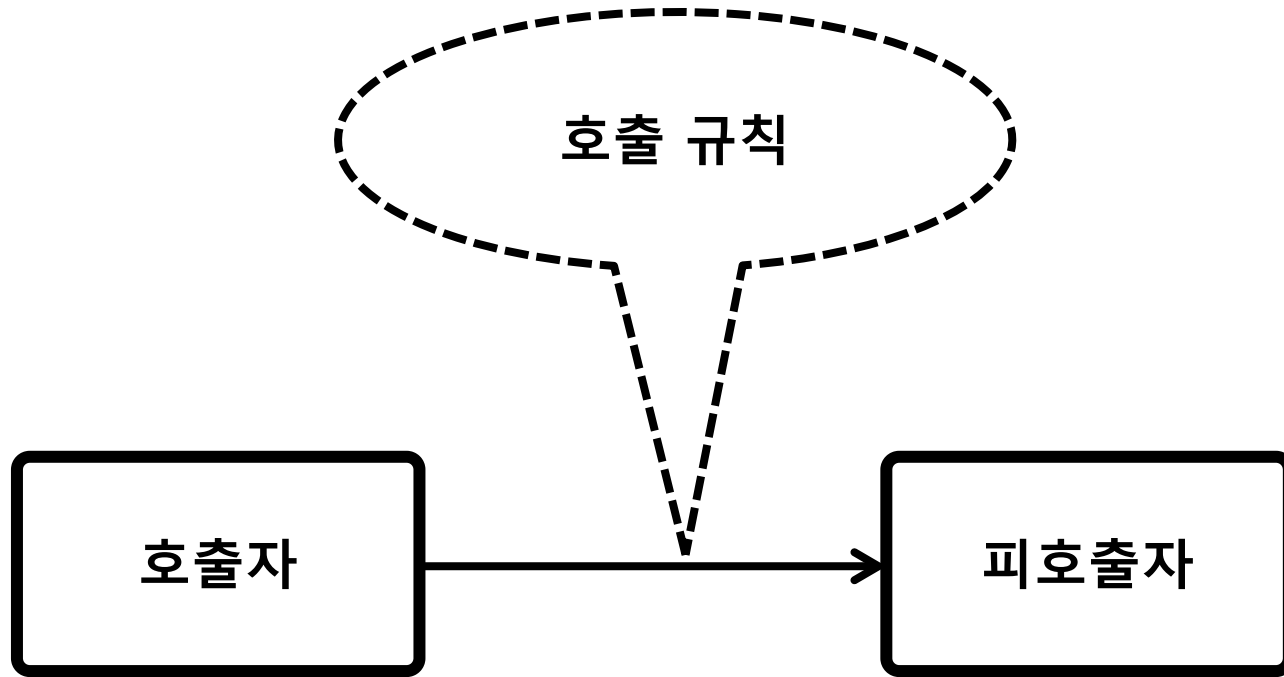
잠깐! 인터페이스란?

인터페이스 문법의 의미

호출자

피호출자

인터페이스 문법의 의미



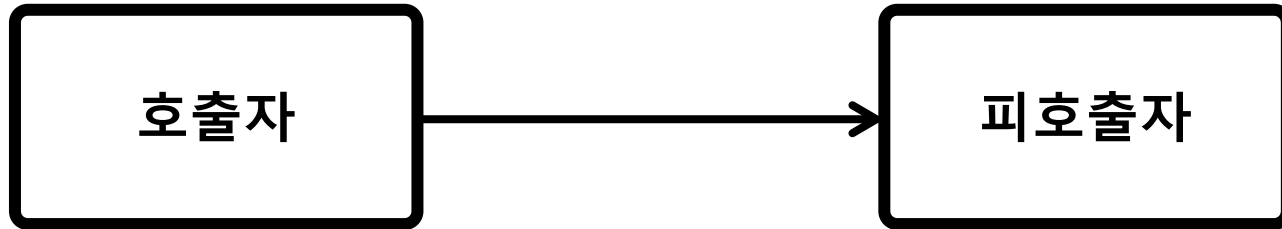
2.1 HTTP 프로토콜의 이해

Servlet 인터페이스

Servlet 인터페이스

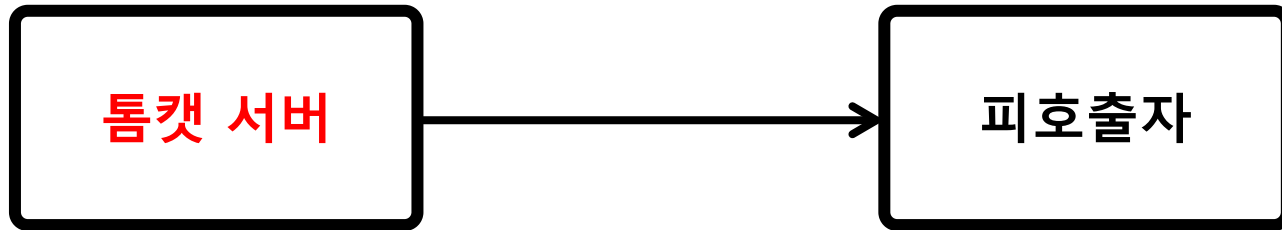
javax.servlet

Servlet



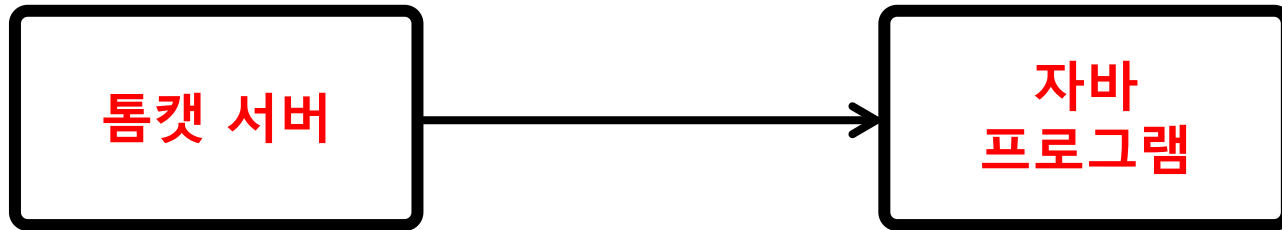
Servlet 인터페이스

javax.servlet
Servlet

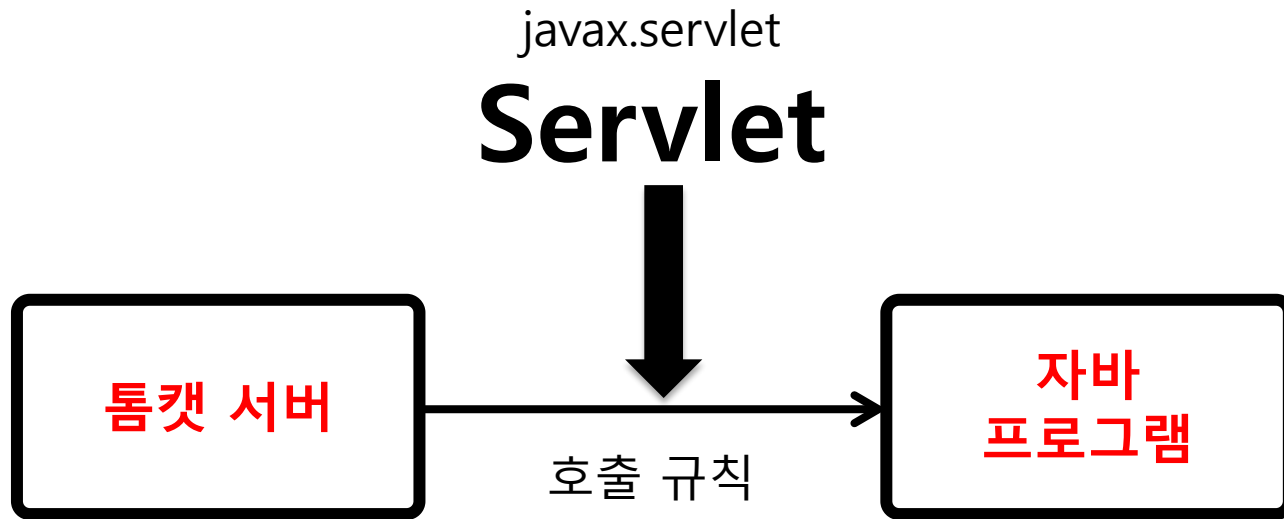


Servlet 인터페이스

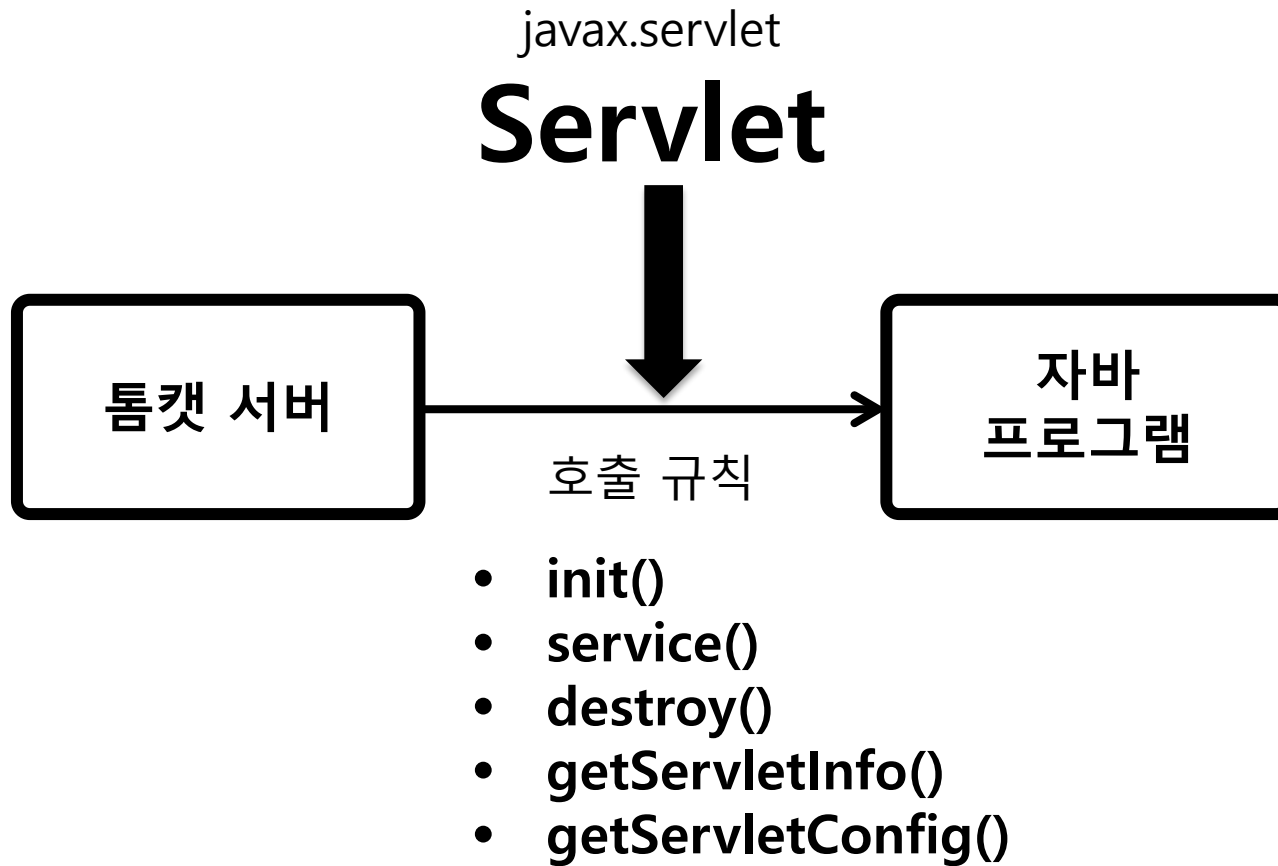
javax.servlet
Servlet



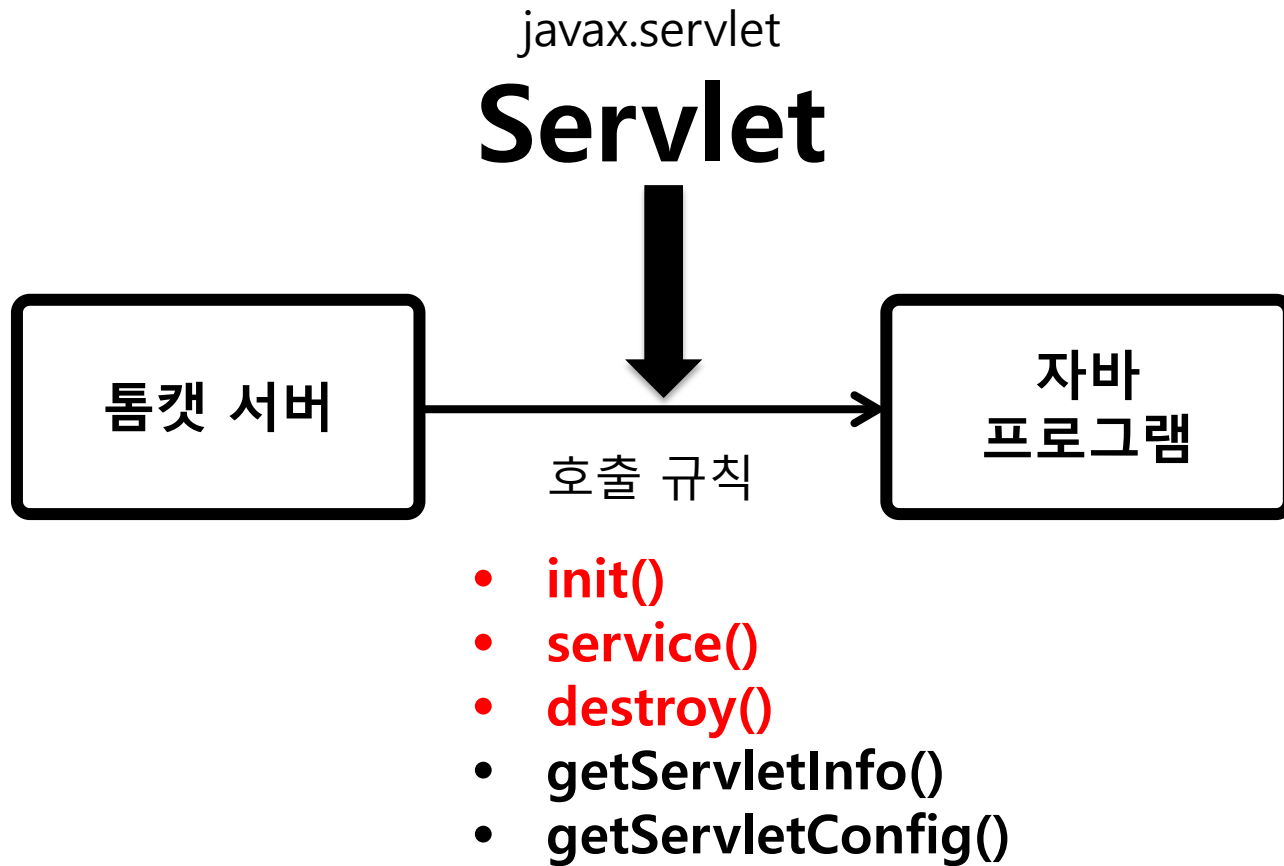
Servlet 인터페이스



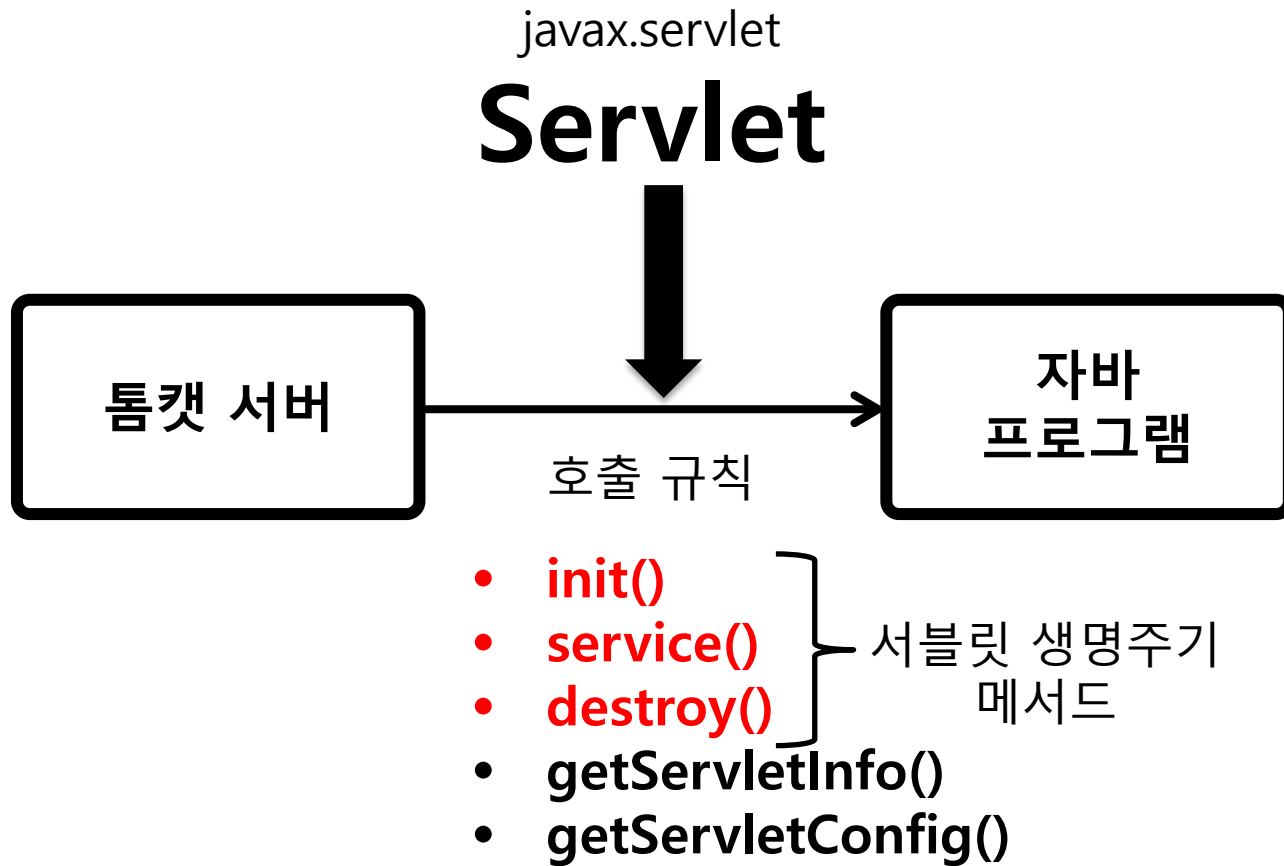
Servlet 인터페이스



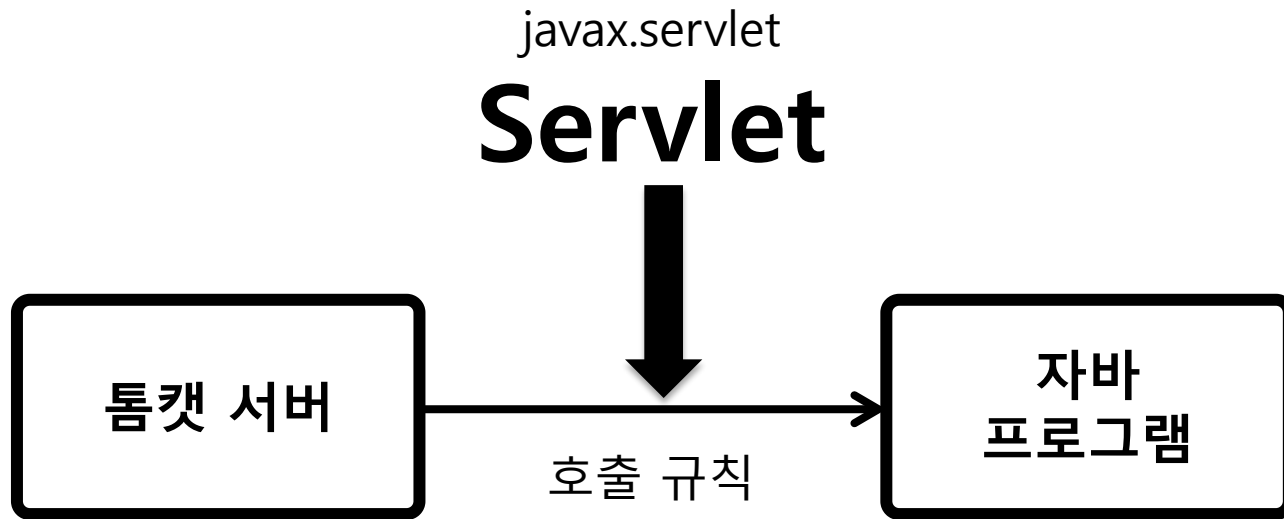
Servlet 인터페이스



Servlet 인터페이스



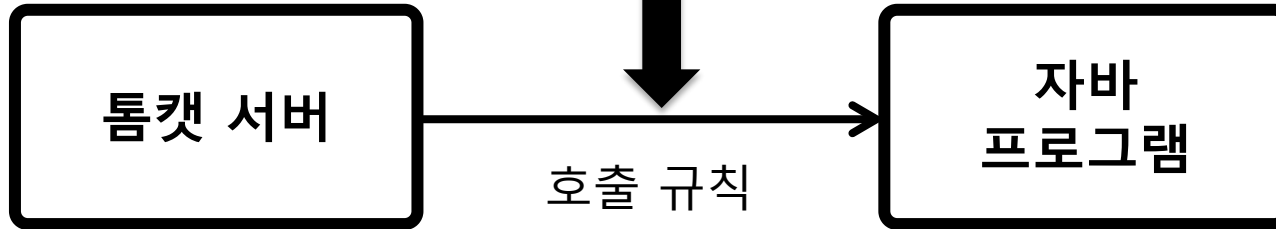
Servlet 인터페이스



- **init()**
- service()
- destroy()
- getServletInfo()
- getServletConfig()

Servlet 인터페이스

javax.servlet
Servlet



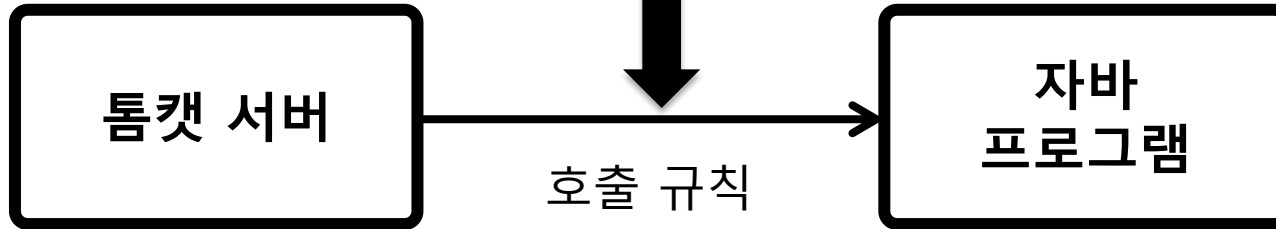
• **init()**

- service()
- destroy()
- getServletInfo()
- getServletConfig()

인스턴스 생성
후 딱 한 번
호출

Servlet 인터페이스

javax.servlet
Servlet



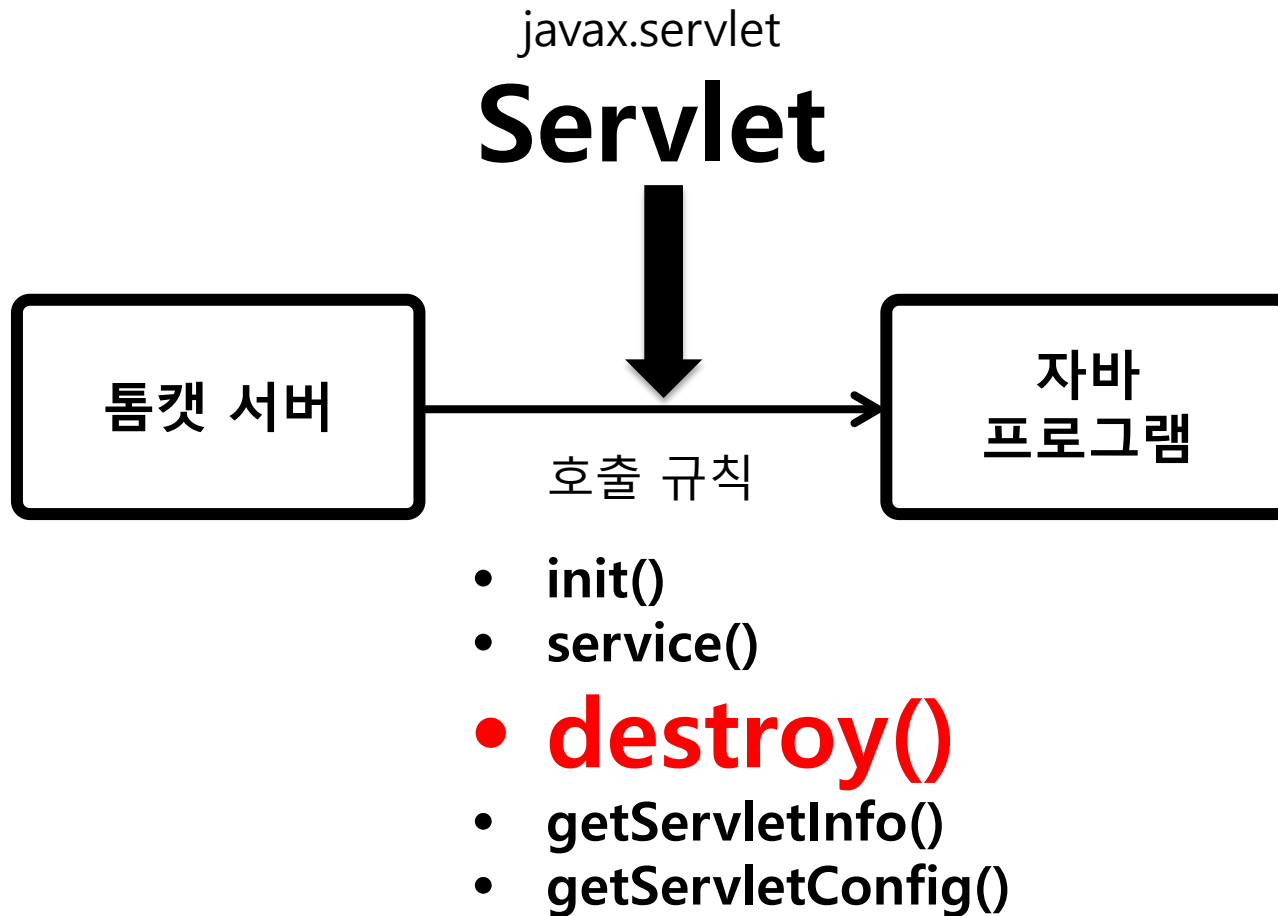
• **init()**

- service()
- destroy()
- getServletInfo()
- getServletConfig()

서비스를
준비하는 코드
작성

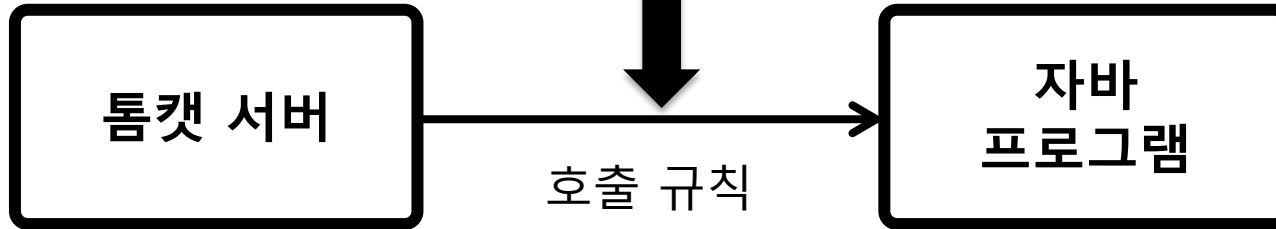
인스턴스 생성
후 딱 한 번
호출

Servlet 인터페이스



Servlet 인터페이스

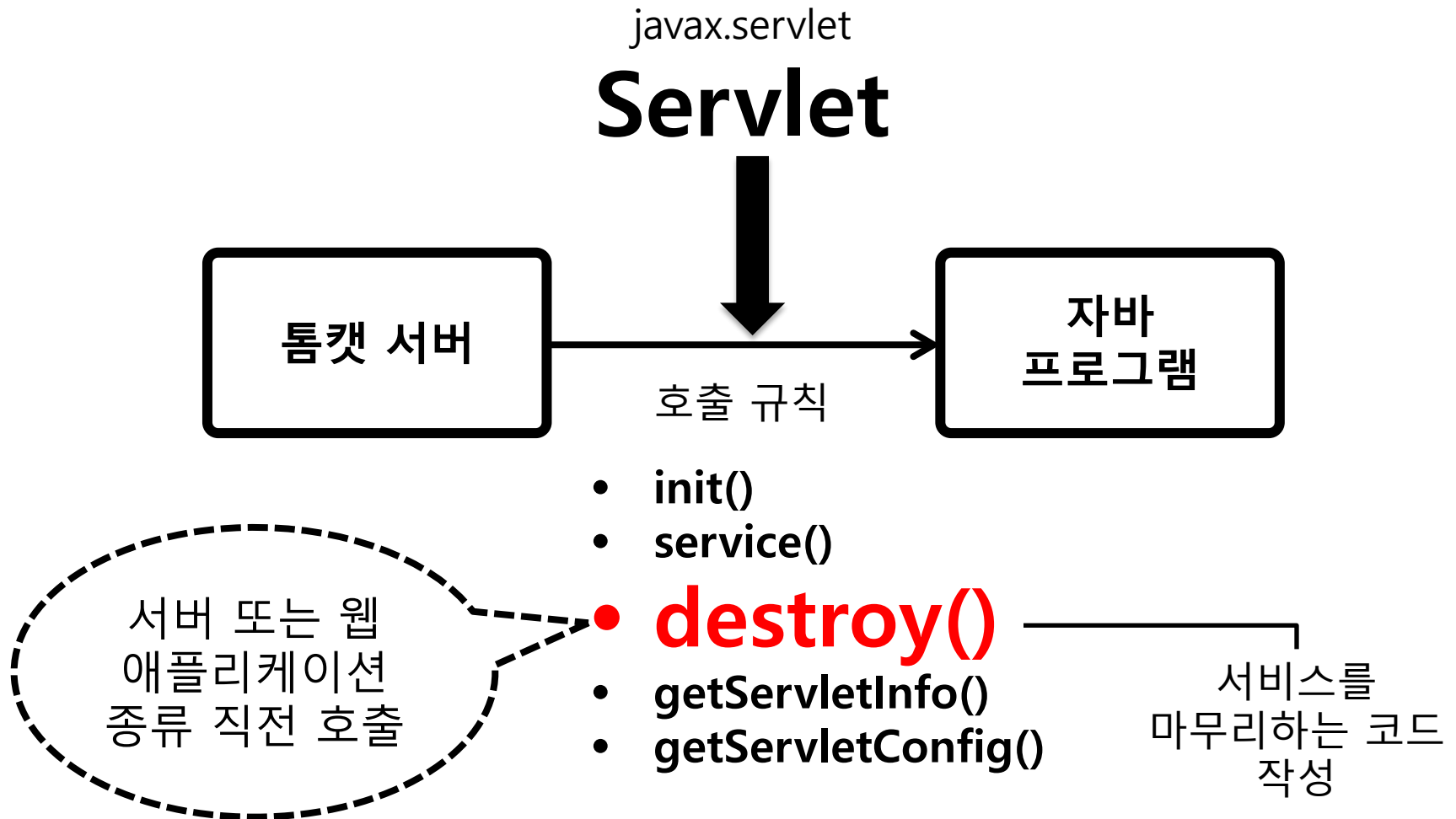
javax.servlet
Servlet



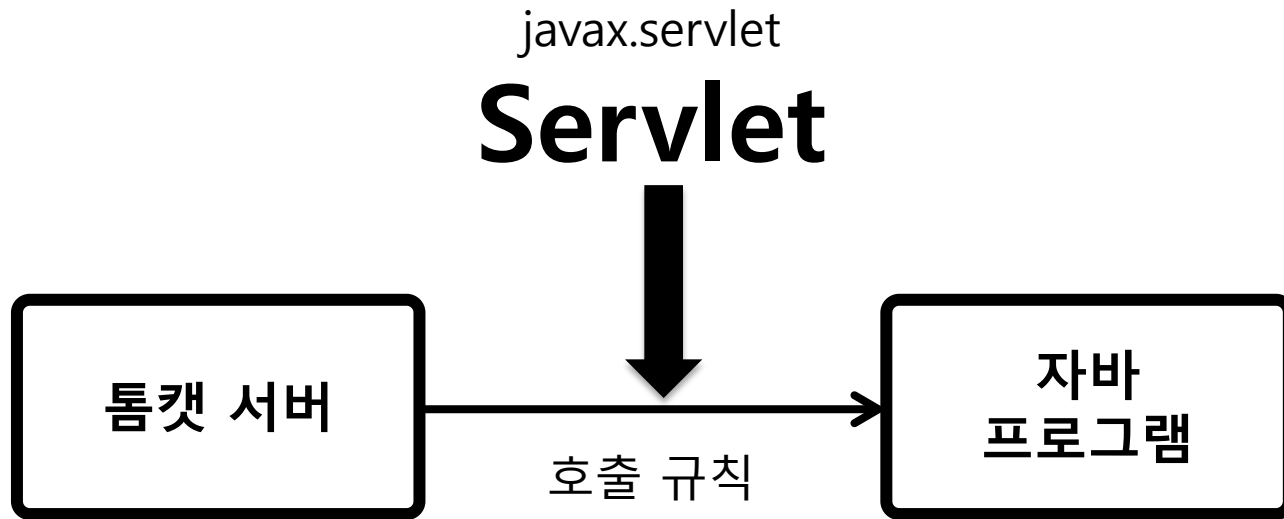
- `init()`
- `service()`
- **`destroy()`**
- `getServletInfo()`
- `getServletConfig()`

서버 또는 웹
애플리케이션
종료 직전 호출

Servlet 인터페이스



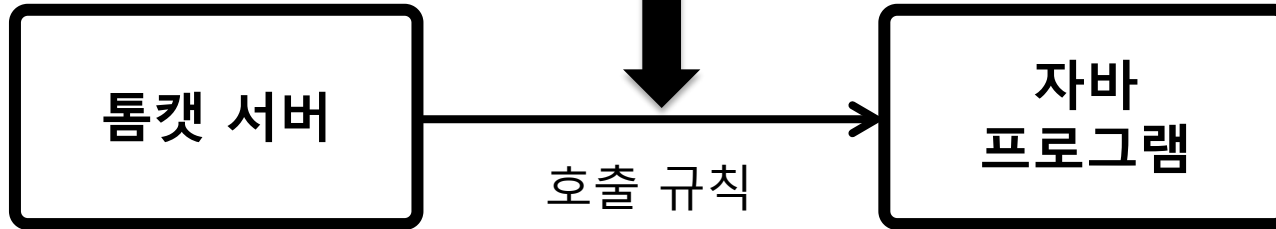
Servlet 인터페이스



- `init()`
- **`service()`**
- `destroy()`
- `getServletInfo()`
- `getServletConfig()`

Servlet 인터페이스

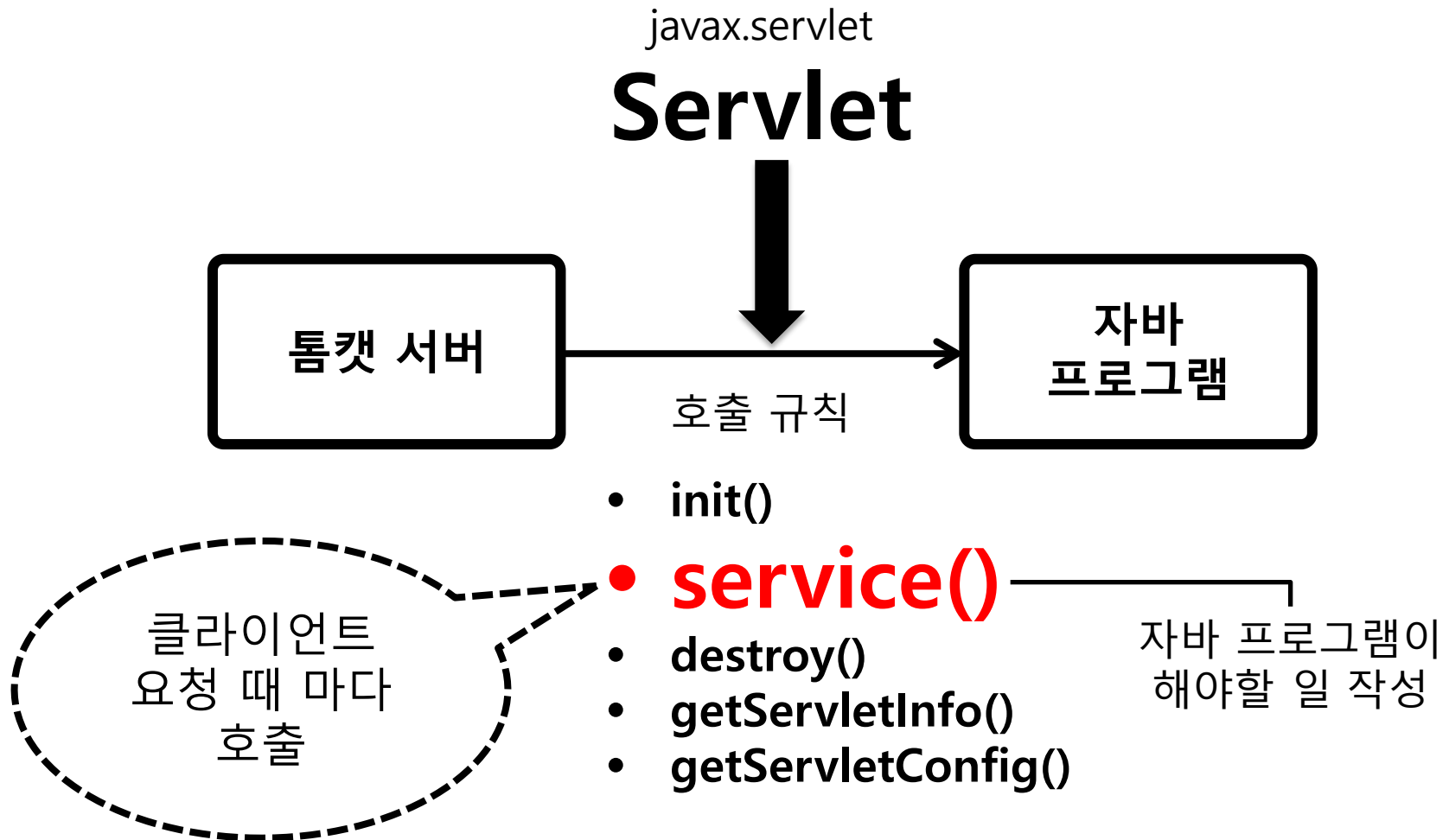
javax.servlet
Servlet



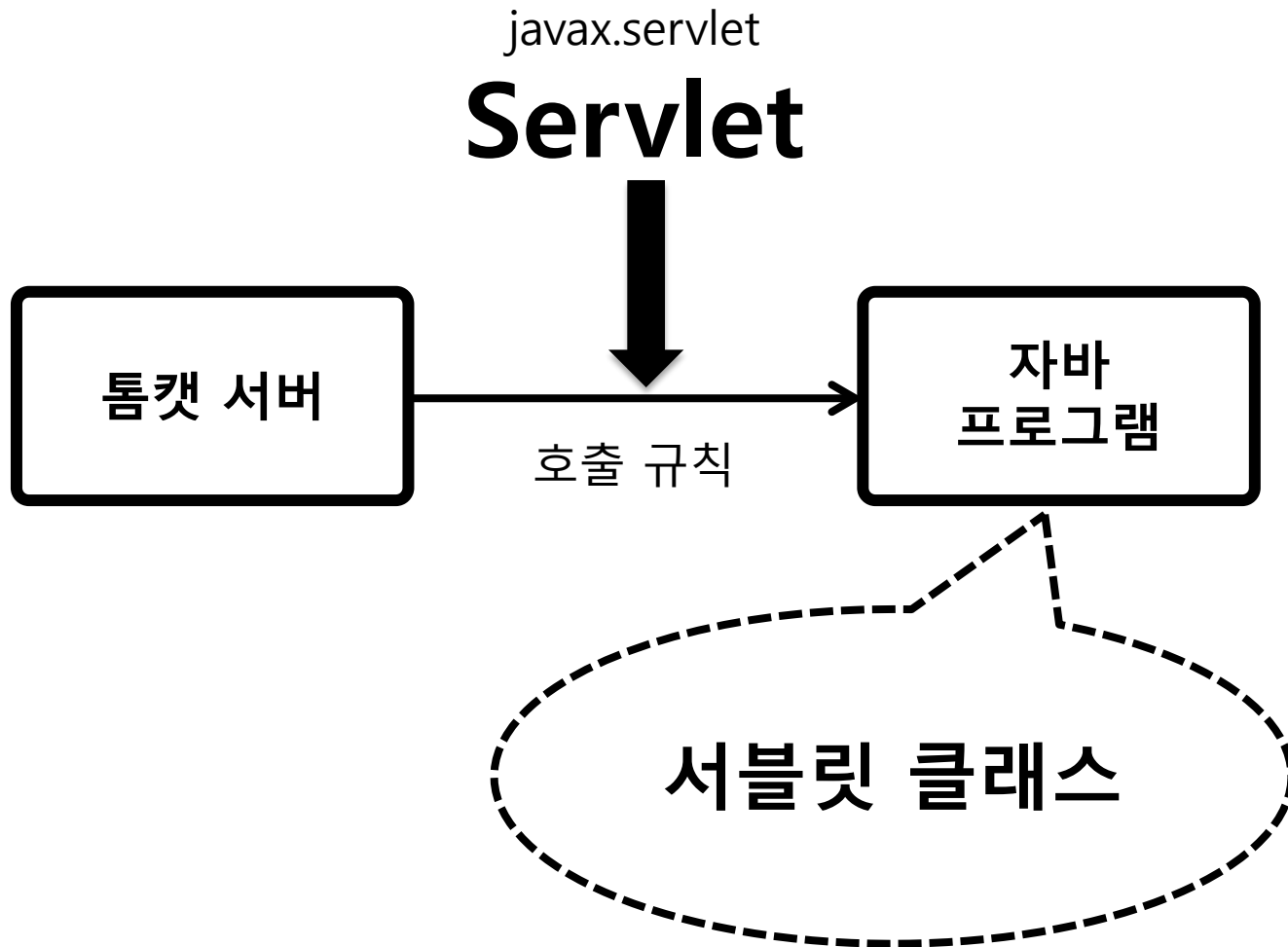
- init()
- **service()**
- destroy()
- getServletInfo()
- getServletConfig()

클라이언트
요청 때 마다
호출

Servlet 인터페이스

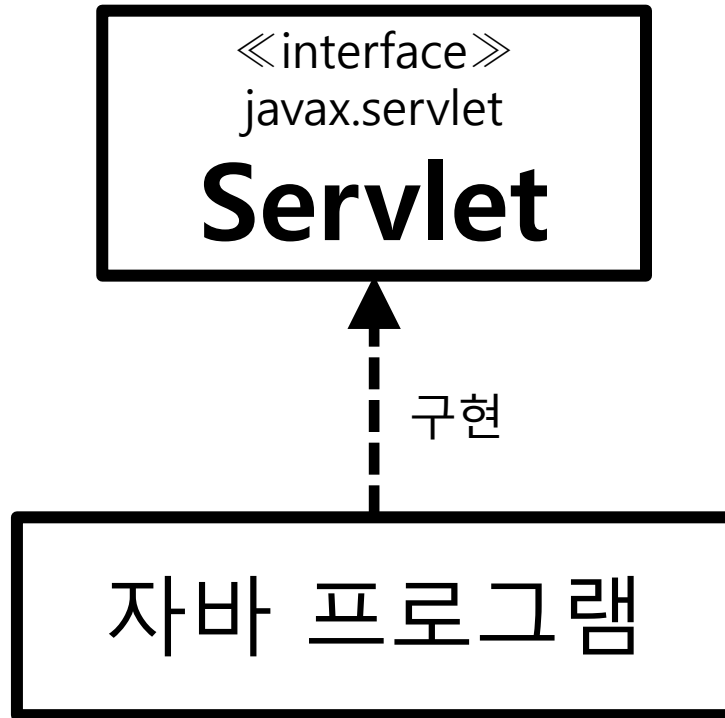


Servlet 인터페이스

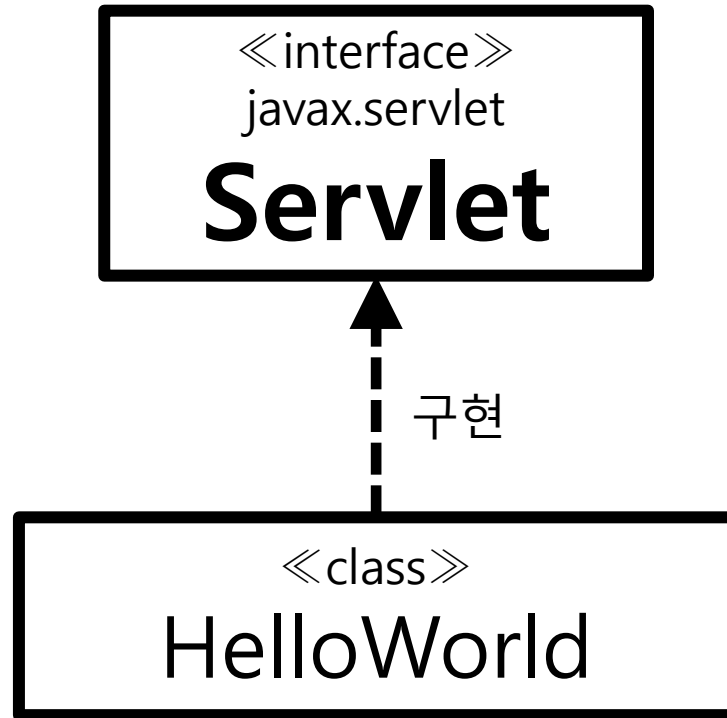


서블릿 클래스

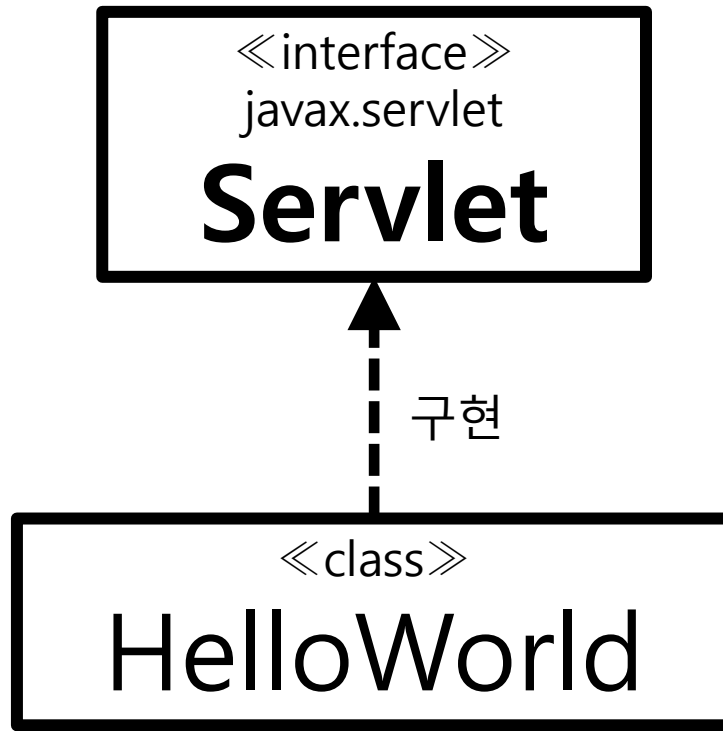
서블릿 클래스



서블릿 클래스

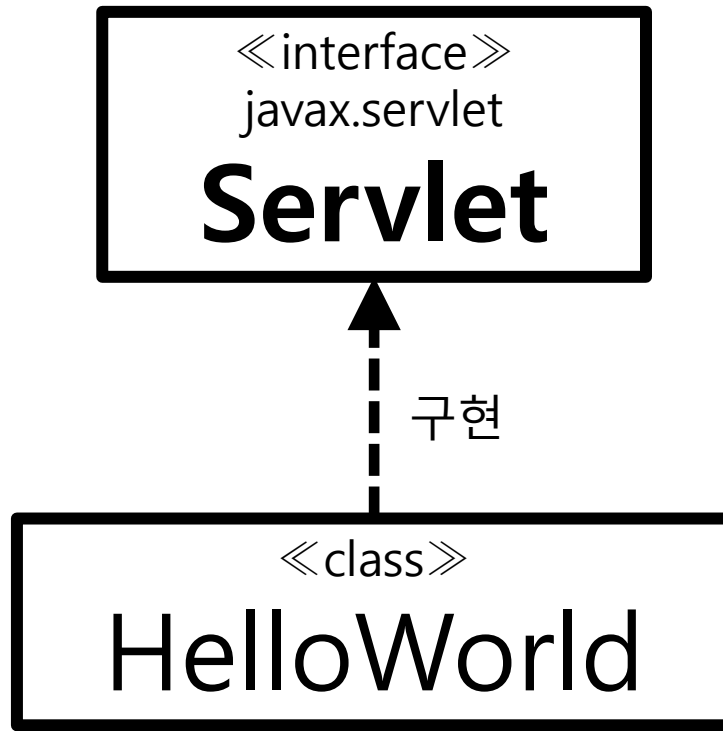


서블릿 클래스



- `init() { ... }`
- `service() { ... }`
- `destroy() { ... }`
- `getServletInfo() { ... }`
- `getServletConfig() { ... }`

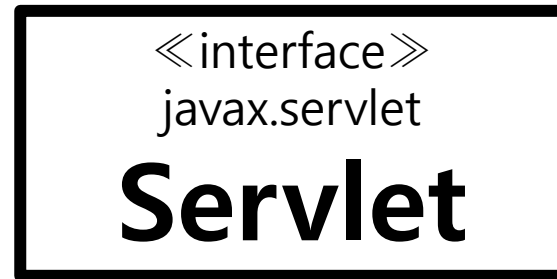
서블릿 클래스



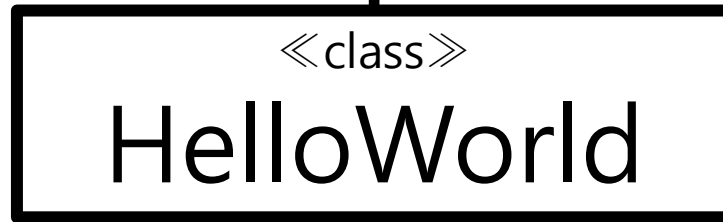
- `init() { ... }`
- **`service() { ... }`**
- `destroy() { ... }`
- `getServletInfo() { ... }`
- `getServletConfig() { ... }`

반드시
구현!

서블릿 클래스



구현

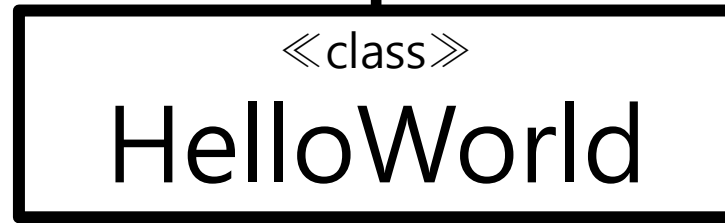
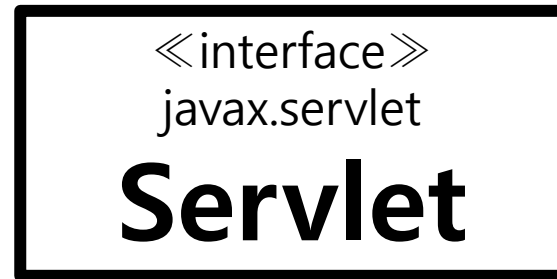


필요할 때만 구현

- `init() { ... }`
- **`service() { ... }`**
- `destroy() { ... }`
- `getServletInfo() { ... }`
- `getServletConfig() { ... }`

반드시
구현!

서블릿 클래스



구현

번거롭다!
불편하다!

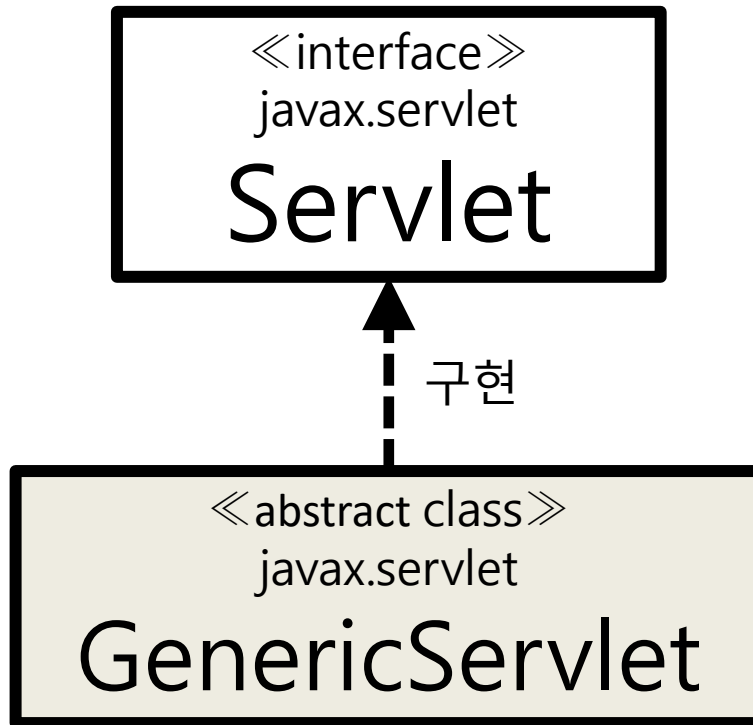
필요할 때만 구현

- `init() { ... }`
- **`service() { ... }`**
- `destroy() { ... }`
- `getServletInfo() { ... }`
- `getServletConfig() { ... }`

반드시
구현!

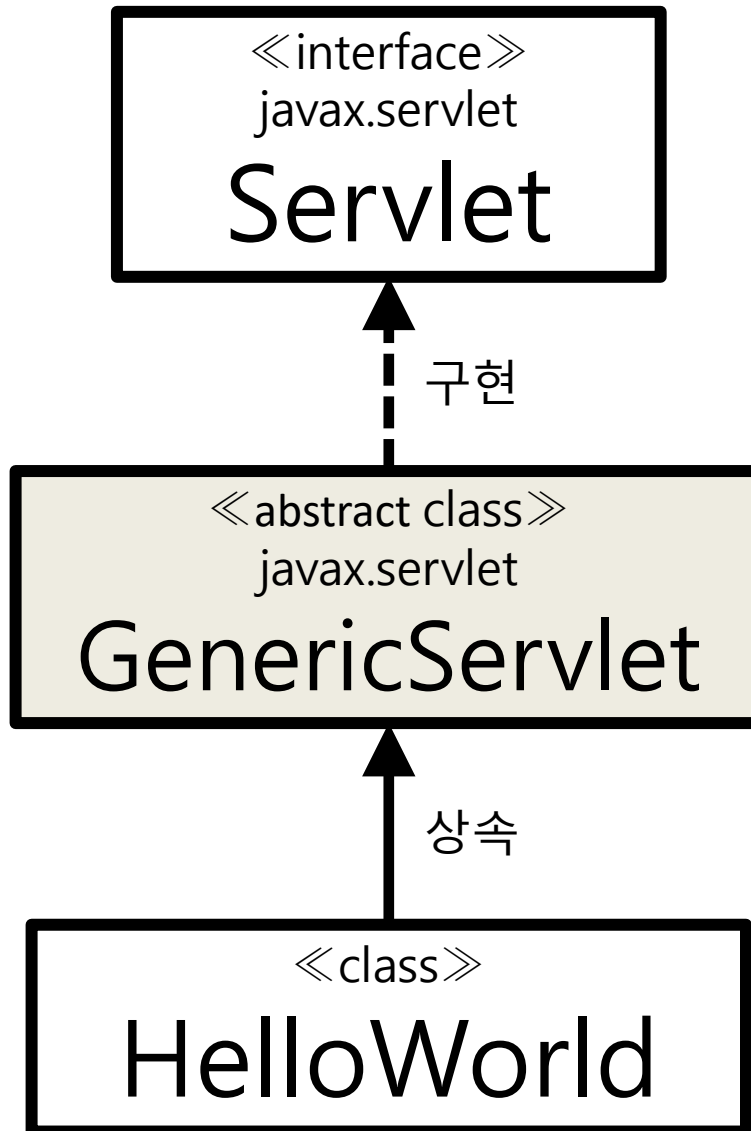
GenericServlet 추상 클래스!

서블릿 클래스



- `init() { ... }`
- `destroy() { ... }`
- `getServletInfo() { ... }`
- `getServletConfig() { ... }`

서블릿 클래스



- `init() { ... }`
- `destroy() { ... }`
- `getServletInfo() { ... }`
- `getServletConfig() { ... }`

- **`service() { ... }`**