

# 삼성 청년 SW 아카데미

Web-BackEnd

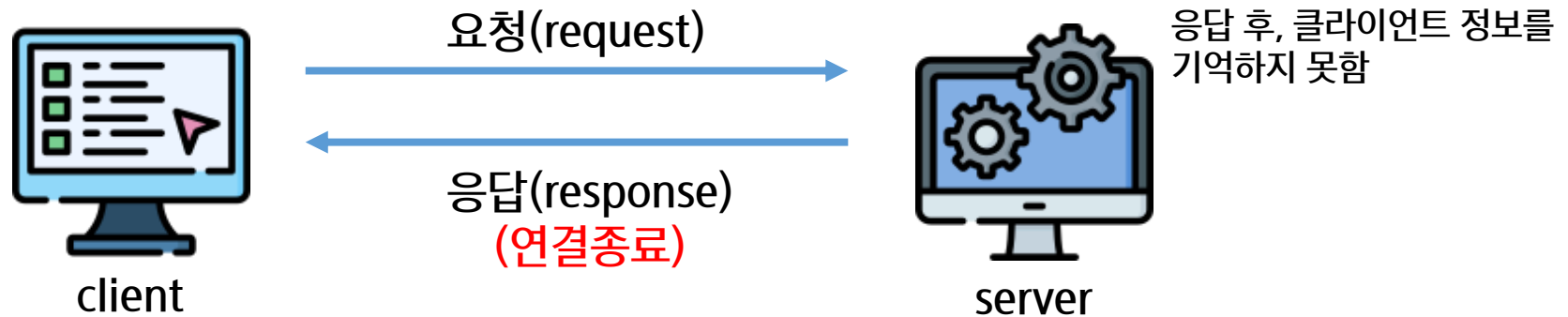
# Cookie & Session

- Cookie
- JSP 기본객체 영역
- Session

# Cookie

### ✓ Http Protocol 특징

#### ✓ 비 연결 (connectionless) 지향형 통신 프로토콜



#### ✓ 응답 후 연결을 종료 (stateless)

- 지속적인 연결 유지로 인한 자원낭비 방지를 위해 연결 해제
- 연결이 해제되면 서버는 클라이언트 정보를 알 수 없음

예) 장바구니에 상품을 담아 두었는데, 페이지가 바뀌면 정보가 없어짐, 로그인 정보가 없음 등

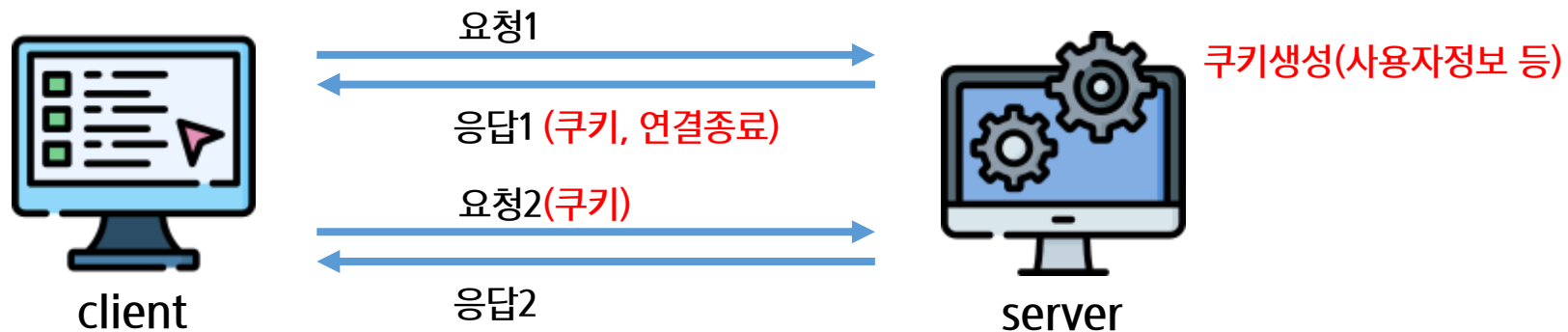
- 이러한 문제를 해결하기 위해서 Cookie와 HttpSession을 활용한다.

## ✓ Cookie : javax.servlet.http.Cookie

- 서버가 생성하는 클라이언트 정보를 가지고 있는 파일
- 클라이언트 컴퓨터에 저장되며, 필요에 따라 요청 시 서버로 같이 전송된다.
- key-value 형태로 구성되며 문자열 데이터
- 브라우저(클라이언트) 별로 별도의 쿠키가 생성된다. ( 브라우저가 다르면 다른 사용자로 처리)
- 세션관리(사용자 아이디, 접속시간, 장바구니 등)을 위해 사용된다
- 사용자마다 다른 페이지를 보여줄 수 있다.
- 사용자의 행동과 패턴을 분석하고 기록하는데 사용된다. ( 사용자가 클릭한 상품 관련 광고 배너 )

## ✓ Cookie 동작순서

- Client가 요청 생성
- WAS는 Cookie를 생성하고 Http Header에 Cookie를 넣어 응답
- Client(Browser)는 Cookie를 저장, 해당 서버에 요청할 때 요청과 함께 Cookie를 전송
- Cookie는 브라우저가 종료되더라도 계속 저장되기 때문에(만료 기간 전까지) 동일 사이트 재 방문하여 요청 시 필요에 따라 Cookie가 재 전송된다.



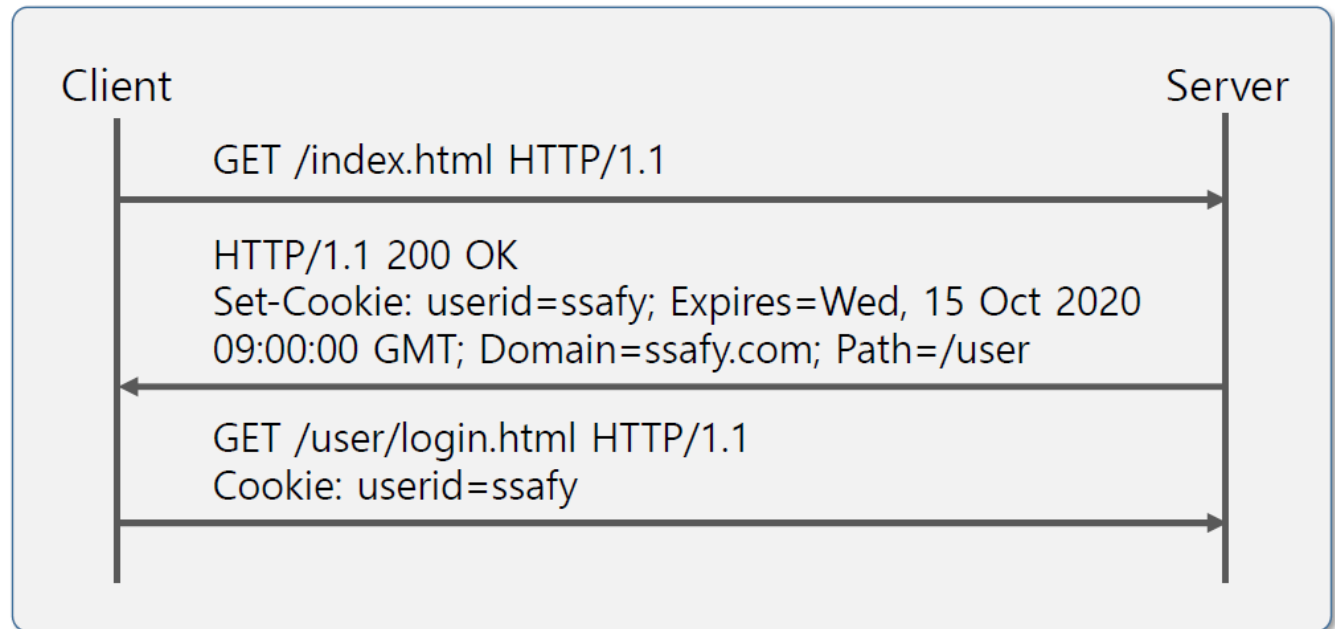
## ✓ Cookie 특징

- 이름(key), 값(value), 만료일(Expire date, 저장기간), 경로정보 로 구성된다.
- 클라이언트에 최대 300개의 쿠키를 저장할 수 있다.
- 하나의 도메인당 20개의 쿠키를 저장할 수 있다.
- 쿠키 하나는 4KB(=4096byte) 까지 저장가능

userid = ssafy, 만료일 : 2020년 8월 15일

유효 도메인 : ssafy.com, 경로 : /user

인 쿠키 동작 및 설정



## ✓ Cookie 생성 및 추가

쿠키 생성 : `javax.servlet.http.Cookie(java.lang.String name, java.lang.String value)`

```
Cookie cookie = new Cookie("userid", "ssafy");
```

쿠키 응답에 추가 : `void javax.servlet.http.HttpServletResponse.addCookie(Cookie cookie)`

```
response.addCookie(cookie);
```

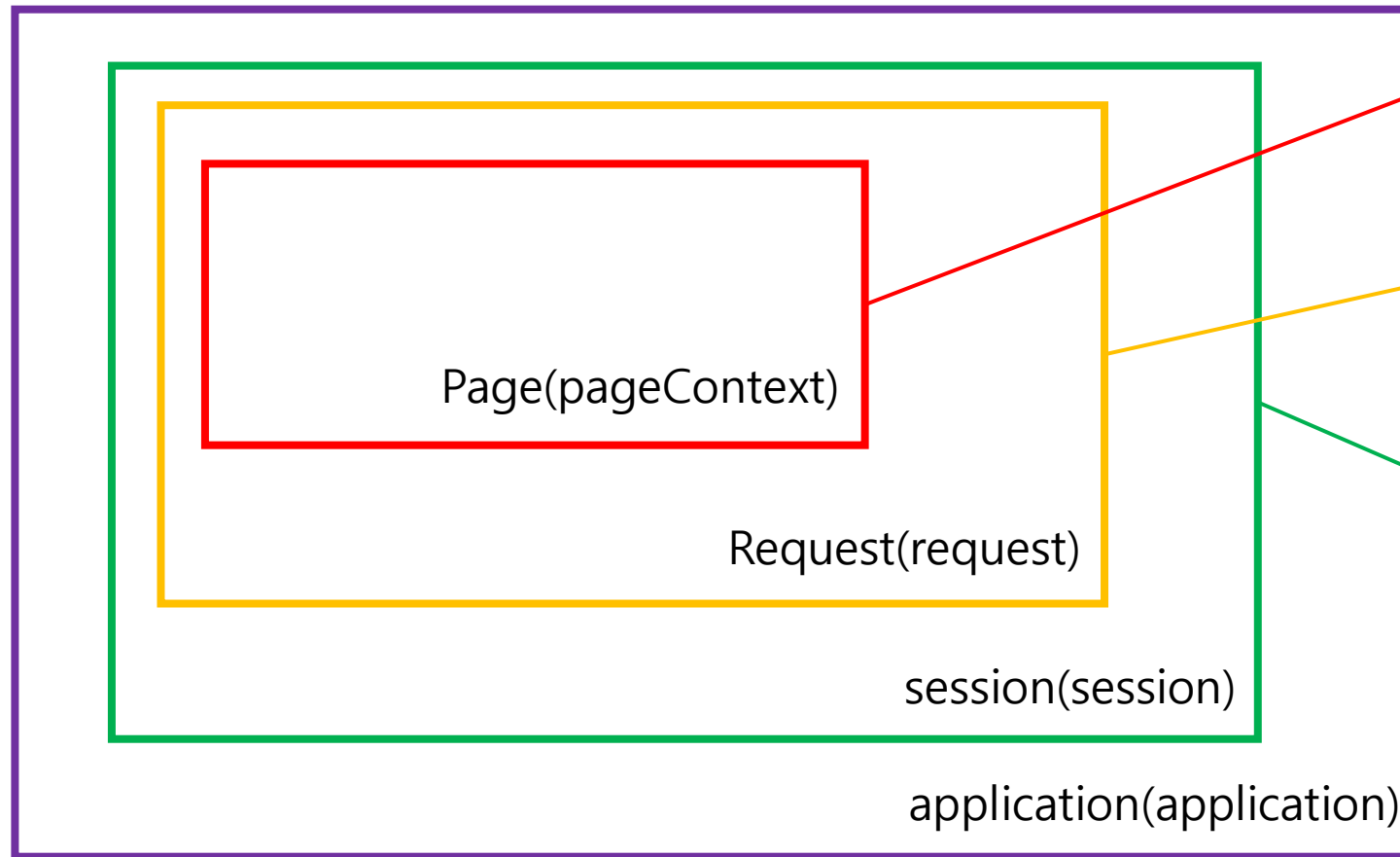


## ✓ Cookie 주요 메서드

메서드	설명
void setComment(String comment)	쿠키에 대한 설명 설정
void setDomain(String domain)	쿠키의 유효한 도메인 설정
void setMaxAge(int expiry)	쿠키 유효기간 설정
void setPath(String path)	쿠키 유효 디렉토리 설정
void setValue(String value)	쿠키 값 설정
String getComment()	쿠키 설명 반환
String getDomain()	쿠키 유효 도메인 반환
int getMaxAge()	쿠키 유효기간 반환
String getPath()	쿠키 유효 디렉토리 반환
String getValue()	쿠키 값 반환

# JSP 기본객체 영역

### ✓ JSP 기본 객체 영역 (Scope)



기본객체 영역과 기본객체

하나의 페이지 정보를 담고 있는 영역, 페이지가 바뀌면 새로운 객체가 생성됨

하나의 요청을 처리할 때 사용되는 영역, 응답이 완료되면 사라진다.

하나의 웹 브라우저와 관련된 영역, 로그인 정보 등을 저장한다.

웹 어플리케이션 영역, 어플리케이션이 시작되면 종료될 때 까지 유지 된다.

### ✓ JSP 기본 객체 영역 (Scope) 메서드

- servlet과 페이지간 정보를 공유하기 위해서 메서드를 지원한다.

아래는 각 영역에서 사용할 수 있는 공통 메서드

메서드	반환형	설명
setAttribute(String name, Object value)	void	key-value 형태로 각 영역에 데이터를 저장. name이 value를 얻어 오기 위한 key 가 된다.
getAttribute(String name)	Object	현재 객체에서 인자로 받은 이름으로 설정된 값을 반환
getAttributeNames()	Enumeration	현재 객체에서 설정된 값의 모든 속성의 이름을 반환
removeAttribute(String name)	void	현재 객체에서 인자로 받은 이름으로 설정된 값을 삭제

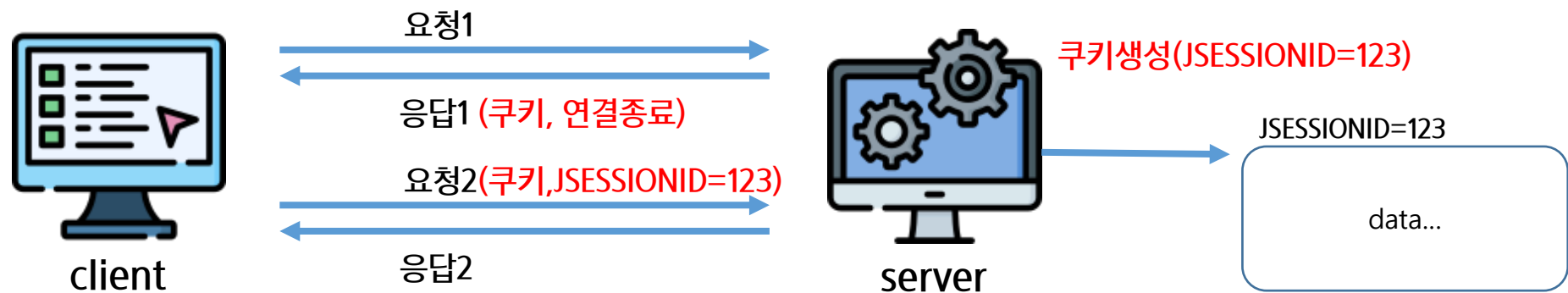
# HttpSession

### ✓ session : javax.servlet.http.HttpSession

- 사용자가 웹 서버에 접속해 있는 상태를 하나의 단위보고 세션이라고 한다.
- 각 세션은 sessionid를 이용해 구분한다.
- WAS의 메모리에 객체 형태로 저장
- 메모리가 허용하는 용량 까지 제한없이 저장가능
- 쿠키는 클라이언트에 저장되기 때문에 공유 PC의 경우 보안에 취약할 수 있다.  
하지만 세션은 서버에 저장되기 때문에 쿠키에 비해 보안이 좋다.
- 사용자(로그인)정보 및 장바구니 등에 사용한다.

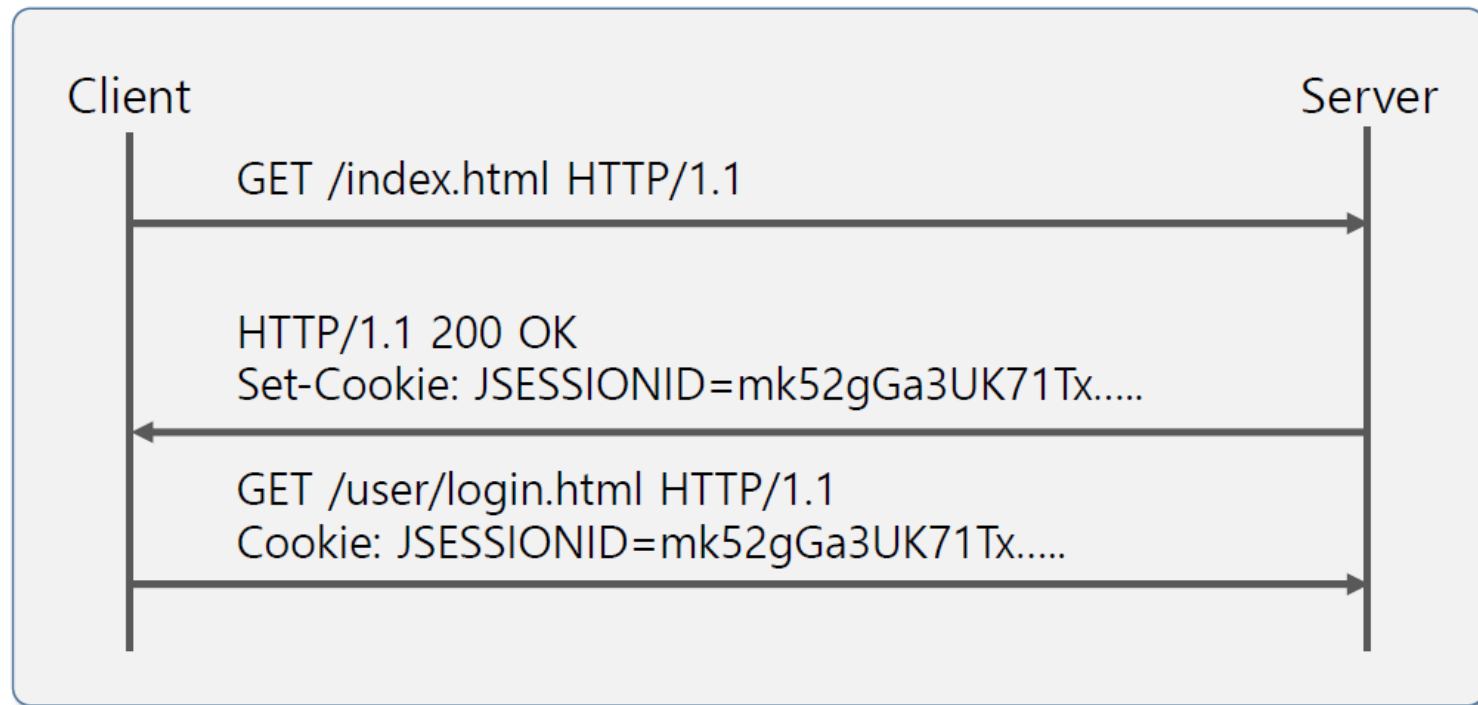
### ✓ session 동작 순서

- 클라이언트가 페이지를 요청
- 서버는 쿠키에 session id 가 있는지 확인한다.
- session id 가 존재하지 않으면 session id를 생성해 쿠키에 쓴 다음 클라이언트로 반환
- 생성된 session id를 이용하여 서버 내 메모리를 생성
- 클라이언트가 다음 요청 시 쿠키에 session id(JSESSIONID)를 포함해 전달하면 서버내에 저장된 session id와 비교하여 데이터를 조회



### ✓ session 설정

- 브라우저 당 하나의 JSESSIONID를 할당 받음
- 아이디 또는 닉네임과 같이 로그인했을 경우 자주 사용되는 정보를 session에 저장하면 db에 접근할 필요가 없으므로 효율적이다.





### ✓ session 사용하기

- 요청 객체로부터 session 객체를 얻어온다.
- session에 데이터를 설정하여 저장한다.

```
HttpSession session = request.getSession();  
session.setAttribute("userid", "ssafy");
```

- session에서 값을 반환하는 getAttribute 메서드는 반환형이 Object이다.

```
String userid = (String)session.getAttribute("userid");
```

## ✓ Session 주요 메서드

메서드	설명
<code>void setAttribute(String name, Object value)</code>	session 에 지정한 name에 해당하는 객체를 추가
<code>void setMaxInactiveInterval(int interval)</code>	사용자가 다음 요청을 보낼 때 까지 세션을 유지하는 최대시간(초단위)를 설정
<code>void invalidate()</code>	현재 세션을 없애고, 속성도 삭제한다.
<code>String getId()</code>	현재 세션의 고유 id를 반환
<code>long getLastAccessTime()</code>	현재 세션에 클라이언트가 마지막으로 요청을 보낸 시간을 반환(long)
<code>Object getAttribute(String name)</code>	name에 해당하는 속성값 반환, 반환형이 Object 임에 유의
<code>long getCreationTime()</code>	세션이 만들어진 시간 반환
<code>void removeAttribute(String name)</code>	세션에서 지정한 이름의 객체를 제거
<code>Enumeration getAttributeNames()</code>	세션에서 모든 객체의 이름을 Enumeration형으로 반환

# 다음 방송에서 만나요!

삼성 청년 SW 아카데미