Session

1. Session의 개념

- HTTP 프로토콜은 클라이언트의 요청(request)과 서버의 응답(response)이 이루어지고 나면 더 이상 연결상태를 '지속'하지 않는다.

- 클라이언트와 서버의 연결 정보를 계속 유지할 방법이 필요한데 대표적으로 '쿠키(Cookie)'와 '세션(Session)'이 있다.

- session은 클라이언트가 서버에 처음 연결될 때 생성되며 여러 요청(request)에 걸쳐 사용자 정보(사용자 로그인 상태 유지, 사용자별 설정 , 등)를 유지한다.

- 세션은 클라이언트와의 연결 정보를 유지하기 위한 정보를 웹 컨테이너(웹 서버)에 저장한다.

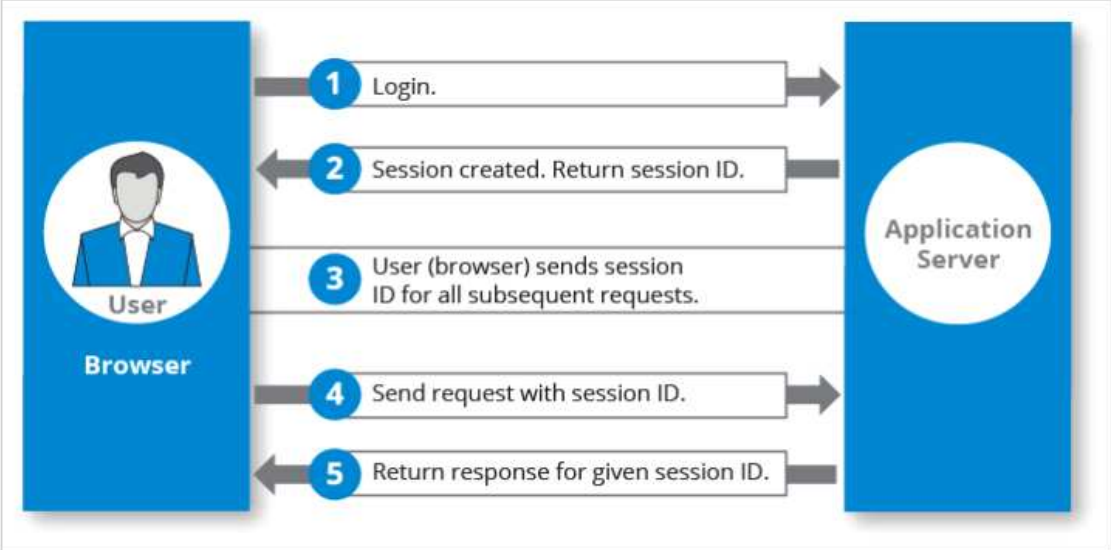
- 쿠키는 클라이언트와의 연결정보를 유지하기 위한 정보를 클라이언트측에 저장한 뒤에 웹서버가 클라이언트의 웹브라우저에서 쿠키를 읽어서 사용한다.

- 클라이언트측에 저장된 연결정보(쿠키)는 연결정보가 클라이언트에서 관리되고 있다는 점에서 연결정보를 서버측에서 관리하는 것보다 보안상 문제가 생길 확률이 많다.

- 세션은 클라이언트와의 연결정보를 서버에서 관리하므로 보안적인 측면에서 쿠키보다 안전하다.

- 세션만을 사용하면 서버에 부하를 줄 수도 있으므로 쿠키와 세션을 각각 목적에 맞게 사용하도록 권장한다.

2. 세션의 작동 원리



이미지 출처: https://hazelcast.com/glossary/web-session/

1) 생성: 사용자가 웹 사이트에 접속하여 서버와의 첫 번째 요청을 보낼 때 JSP에서 세션을 생성할 수 있다.

2) 식별자: 각 세션은 고유한 ID(세션 ID)를 가지며 이 ID를 통해 서버는 사용자를 식별한다.

3) 저장 정보: 세션에는 로그인 정보, 언어 설정 등 사용자 관련 정보가 저장된다.

4) 종료: 사용자가 로그아웃하거나 설정된 시간 동안 활동이 없으면 세션은 종료된다.

3. JSP에서 자주 사용하는 메서드 예시

HttpSession session = request.getSession();
- JSP에서 세션 객체에 접근하기 위해 session 내장 객체를 사용한다.

setAttribute(속성명, 값)
- 세션 속성명과 속성값으로 value를 할당한다.

getAttribute(속성명)
- 세션 속성명의 값을 Object 타입으로 리턴한다.(할당했던 데이터 타입으로 형변환 후 사용)
  해당 되는 속성명이 없을 경우에는 null 값을 리턴한다.

removeAttribute(속성명)
- 세션속성을 제거한다. (해당 속성만 제거)

invalidate()
- 세션속성을 제거한다. (초기화 , 주로 로그아웃시 사용)

setMaxInactiveInterval(유지기간(초))
- 세션을 유지하기 위한 세션 유지시간을 초 단위로 설정한다. (기본값은 30분)

4. 세션 관리

1) 유효 시간 설정
- 적절한 세션 유효 시간 설정을 통해 불필요한 리소스 점유를 방지한다.

2) 세션 데이터 최소화
- 필요한 최소한의 정보만 세션에 저장하여 성능을 최적화한다.

5. 주의사항

- 많은 수의 사용자와 데이터를 다룰 경우, 세션 데이터로 인한 서버 메모리 사용 증가에 주의 해야하며 , 대체 인증 수단을 고려해 볼 수 있다.