

LOGIN

세션을 이용한 로그인과 JWT를 이용한 로그인 방식의 차이

향해 14기 R반 홍종훈

SESSION

SESSION이란?

- 사용자가 웹 사이트에 접속하여 해당 사이트를 이용하는 동안의 상태 정보를 저장하는 방법을 말합니다.
- 이 정보를 이용해 웹 애플리케이션의 상태를 유지합니다.
이 정보는 서버 측에 저장되며, 브라우저를 닫거나 세션이 만료될 때까지 유지됩니다.

SESSION 저장 방식에 따른 차이

저장소

브라우저

서버

저장위치

- 브라우저 측에서 관리됩니다.
- 브라우저마다 독립적으로 관리되며,
- 브라우저가 종료될 때까지 유지됩니다.

- 서버 측에서 관리됩니다.
- 여러 클라이언트가 공유하는 데이터를 저장할 수 있으며,
- 서버가 종료될 때까지 유지됩니다.

SESSION 저장 방식에 따른 차이

저장소

브라우저

서버

저장데이터

주로 로그인 정보나
사용자 환경 설정 등의
클라이언트 측 정보를 저장
합니다.

클라이언트의 상태 정보나
사용자 데이터 등
서버 측에서 필요한 정보를
저장합니다.

SESSION 저장 방식에 따른 차이

저장소

브라우저

서버

보안

클라이언트 측에서 해당 데이터를 확인하거나 조작하는 것이 가능합니다.

보안 상 중요한 데이터는 브라우저의 세션 저장소에 저장하면 안됩니다.

서버의 세션 저장소는 서버 측에서 관리되므로, 보안성이 높습니다.

서버에서 적절한 보안 조치를 취하면 데이터의 안전성을 보장할 수 있습니다.

SESSION 저장 방식에 따른 차이

저장소	브라우저	서버
세션 ID의 전달 방식	쿠키(cookie)나 웹 스토리지(localStorage/sessionStorage) 등을 이용하여 전달	HTTP 헤더 등을 이용하여 전달

그래서 SESSION으로 로그인 구현 가능?

- 저장 위치, 저장 데이터, 보안성, 세션 ID의 전달 방식 등에서 차이가 있습니다.
- 그러나 두 저장소 모두 세션 기반 인증 방식에서 사용될 수 있으며,
- 이를 이용하여 사용자 인증을 구현할 수 있습니다.

JWT

JWT란?

- 웹 어플리케이션에서 사용자 인증과 정보 교환을 위한 토큰 기반의 인증 방식입니다.
- JWT는 토큰 기반 인증 시스템으로, 토큰에 사용자 정보를 담아서 클라이언트와 서버간에 정보를 안전하게 전달합니다.
- JWT는 JSON 형태의 데이터를 사용하여 정보를 전달하며, 일반적으로 3개의 파트로 이루어져 있습니다.

JWT 3개의 파트

1. JWT 토큰의 헤더(Header)로,
토큰의 타입과 해싱 알고리즘 정보를 담고 있음.
2. JWT 토큰의 페이로드(Payload)로,
인증을 위한 정보를 담고있음.
3. JWT 토큰의 서명(Signature)으로,
토큰의 유효성 검증을 위한 서명 정보를 담고 있음.

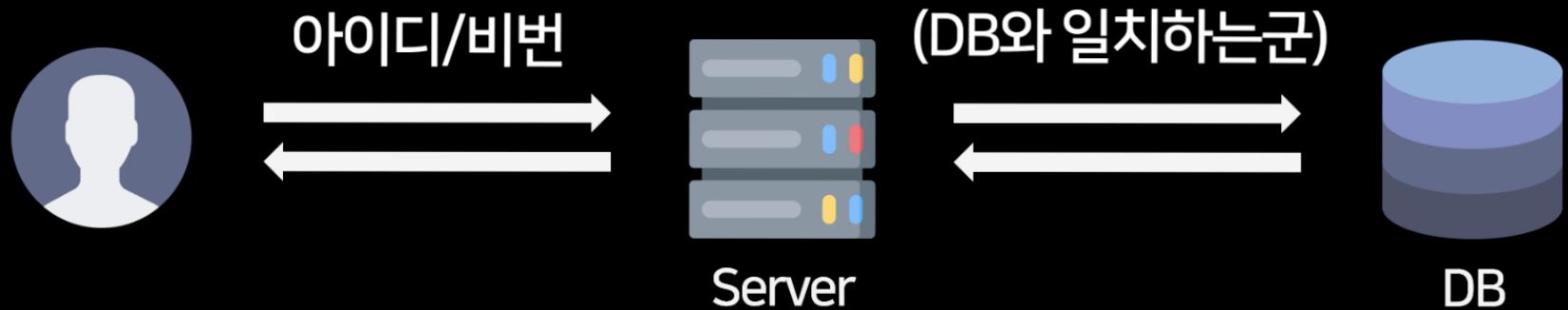
회원가입



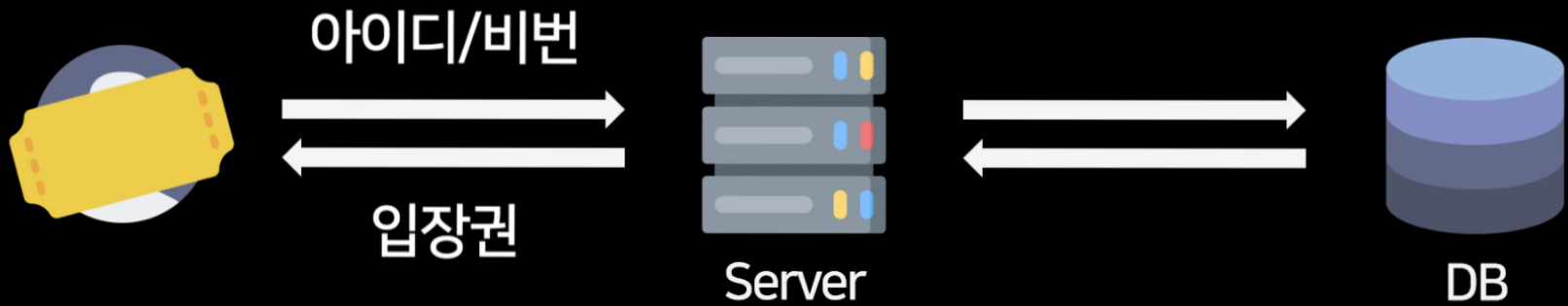
로그인



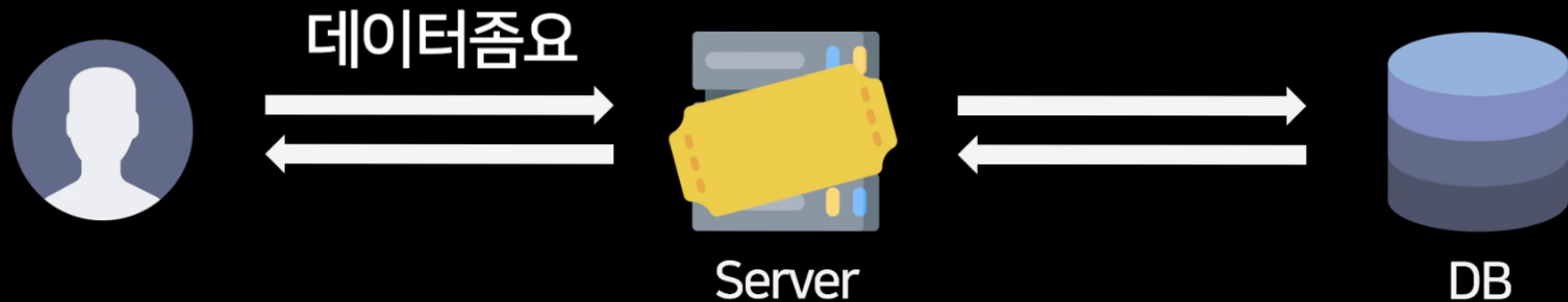
로그인



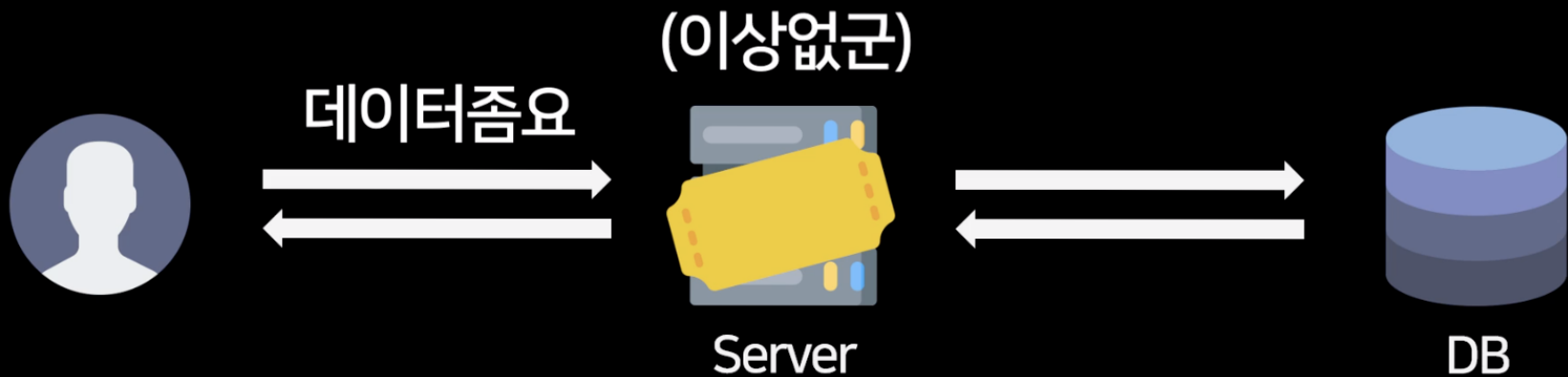
로그인



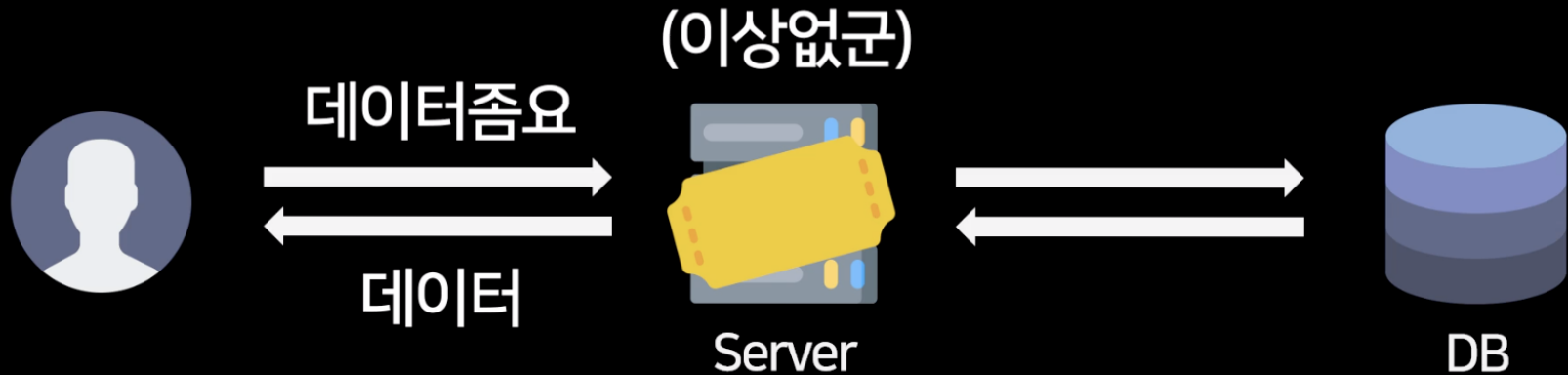
서버에 데이터 요청하려면?



서버에 데이터 요청하려면?



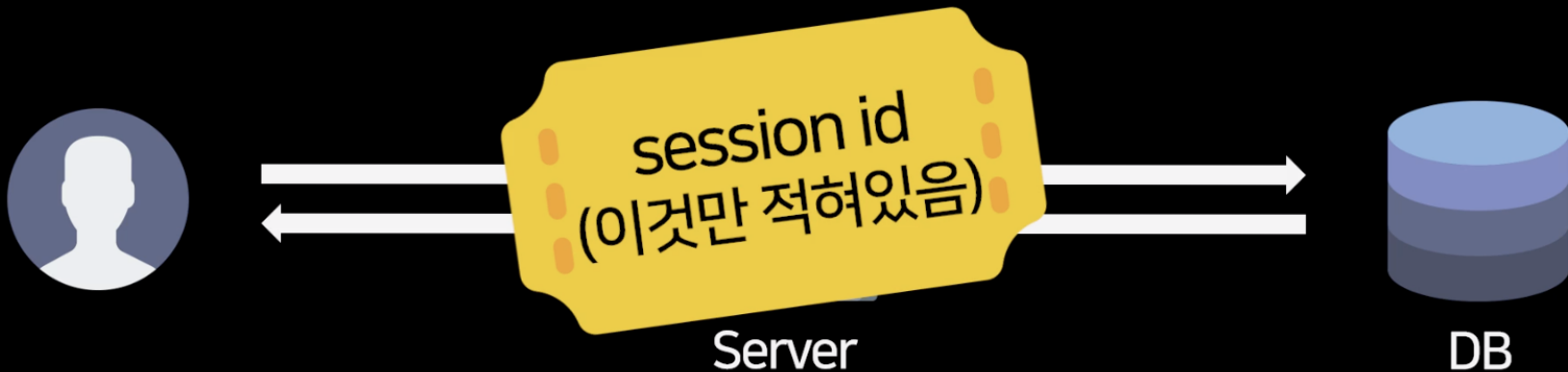
서버에 데이터 요청하려면?



SESSION을 이용한 로그인

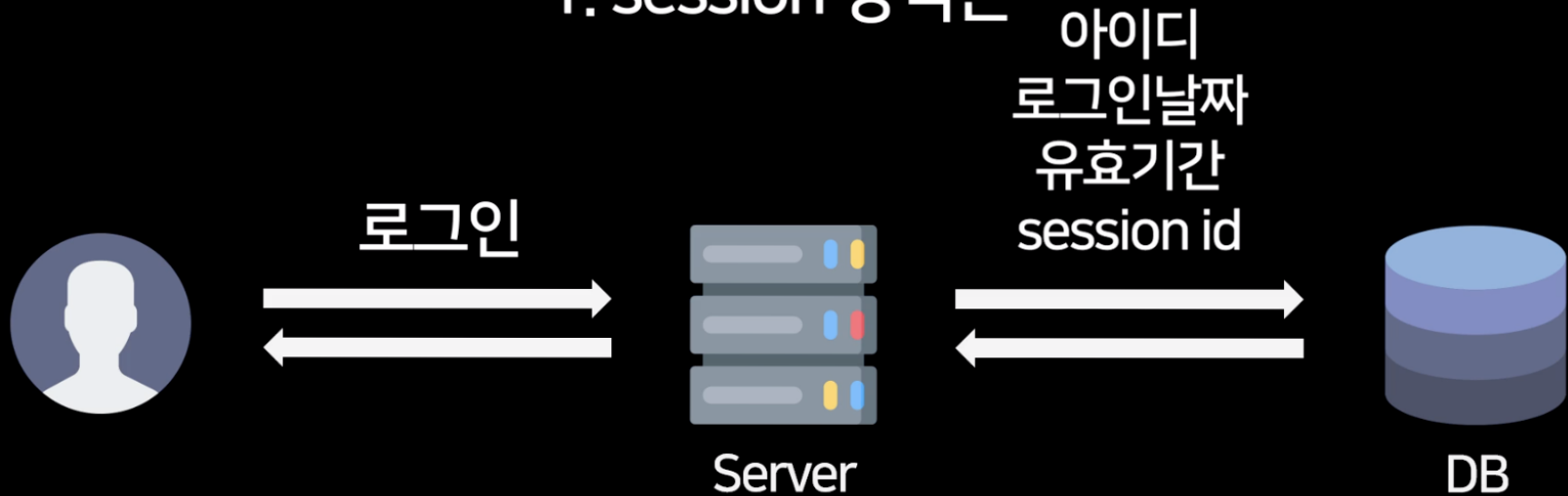
SESSION을 이용한 로그인

1. session 방식은



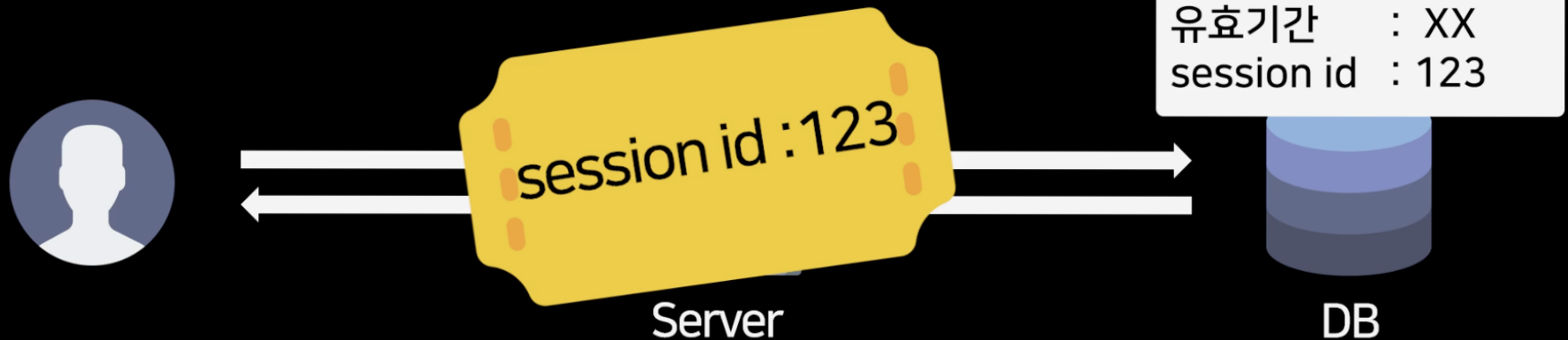
SESSION을 이용한 로그인

1. session 방식은



SESSION을 이용한 로그인

1. session 방식은



SESSION을 이용한 로그인

1. session 방식은



SESSION을 이용한 로그인

1. session 방식은

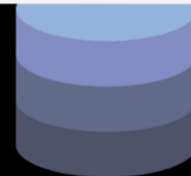


데이터요요

session id : 123

Server

아이디 : XX
로그인날짜 : XX
유효기간 : XX
session id : 123



DB

SESSION을 이용한 로그인

1. session 방식은

별 문제 없군



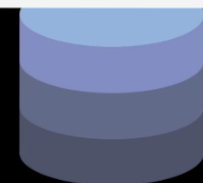
데이터요청



데이터보냄



Server



DB

아이디 : XX
로그인날짜 : XX
유효기간 : XX
session id : 123

SESSION을 이용한 로그인

1. session 방식은

(session id 보관용 DB)



Server

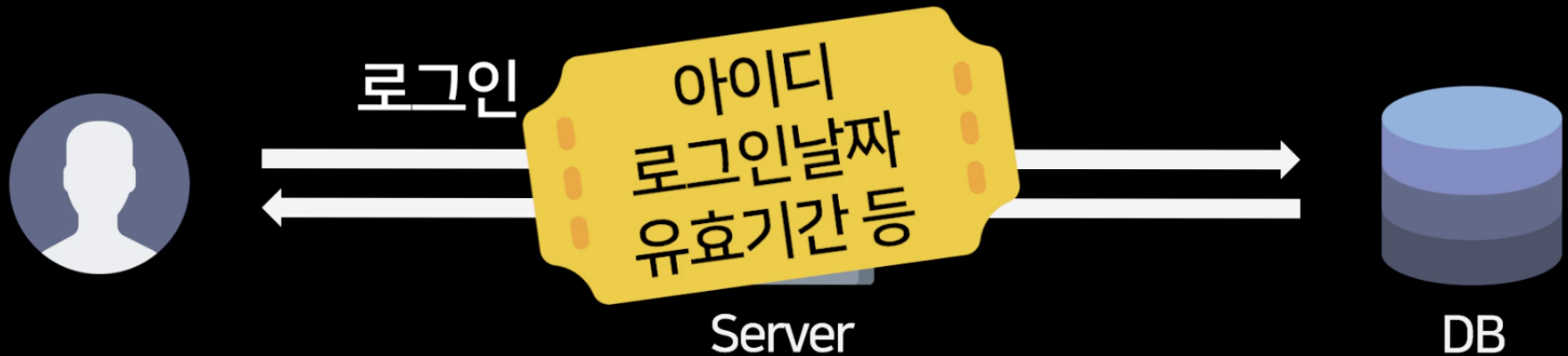


장점 : 입출력 빠름

JWT를 이용한 로그인

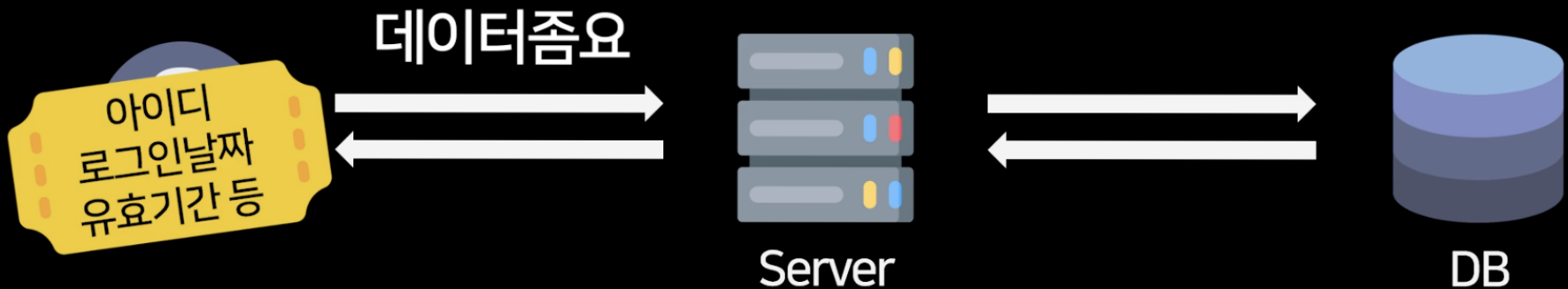
JWT를 이용한 로그인

2. token 방식은



JWT를 이용한 로그인

2. token 방식은



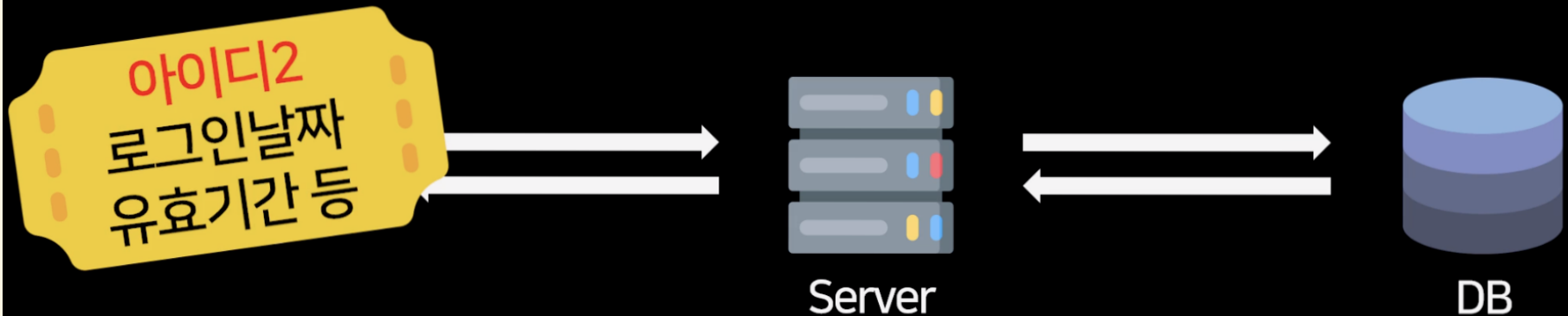
JWT를 이용한 로그인

2. token 방식은



JWT를 이용한 로그인

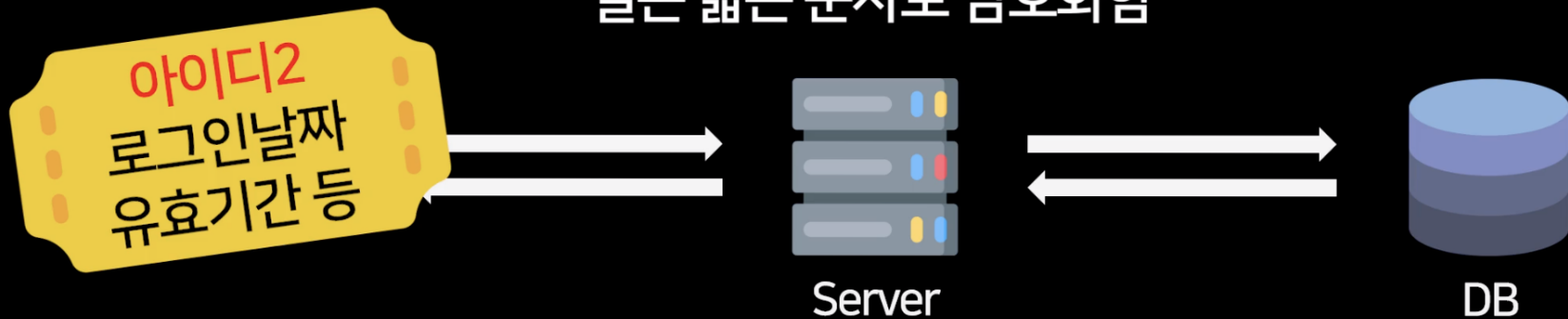
2. token 방식은



JWT를 이용한 로그인

2. token 방식은

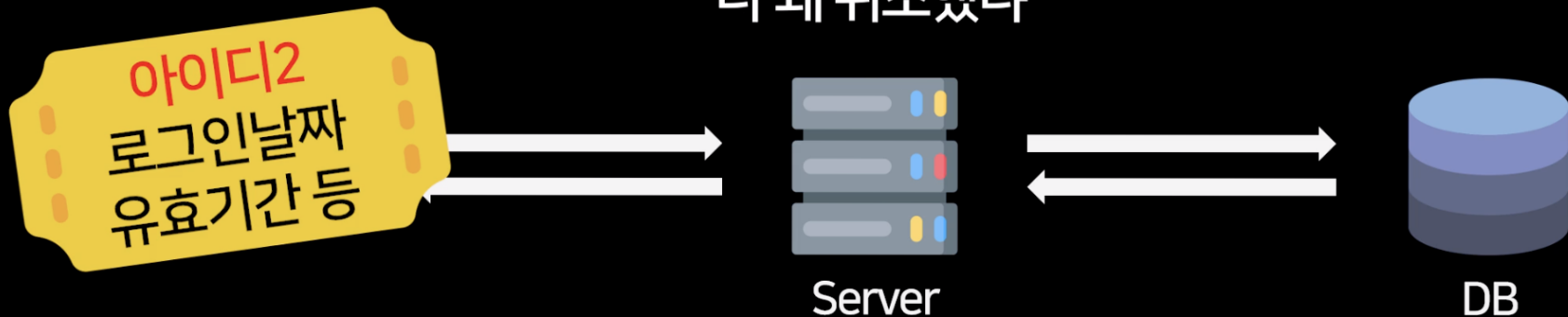
실은 짧은 문자로 암호화함



JWT를 이용한 로그인

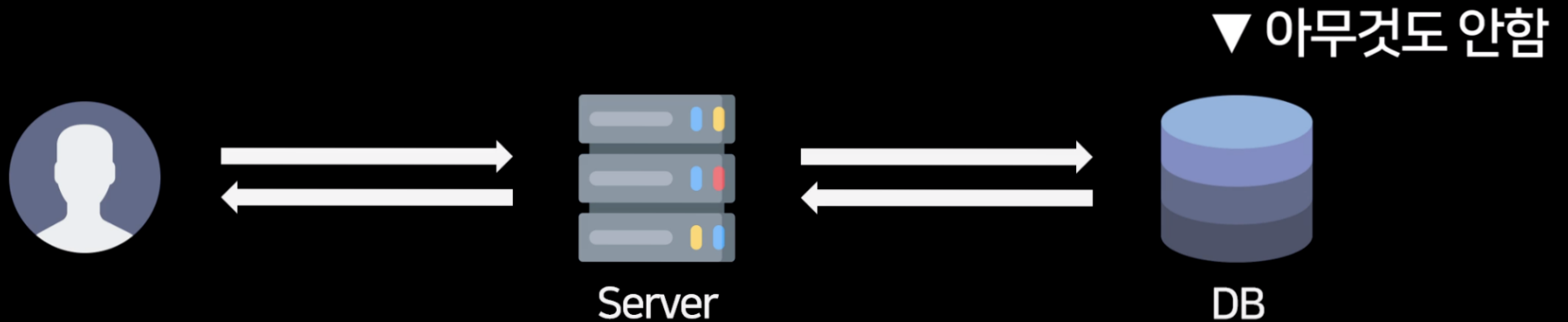
2. token 방식은

“너 왜 위조했냐”



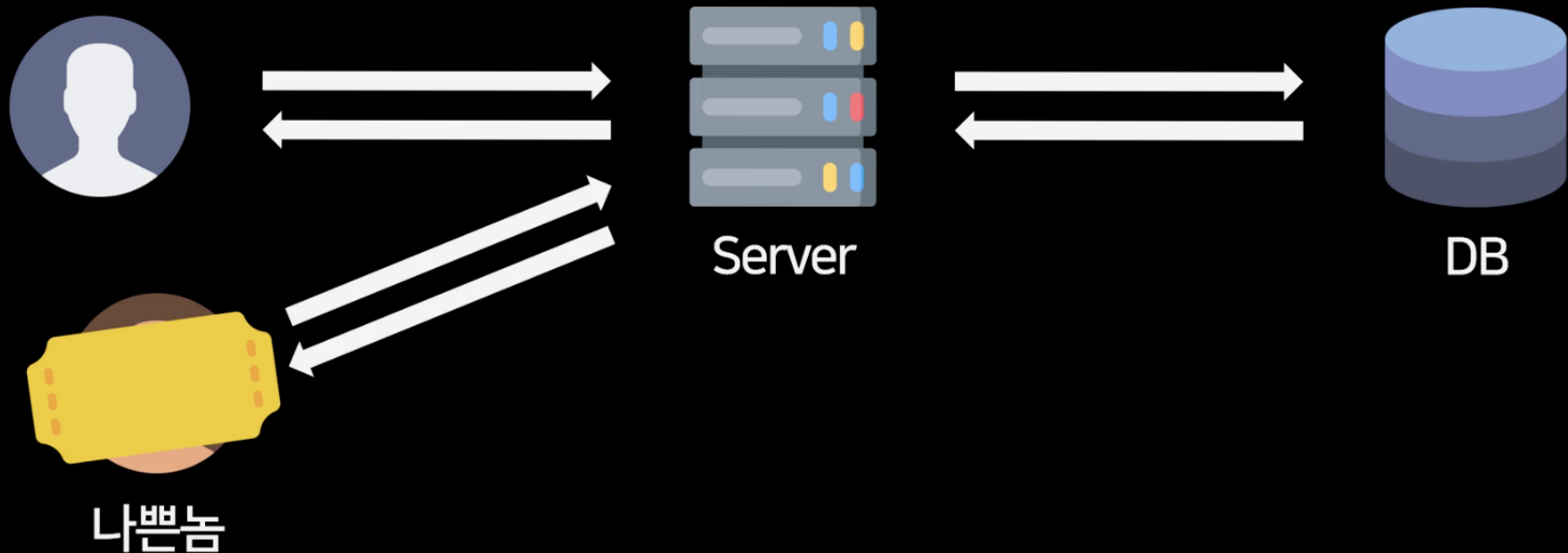
JWT를 이용한 로그인

2. token 방식은



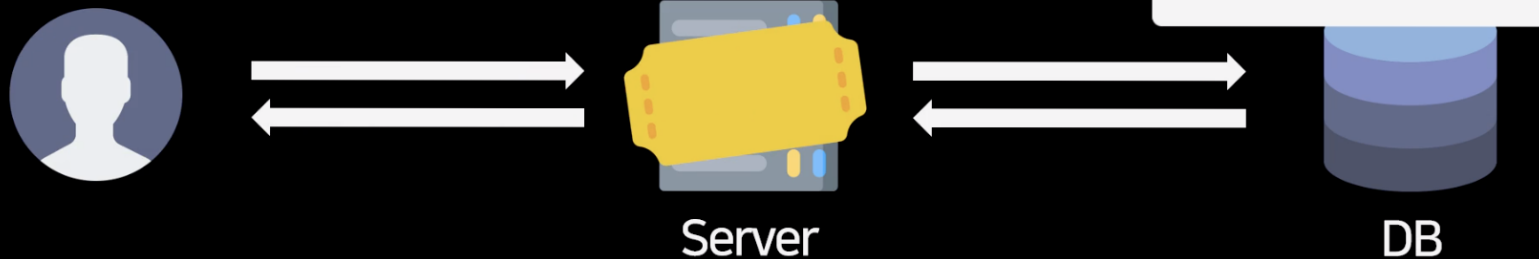
JWT를 이용한 로그인

2. token 방식은



JWT를 이용한 로그인

2. token 방식은



SESSION방식과 JWT방식의 차이점

- Session

DB를 통해 세션 상태를 관리합니다. 서버는 세션 ID를 발급하여 클라이언트에게 전달하고, 이후 클라이언트가 요청을 보낼 때마다 해당 세션 ID를 이용하여 세션 상태를 유지합니다.

- JWT

DB에서 토큰 상태를 관리하지 않습니다. 클라이언트 측에서 토큰을 관리하고, 서버에서는 해당 토큰의 유효성만 검증합니다.

END