Code Factory

Token vs Session

토큰과 세션

Code Factory

Session 이란?

유저의 정보를 데이터베이스에 저장하고
상태를 유지하는 도구

Code Factory

Session의 특징

- Session은 특수한 ID 값으로 구성되어있다.
- Session은 서버에서 생성되며 클라이언트에서 쿠키를 통해 저장된다.
- 클라이언트에서 요청을 보낼때 Session ID를 같이 보내면 현재 요청을 보내는 사용자가 누구인지 서버에서 알 수 있다. (요청마다 매번 아이디와 비밀번호를 물어볼 필요 없음)
- Session ID는 데이터베이스에 저장되기때문에 요청이 있을때마다 매번 데이터베이스를 확인해야한다.
- 서버에서 데이터가 저장되기때문에 클라이언트에 사용자 정보가 노출될 위험이 없다.
- 데이터베이스에 Session을 저장해야하기때문에 Horizontal Scaling이 어렵다.

Token 이란?

유저의 정보를 Base 64로 인코딩된
String 값에 저장하는 도구

Code Factory

Token의 특징

- Token은 Header, Payload, Signature로 구성되어있으며 Base 64로 인코딩 되어있다.
- Token은 서버에서 생성되며 클라이언트에서 저장된다.
- 클라이언트에서 요청을 보낼때 Token ID를 같이 보내면 현재 요청을 보내는 사용자가 누구인지 서버에서 알 수 있다. (요청마다 매번 아이디와 비밀번호를 물어볼 필요 없음)
- Token은 데이터베이스에 저장되지않고 Signature 값을 이용해서 검증할 수 있다. 그래서 검증할때마다 데이터베이스를 매번 들여다볼 필요가 없다.
- 정보가 모두 토큰에 담겨있고 클라이언트에서 토큰을 저장하기 때문에 정보 유출의 위험이 있다.
- 데이터베이스가 필요없기때문에 Horizontal Scaling이 쉽다.

Session 생성 방식

Client

API 서버

Database

❶ ID / Pass 전송

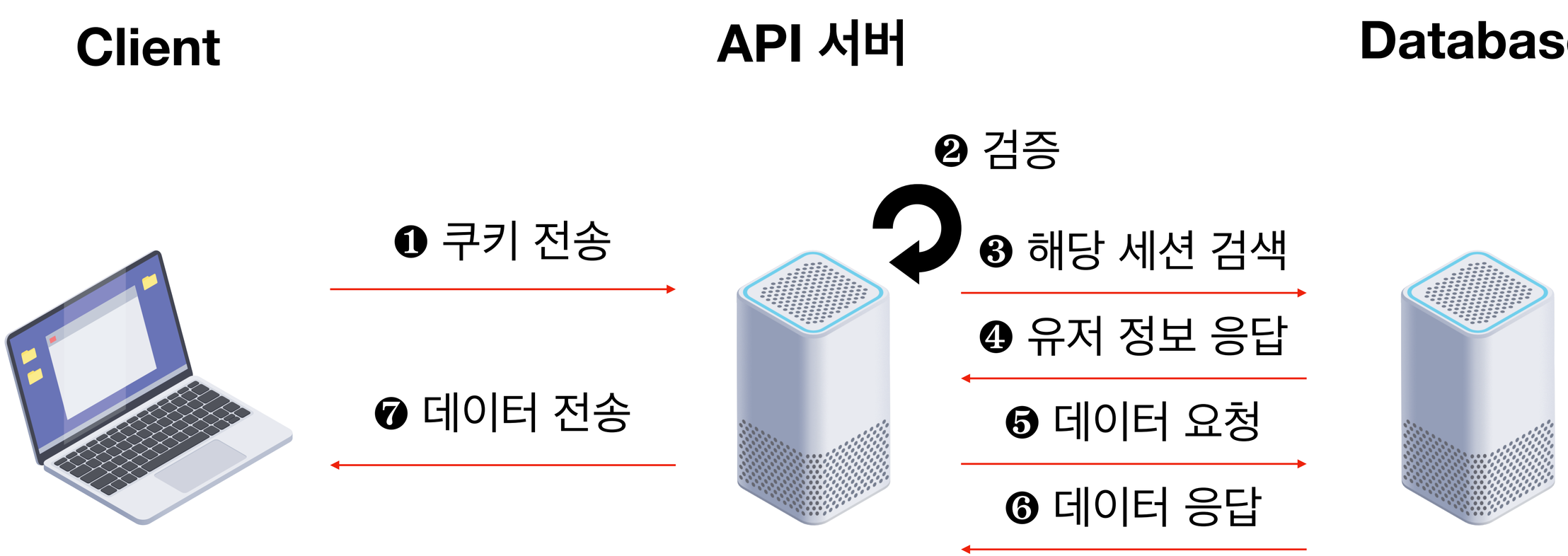
❷ 검증

❸ 세션 생성 및 저장

❹ 쿠키 전송

Code Factory

Session 사용 방식



Code Factory

Token 생성 방식

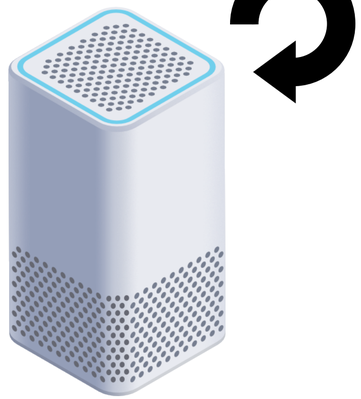
Client

API 서버

❷ 검증

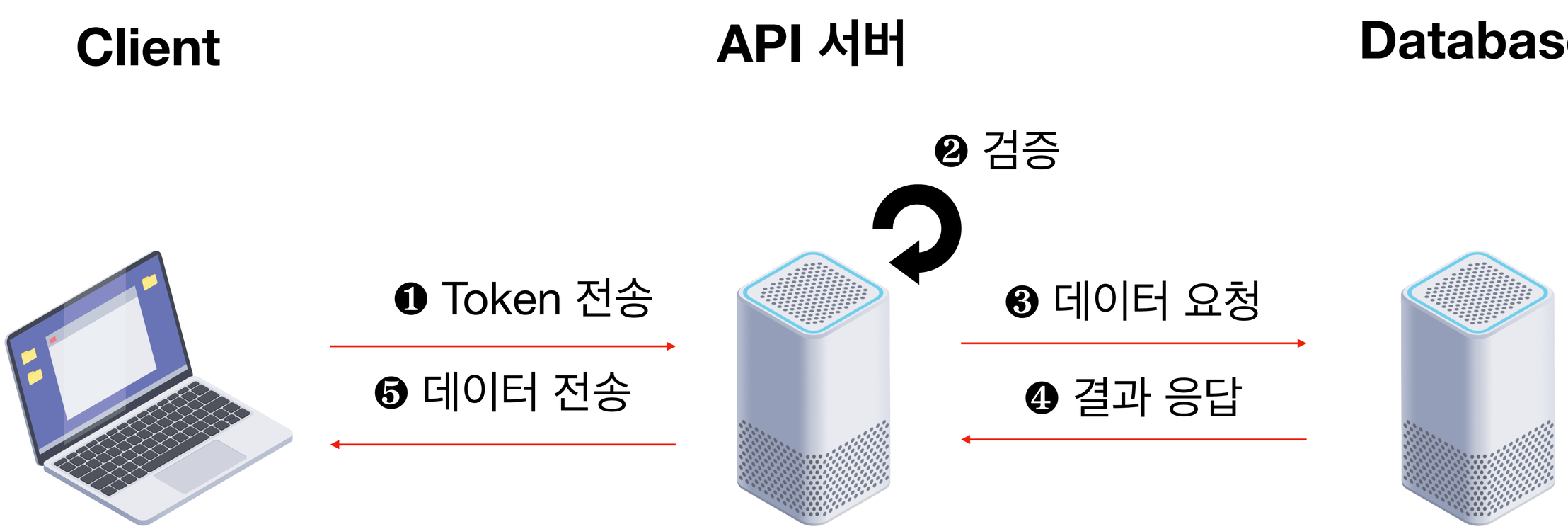
❶ ID / Pass 전송

❹ Token 전송

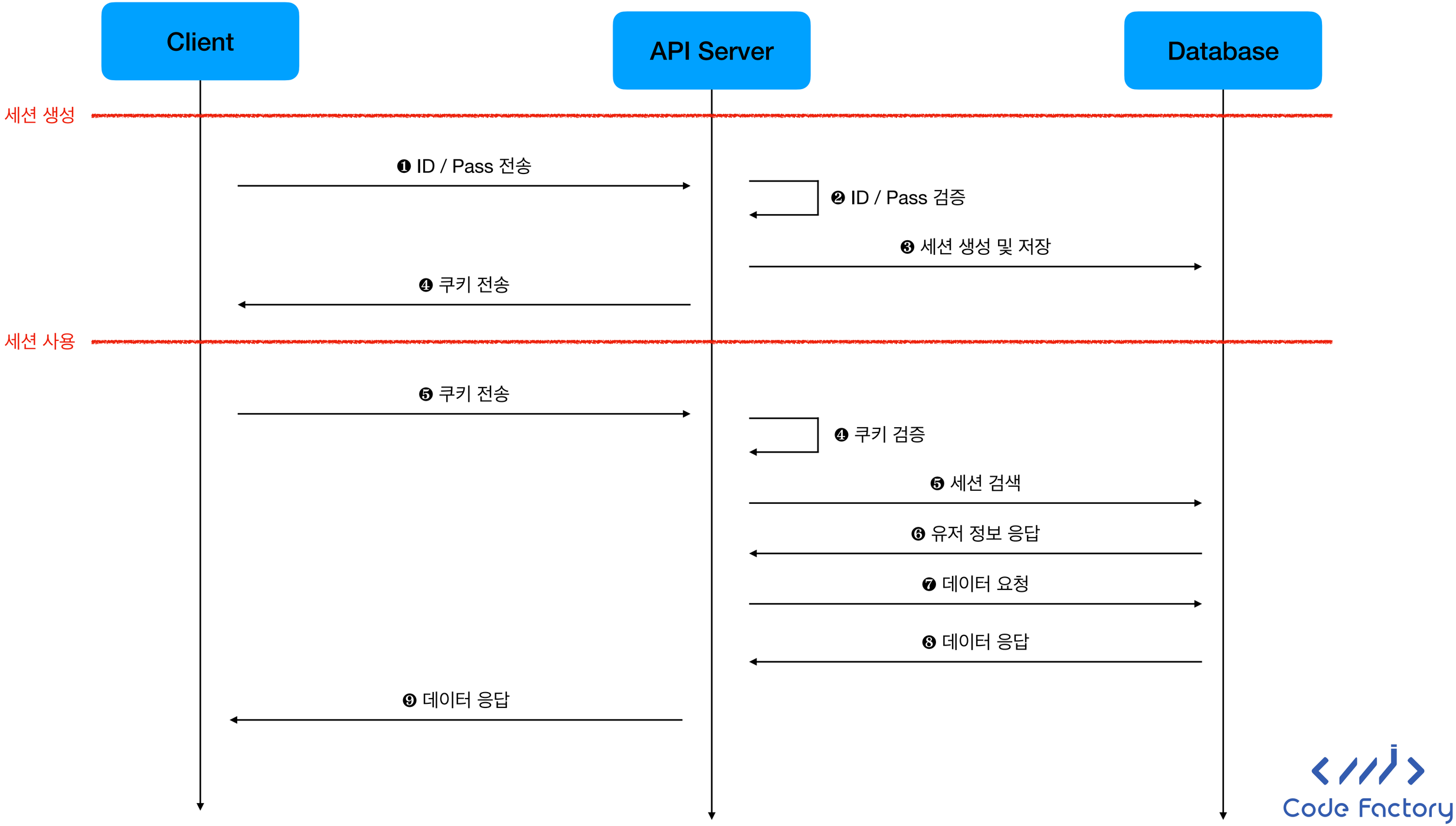


Code Factory

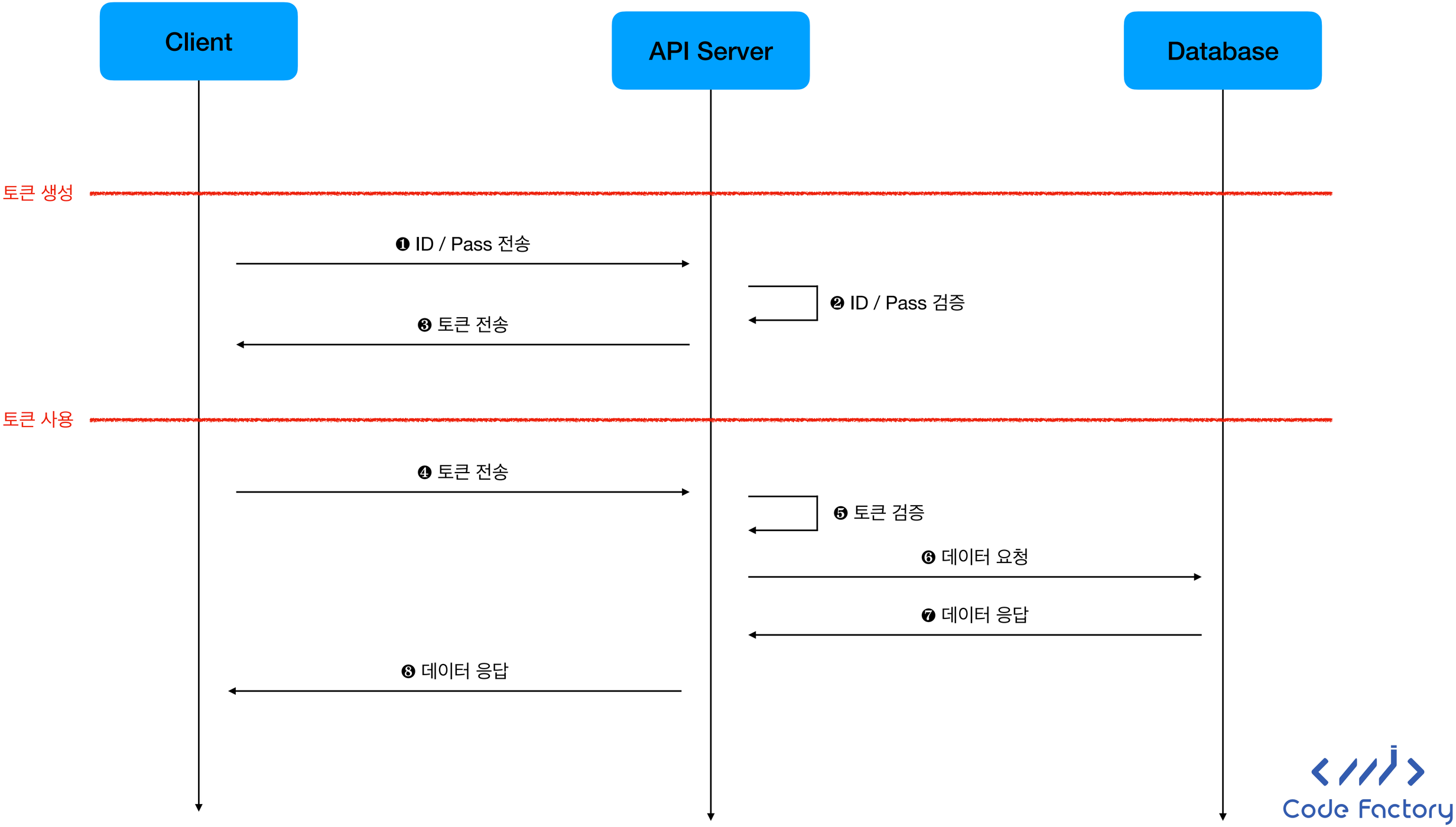
Token 사용 방식



Code Factory



Code Factory



Code Factory

Session vs Token 비교

비교 요소	Session	Token
유저의 정보를 어디에서 저장하고 있는가?	서버	클라이언트
클라이언트에서 서버로 보내는 정보는?	쿠키	토큰
유저 정보를 가져올때 데이터베이스를 확인해야하는가?	확인필요	토큰의 Payload에 들어있는 정보만 필요할경우 확인 불필요
클라이언트에서 인증 정보를 읽을 수 있는가?	불가능	가능
Horizontal Scaling이 쉬운가?	어려움	쉬움