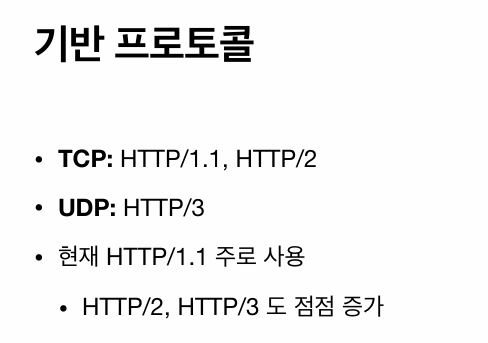


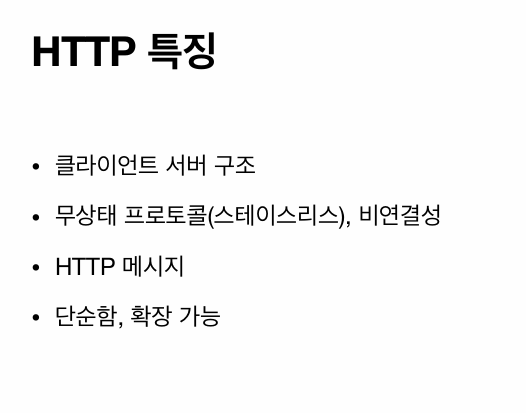
HTTP 1.1 에 대부분의 기능이 들어있고, HTTP2와 3에서는 성능개선이 됨.

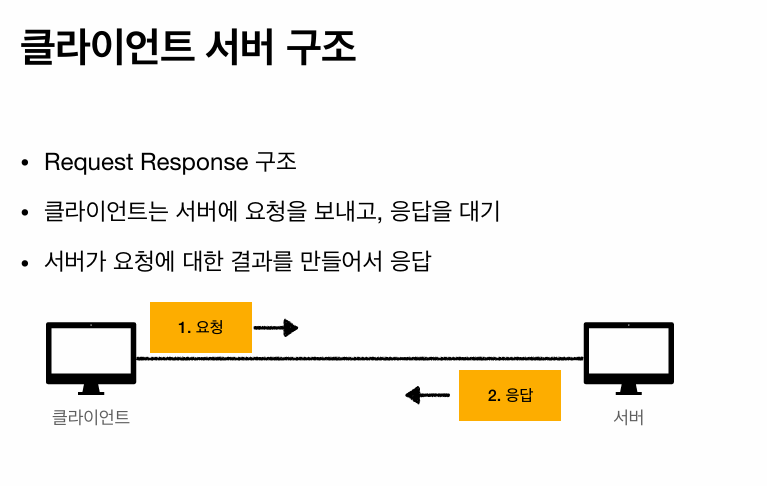


TCP : 속도가 느림.

UDP : 속도가 빠름. -> 속도를 빠르게 하기 위해 UDP레벨에 HTTP를 올림. 이것이 HTTP3

최근에는 HTTP2와 3이 많이 쓰이고 있음.





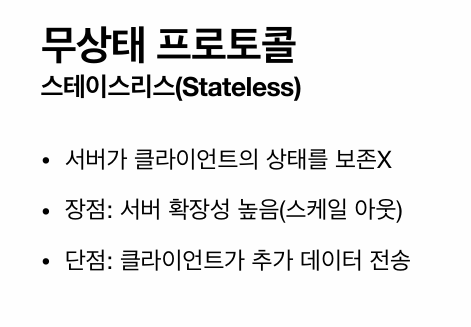
클라이언트가 HTTP메시지를 통해 서버에 요청을 보냄. 클라이언트는 서버의 요청이 올 때까지 대기를 함. 서버가 요청에 대한 결과를 만들어서 응답하면, 응답 결과를 열어서 클라이언트가 동작하게 됨.

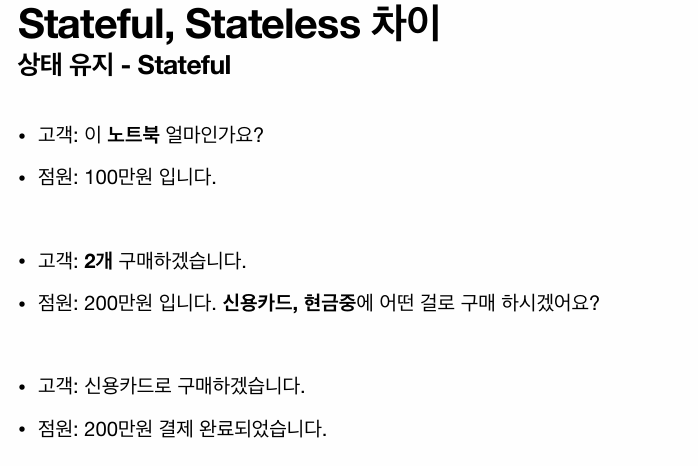
클라이언트와 서버를 분리하는 것이 중요함. 이전에는 클라이언트와 서버가 분리되어있지 않았음. 데이터나 비즈니스 로직은 서버에 넣고, 클라이언트에는 UI와 사용성에 집중.

클라이언트와 서버가 각각 독립적으로 발전할 수 있음.

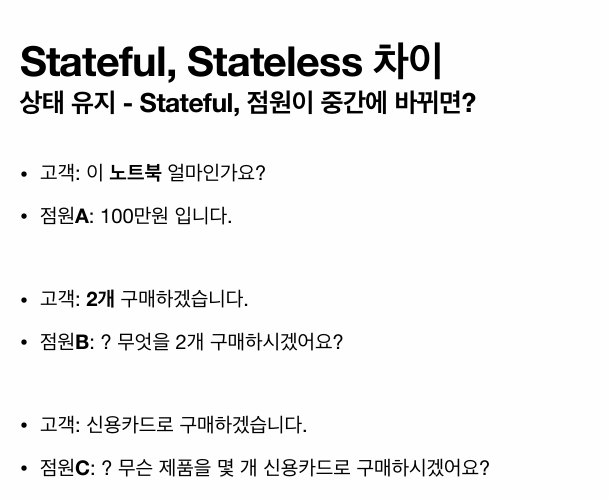
클라이언트는 UI를 어떻게 만들지만 고민. 복잡한 비즈니스로직은 서버에서 처리하게 함.

비즈니스가 잘되서 트래픽이 많이 증가함 -> 클라이언트는 건들지 않고, 서버쪽만 수정하면 됨.

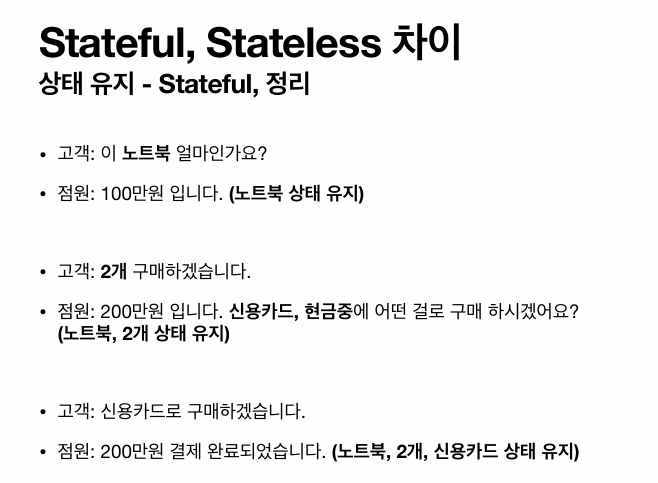


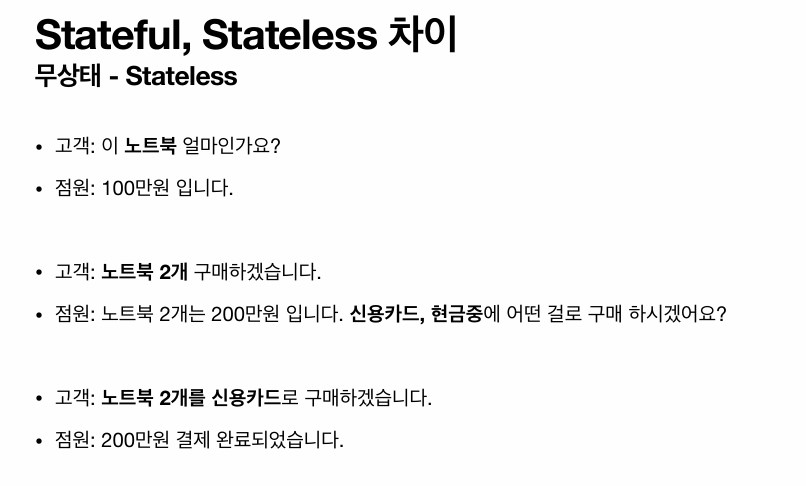


Stateful은 문맥을 보존하는 개념.

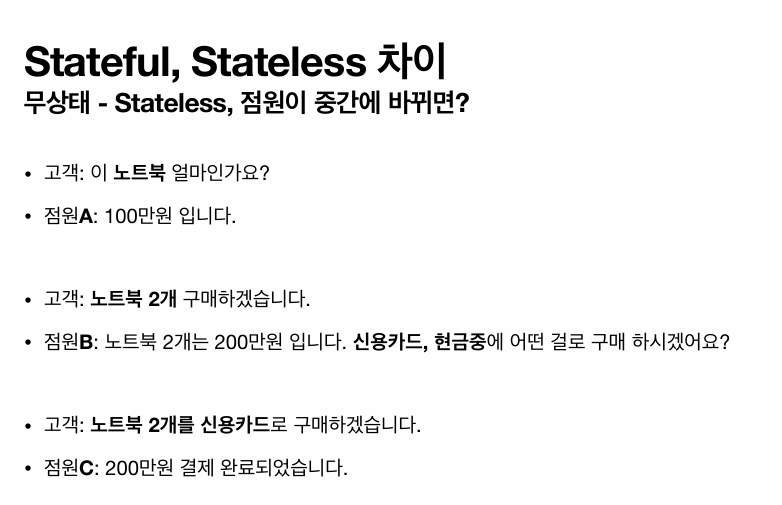


점원(서버)이 상태를 유지하는 경우 다른 점원(서버)은 유지된 상태값을 모름.

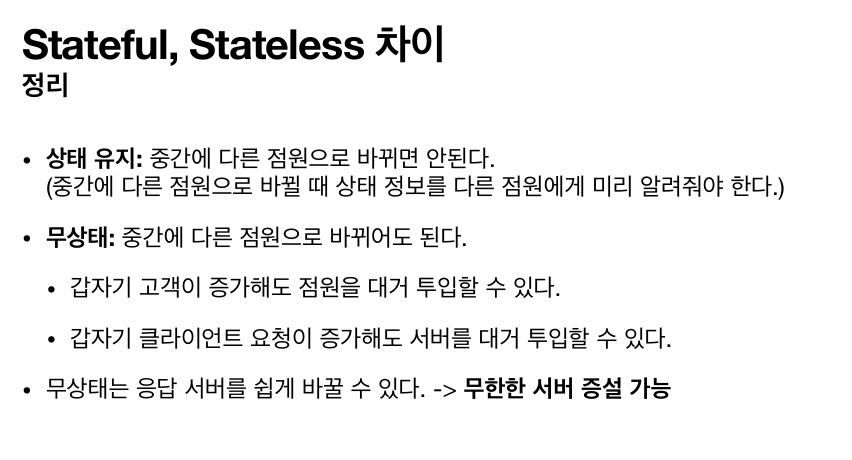


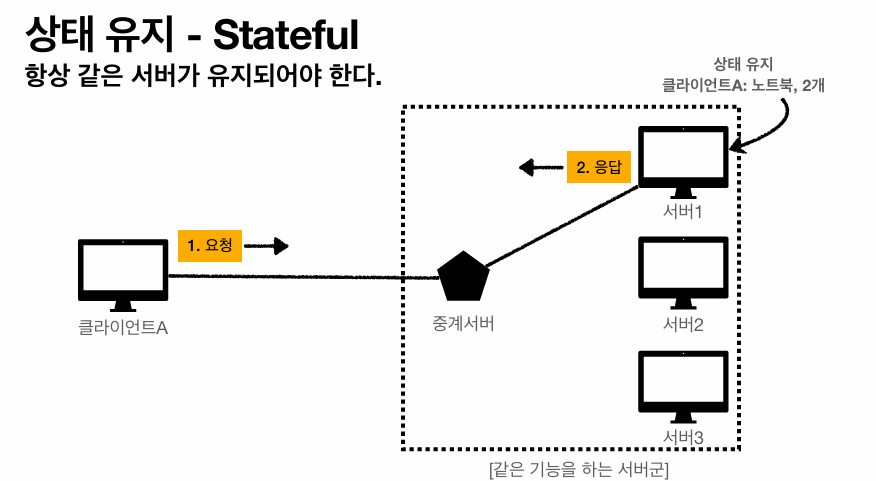


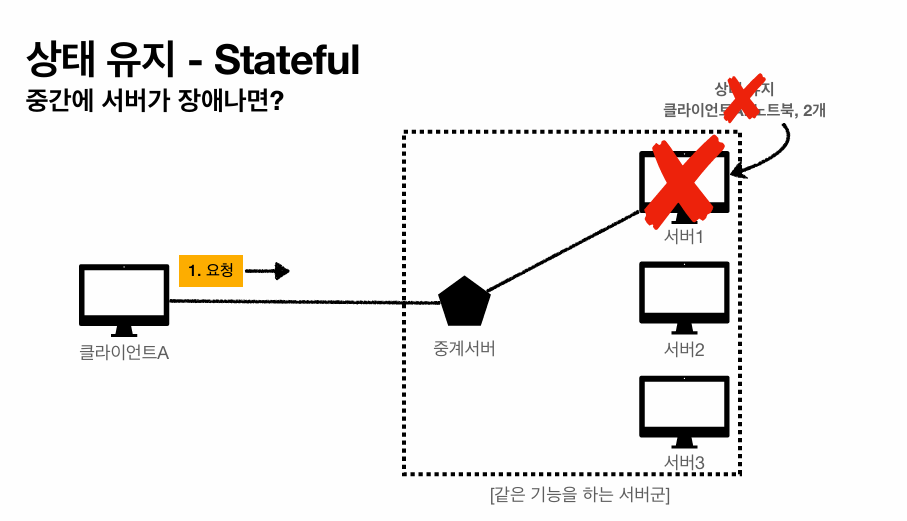
고객이 상태를 저장한다.



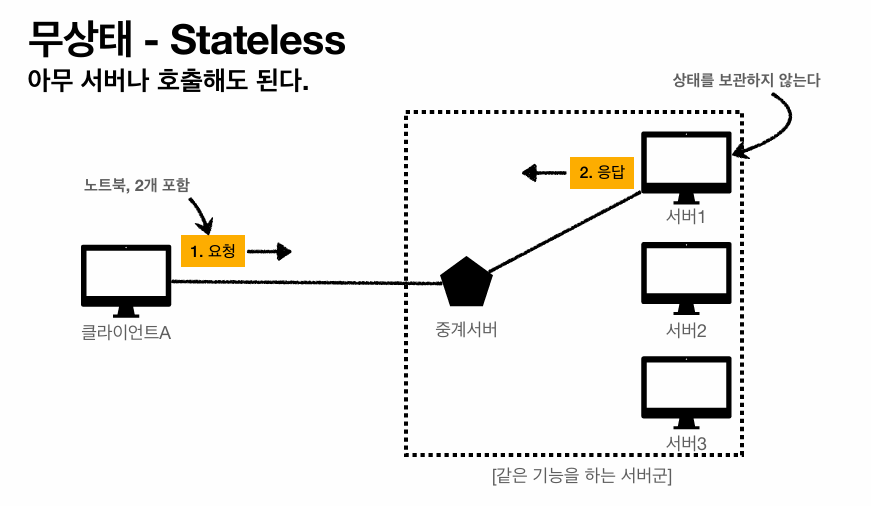
점원이 바뀌더라도 대화가 이어짐. -> 서버를 이중화 삼중화 가능. 서버가 무한 증식이 가능하다.

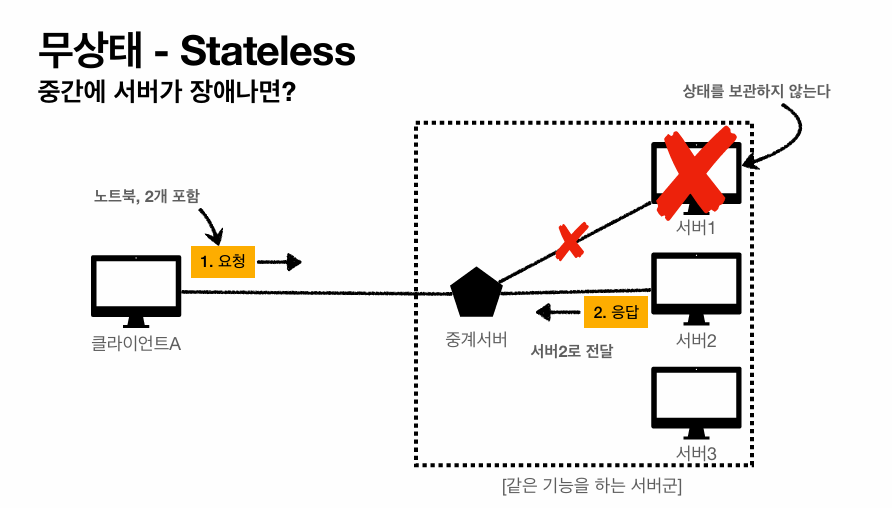


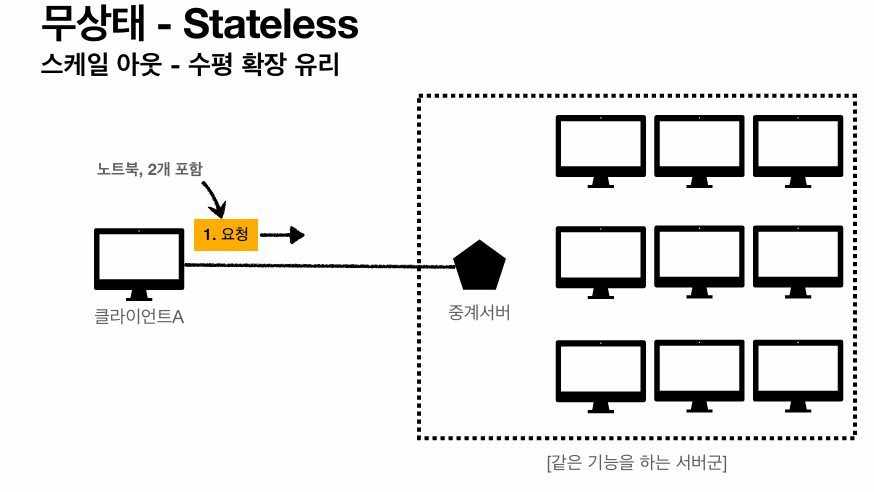


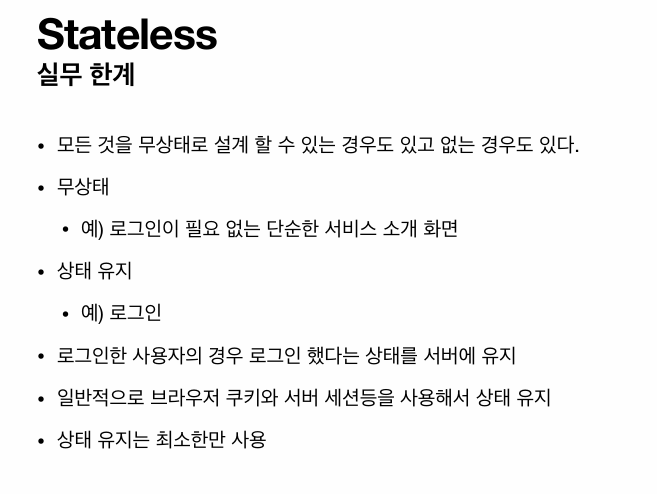


클라이언트A는 처음부터 다시 소통해야 함.









단점 : 클라이언트는 데이터를 너무 많이 보냄.

최대한 무상태로 설계를 하고, 상태 유지는 최소한만 사용해야 한다.

