

P R E S E N T A T I O N

13주차 주제 **먹등성**

~HTTP 먹등성~

by mun



HTTP의 속성

요약표

HTTP 메소드	RFC	요청에 Body가 있음	응답에 Body가 있음	안전	멱등(Idempotent)	캐시 가능
GET	RFC 7231	아니요	예	예	예	예
HEAD	RFC 7231	아니요	아니요	예	예	예
POST	RFC 7231	예	예	아니요	아니요	예
PUT	RFC 7231	예	예	아니요	예	아니요
DELETE	RFC 7231	아니요	예	아니요	예	아니요
CONNECT	RFC 7231	예	예	아니요	아니요	아니요
OPTIONS	RFC 7231	선택 사항	예	예	예	아니요
TRACE	RFC 7231	아니요	예	예	예	아니요
PATCH	RFC 5789	예	예	아니요	아니요	예

안전
멱등
캐시가능

<https://ko.wikipedia.org/wiki/HTTP>

멱등 (Idempotent)

$f(f(x)) = f(x)$ 여러번 요청하더라도 결과는 같다

요약표

HTTP 메소드		안전	멱등(Idempotent)	캐시 가능
GET	RF	예	예	예
HEAD	RF	예	예	예
POST	RF	아니요	아니요	예
PUT	RF	아니요	예	아니요
DELETE	RF	아니요	예	아니요
CONNECT	RF	아니요	아니요	아니요
OPTIONS	RF	예	예	아니요
TRACE	RF	예	예	아니요
PATCH	RF	아니요	아니요	예

멱등

GET 몇번 조회하든 똑같은 결과 조회

PUT 결과를 대체 - 최종 결과는 같음

DELETE 결과를 삭제 - 몇번 호출해도 같음

멱등아님

POST 중복 발생 가능

멱등 (Idempotent)

$f(f(x)) = f(x)$ 여러번 요청하더라도 결과는 같다

요약표

HTTP 메소드 ◆		안전 ◆	멱등
GET	RF	예	
HEAD	RF	예	
POST	RF	아니요	
PUT	RF	아니요	
DELETE	RF	아니요	
CONNECT	RF	아니요	
OPTIONS	RF	예	
TRACE	RF	예	
PATCH	RF	아니요	

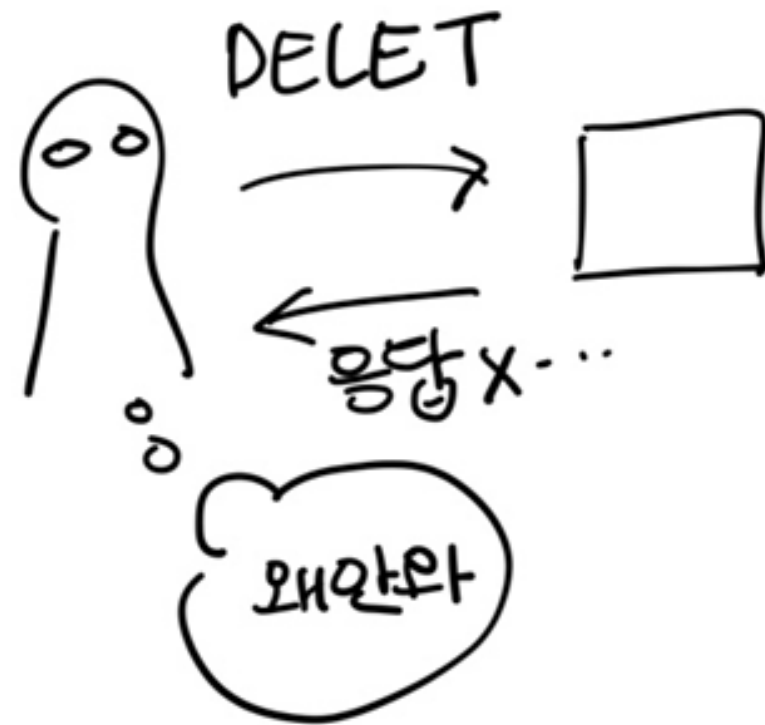


멱등 (Idempotent)

$f(f(x)) = f(x)$ 여러번 요청하더라도 결과는 같다

요약표

HTTP 메소드 ◆		안전 ◆	멱등
GET	RF	예	
HEAD	RF	예	
POST	RF	아니요	
PUT	RF	아니요	
DELETE	RF	아니요	
CONNECT	RF	아니요	
OPTIONS	RF	예	
TRACE	RF	예	
PATCH	RF	아니요	아니요



같은 요청을 다시 보내도 되는가
판단근거가 된다

이런 경우는요?

User1 **GET** userprofile - usernickname : 문쿨

User2 **PUT** userprofile - usernickname : 달선

User1 **GET** userprofile - usernickname : 달선



어? 결과가 다른데요?

이런 경우는요?

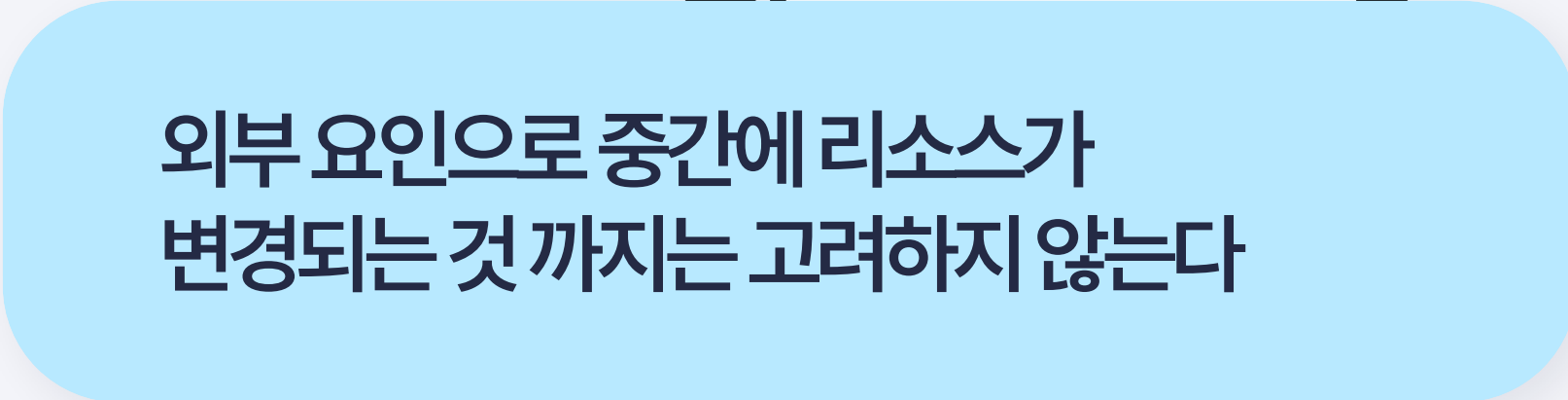
User1 **GET** userprofile - usernickname : 문쿨

User2 **PUT** userprofile - usernickname : 달선

User1 **GET** userprofile - usernickname : 달선



어? 결과가 다른데요?



외부 요인으로 중간에 리소스가
변경되는 것까지는 고려하지 않는다

동일 사용자의 요청으로 이해하면 됩니다

멱등성 보장하기 멱등키

멱등키를 API 요청에 포함시킨다

```
Idempotency-Key: {IDEMPOTENCY_KEY}
```

```
curl --request POST \  
  --url https://api.tosspayments.com/v1/payments/5zJ4xY7m0kODnyRp/cancel \  
  --header 'Authorization: Basic dGVzdF9za196WExrS0V5cE5BcldtbzUwblgzG1lYXhZRzVS0g==' \  
  --header 'Content-Type: application/json' \  
  --header 'Idempotency-Key: SAAABPQbcqjEXiDL' \  
  --data '{"cancelReason":"고객 변심"}'
```

토스페이먼트의 멱등키 지원

멱등성 보장하기 멱등키

멱등키를 API 요청에 포함시킨다

```
Idempotency-Key: {IDEMPOTENCY_KEY}
```

```
curl --request POST \  
  --url https://api.tosspayments.com/v1/payments/5zJ4xY7m0kODnyRp/cancel \  
  --header 'Authorization: Basic dGVzdF9za196WExrS0V5cE5BcldtbzUwblgzG1lYXhZRzVS0g==' \  
  --header 'Content-Type: application/json' \  
  --header 'Idempotency-Key: SAAABPQbcqjEXiDL' \  
  --data '{"cancelReason":"고객 변심"}'
```

토스페이먼트의 멱등키 지원

idempotency-key: {IDEMPOTENCY_KEY}

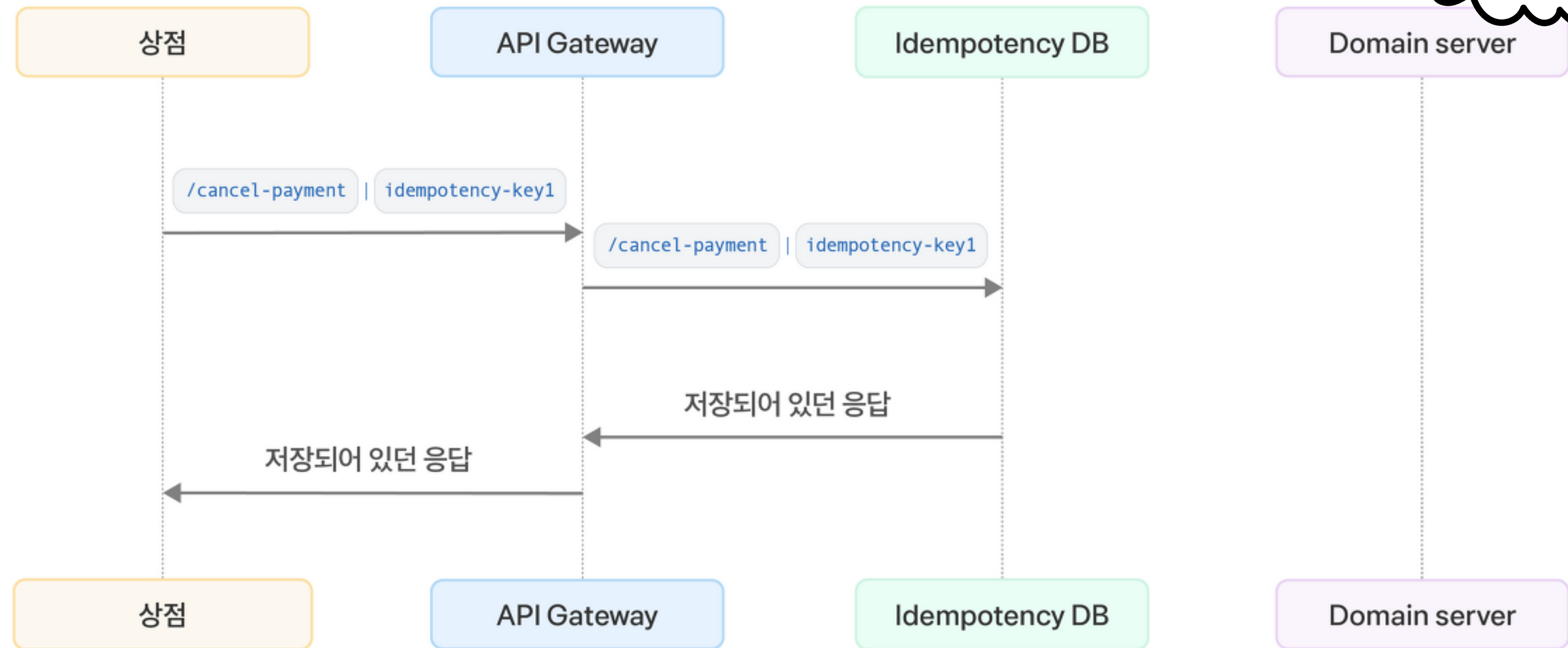
*UUID란?

범용 고유 식별자(Universally Unique Identifier, UUID)로
정규적으로 123e4567-e89b-12d3-a456-426614174000
(하이픈으로 구분된 5개의 16진수 문자열) 형식의
36자 문자열로 표시되는 128비트 값

UUID 사용을 권장

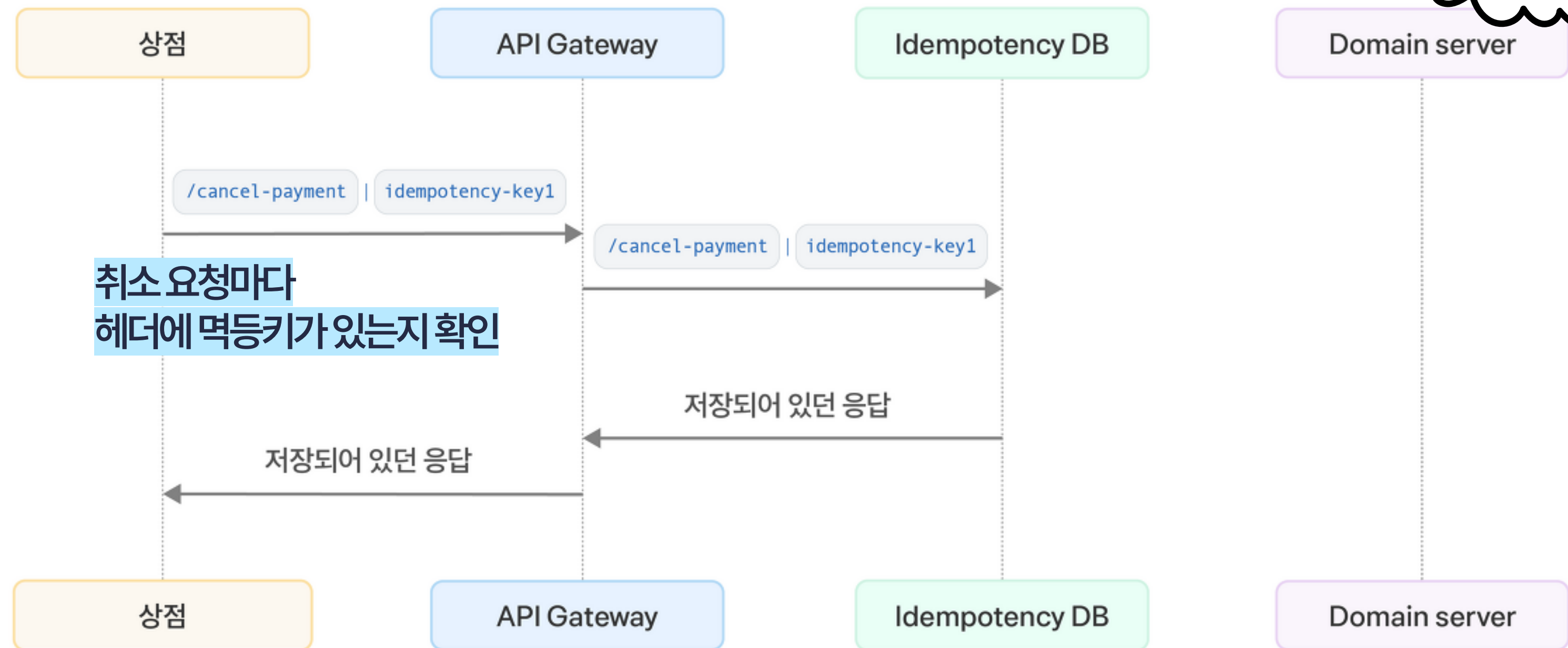
멱등성 보장하기 멱등키

결제 취소 플로우



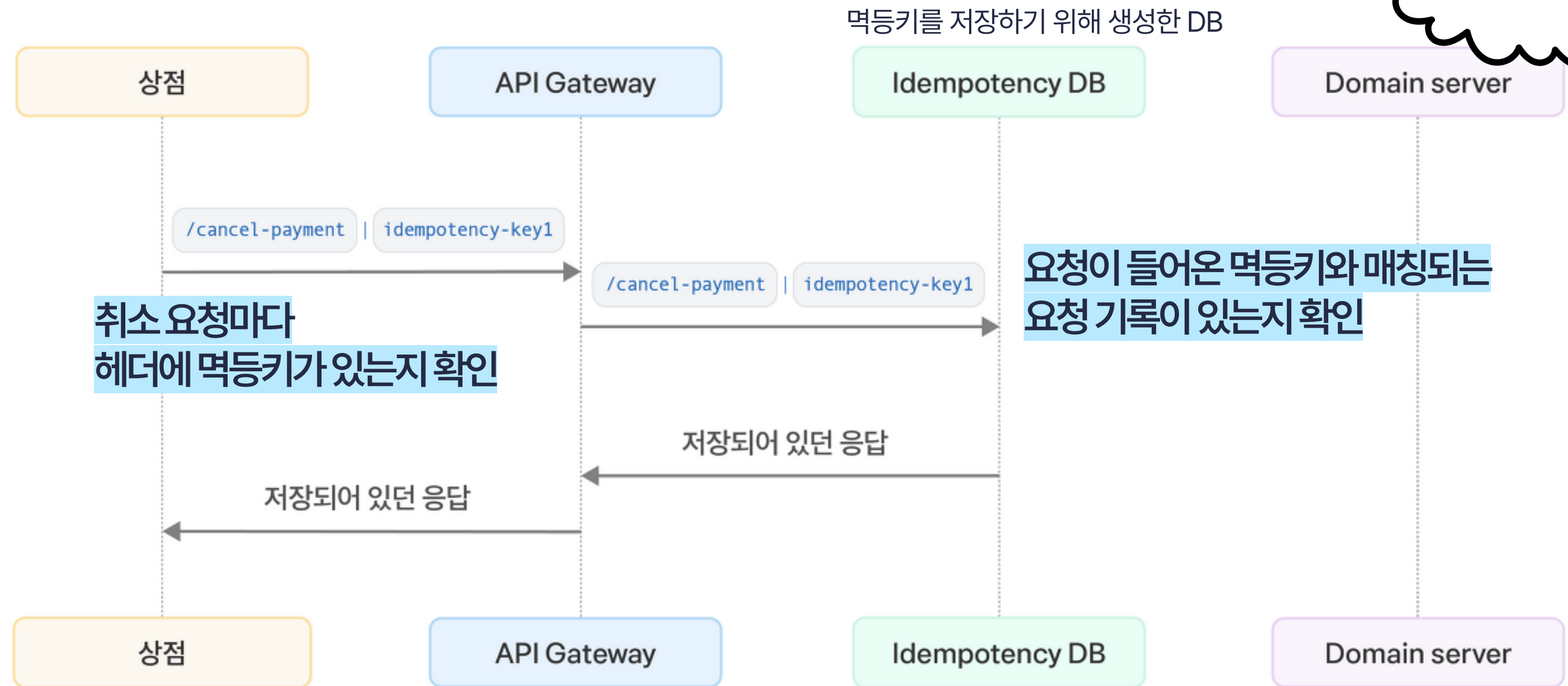
멱등성 보장하기 멱등키

결제 취소 플로우



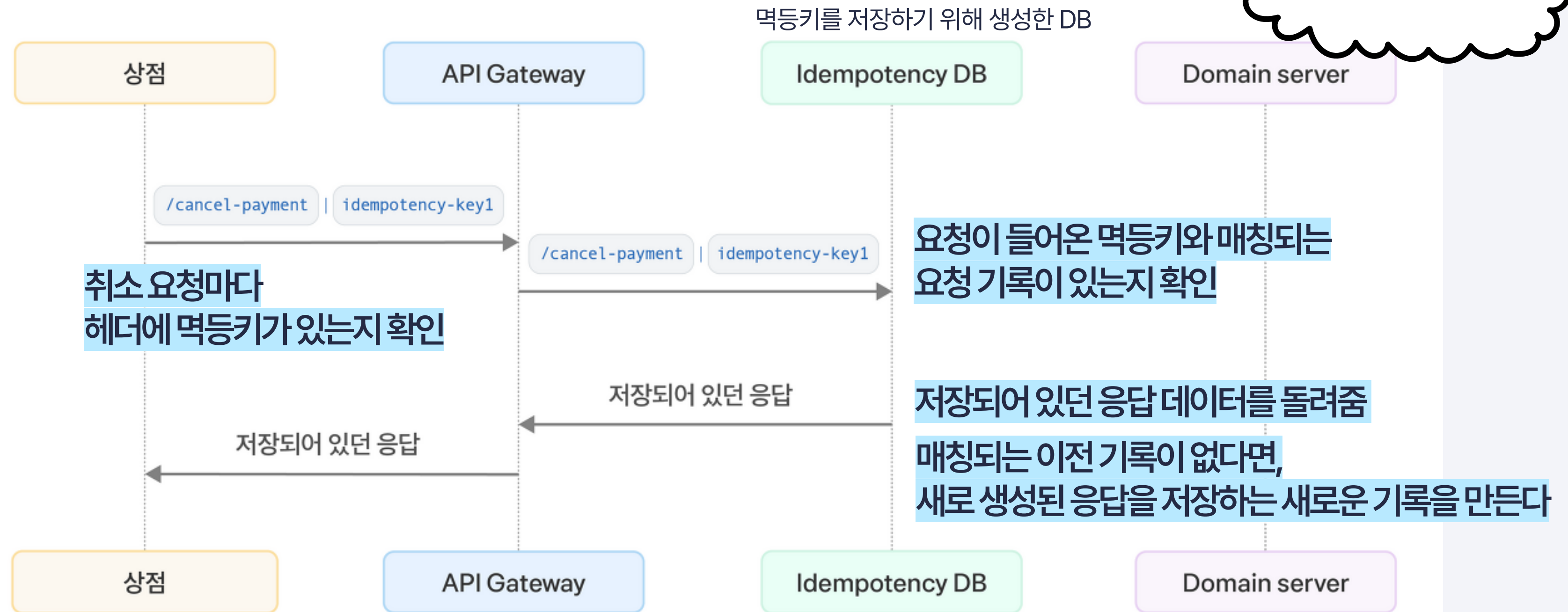
멱등성 보장하기 멱등키

결제 취소 플로우



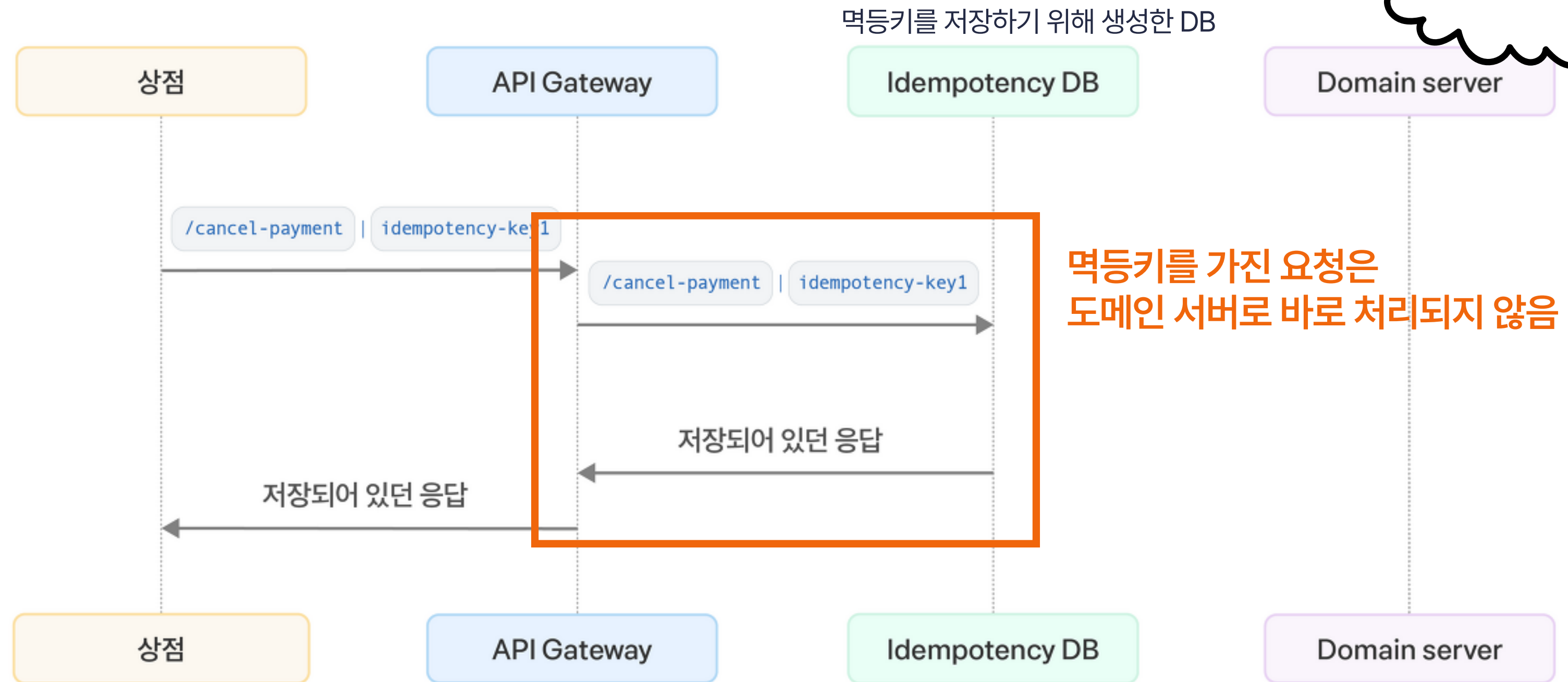
멱등성 보장하기 멱등키

결제 취소 플로우



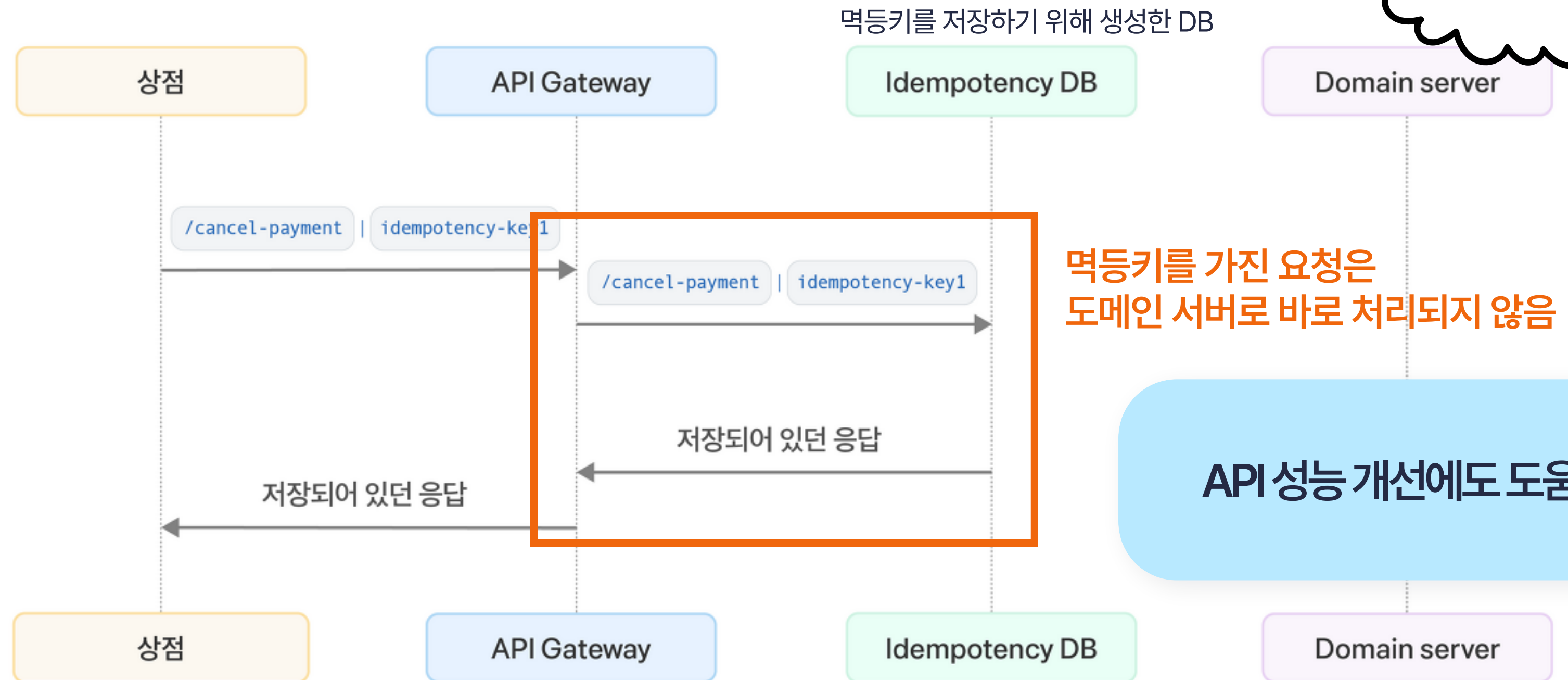
멱등성 보장하기 멱등키

결제 취소 플로우



멱등성 보장하기 멱등키

결제 취소 플로우



적등성을 보장하며
좋은 API를 만들어보아요

감사합니다



참고 : 모든 개발자를 위한 HTTP 웹 기본 지식 (김영한),
<https://velog.io/@tosspayments/먹등성이-뭔가요>