

Mutation 패턴

타입스크립트 해적단

Mutation 이란?

- 서버에 데이터를 변경 요청하는 작업
- 사용자가 게시판에서 게시글을 작성하고, '작성 완료' 버튼을 눌렀을 때 이뤄지는 작업을 말한다.

Mutation 흐름

게시글 목록

test10

test9

test8

test7

게시글 작성

제목

내용

작성 완료

Mutation 흐름

게시글 목록

test10

test9

test8

test7

게시글 작성

제목

내용

작성 완료

1. 사용자가 제목, 게시글 내용 등을 입력함 (필드 입력)
2. 사용자가 '작성 완료' 버튼을 클릭함 (submit)
3. 입력된 게시글이 즉각적으로 등록된 것으로 보이도록 업데이트함 (optimistic update)
4. 입력된 필드에 문제가 있으면 에러를 표시함 (validation error)
5. 서버에서 에러가 발생하면 에러를 표시함 (error)
6. 4또는 5이 만족하면 3에서 수행한 업데이트를 롤백함 (rollback)
7. 입력된 정보가 서버에 저장됨 (mutation)

댓글 등록, 프로필 수정 등의
경우에도 내용이 비슷함

Mutation 흐름

게시글 목록

test10

test9

test8

test7

게시글 작성

제목

내용

작성 완료

1. 사용자가 제목, 게시글 내용 등을 입력함 (필드 입력)
2. 사용자가 '작성 완료' 버튼을 클릭함 (submit)
3. 입력된 게시글이 즉각적으로 등록된 것으로 보이도록 업데이트함 (optimistic update)
4. 입력된 필드에 문제가 있으면 에러를 표시함 (validation error)
5. 서버에서 에러가 발생하면 에러를 표시함 (error)
6. 4또는 5이 만족하면 3에서 수행한 업데이트를 롤백함 (rollback)
7. 입력된 정보가 서버에 저장됨 (mutation)

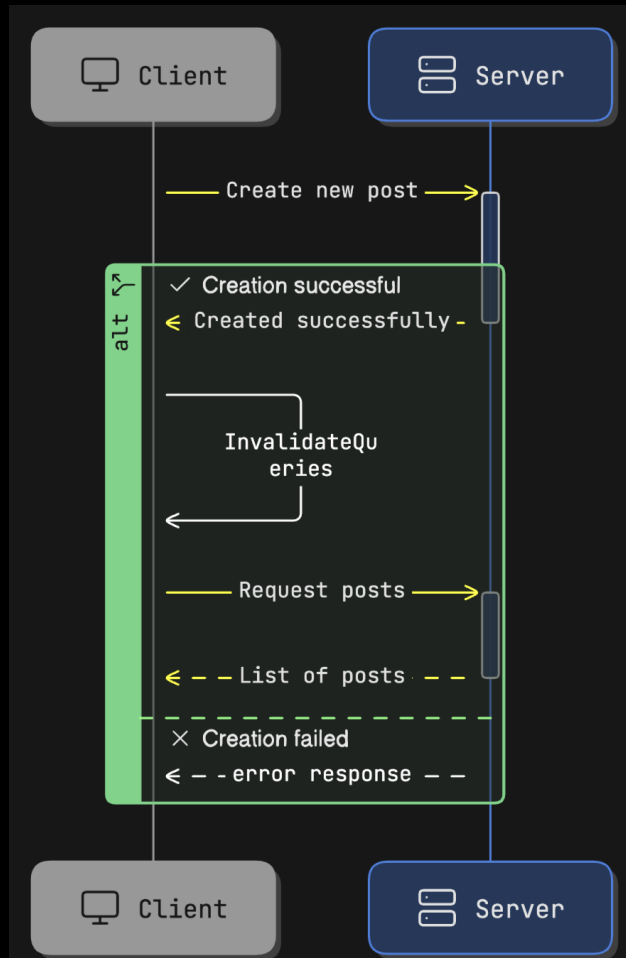
댓글 등록, 프로필 수정 등의
경우에도 내용이 비슷함

캐시 업데이트

- 우선 기본적인 예제부터 같이 봅시다

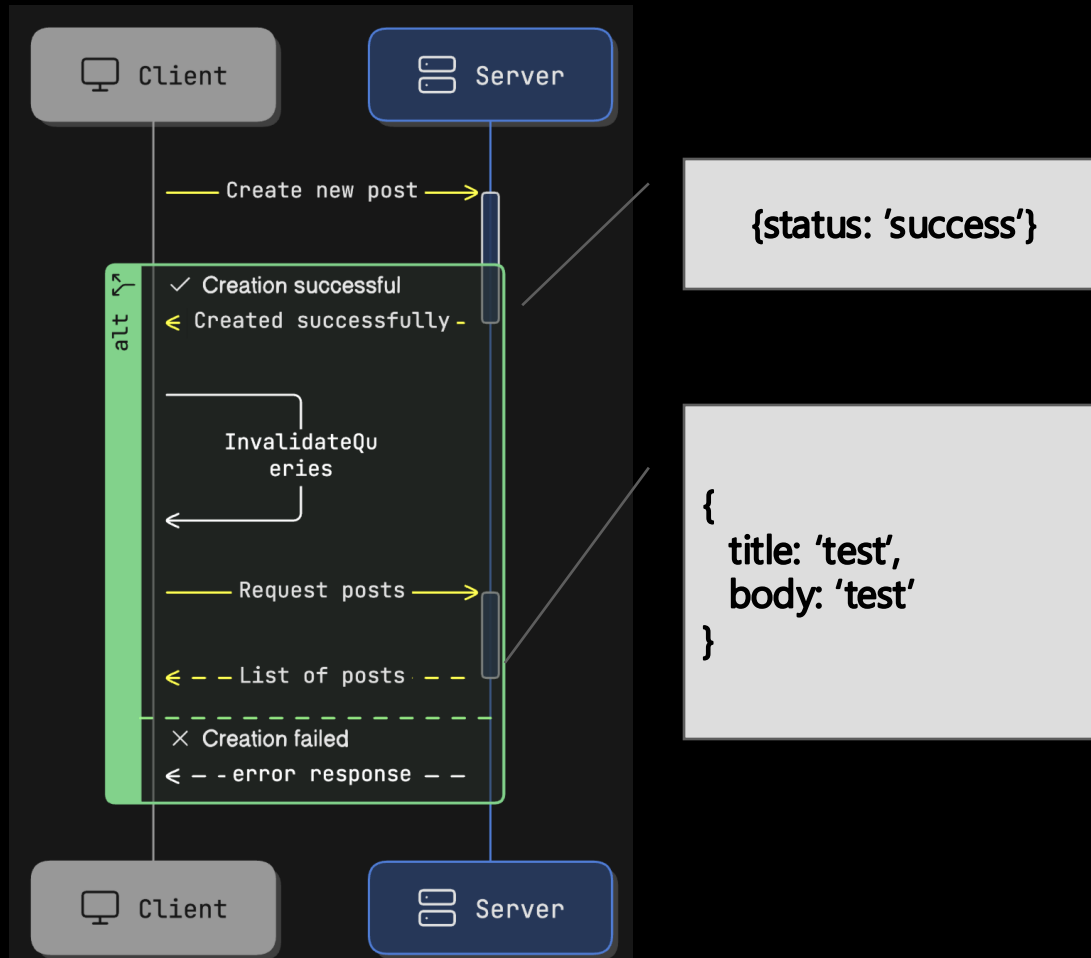
캐시 업데이트

- invalidateQueries의 문제점

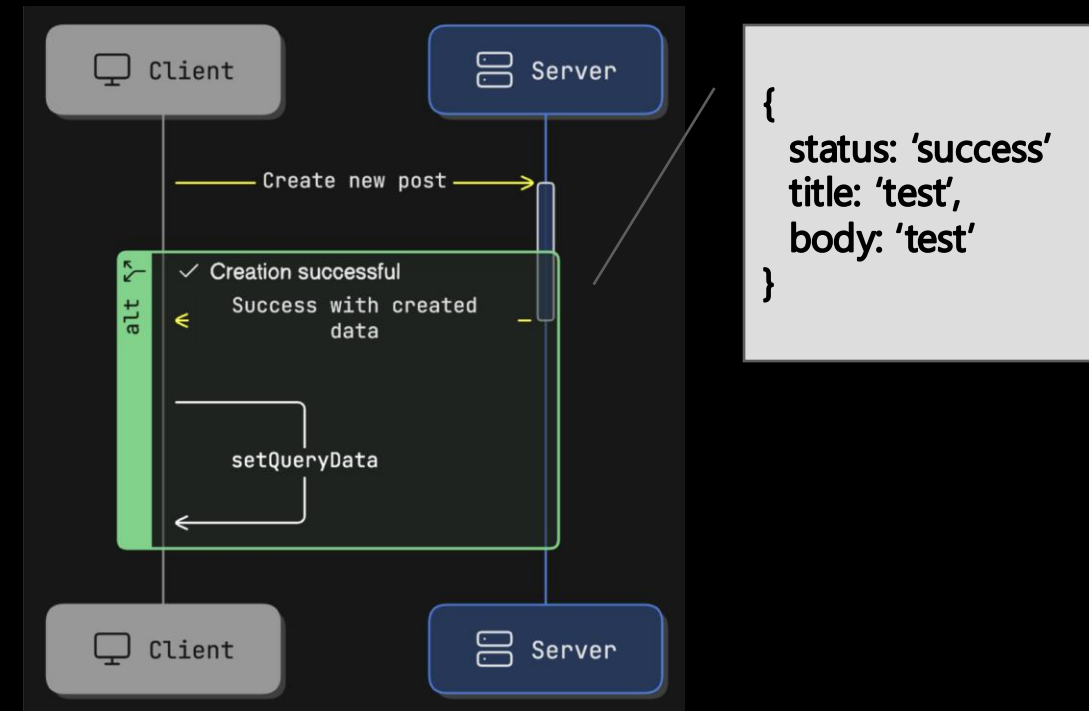
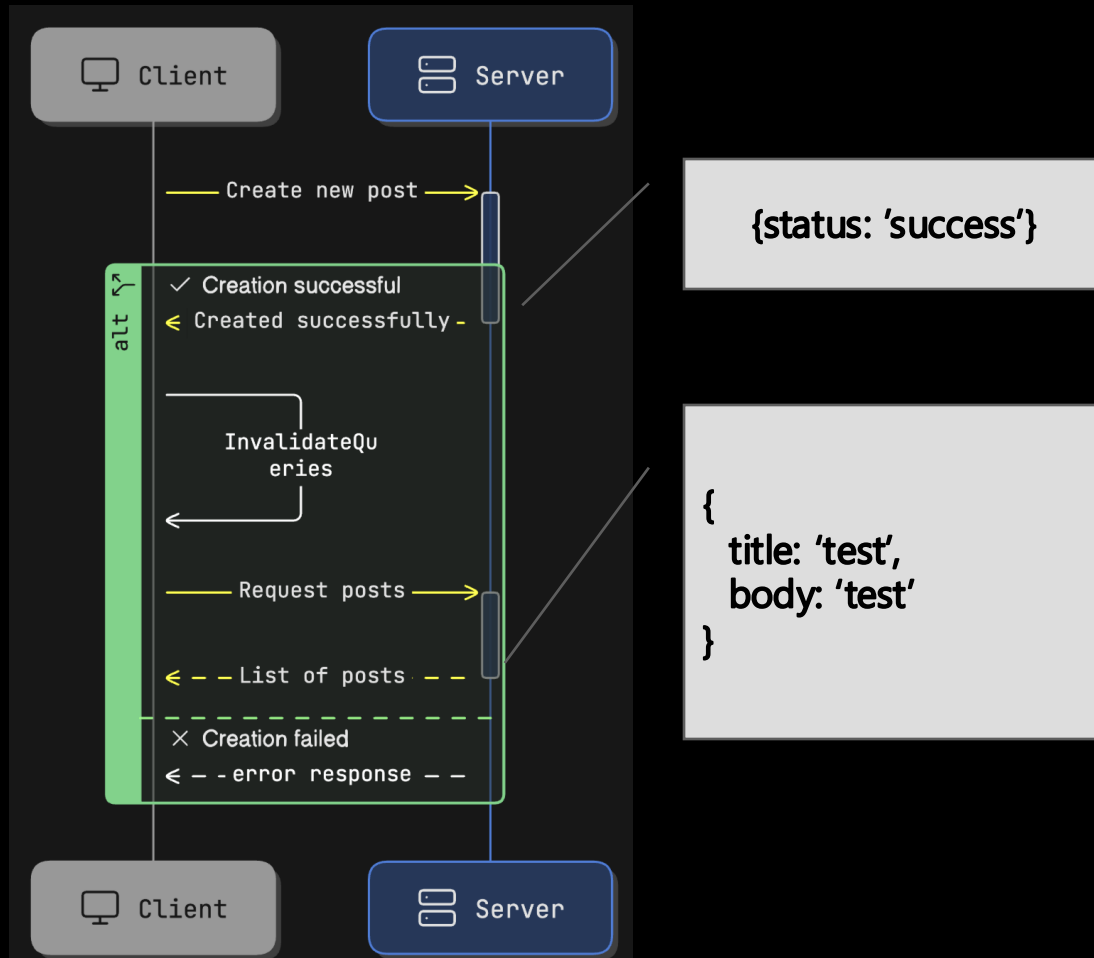


캐시 업데이트

- invalidateQueries의 문제점

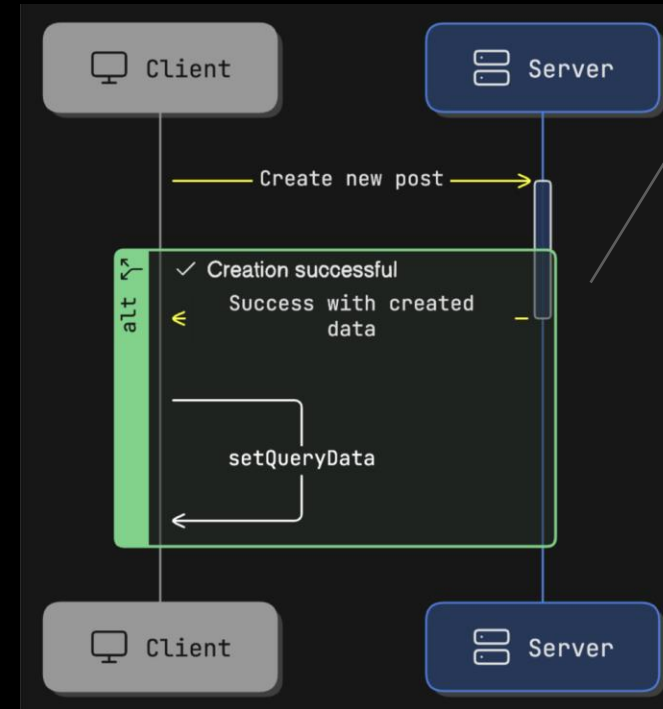
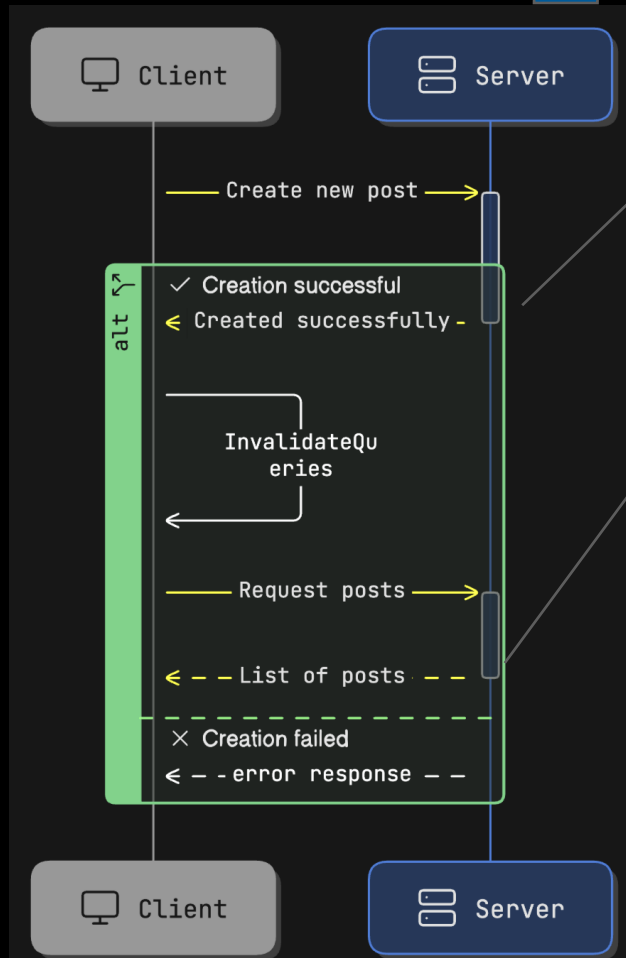


캐시 업데이트

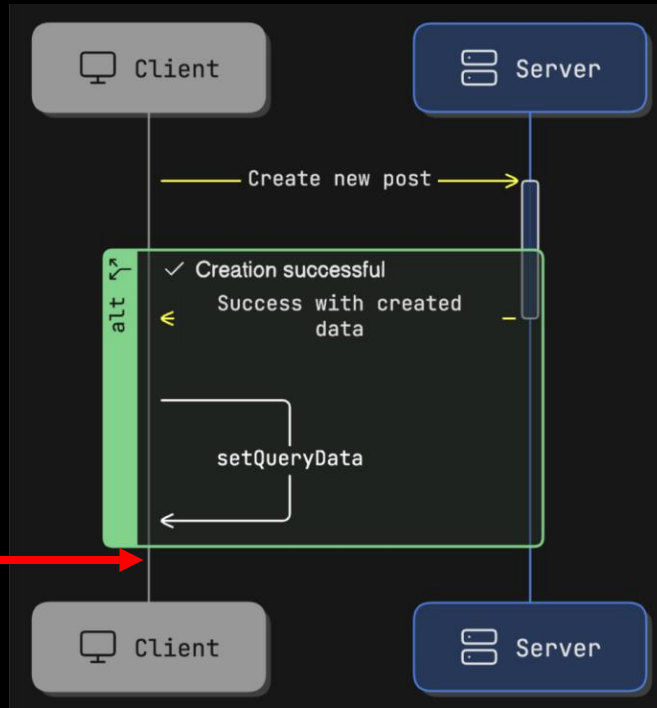


캐시 업데이트

한번의 네트워크 왕복

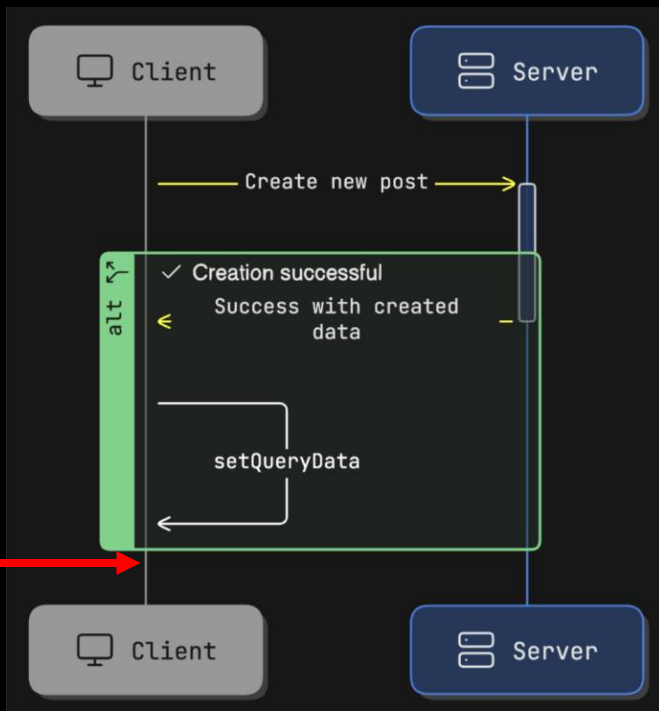


낙관적 업데이트

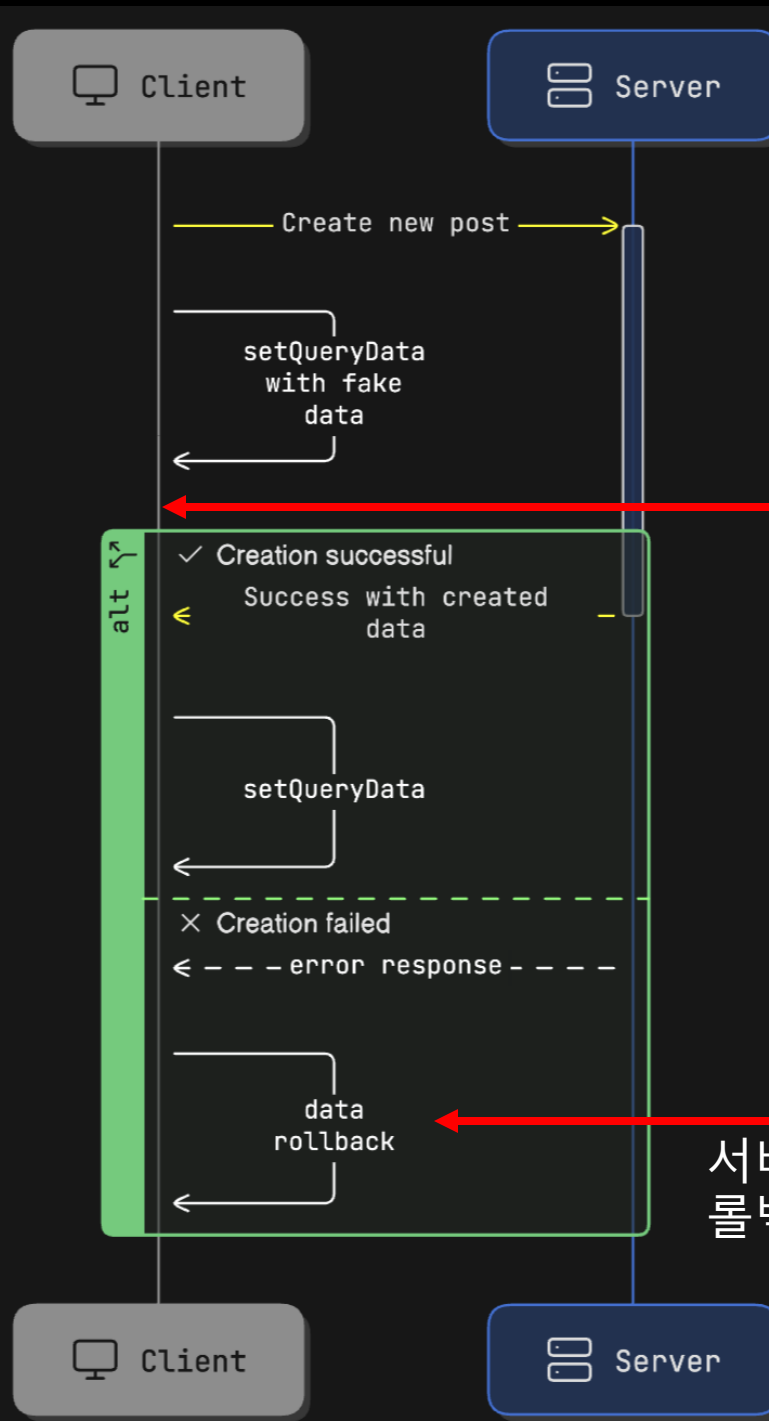


이 시점부터
UI에 반영됨

낙관적 업데이트



이 시점부터
UI에 반영됨



UI에 먼저
반영됨

서버 에러 발생시 데이터
롤백이 필요함

useSmartMutation

게시글 목록

test10

test9

test8

te

게시글 작성

test11

test11

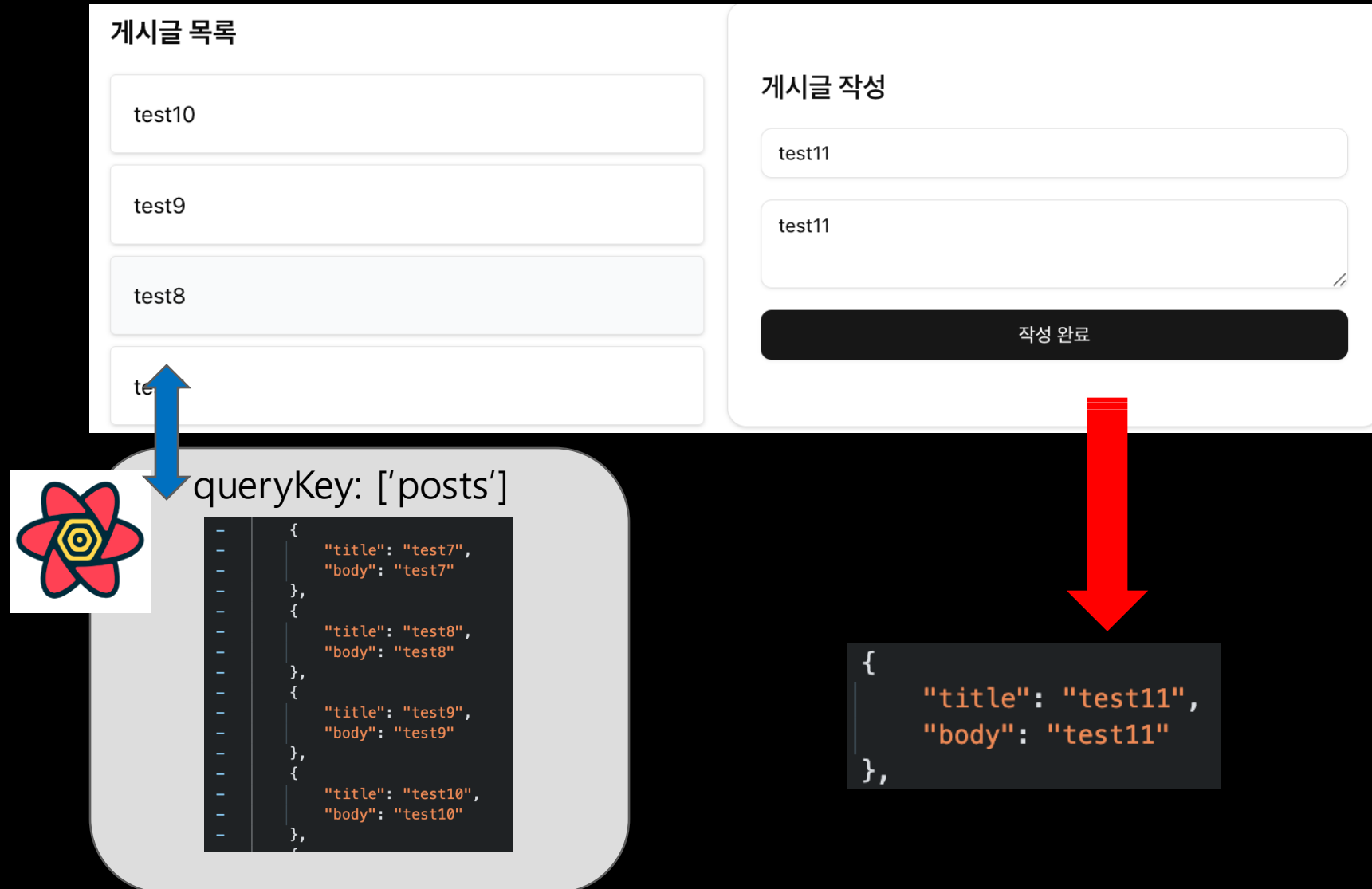
작성 완료



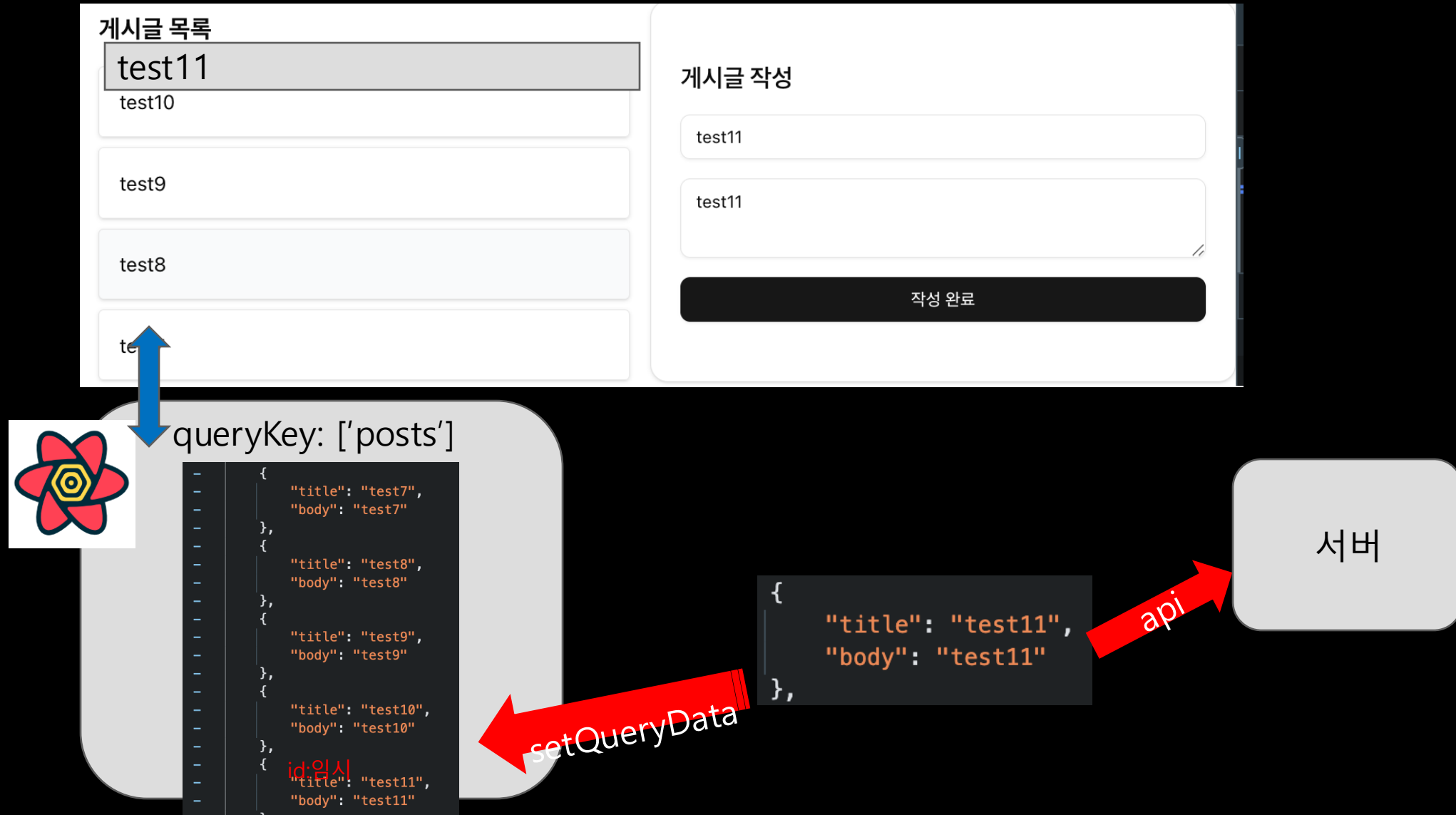
queryKey: ['posts']

```
- {  
-   "title": "test7",  
-   "body": "test7"  
- },  
- {  
-   "title": "test8",  
-   "body": "test8"  
- },  
- {  
-   "title": "test9",  
-   "body": "test9"  
- },  
- {  
-   "title": "test10",  
-   "body": "test10"  
- },  
- }
```

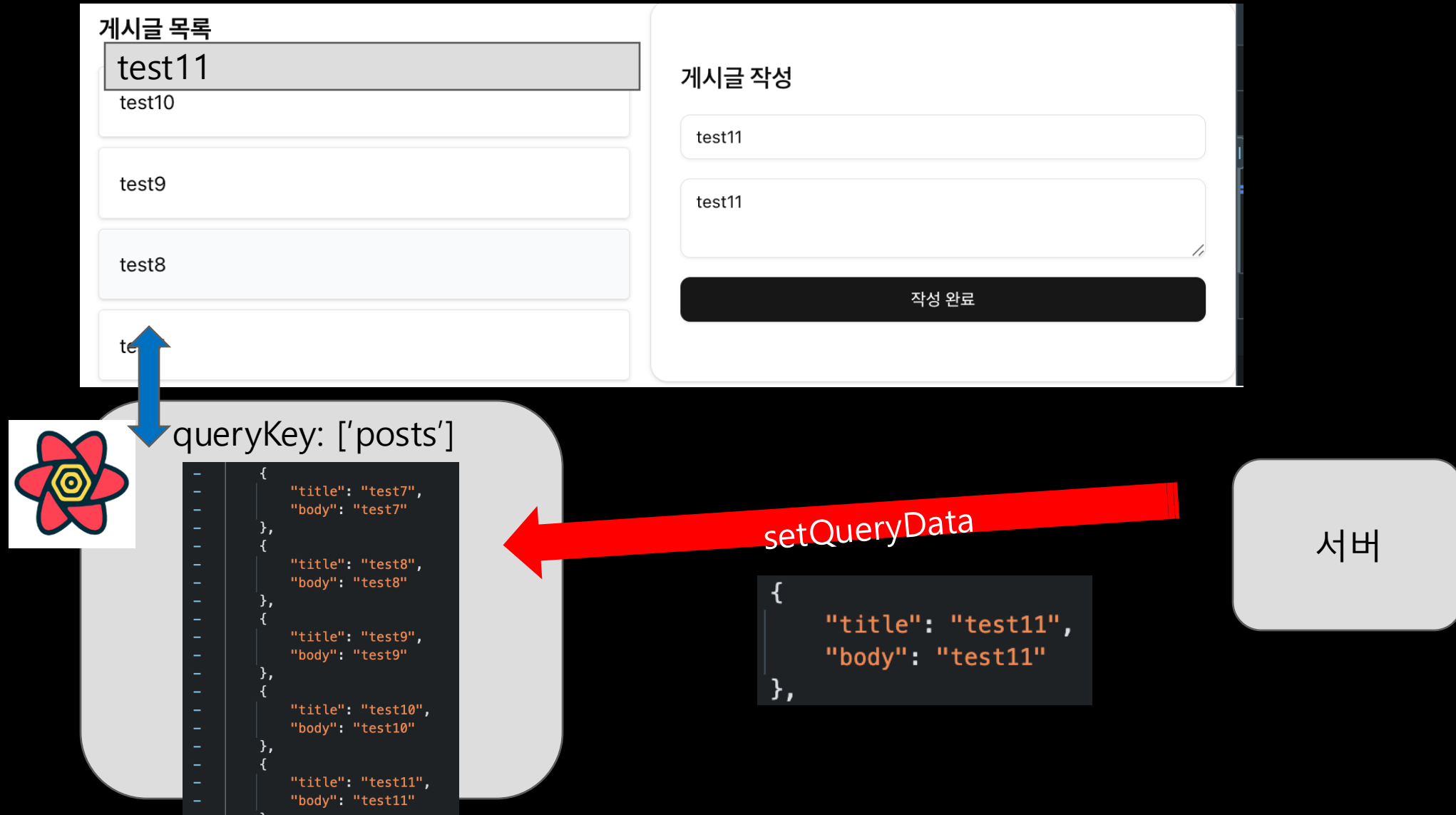
useSmartMutation



useSmartMutation



useSmartMutation



useSmartMutation

1. onMutate에서 임시 데이터를 추가하여 optimistic update
2. onSuccess에서 임시 데이터를 response body 로 교체
3. onError에서 임시 데이터를 삭제