테이블에는 어떤 row가 추가되거나 갱신되었을 때 그 추가 혹은 갱신 시각을 저장해야할 때가 있습니다. 예를 들어,

- 게시글 업로드 시각,

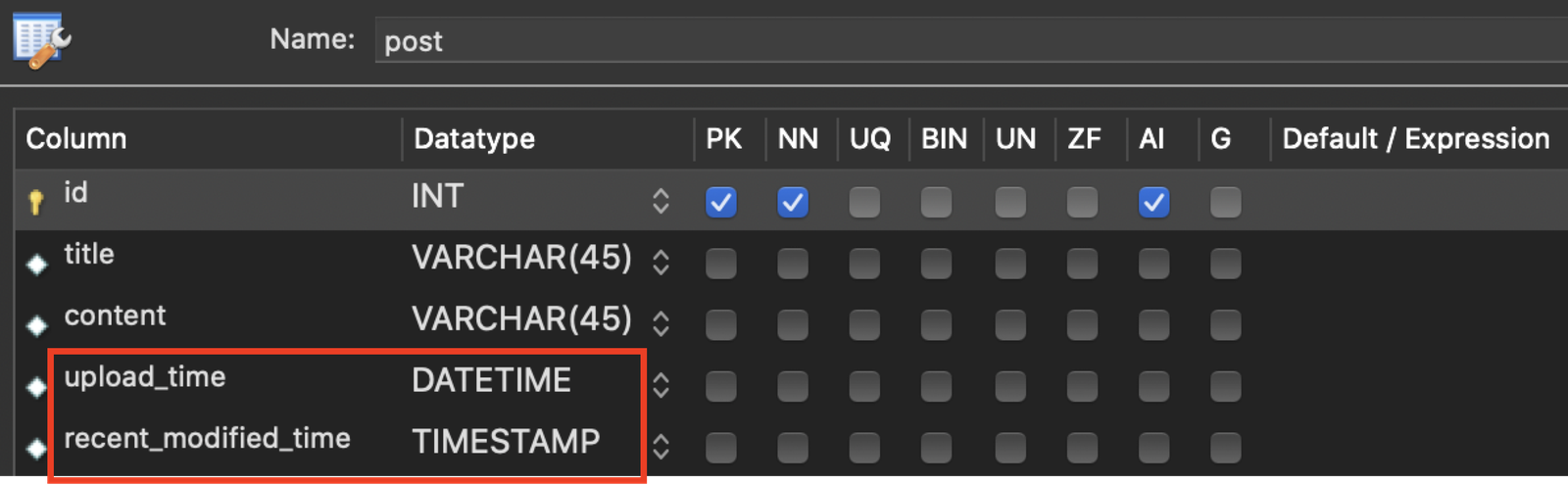
- 댓글이 달린 시각,

- 댓글을 수정한 시각

등의 정보가 그 예인데요. 이런 정보를 저장하는 컬럼에 현재 시간 값을 넣어줄 때는 크게 두 가지 방법이 있습니다. 하나씩 살펴볼게요.

**1. NOW() 함수 사용하기**

SQL 문 안에서 **NOW()**라는 함수를 사용하는 방법입니다. 예를 들어, 어떤 SNS에서 사용자가 업로드한 게시물 정보를 저장한 **post 테이블**이 아래와 같이 있다고 해봅시다.



각 컬럼에 대해 간단히 설명하면,

**id** : PRIMARY KEY 역할을 하는 컬럼

**title** : 게시물의 제목

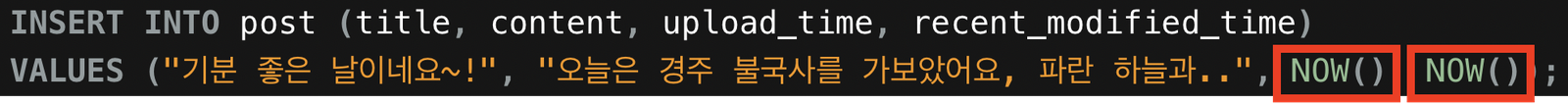
**content** : 게시물의 내용

**upload\_time** : 게시물 최초 업로드 시각

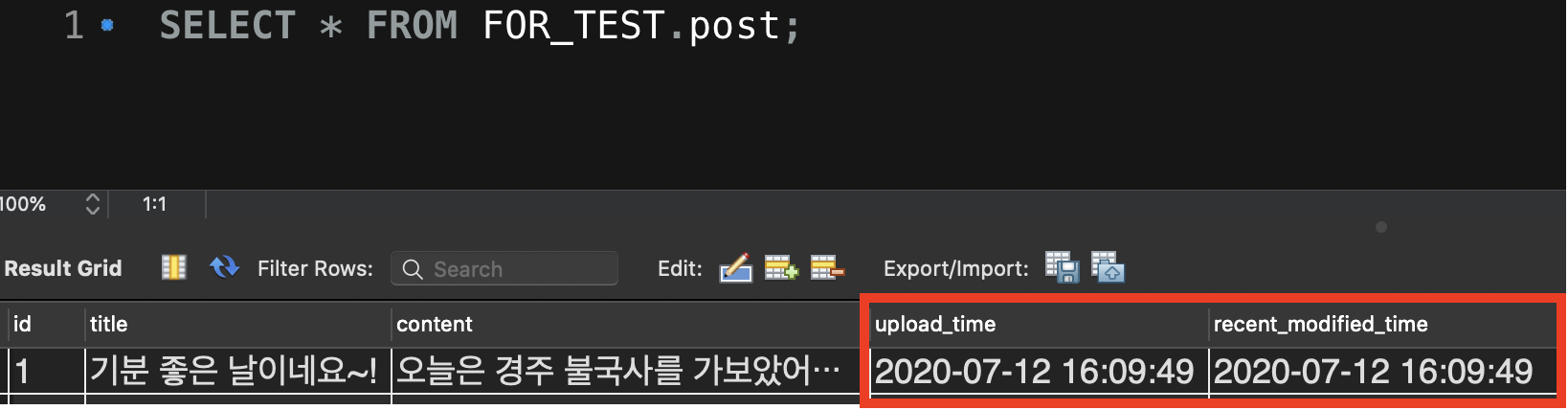
**recent\_modified\_time** : 게시물 최근 갱신 시각

인데요.

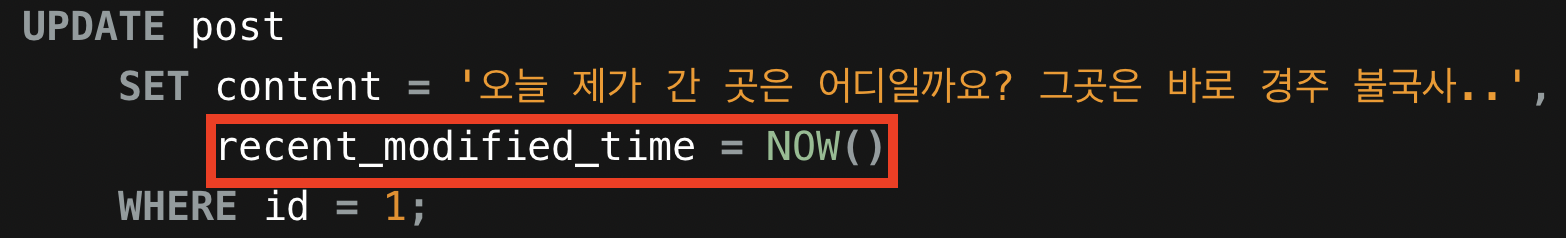
자, 이 테이블에 게시물 하나를 추가한다고 해봅시다. 이렇게 쓰면 되겠죠?



지금 upload\_time 컬럼, recent\_modified\_time 컬럼에 **NOW()라는 함수의 리턴값**을 넣었습니다. 테이블을 조회해보면,

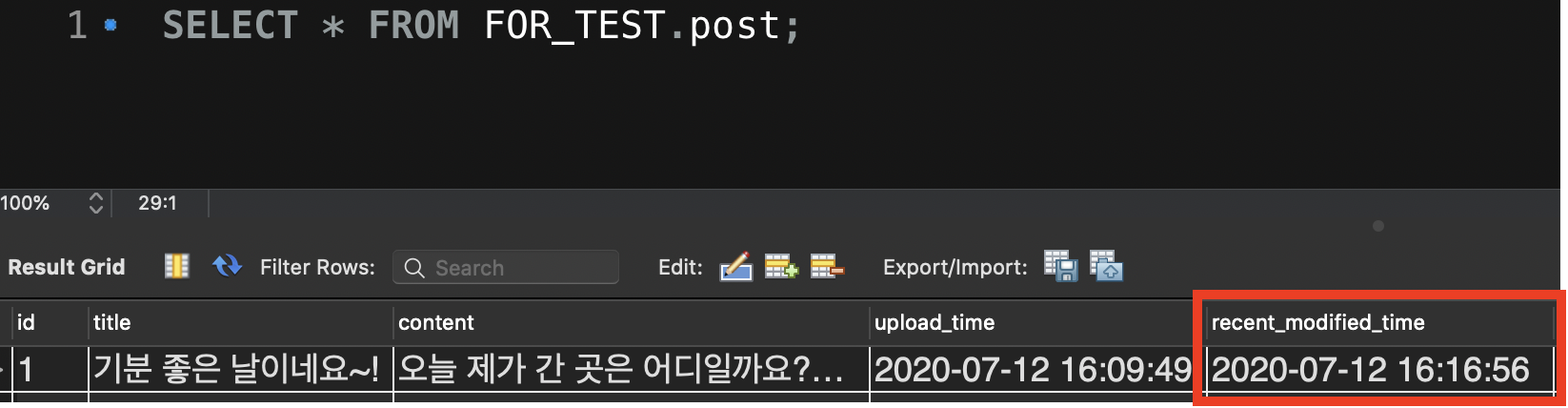


두 컬럼 모두 현재 시각이 잘 들어간 것을 볼 수 있습니다.



만약 이 포스트의 내용을 다시 갱신한다고 해봅시다. 그럼 UDDATE 문을 사용하면 되겠죠?

content 컬럼을 갱신할 때 동시에, recent\_modified\_time 컬럼의 값도 NOW 함수의 리턴값으로 갱신해주면 됩니다. 실행하면



recent\_modified\_time 컬럼의 값이 잘 갱신됩니다.

지금 upload\_time 컬럼의 값에서 약 7분 이후의 값인데요. 최초 업로드 후 약 7분 후에 게시물이 다시 수정된 겁니다. 이렇게 NOW라는 함수를 쓰면 현재 시간을 편하게 구할 수 있습니다. 그런데 이 방법 말고 컬럼에 현재 시간 값을 넣는 다른 방법도 있습니다.

**2. 컬럼에 DEFAULT CURRENT\_TIMESTAMP / ON UPDATE CURRENT\_TIMESTAMP 속성 설정하기**

**DATETIME 타입** 또는 **TIMESTAMP 타입**의 컬럼에는

**DEFAULT CURRENT\_TIMESTAMP** 라는 속성과,

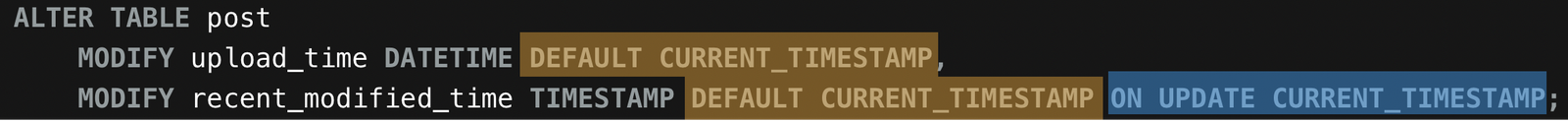
**ON UPDATE CURRENT\_TIMESTAMP** 라는 속성을 줄 수 있습니다.

DEFAULT CURRENT\_TIMESTAMP 속성은 테이블에 새 row를 추가할 때 따로 그 컬럼에 값을 주지 않아도 현재 시간이 설정되도록 하는 속성입니다.

그리고 ON UPDATE CURRENT\_TIMESTAMP 속성은 기존 row에서 단 하나의 컬럼이라도 갱신되면 갱신될 때의 시간이 설정되도록 하는 속성인데요.

정확히 어떤 속성들인지 직접 보여드릴게요.

일단 post 테이블의 upload\_time 컬럼과 recent\_modified\_time 컬럼에 위에서 말한 두 속성을 추가해보겠습니다.



이 SQL 문을 보면

**upload time 컬럼에는 DEFAULT CURRENT\_TIMESTAMP 속성만 줬고,**

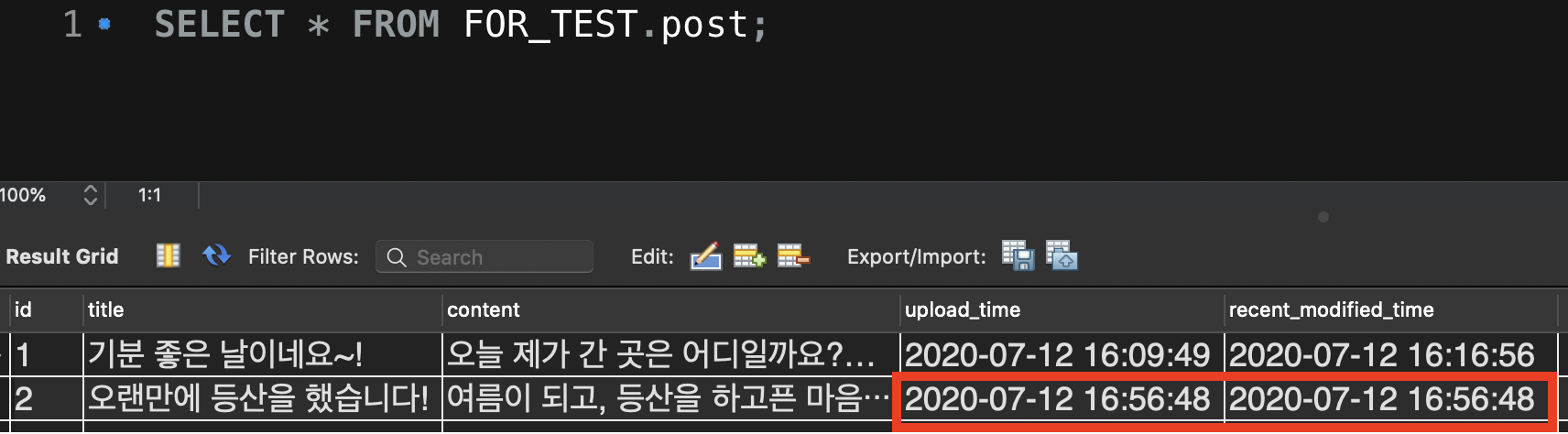
**recent\_modified\_tiem 컬럼에는 DEFAULT CURRENT\_TIMESTAMP 속성과 ON UPDATE CURRENT\_TIMESTAMP 속성을 둘다 줬습니다.**

 이제 테이블에 새로운 row를 추가해볼까요? 아래와 같이 새 row를 추가하겠습니다.



**그런데 지금 title, content 컬럼에만 값을 주고, upload\_time, recent\_modified\_time 컬럼에는 값을 주지 않았습니다. 이 사실이 아주 중요한데요.**

다시 post 테이블을 보면

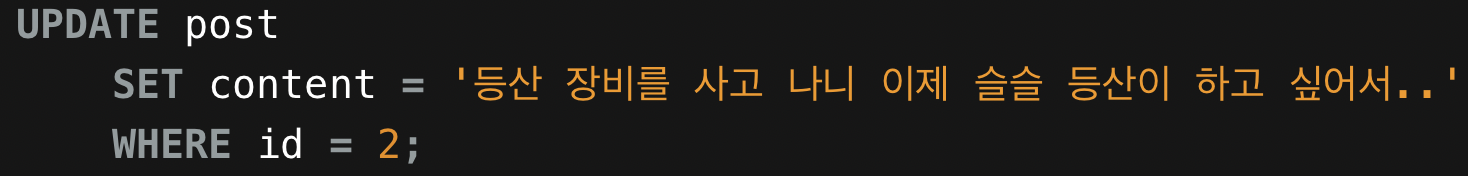


새로운 row가 잘 추가된 것을 볼 수 있습니다.

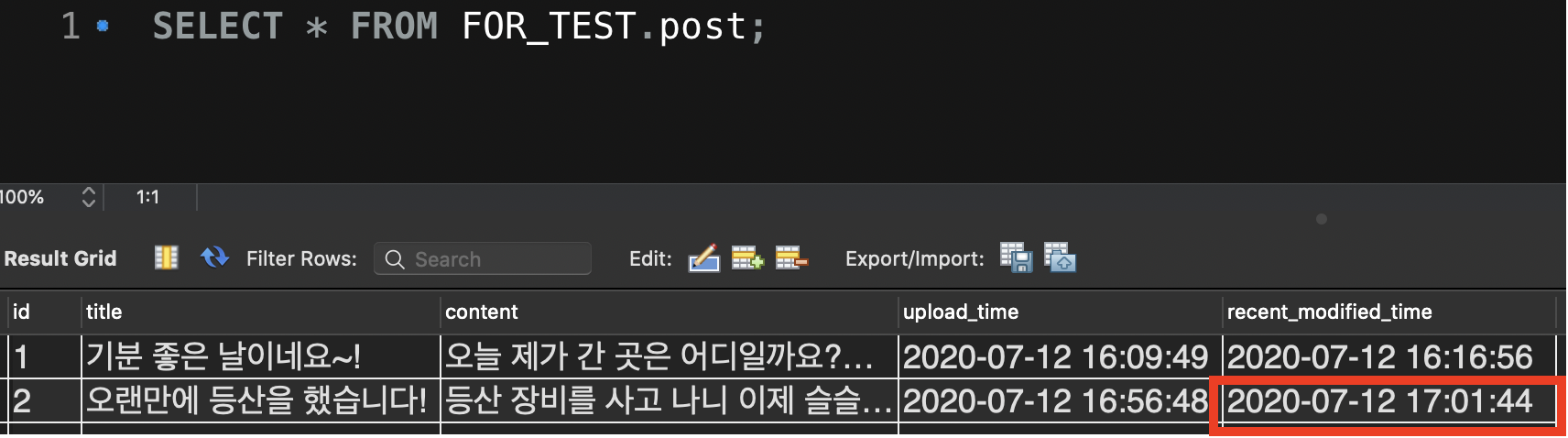
**그리고 upload\_time, recent\_modified\_time 컬럼에는 별도로 값을 주지 않았는데도 현재 시간이 값으로 잘 들어갔네요.**

**두 컬럼에 DEFAULT CURRENT\_TIMESTAMP 속성을 줬기 때문입니다.**

그리고 이 row의 content 컬럼값을 갱신할게요.



그런데 지금 최근 갱신 시각을 나타내는 recent\_modified\_time 컬럼에는 따로 값을 지정하지 않았는데요. 과연 어떤 결과가 나올까요? 이 SQL 문을 실행하고 다시 post 테이블을 보면



2번 row의 content 컬럼의 값이 잘 갱신되었고, recent\_modified\_time 컬럼도 잘 갱신된 것을 알 수 있습니다. **recent\_modified\_time 컬럼에 값을 주지 않았는데도 이렇게 최근 갱신 시각이 잘 들어가게 된 것은, 아까 recent\_modified\_time 컬럼에 ON UPDATE CURRENT\_TIMESTAMP라는 속성을 주었기 때문입니다.**

자, 어떤가요? 두 속성의 의미를 이제 잘 아시겠죠?

이때까지 컬럼의 데이터 타입이 DATETIME 또는 TIMESTAMP 타입일 때 날짜/시간 값을 넣을 수 있는 두 가지 방법을 배웠습니다.

만약 각 row마다 시간값에 관한 처리를 다르게 해줘야하는 경우라면 (예를 들어, 어떤 row는 현재 시간에서 +3시간, 어떤 row는 현재 시간에서 -5시간을 해줘야하는 경우) NOW 함수를 쓰는 게 좋을 겁니다.

하지만 그럴 필요가 없는 상황이고, 굳이 날짜/시간 값을 별도로 신경쓰기가 싫다면 해당 컬럼에 DEFAULT CURRENT\_TIMESTAMP 속성, ON UPDATE CURRENT\_TIMESTAMP 속성을 설정해서 DBMS가 알아서 관리하도록 하는게 좋겠죠?

ㄹ