

4.1 트랜잭션

🕒 생성일	@2021년 5월 30일 오후 10:26
☰ 태그	

MyISAM, MEMORY는 트랜잭션을 지원하지 않는 스토리지 엔진



InnoDB는 트랜잭션을 지원한다!

1. MySQL에서 트랜잭션



트랜잭션은 하나의 논리적인 작업 셋(Set)에 쿼리 수에 상관없이 셋 전체가 100% 적용되거나, 아무것도 적용되는 것을 보장해 주는것

쿼리 1, 2, 3을 실행하는데 → 쿼리 3에서 duplicate key 오류가 나는 경우

MyISAM의 경우

- primary key 중복 오류로 쿼리가 실패해도 1,2는 INSERT 된 상태로 남아있다
- Partial Update : 테이블 정합성을 맞추는데 어려운 문제를 만들어냄

InnoDB의 경우

트랜잭션이 지원됨

- 1, 2, 3 쿼리 전체가 업데이트 되지 않는다
- 실패에 대한 재처리가 쉬워진다

2. 주의 사항

트랜잭션의 범위를 최소화 해야한다!!!

```
1) 처리 시작
=> 데이터베이스 커넥션 생성
=> 트랜잭션 시작
2) 사용자의 로그인 여부 확인
3) 사용자의 글쓰기 내용의 오류 여부 확인
4) 첨부로 업로드된 파일 확인 및 저장
5) 사용자의 입력 내용을 DBMS에 저장
6) 첨부 파일 정보를 DBMS에 저장
7) 저장된 내용 또는 기타 정보를 DBMS에서 조회
8) 게시물 등록에 대한 알림 메일 발송
9) 알림 메일 발송 이력을 DBMS에 저장
<= 트랜잭션 종료(COMMIT)
<= 데이터베이스 커넥션 반납
10) 처리 완료
```

- 실제로 DBMS에 데이터를 저장하는 건 5번이기때문에, 저장할때만 커넥션을 생성해야 한다
- 8번 작업에서는 FTP를 사용하기 때문에, 오류 발생시 DB서버까지 위험해지므로 지양
- 5, 6번 작업은 하나의 트랜잭션으로 묶음. 9번 작업은 성격이 다르므로 트랜잭션 분리
- 7번은 단순작업이라 트랜잭션 사용X

-
- 1) 처리 시작
 - 2) 사용자의 로그인 여부 확인
 - 3) 사용자의 글쓰기 내용의 오류 발생 여부 확인
 - 4) 첨부로 업로드된 파일 확인 및 저장
 - ⇒ 데이터베이스 커넥션 생성(또는 커넥션 풀에서 가져오기)
 - ⇒ 트랜잭션 시작
 - 5) 사용자의 입력 내용을 DBMS에 저장
 - 6) 첨부 파일 정보를 DBMS에 저장
 - ⇐ 트랜잭션 종료 (COMMIT)
 - 7) 저장된 내용 또는 기타 정보를 DBMS에서 조회
 - 8) 게시물 등록에 대한 알림 메일 발송
 - ⇒ 트랜잭션 시작
 - 9) 알림 메일 발송 이력을 DBMS에 저장
 - ⇐ 트랜잭션 종료(COMMIT)
 - ⇐ 데이터베이스 커넥션 종료(또는 커넥션 풀에 반납)
 - 10) 처리 완료
-