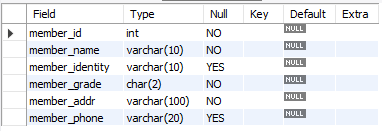
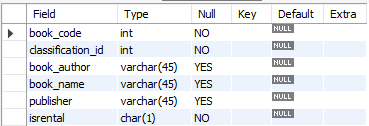
SQL활용

1) 각 테이블을 생성합니다.

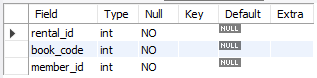
- tbl\_member 생성



- tbl\_book 생성

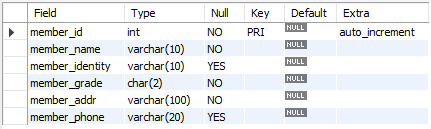


- tbl\_rental 생성



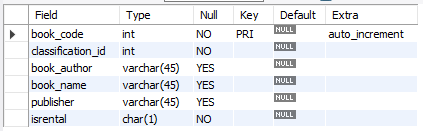
2) tbl\_member의 member\_id에 Primary key 제약 조건 설정하세요(alter 사용)

- PK는 auto\_increment 설정을 합니다.



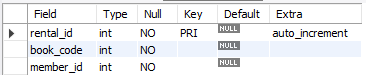
3) tbl\_book의 book\_code에 Primary key 제약 조건 설정하세요(alter 사용)

- PK는 auto\_increment 설정을 합니다.



4) tbl\_rental의 rental\_id에 Primary key 제약 조건 설정하세요(alter 사용)

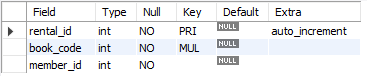
- PK는 auto\_increment 설정을 합니다.

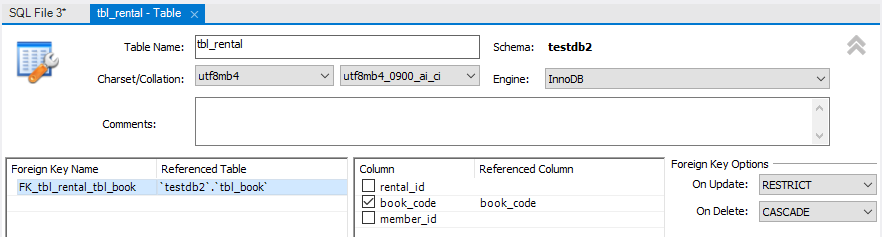


5) tbl\_rental의 book\_code에 tbl\_book의 book\_code(PK)를 참조하는 FK 설정하세요(alter사용)

- Update 시 Restrict 옵션 적용합니다.

- Delete 시 cascade 옵션 적용합니다.

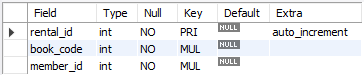


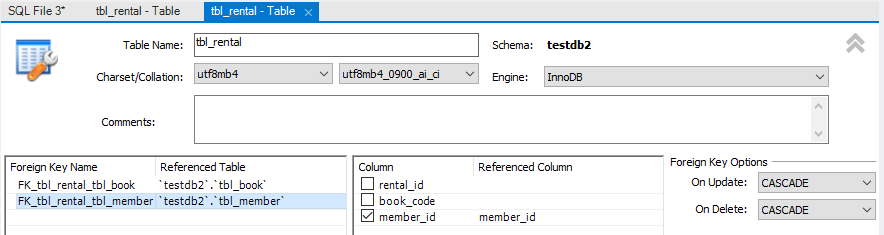


6) tbl\_rental의 member\_id에 tbl\_member의 member\_id(PK)를 참조하는 FK 설정하세요(alter사용)

- Update 시 cascade 옵션 적용합니다.

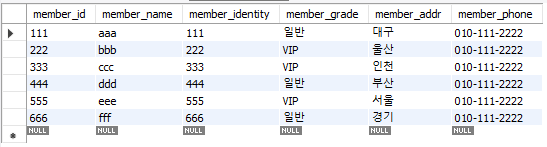
- Delete 시 cascade 옵션 적용합니다.



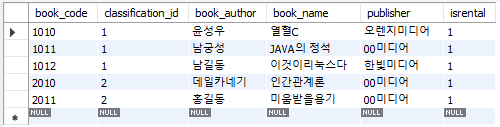


7) 각 테이블에 다음 값을 넣어보세요

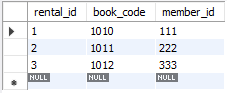
-- tbl\_member



-- tbl\_book



-- tbl\_rental



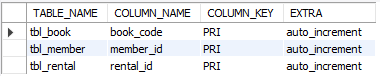
8) 각 테이블의 제약 조건을 확인합니다.

- tbl\_member의 PK 확인이 되어야 합니다.

- tbl\_book의 PK 확인이 되어야 합니다.

- tbl\_rental의 PK 확인이 되어야 합니다.

- Information\_Schema를 이용합니다.

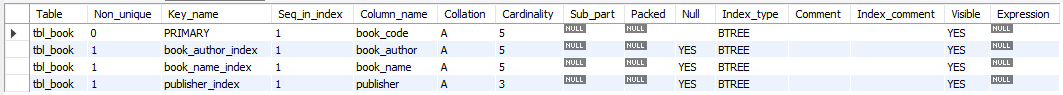


9) Index 설정합니다.

- tbl\_member의 member\_addr열에 non\_unique Index를 설정합니다.



- tbl\_book의 book\_author, book\_name, publisher 각각에 non\_unique Index 설정합니다.



10) 다음과 같이 내용이 나오도록 하는 View 테이블을 만들고 사용해 봅니다.

(조인+뷰사용)

- 뷰테이블명: View\_ShowRental

- JOIN 종류: Inner Join 사용할 것

- 책을 대여한 회원에 대한 정보를 표시

- rental\_id, member\_name, book\_name만 표시

\* ERD

