

과제 #1

- MovieExec 테이블의 튜플을 변경하는 경우 다음과 같이 동작하는 Exec_Update라는 Trigger를 작성 하고 **1.sql**로 저장한다.
 - ① 이 MovieExec 튜플이 사장 또는 영화 제작자인 경우 name이 변경되지 않게 한다.
 - ② netWorth가 null로 변경되는 경우 MovieExec 튜플들의 가장 큰 netWorth로 한다.
 - ③ MovieExec 튜플이 사장도 아니고 영화 제작자도 아닌 경우 이 튜플의 netWorth가 MovieExec 테이블의 평균 netWorth 보다 큰 경우 이 MovieExec 튜플을 random하게 선택한 영화사의 사장이 되게 한다.
 - ④ 이 MovieExec 튜플이 **영화에 출연했던 배우(StarsIn 테이블에 있음)**인 경우 address 애트리뷰트를 '[address]에 배우가 삽니다!'가 되게 한다. address는 MovieExec 튜플의 주소값임.

과제 #2 (1/2)

- Movie 테이블에 새 튜플을 삽입하는 **Movie_Insert**라는 Trigger를 **2_1.sql**에 다음과 같이 작성한다.
 - ▣ length가 NULL인 경우 **평균 상영시간**
 - ▣ inColor가 NULL인 경우 **'true'**
 - ▣ studioname이 NULL인 경우 영화를 가장 적게 제작한 영화사 (이 영화사들이 여럿이면 random하게 선택. 또한, **영화를 제작한 적이 없는 영화사는 제외**)
 - ▣ producerNo가 NULL인 경우 MovieExec 테이블에서 random하게 선택한 튜플의 certno로 설정

과제 #2 (2/2)

- 1120.zip 파일의 StarPlays 뷰에 (title, year, name)의 튜플을 삽입하기 위한 다음 조건을 만족하는 **StarPlays_Trigger**라는 트리거를 **2.sql**에 작성한다.
 - (title,year)에 해당하는 영화에 name의 영화배우가 출연하도록 설정하는 Trigger이다.
 - (title,year)가 Movie 튜플에 없다면 **2번 페이지**의 **Movie Insert** 트리거에 의해서 자동으로 Movie 튜플이 삽입되게 한다. 단, producerNo는 **가장 많은 영화를 제작한 제작자**의 certNo로 설정한다. 단, 이 제작자가 다수면 이들 중 임의로 설정한다.
 - name의 MovieStar 튜플이 없다면 다음과 같은 애트리뷰트를 갖도록 새 MovieStar 튜플을 생성한다.
 - address : 임의의 주소(**한국식 주소 형식을 가지는 random한 주소값임**)
 - gender : 가장 어린 배우의 성별 (여러 명일 경우 이 중에서 random하게 선택)
 - birthdate : 임의의 생년월일 (1980년 이후일 것)

실습과제 #9 제출 방법

- 과제 번호.sql로 각 실습과제를 SQL 스크립트 파일로 저장하고 “학번.zip”으로 묶어서 “실습과제 #9”의 제목으로 oracle@cs.ks.ac.kr로 첨부하여 제출
- 제출기한 : 11월 24일 (일) 24:00