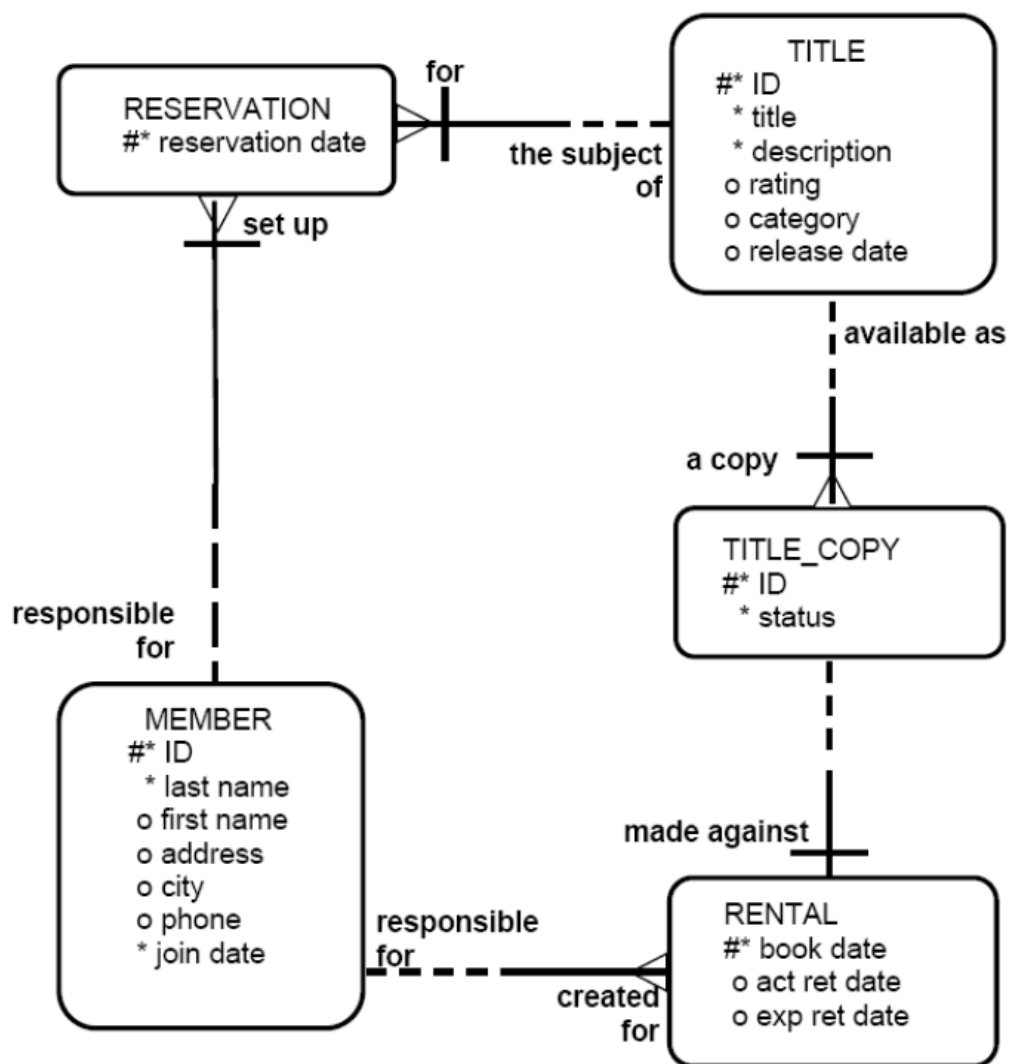


## 1.개요

이 사례 연구에서는 비디오 대여점 응용 프로그램에 대한 프로시저와 함수를 포함하는 **VIDEO\_PKG** 라는 패키지를 생성합니다. 이 응용 프로그램을 사용하여 고객은 비디오 대여점의 회원이 될 수 있습니다. 회원이 되면 영화 대여, 대여한 영화 반납, 영화 예약 등이 가능합니다. 또한 비디오 테이블에 있는 데이터를 영업 시간에만 수정할 수 있도록 하는 트리거를 생성합니다.

SQL\*Plus 를 사용하여 패키지를 생성하고 오라클 제공 패키지 DBMS\_OUTPUT 을 사용하여 메시지를 표시합니다. 비디오 대여점 데이터베이스에는 TITLE, TITLE\_COPY, RENTAL, RESERVATION, MEMBER 등의 테이블이 있습니다.

비디오 대여점 데이터베이스 엔티티 관계 도표



## 2.환경 설정

```
C:\#Users#COM> sqlplus system/oracle@localhost:1521/xepdb1
```

```
SQL> create user video  
      identified by video  
      default tablespace users  
      quota unlimited on users  
      temporary tablespace temp;
```

```
SQL> grant connect, resource to video;
```

```
SQL> connect video/video@localhost:1521/xepdb1
```

```
SQL> alter session set nls_territory = 'america';  
SQL> alter session set nls_language = 'american';  
SQL> alter session set nls_date_format = 'DD-MON-RR';
```

```
SQL> @video1.sql
```

```
SQL> @video2.sql
```

### 3.패키지 생성

이 사례 연구에서는 비디오 대여점 응용 프로그램에 대한 프로시저와 함수를 포함하는 **VIDEO\_PKG** 라는 패키지를 생성합니다.

다음 프로시저와 함수를 사용하여 **VIDEO\_PKG** 라는 패키지를 생성합니다.

**NEW\_MEMBER:** MEMBER 테이블에 새로운 회원을 추가하는 공용(public) 프로시저입니다. 회원 ID 번호에는 MEMBER\_ID\_SEQ 시퀀스를 사용하고, 가입 날짜에는 SYSDATE 를 사용합니다. 새 행에 삽입될 다른 모든 값은 파라미터로 전달합니다.

**NEW\_RENTAL:** 새로운 대여 내용을 기록하는 오버로드 공용(public) 함수입니다. 고객의 성이나 회원 ID 번호 중 하나와 고객이 대여하려고 하는 비디오의 제목 ID 번호를 함수에 전달합니다. 함수는 비디오 반납 기한을 반환해야 합니다. 반납 기한은 비디오를 대여한 날짜로부터 3 일입니다. TITLE\_COPY 테이블에 요청한 영화의 한 사본에 대한 상태가 AVAILABLE 로 표시될 경우 이 TITLE\_COPY 테이블을 갱신한 다음 상태를 RENTED 로 설정합니다. 대여 가능한 비디오가 없을 경우 함수는 NULL 을 반환해야 합니다. 그런 다음 RENTAL 테이블에 오늘 날짜로 된 예약 날짜, 사본 ID 번호, 회원 ID 번호, 제목 ID 번호, 예상 반납 일자 등을 식별하는 새 레코드를 삽입합니다. 성이 같은 고객이 여러 명일 수 있다는 점을 유념합니다. 이 경우 함수가 NULL 을 반환하면서 일치하는 고객 이름과 ID 번호 리스트를 표시합니다.

**RETURN\_MOVIE:** 비디오 상태(Available, Rented, Damaged)를 갱신하고 반납 일자를 설정하는 공용(public) 프로시저입니다. 이 프로시저에 제목 ID, 사본 ID 및 상태를 전달합니다. 해당 제목에 대해 예약된 사항이 있는지 확인한 다음 예약되어 있을 경우 메시지를 표시합니다. RENTAL 테이블을 갱신한 다음 실제 반납 일자를 오늘 날짜로 설정합니다. 프로시저에 전달된 상태 파라미터에 따라 TITLE\_COPY 테이블의 상태를 갱신합니다.

**RESERVE\_MOVIE:** NEW\_RENTAL 프로시저에 요청된 모든 비디오 사본의 상태가 RENTED 일 경우에만 실행되는 전용(private) 프로시저입니다. 이 프로시저에 회원 ID 번호와 제목 ID 번호를 전달합니다. RESERVATION 테이블에 새 레코드를 삽입한 다음 예약 날짜, 회원 ID 번호 및 제목 ID 번호를 기록합니다. 영화가 예약된 상태임과 예상 반납 일자를 나타내는 메시지를 출력합니다.

**EXCEPTION\_HANDLER:** 공용(public) 프로그램의 예외 처리기에서 호출되는 전용(private) 프로시저입니다. 이 프로시저에 SQLCODE 번호와 오류가 발생한 프로그램 이름(텍스트 문자열)을 전달합니다. RAISE\_APPLICATION\_ERROR 를 사용하여 사용자가 정의한 오류를 발생시킵니다. Unique Key 위반(-1) 및 Foreign Key 위반(-2292)으로 시작합니다. 다른 오류에 대해서는 예외 처리기가 일반 오류를 발생시키도록 허용합니다.

```

CREATE OR REPLACE PACKAGE video_pkg IS
  PROCEDURE new_member
    (p_lname      IN member.last_name%TYPE,
     p_fname      IN member.first_name%TYPE  DEFAULT NULL,
     p_address    IN member.address%TYPE     DEFAULT NULL,
     p_city       IN member.city%TYPE        DEFAULT NULL,
     p_phone      IN member.phone%TYPE       DEFAULT NULL);

  FUNCTION new_rental
    (p_memberid   IN rental.member_id%TYPE,
     p_titleid    IN rental.title_id%TYPE)
    RETURN DATE;

  FUNCTION new_rental
    (p_membername IN member.last_name%TYPE,
     p_titleid    IN rental.title_id%TYPE)
    RETURN DATE;

  PROCEDURE return_movie
    (p_titleid    IN rental.title_id%TYPE,
     p_copyid     IN rental.copy_id%TYPE,
     p_sts        IN title_copy.status%TYPE);
END video_pkg;
/

```

```

CREATE OR REPLACE PACKAGE BODY video_pkg IS

```

패키지를 완성하세요!

```

END video_pkg;
/

```

## 4.패키지 활용

다음 명령을 사용하여 회원 두 명을 추가하세요.

```
EXECUTE video_pkg.new_member('Haas', 'James', 'Chestnut Street', 'Boston',  
'617-123-4567')  
EXECUTE video_pkg.new_member('Biri', 'Allan', 'Hiawatha Drive', 'New York',  
'516-123-4567')
```

다음 명령을 사용하여 사용하여 새로운 비디오 대여 기록을 추가하세요.

```
EXEC DBMS_OUTPUT.PUT_LINE(video_pkg.new_rental(110, 98))  
EXEC DBMS_OUTPUT.PUT_LINE(video_pkg.new_rental(109, 93))  
EXEC DBMS_OUTPUT.PUT_LINE(video_pkg.new_rental(107, 98))  
EXEC DBMS_OUTPUT.PUT_LINE(video_pkg.new_rental('Biri', 97))
```

다음 명령을 사용하여 사용하여 사용하여 영화를 반납하세요.

```
EXECUTE video_pkg.return_movie(92, 3, 'AVAILABLE')  
EXECUTE video_pkg.return_movie(95, 3, 'AVAILABLE')  
EXECUTE video_pkg.return_movie(93, 1, 'RENTED')
```

비디오 대여점의 영업 시간은 일요일에서 금요일까지는 오전 8:00 - 오후 10:00 이고 토요일은 오전 8:00 - 오후 12:00 입니다. 이 시간 동안에만 테이블을 수정할 수 있도록 트리거에 의해 호출되는 내장 프로시저를 테이블에 생성합니다.

현재 시간이 영업 시간인지 확인하는 TIME\_CHECK 라는 내장 프로시저를 생성합니다. 현재 시간이 영업 시간이 아닐 경우 RAISE\_APPLICATION\_ERROR 프로시저를 사용하여 해당 메시지를 제공합니다.

```
CREATE OR REPLACE PROCEDURE time_check IS
BEGIN
    IF ((TO_CHAR(SYSDATE, 'D') BETWEEN 1 AND 6) AND
        (TO_DATE(TO_CHAR(SYSDATE, 'hh24:mi '), 'hh24:mi ') NOT BETWEEN
            TO_DATE('08:00', 'hh24:mi ') AND TO_DATE('22:00', 'hh24:mi ')))
        OR ((TO_CHAR(SYSDATE, 'D') = 7)
            AND (TO_DATE(TO_CHAR(SYSDATE, 'hh24:mi '), 'hh24:mi ') NOT BETWEEN
                TO_DATE('08:00', 'hh24:mi ') AND TO_DATE('24:00', 'hh24:mi '))) THEN
        RAISE_APPLICATION_ERROR(-20999,
            'Data changes restricted to office hours. ');
    END IF;
END time_check;
/
```

다섯 개의 테이블 각각에 대한 트리거를 생성합니다. 데이터가 테이블에서 삽입, 갱신 및 삭제되기 전에 트리거를 실행합니다. 이러한 트리거 각각에서 TIME\_CHECK 프로시저를 호출합니다.

```
CREATE OR REPLACE TRIGGER member_trig
BEFORE INSERT OR UPDATE OR DELETE ON member
CALL time_check
/
CREATE OR REPLACE TRIGGER rental_trig
BEFORE INSERT OR UPDATE OR DELETE ON rental
CALL time_check
/
CREATE OR REPLACE TRIGGER title_copy_trig
BEFORE INSERT OR UPDATE OR DELETE ON title_copy
CALL time_check
/
```

```

CREATE OR REPLACE TRIGGER title_trig
  BEFORE INSERT OR UPDATE OR DELETE ON title
  CALL time_check
/
CREATE OR REPLACE TRIGGER reservation_trig
  BEFORE INSERT OR UPDATE OR DELETE ON reservation
  CALL time_check
/

```

트리거를 테스트합니다.

참고: **트리거가 실패하게 하려면 시간을 클래스의 현재 시간 범위 밖으로 변경해야 합니다.** 예를 들면, 테스트하는 동안 트리거의 적합한 비디오 영업 시간을 오후 6:00 에서 오전 8:00 로 설정할 수 있습니다.

```

-- First determine current timezone and time
SELECT SESSIONTIMEZONE,
       TO_CHAR(CURRENT_DATE, 'DD-MON-YYYY HH24:MI') CURR_DATE
FROM DUAL;

-- Change your time zone usinge [+/-]HH:MI format such that the current
-- time returns a time between 6pm and 8am
ALTER SESSION SET TIME_ZONE='-7:00';

-- Add a new member (for a sample test)
EXECUTE video_pkg.new_member('Elias', 'Elliane', 'Vine Street',
                             'California', '789-123-4567')

-- Restore the original time zone for your session.
ALTER SESSION SET TIME_ZONE='-00:00';

```