https://www.sublimetext.com/3

기준:
http://www.kobis.or.kr/kobisopenapi/homepg/apiservice/searchServiceInfo.do?serviceId=searchDailyBoxOffice

주의하실점은 customer 테이블에 데이터 넣기 전에 movieinfo에 데이터를 먼저 넣어주셔야해요

1. 영화관 예매 발행 시스템
1) 어떤 데이터를 사용할 것인지

(완) 가. CUSTOMER - 고객 데이터 (시퀀스 사용)

예매번호(reservation_num, PK), 고객명(username), 영화번호(movienum, FK), 영화제목(moviename), 상영관(theater), 상영일(running_date), 상영시간(running_time), 결제가격(price), 할인여부(isdiscount),

고객번호 없애고 예매번호를 PK로 쓰겠습니다

(완) 나. MOVIE_INFO - 영화 종류 데이터 (뷰 사용)

movienum
1 - 해리포터 : 죽음의 성물 2
2 - 스파이더맨 : 뉴 유니버스
3 - 블레어 위치 프로젝트 2
4 - 어벤져스 2 : 에이지 오브 울트론
5 - 닥터 스트레인지

- 고객 관점

영화번호(movienum, PK), 영화제목(moviename), 배급(distributor), 출연진(actor), 감독(director), 장르(genre), 등급(class), 결제가격(price), 월드박스오피스(worldboxoffice_rank), 할인여부(isdiscount) 평균 별점(avg_star)
(뷰) 영화추천서비스 - 가격 생략하고 위에서 3개만 ROWRUM

// 할인 설정 넣으려면 여기 할인여부 들어가는 게 맞지 않은가 생각이 듭니다 ?
원래 영화가격은 다 똑같은데 블레어윗치만 할인쿠폰을 주니까 그거 예매했다가 취소하는 컨셉?
그러면 그냥 탭 하나 discount_ability 만들어서 o, x만 넣으면 될 거 같아여
O x 보다는 isdiscount 컬럼명 넣어서 할인 되면 1 안되면 0으로 넣는거 어떠세요?
좋습니다~
괜찮습니다.

(완) 다. THEATERALL - 시간별 영화 예약 상황을 볼 수 있는 데이터

- 영화별 시간 (영화별 2-3개씩)

영화번호(movienum, PK), 영화제목(moviename), 상영관(theater), 상영시간(running_time)

(완) 라. BLAIRWITCH_SEAT - 특정 영화의 좌석 데이터

- 블레어 위치 프로젝트 2 1개 테이블 만들기
- 좌석번호(seat_num, PK), 좌석있는지여부(isseat),영화번호(movienum, FK), 예매번호(reservation_num, FK),
- 좌석 예시 : A열 1~46, B열 1~46, C열 1~46

(완) 마. MOVIE_REVIEW - 특정 영화의 후기 있는 데이터 (별점 여러개, 후기 여러개)

- 영화번호(movienum, FK), 영화제목(moviename), 별점(star), 후기(review), 리뷰번호(reviewnum, PK)
  (해리포터 : 죽음의 성물 2, 스파이더맨 : 뉴 유니버스, 어벤져스 2 : 에이지 오브 울트론, 닥터 스트레인지, 블레어 윗치 프로젝트2 영화당 5개)

작업 순서

1. 주제 선정 :
2.
3. 엑셀로 데이터 만들기
4. 모여서 확인
   - 변수 종류 매기기 (varchar2, number, date etc…)

- starUML 로 RDBMS 화
- 문제 만들기

5. 테이블로 변환
6. 데이터 INSERT
7. 시나리오 재검토
8. 시나리오 보면서 SQL 쿼리문 작성
9. 재확인

10. 발표자료 만들기

https://docs.google.com/spreadsheets/d/14lcs5NwP93An-m6AP_Uappe1ixi7u-x1rQJaP3LThkY/edit#gid=1152882001

https://ko.wikipedia.org/wiki/2020%EB%85%84_%EB%8C%80%ED%95%9C%EB%AF%BC%EA%B5%AD%EC%9D%98_%EC%98%81%ED%99%94_%EB%AA%A9%EB%A1%9D

2) 고객 관점에서 진행(스토리,시나리오, 프로세스에 따른 SQL문 작성 필요)

1. 고객이 보고 싶은 영화를 검색한다.
2. 보고 싶고, 끌리는 영화가 없어서 최근에 가장 인기가 많은 영화를 찾는다.
3. 가장 인기가 많은 1위~3위 영화를 선택해서 예약을 하려고 한다. view
4. 영화에 대해 찾아보던 중 할인을 받을 수 있는 영화를 발견했다! (3위)
5. 고객은 가장 인기가 많은 영화 1~3위 중 가장 저렴하게 볼 수 있는 영화를 선택한다.
6. 영화를 예매하려고 하자, 이미 영화를 온라인상으로 예약한 사람이 꽤 있었다. Null
7. 고객은 영화를 되도록이면 가운데, 맨 앞줄로부터 중간 위치에 예약을 한다.
8. 고객은 후기를 본다. 드럽게 재미없다고 한다. 역시 투자한만큼 영화도 재미있는것이다.
9. 고객은 예약했었던 영화 티켓을 취소하고, 가장 인기가 많은 순위 1~3위 중 가장 비싼 영화를 본다.

3) table구조 + 제약조건 + test data4) 배운 내용 활용하기
- 영화 최고 인기 순위 출력 (rownum 활용)
- 영화 추천 내역 공유 (뷰를 활용하기)
- GROUP BY (AVERAGE 영화별 별점)

0. INTRO

연지는 지난주에 한 소개팅에서 애프터 신청을 받았다.
크리스마스가 다가오고 있다.
이번 소개팅은 꼭 성공해야 한다.

연지는 팀프로젝트를 함께 하고 있는
플레이데이터 최고의 연애 마스터 민재에게 조언을 구하기로 했다.

아래는 김혜경 강사님이
연지, 민재, 종욱이 팀프로젝트를 하는 소회의실에 들어가 엿들은
셋이서 팀프로젝트에 삽입할 insert 구문을 만드는 노가다를 하며 나눈 대화다.

민재 : 빨리 사귀려면 공포영화가 국룰이지!!!
귀신 나올 때 아아악 소리지르면서 니가 먼저 손 꼭 잡아버려~
화장실 들어갈 때랑 나올 때 마음 다르듯
영화관 들어갈 때 썸남썸녀가 나올 땐 남친여친된다.
연지 : 오, 비유는 이상하지만 뭔가 그럴 듯 한데?
종욱 : 어! 나 얼마전에 개봉한 영화 블레어윗치인가 그거 할인쿠폰 있는데 너 줄까?
연지 : 오, 그것도 좋지!!
종욱 : 슬랙으로 보내줄게. 잘 되면 새끼쳐라!! ㅎㅎ

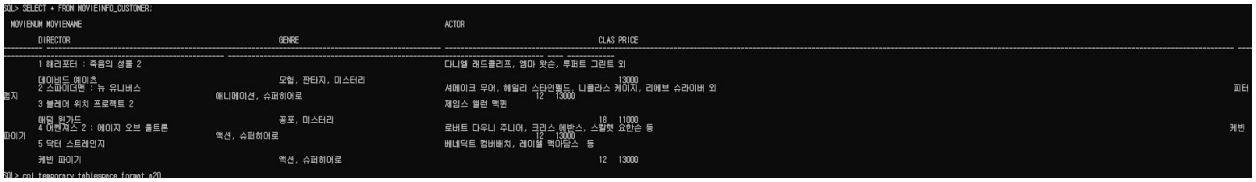
1. MOVIEINFO 연 / 완료

연지는 영화 사이트에 들어가 현재 상영 중인 영화들의 이름, 배우, 감독, 장르, 등급, 가격만 뽑아 view를 만들어 확인한다. (view)

-- Ans)
CREATE VIEW MOVIEINFO_CUSTOMER

AS SELECT movienum, moviename, actor, director, genre, class, price
FROM MOVIEINFO;

SELECT * FROM MOVIEINFO_CUSTOMER;



이름만 봐서는 뭔지 모르겠어서 MOVIE_REVIEW 테이블에서가장 별점이 높은 영화 3개를 찾아 MOVIEINFO와 함께 보기로 한다. (rownum)

-- Ans)
SELECT rownum, moviename, avg
FROM (
SELECT moviename, AVG(star) as avg
FROM MOVIE_REVIEW
GROUP BY moviename
ORDER BY AVG(star) DESC
)
WHERE ROWNUM < 4;

종욱이 준 할인쿠폰을 쓸 수 있는 영화는 없다. (아우터 조인)

연지는 그래도 블레어 위치가 어떤 장르 영화인지 확인하려 블레어 위치의 MOVIENUM(3)만을 선택하여 영화 이름, 장르 정보를 본다.

-- Ans)
SELECT moviename, genre
FROM MOVIEINFO_CUSTOMER
WHERE movienum = 3;

두둥,

```
GENRE
--------------------------------------------------------------------------------
공포, 미스터리
```

```
GENRE
-----------------
공포, 미스터리
```

```
GENRE
-----
공포,
```

별점 꺼지라 그래… 돈이 최고시다! 연지는 상영중인 모든 영화의 이름과 가격을 가장 저렴한 영화부터 비교한다.

-Ans)
SELECT moviename, price
FROM MOVIEINFO_CUSTOMER
ORDER BY PRICE ASC;

```
MOVIENAME                                                                                          PRICE
-------------------------------------------------------------------------------------------------- -----------
블레어 위치 프로젝트 2                                                                                 11000
스파이더맨 : 뉴 유니버스                                                                               13000
닥터 스트레인지                                                                                        13000
어벤져스 2 : 에이지 오브 울트론                                                                        13000
해리포터 : 죽음의 성물 2                                                                              13000
```

2. THEATERALL 종/완료

연지는 블레어윗치프로젝트2를 예매하기 위해 THEATERALL에서 블레어윗치프로젝트 2의 오늘 날짜 영화상영정보를 확인한다.

블레어윗치프로젝트2는 공포 영화니까 저녁에 보는 게 좋겠지? 몇시에 하더라…

연지는 영화이름이 블레어 윗치 프로젝트 2를 검색해 영화이름과 상영시간을 확인한다.

-- Ans)
SELECT moviename, running_time
FROM THEATERALL
WHERE moviename='블레어 윗치 프로젝트 2';

```
SQL> select moviename, running_time
  2  from theaterall
  3  where moviename='블레어 윗치 프로젝트 2';

MOVIENAME                                                                                          RUNNING_TIME
-------------------------------------------------------------------------------------------------- --------------------
블레어 윗치 프로젝트 2                                                                                 15:30  20:50
```

<다른 풀이>
SELECT moviename, running_time
FROM THEATERALL
WHERE theater = ‘3관’;

```
SQL> select moviename, running_time
  2  from theaterall
  3  where theater = '3관';

MOVIENAME                                                                                          RUNNING_TIME
-------------------------------------------------------------------------------------------------- ------------------------------
블레어 윗치 프로젝트 2                                                                                     15:30  20:50
```

3. BLAIREWITCH_SEAT 종 / 완료

벌써 좌석이 꽤 차 있다.

-- Ans)
SELECT count(isseat) AS 예매된좌석수
FROM (
  SELECT *
  FROM movieinfo i, blairwitch_seat b
  WHERE isseat = 1 AND i.movienum = b.movienum
)
GROUP BY isseat;

SELECT count(isseat) AS 예매가능한좌석수
FROM (
  SELECT *
  FROM movieinfo i, blairwitch_seat b
  WHERE isseat = 0 AND i.movienum = b.movienum
)
GROUP BY isseat;

```
SQL> select count(isseat) as 예매된좌석수
  2  from (
  3      select *
  4      from movieinfo i, blairwitch_seat b
  5      where isseat = 1 and i.movienum = b.movienum
  6  )group by isseat;

예매된좌석수
------------
          41

SQL> select count(isseat) as 예매가능한좌석수
  2  from (
  3      select *
  4      from movieinfo i, blairwitch_seat b
  5      where isseat = 0 and i.movienum = b.movienum
  6  )group by isseat;

예매가능한좌석수
----------------
              97
```

영화를 좋은 자리에서 보려고 정 중앙인 B줄의 가운데인 B25를 선택한다.
그 자리는 벌써 예매되어 있었다.

-- Ans)
SELECT moviename, seat_num, isseat
FROM movieinfo i, blairwitch_seat b
WHERE b.seat_num = 'B25' AND i.movienum = b.movienum;

```
SQL> select moviename, seat_num, isseat
  2  from movieinfo i, blairwitch_seat b
  3  where b.seat_num = 'B25' and i.movienum = b.movienum;

MOVIENAME                                                                                                        SEAT_N     ISSEAT
---------------------------------------------------------------------------------------------------------------- ------ ----------
블레어 위치 프로젝트 2                                                                                           B25             1
```

SELECT moviename, seat_num, isseat, reservation_num
FROM movieinfo i, blairwitch_seat b
WHERE b.isseat = 0 AND b.seat_num like 'B2_' AND i.movienum = b.movienum;

```
SQL> select moviename, seat_num, isseat, reservation_num
  2  from movieinfo i, blairwitch_seat b
  3  where b.isseat = 0 and b.seat_num like 'B2_' and i.movienum = b.movienum;

MOVIENAME                                                                                                        SEAT_N     ISSEAT RESERVATION_NUM
---------------------------------------------------------------------------------------------------------------- ------ ---------- --------------
블레어 위치 프로젝트 2                                                                                           B21             0
블레어 위치 프로젝트 2                                                                                           B22             0
블레어 위치 프로젝트 2                                                                                           B23             0
블레어 위치 프로젝트 2                                                                                           B24             0
블레어 위치 프로젝트 2                                                                                           B26             0
블레어 위치 프로젝트 2                                                                                           B27             0
블레어 위치 프로젝트 2                                                                                           B28             0

7 rows selected.
```

바로 옆 좌석인 B24를 예매한다.

-- Ans)
SELECT moviename, seat_num, isseat, reservation_num
FROM movieinfo i, blairwitch_seat b
WHERE b.seat_num ='B24' AND reservation_num is null AND i.movienum = b.movienum;

```
SQL> select moviename, seat_num, isseat, reservation_num
  2  from movieinfo i, blairwitch_seat b
  3  where b.seat_num ='B24'and reservation_num is null and i.movienum = b.movienum;

MOVIENAME                                                                                                        SEAT_N     ISSEAT RESERVATION_NUM
---------------------------------------------------------------------------------------------------------------- ------ ---------- --------------
블레어 위치 프로젝트 2                                                                                           B24             0
```

4. CUSTOMER 연 / 완료

연지는 아래의 INSERT 구문을 참고하여 1231번에 <블레어 위치 프로젝트 2>를 예매한다.

-- Ans)
INSERT INTO CUSTOMER VALUES(1231,'김연지',3,'블레어 위치 프로젝트 2','3관',to_date('11-27-2020','mm-dd-yyyy'),'15:30',11000,1);

```
SQL> INSERT INTO CUSTOMER VALUES(1231,'김연지',3,'블레어 위치 프로젝트 2','3관',to_date('11-27-2020','mm-dd-yyyy'),'15:30',11000,1);

1 row created.
```

연지는 예매번호 1231번을 검색해 예매번호와 상영관, 예매시간 등을 확인한다.

-- Ans)
SELECT * FROM CUSTOMER
WHERE reservation_num = 1231;

```
SQL> SELECT * FROM CUSTOMER
  2  WHERE reservation_num = 1231;

RESERVATION_NUM USERNAME                                 MOVIENUM MOVIENAME                                                                                       THEATER                                            RUNNING_ RUNNING_TIME                                       PRICE ISDISCOUNT
--------------- ---------------------------------------- -------- ---------------------------------------- -------------------------------------------------- ---------- ---------- ---------- ---------- ---------- ----------
           1231 김연지                                          1 블레어 위치 프로젝트 2                   3관                                                20/11/27 15:30                                       11000          1
```

아 맞다! 할인쿠폰을 제대로 썼는지 특별히 따로 확인한다. 1이면 사용이 된 것이다.

-- Ans)
SELECT isdiscount FROM CUSTOMER
WHERE reservation_num = 1231;

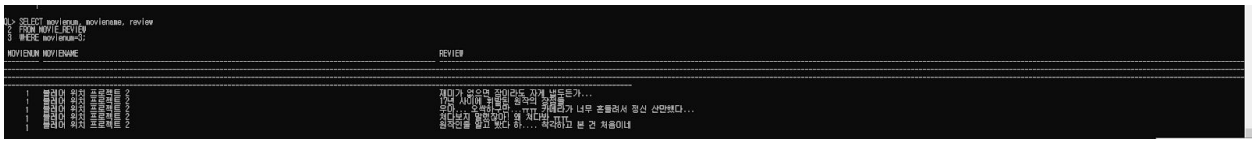
```
SQL> SELECT isdiscount  FROM CUSTOMER
  2  WHERE reservation_num=1231;

ISDISCOUNT
----------
         1
```

5. MOVIE_REVIEW 연

연지는 “데이트, 로맨틱, 성공적!” 하기 위해
썸남에게 영화 예매 내역을 보내기 전에 오늘 볼 영화(movienum=3)에 대해 다른사람이 쓴 리뷰들(REVIEW)을 찾아 본다.

악평 일색이다.

-- Ans)
SELECT movienum, moviename, review
FROM MOVIE_REVIEW
WHERE movienum = 3;



6. MOVIEINFO 민

다른 영화를 예매하기 위해 다시 MOVIEINFO에서 가장 평균별점이 높은 영화를 선택한다.

-- Ans)
SELECT ROWNUM, 영화제목, 평균별점
FROM (
SELECT moviename AS 영화제목, AVG(star) AS 평균별점
FROM MOVIE_REVIEW
GROUP BY moviename
ORDER BY AVG(star) DESC
)
WHERE ROWNUM < 2;

```
SQL> SELECT ROWNUM, 영화제목, 평균별점
  2  FROM (
  3  SELECT moviename AS 영화제목, AVG(star) AS 평균별점
  4  FROM MOVIE_REVIEW
  5  GROUP BY moviename
  6  ORDER BY AVG(star) DESC
  7  )
  8  WHERE ROWNUM < 2;

    ROWNUM 영화제목                                          평균별점
---------- ---------------------------------------- ----------
         1 스파이더맨 : 뉴 유니버스                          4.7
```

7. THEATERALL 민

연지는 이전에 했던 예매번호 1231번 영화예매를 취소한다.

-- Ans)
DELETE FROM CUSTOMER WHERE reservation_num = 1231;

```
SQL> DELETE FROM CUSTOMER
  2  WHERE reservation_num = 1231;

1 row deleted.
```

그 후, 새로 <스파이더맨 : 뉴 유니버스> 영화를 예매한다.

-- Ans)
INSERT INTO CUSTOMER VALUES(1231,'김연지',2,'스파이더맨 : 뉴 유니버스','2관',to_date('11-27-2020','mm-dd-yyyy'),'17:20',13000,1);

```
SQL> INSERT INTO CUSTOMER VALUES(1231,'김연지',2,'스파이더맨 : 뉴 유니버스','2관',to_date('11-27-2020','mm-dd-yyyy'),'17:20',13000,1);

1 row created.
```

연지는 다시 한 번 예매번호 1231번을 검색해 예매번호와 상영관, 예매시간 등을 확인한다.

-- Ans)
SELECT * FROM CUSTOMER WHERE reservation_num = 1231;

```
SQL> SELECT *
  2  FROM CUSTOMER
  3  WHERE reservation_num = 1231;

RESERVATION_NUM USERNAME                       MOVIENUM MOVIENAME                                THEATER    RUNNING_DATE RUNNING_TIME       PRICE ISDISCOUNT
--------------- ------------------------------ -------- ---------------------------------------- ---------- ------------ ------------ ---------- ----------
          1,231 김연지                                2 스파이더맨 : 뉴 유니버스                 2관         20/11/27     17:20            13,000          1
```

8. CUSTOMER 연

연지는 예매 후 이름을 검색하여 자신의 예매번호와 상영관, 예매시간 등을 확인한다.

-- Ans)
1) SELECT * FROM CUSTOMER WHERE username LIKE '김연지';
2) SELECT * FROM CUSTOMER WHERE username = '김연지';

```
SQL> SELECT * FROM CUSTOMER WHERE username LIKE '김연지';
RESERVATION_NUM USERNAME                       MOVIENUM MOVIENAME                                THEATER    RUNNING_ RUNNING_TIME       PRICE ISDISCOUNT
--------------- ------------------------------ -------- ---------------------------------------- ---------- -------- ------------ ---------- ----------
           1231 김연지                                1 블레어 위치 프로젝트 2                   3관         20/11/27 15:30            11000          1
```

9. OUTRO

연지의 썸남은 애니메이션 덕후였고 심사숙고한 영화는 딱 그 남자의 취향이었다.
2020년 크리스마스, 연지는 외롭지 않다.

종욱도 외롭지 않다.
연지는 쿠폰은 쓰지 않았지만 새끼를 쳤기 때문이다.

민재는 원래 외롭지 않다.
플레이데이터 최고의 연애마스터이기 때문이다.

==========

2. 숙소 예약 관련 DB
1) 어떤 데이터를 사용할 것인지
- 고객 데이터(생년 월일, 이름, 등급, 적립률, 마일리지 등)
- 숙소방 데이터 (숙소번호, 방번호, 소재지, 업소전화번호, 객실등급)
- 리뷰 데이터(리뷰번호, 리뷰내용, 평점)
- 지역별 숙소 예약 현황을 볼 수 있는 데이터2) 고객 관점에서 진행(스토리,시나리오)3) table구조 + 제약조건 + test data4) 배운 내용 활용하기
- 숙소 최고 인기 순위 출력 (rownum 활용)
2) 고객 관점 진행
1. 고객은 회원가입을 할 수 있음
2. 회원가입 시 전화번호, 이름, 비밀번호, 생년월일, 주소를 입력하면 회원번호를 할당받음
3. 고객은 회원정보를 변경할 수 있음
4. 고객은 숙소 검색이 가능함
5. 고객은 숙소 예약 시 등급별 마일리지를 쌓을 수 있음

6. 고객은 숙소 예약 시 예약번호를 할당받음
7. 고객은 투숙한 숙박에 대하여 평점입력이 가능함