

■ 문제

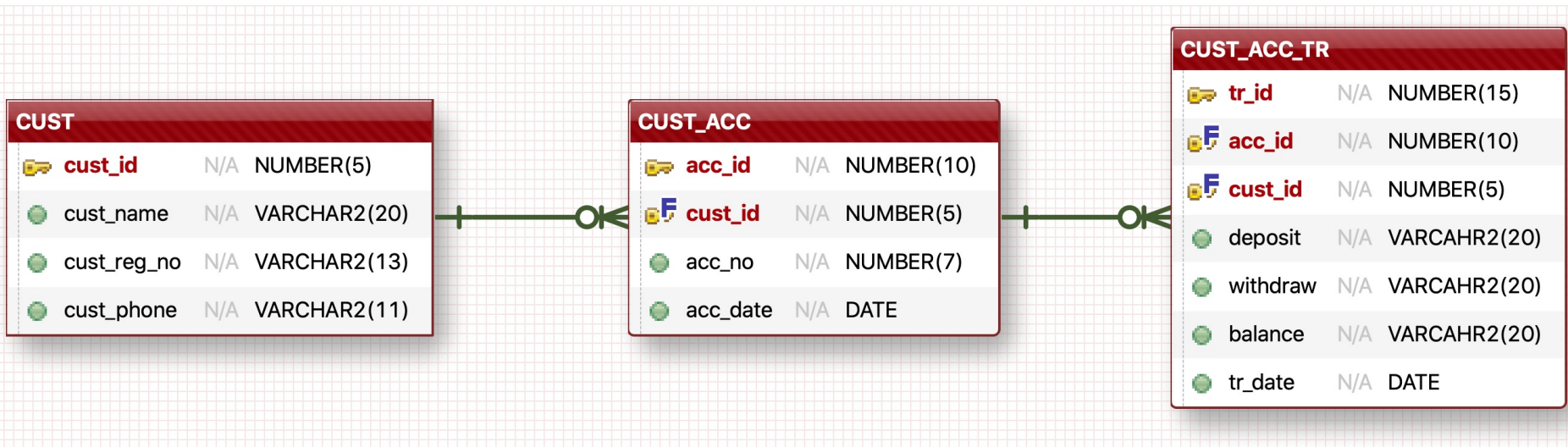
- Scanner 를 사용해서 키워드를 입력받아 **계좌거래** 및 **거래내역 검색** 프로그램을 작성해 보세요
- **거래일(특정일) 거래내역 조회, 거래일 기간조회**가 가능 하도록 해당 메소드를 구현 하세요
- Dao, Vo 클래스를 사용하세요

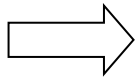
일련번호	예금액	출금액	거래일	잔액
seq_id	deposit	withdraw	tr_date	balanace
1	0	0	2020-02-22	0
2	1000	0	2020-03-01	1000
3	10000	0	2020-08-15	11000
4	0	1000	2020-08-20	10000
5	0	5000	2020-09-22	5000
6	1000	0	2020-11-03	6000

논리 ERD



물리 ERD



AccountApp**현재잔액출력()**
거래전체내역출력()**AccountDAO****getBalance()**
getListAll()**AccountVO****int seq_id**
int deposit
int withdraw
String tr_date
int balance

일련번호

예금액

출금액

거래일

현재잔액

seq id	deposit	withdraw	tr date	balanace
1	0	0	2020-02-22	0
2	1000	0	2020-03-01	1000
3	10000	0	2020-08-15	11000
4	0	1000	2020-08-20	10000
5	0	5000	2020-09-22	5000
6	1000	0	2020-11-03	6000

List<AccountVO> accList**AccountVO accVO**

■ 데이터베이스에 테이블, 시퀀스 객체 생성

```
create table account(  
    seq_id number(10) not null,  
    deposit number(15) not null,  
    withdraw number(15) not null,  
    tr_date date,  
    balanace number(15) not null,  
    primary key(seq_id)  
);  
  
CREATE SEQUENCE ACCOUNT_SEQ_ID  
MINVALUE 1 INCREMENT BY 1 ;  
  
commit;
```

```
1 package com.javaex.bank;  
2  
3 public class AccountVO {  
4     private int seqId;  
5     private int deposit;  
6     private int withdraw;  
7     private String trDate;  
8     private int balance;  
9  
10    public int getSeqId() {  
11        return this.seqId;  
12    }  
13    public void setSeqId(int seqId) {  
14        this.seqId = seqId;  
15    }  
16    { ...생략... }
```

■ VO 작성

- 거래정보를 저장한 테이블(account)의 seq_id, deposit, withdraw, tr_date, balance 컬럼에 해당하는 VO 작성

일련번호	예금액	출금액	거래일	잔액
seq_id	deposit	withdraw	tr_date	balanace
1	0	0	2020-02-22	0
2	1000	0	2020-03-01	1000

- AccountVO Class 의 setter(), getter(), to_String(), AccountVO() 등의 method 들도 작성

■ DAO 작성

- AccountDAO Class 에 getBalance(), getList(거래일), getList(검색시작일, 검색종료일), getListAll(), insertTradeInfo(예금,예금액) or insertTradeInfo(출금,출금액)
- public AccountVO getBalance() { ... }
- public List<AccountVO> getListAll() { ... }
- public List<AccountVO> getList(searchStartDate, searchEndDate) { ... }
- public List<AccountVO> getList(TradingDate) { ... }
- public int insertTradeInfo("Deposit", 1000) { ... }

■ DAO의 insertTradeInfo() Method 구현시 참고

```
-- 거래내역 기간 조회 SQL
select seq_id, deposit, withdraw, to_char(tr_date,'YYYY-MM-DD ' ), balance
from account where to_char(tr_date,'YYYYMMDD') BETWEEN '2020-08-01'
AND '2020-08-31';

-- 입금SQL
INSERT INTO account
(seq_id, deposit, withdraw, tr_date, balance)
select ACCOUNT_SEQ_ID.nextval, 3000, 0, sysdate, balance+3000
from account
where seq_id = (select max(seq_id) from account);
```


■ App 작성

- AccountApp Class 에 main method 에 다음 내용을 구현
- Scanner 를 통해 메뉴 선택 (1.예금, 2.출금, 3.잔액조회, 4.거래일 조회, 5.기간별 조회)
- 1. 선택후 "예금액>" 출력후 Scanner 통해 금액 입력 받아 **insertTradeInfo()** 호출
- 2. 선택후 "출금액>" 출력후 Scanner 통해 금액 입력 받아 **insertTradeInfo()** 호출
- 3. 선택후 **getBalance()** 호출
- 4. 선택후 "조회일(YYYYMMDD)>" 출력후 Scanner 통해 날짜 입력 받아 **getList(TradingDate)** 호출
- 5. 선택후 "조회일(YYYYMMDD-YYYYMMDD)>" 출력후 Scanner 통해 날짜-날짜 입력 받아 **getList(searchStartDate, searchEndDate)** 호출
- 작성후 jongukjeong@gmail.com 로...
- 작성할 소스의 패키지 명 **com.kosta.jdbcdao.본인이름이니셜**