

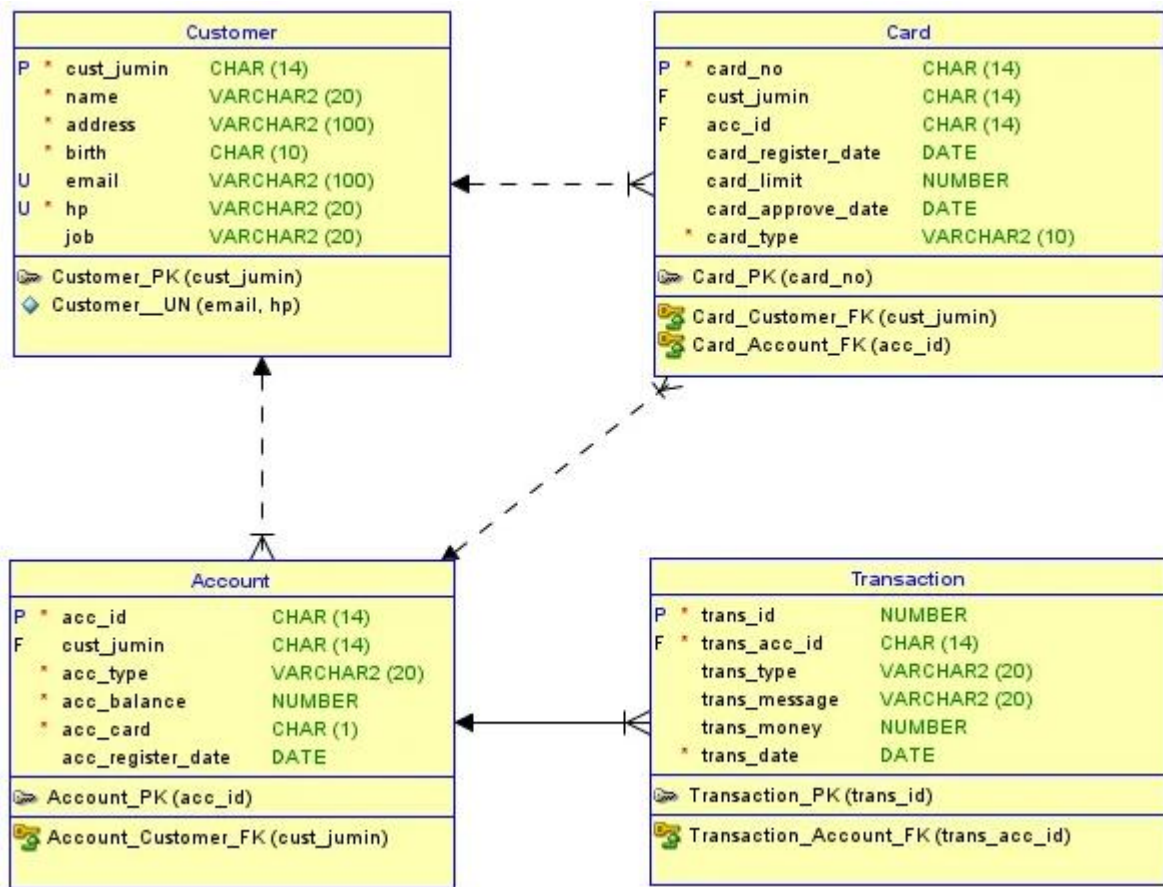
(부산은행) 금융 DT아카데미 개발자 양성과정

SQL(ORACLE DBMS)

과제제출

최민수

작업1) ERD 작업 및 테이블 생성



사용자명 입력: system

비밀번호 입력:

마지막 성공한 로그인 시간: 화 8월 12 2025 11:48:09 +09:00

다음에 접속됨:

Oracle Database 21c Express Edition Release 21.0.0.0.0 - Production
Version 21.3.0.0.0

```
SQL> ALTER SESSION SET "_ORACLE_SCRIPT"=TRUE;
```

세션이 변경되었습니다.

```
SQL> CREATE USER BANKTEST IDENTIFIED BY 1234;
```

사용자가 생성되었습니다.

```
SQL> GRANT CONNECT, RESOURCE, UNLIMITED TABLESPACE TO BANKTEST;
```

권한이 부여되었습니다.

작업1) ERD 작업 및 테이블 생성

Table ACCOUNT이(가) 생성되었습니다.

Table ACCOUNT이(가) 변경되었습니다.

Table CARD이(가) 생성되었습니다.

Table CARD이(가) 변경되었습니다.

Table CUSTOMER이(가) 생성되었습니다.

Table CUSTOMER이(가) 변경되었습니다.

Table CUSTOMER이(가) 변경되었습니다.

Table TRANSACTION이(가) 생성되었습니다.

Table TRANSACTION이(가) 변경되었습니다.

Table ACCOUNT이(가) 변경되었습니다.

Table CARD이(가) 변경되었습니다.

Table CARD이(가) 변경되었습니다.

Table TRANSACTION이(가) 변경되었습니다.

Sequence TRANSACTION_TRANS_ID_SEQ이(가) 생성되었습니다.

Trigger TRANSACTION_TRANS_ID_TRG이(가) 컴파일되었습니다.

작업2) 데이터 입력

-- ■■■ CUSTOMER

```
INSERT INTO CUSTOMER VALUES ('760121-1234567', '정우성', '서울', '1976-01-21', '-', '010-1101-7601', '배우');
INSERT INTO CUSTOMER VALUES ('750611-1234567', '이정재', '서울', '1975-06-11', '-', '010-1102-7506', '배우');
INSERT INTO CUSTOMER VALUES ('890530-1234567', '전지현', '대전', '1989-05-30', 'jjh@naver.com', '010-1103-8905', '자영업');
INSERT INTO CUSTOMER VALUES ('790413-1234567', '이나영', '대전', '1979-04-13', 'lee@naver.com', '010-2101-7904', '회사원');
INSERT INTO CUSTOMER VALUES ('660912-1234567', '원빈', '대전', '1966-09-12', 'one@daum.net', '010-2104-6609', '배우');
```

-- ■■■ ACCOUNT

```
INSERT INTO ACCOUNT VALUES ('1011-1001-1001', '760121-1234567', '자유입출금', 4160000, 'Y', '2020-01-21 13:00:02');
INSERT INTO ACCOUNT VALUES ('1011-1001-1002', '890530-1234567', '자유입출금', 376000, 'Y', '2020-06-11 13:00:02');
INSERT INTO ACCOUNT VALUES ('1011-1001-1003', '790413-1234567', '자유입출금', 1200000, 'Y', '2020-05-30 13:00:02');
INSERT INTO ACCOUNT VALUES ('1011-2001-1004', '660912-1234567', '정기적금', 1000000, 'N', '2020-04-13 13:00:02');
INSERT INTO ACCOUNT VALUES ('1011-1002-1005', '750611-1234567', '자유입출금', 820000, 'Y', '2020-09-12 13:00:02');
```

-- ■■■ CARD

```
INSERT INTO CARD VALUES ('2111-1001-1001', '760121-1234567', '1011-1001-1001', '2020-01-21', 1000000, '2020-02-10', 'check');
INSERT INTO CARD VALUES ('2041-1001-1002', '890530-1234567', '1011-1001-1002', '2020-06-11', 3000000, '2020-06-15', 'check');
INSERT INTO CARD VALUES ('2011-1001-1003', '790413-1234567', '1011-1001-1003', '2020-05-30', 5000000, '2020-06-25', 'check');
INSERT INTO CARD VALUES ('2611-1001-1005', '750611-1234567', '1011-1002-1005', '2020-09-12', 1500000, '2020-10-10', 'check');
```

-- ■■■ TRANSACTION

```
INSERT INTO TRANSACTION (TRANS_ACC_ID, TRANS_TYPE, TRANS_MESSAGE, TRANS_MONEY, TRANS_DATE) VALUES ('1011-1001-1001', '입금', '2월 정기급여', 3500000, '2020-02-10 12:36:21');
INSERT INTO TRANSACTION (TRANS_ACC_ID, TRANS_TYPE, TRANS_MESSAGE, TRANS_MONEY, TRANS_DATE) VALUES ('1011-1001-1003', '출금', 'ATM 출금', 300000, '2020-02-10 12:37:21');
INSERT INTO TRANSACTION (TRANS_ACC_ID, TRANS_TYPE, TRANS_MESSAGE, TRANS_MONEY, TRANS_DATE) VALUES ('1011-1001-1002', '입금', '2월 급여', 2800000, '2020-02-10 12:38:21');
INSERT INTO TRANSACTION (TRANS_ACC_ID, TRANS_TYPE, TRANS_MESSAGE, TRANS_MONEY, TRANS_DATE) VALUES ('1011-1001-1001', '출금', '2월 공과금', 116200, '2020-02-10 12:39:21');
INSERT INTO TRANSACTION (TRANS_ACC_ID, TRANS_TYPE, TRANS_MESSAGE, TRANS_MONEY, TRANS_DATE) VALUES ('1011-1002-1005', '출금', 'ATM 출금', 50000, '2020-02-10 12:40:21');
```

작업3) SQL 실행 // 1, 2, 3

-- ■■■ Q.1) "모든 고객" 정보 조회

```
SELECT * FROM CUSTOMER;
```

	CUST_JUMIN	NAME	ADDRESS	BIRTH	EMAIL	HP	JOB
1	760121-1234567	정우성	서울	1976-01-21	-	010-1101-7601	배우
2	750611-1234567	이정재	서울	1975-06-11	-	010-1102-7506	배우
3	890530-1234567	전지현	대전	1989-05-30	jjh@naver.com	010-1103-8905	자영업
4	790413-1234567	이나영	대전	1979-04-13	lee@naver.com	010-2101-7904	회사원
5	660912-1234567	원빈	대전	1966-09-12	one@daum.net	010-2104-6609	배우

-- ■■■ Q.2) "모든 카드" 정보 조회

```
SELECT * FROM CARD;
```

	CARD_NO	CUST_JUMIN	ACC_ID	CARD_REGISTER_DATE	CARD_LIMIT	CARD_APPROVE_DATE	CARD_TYPE
1	2111-1001-1001	760121-1234567	1011-1001-1001	2020-01-21 00:00:00	1000000	2020-02-10 00:00:00	check
2	2041-1001-1002	890530-1234567	1011-1001-1002	2020-06-11 00:00:00	3000000	2020-06-15 00:00:00	check
3	2011-1001-1003	790413-1234567	1011-1001-1003	2020-05-30 00:00:00	5000000	2020-06-25 00:00:00	check
4	2611-1001-1005	750611-1234567	1011-1002-1005	2020-09-12 00:00:00	1500000	2020-10-10 00:00:00	check

-- ■■■ Q.3) "모든 예금계좌" 정보 조회

```
SELECT * FROM ACCOUNT;
```

	ACC_ID	CUST_JUMIN	ACC_TYPE	ACC_BALANCE	ACC_CARD	ACC_REGISTER_DATE
1	1011-1001-1001	760121-1234567	자유입출금	4160000	Y	2020-01-21 13:00:02
2	1011-1001-1002	890530-1234567	자유입출금	376000	Y	2020-06-11 13:00:02
3	1011-1001-1003	790413-1234567	자유입출금	1200000	Y	2020-05-30 13:00:02
4	1011-2001-1004	660912-1234567	정기적금	1000000	N	2020-04-13 13:00:02
5	1011-1002-1005	750611-1234567	자유입출금	820000	Y	2020-09-12 13:00:02

작업3) SQL 실행 // 4, 5, 6

-- ■■■ Q.4) "가장 최근 거래 내역 3건" 조회

```
SELECT
    TRANS_DATE "최근 거래 내역"
FROM TRANSACTION
ORDER BY trans_date DESC
OFFSET 0 ROWS FETCH NEXT 3 ROWS ONLY
;
```

	최근 거래 내역
1	2020-02-10 12:40:21
2	2020-02-10 12:39:21
3	2020-02-10 12:38:21

-- ■■■ Q.5) "카드한도금액 200만원 이상" "고객이름, 카드종류" 조회

```
SELECT
    C.NAME "고객이름",
    CA.CARD_TYPE "카드종류"
FROM CUSTOMER C
JOIN CARD CA ON c.cust_jumin = CA.cust_jumin
WHERE CA.CARD_LIMIT >= 2000000
;
```

	고객이름	카드종류
1	전지현	check
2	이나영	check

-- ■■■ Q.6) "예금계좌별" "거래 건수" 조회

```
SELECT
    A.ACC_ID "예금 계좌",
    COUNT(*) "거래 건수"
FROM ACCOUNT A
JOIN TRANSACTION T ON a.acc_id = t.trans_acc_id
GROUP BY A.ACC_ID
;
```

	예금 계좌	거래 건수
1	1011-1001-1001	2
2	1011-1001-1003	1
3	1011-1001-1002	1
4	1011-1002-1005	1

작업3) SQL 실행 // 7, 8, 9

-- ■■■ Q.7) "거래금액 100만 이상" "거래 내역" "최근 거래순" 조회

```
SELECT *  
FROM TRANSACTION  
WHERE trans_money >= 1000000  
ORDER BY trans_date DESC
```

	↕ TRANS_ID	↕ TRANS_ACC_ID	↕ TRANS_TYPE	↕ TRANS_MESSAGE	↕ TRANS_MONEY	↕ TRANS_DATE
1	3	1011-1001-1002	입금	2월 급여	2800000	2020-02-10 12:38:21
2	1	1011-1001-1001	입금	2월 정기급여	3500000	2020-02-10 12:36:12

;

-- ■■■ Q.8) "계좌와 연결된" "계좌ID, 카드ID, 카드종류" 조회

```
SELECT  
A.ACC_ID "계좌ID",  
C.CARD_NO "카드ID",  
C.CARD_TYPE "카드종류"  
FROM ACCOUNT A  
JOIN CARD C ON A.ACC_ID = C.ACC_ID
```

	↕ 계좌ID	↕ 카드ID	↕ 카드종류
1	1011-1001-1001	2111-1001-1001	check
2	1011-1001-1002	2041-1001-1002	check
3	1011-1001-1003	2011-1001-1003	check
4	1011-1002-1005	2611-1001-1005	check

;

-- ■■■ Q.9) "예금구분 = 입금" "거래 총합" 조회

```
SELECT SUM(TRANS_MONEY) "거래 총합"  
FROM TRANSACTION  
WHERE TRANS_TYPE = '입금'
```

	↕ 거래 총합
1	6300000

;

작업3) SQL 실행 // 10 (END)

```
-- ■■■ Q.10) "잔고 400만 이상 고객" "고객명, 주민번호, 전화번호, 주소" 조회
SELECT
  C.NAME "고객명",
  C.CUST_JUMIN "주민번호",
  C.HP "전화번호",
  C.ADDRESS "주소"
FROM ACCOUNT A
JOIN CUSTOMER C ON A.CUST_JUMIN = C.CUST_JUMIN
WHERE A.acc_balance >= 4000000
```

	고객명	주민번호	전화번호	주소
1	정우성	760121-1234567	010-1101-7601	서울

감사합니다

- 끝 -