

# MariaDB설계\_과정평가형

교사 최선경

# 도서관리Book

Book(도서) 테이블  
도서번호(BookID, PK)  
도서명(Title, NOT NULL)  
저자(Author)  
출판사(Publisher)  
가격(Price, CHECK Price ≥ 0)  
출판년도(PubYear, YYYY 형식)

Rental(대출) 테이블  
대출번호(RentalID, PK)  
회원번호(MemberID, FK → Member)  
도서번호(BookID, FK → Book)  
대출일자(RentDate)  
반납일자(ReturnDate, NULL 가능)

Member(회원) 테이블  
회원번호(MemberID, PK)  
이름(Name, NOT NULL)  
연락처(Phone)  
주소(Address)

1. 테이블을 생성하시오. (제약조건 포함)
2. SQL 문을 작성하시오.
  - (1) 2020년 이상(>=) 출판된 도서를 검색하시오.
  - (2) '홍길동' 회원이 대출한 도서 목록을 출력하시오.
  - (3) 반납하지 않은 도서를 검색하시오.
  - (4) 도서별 대출 횟수를 출력하시오.
  - (5) 가격이 가장 비싼 도서를 출력하시오.

# 사원관리Employee

Employee(사원관리) 테이블

컬럼:

EmpNo (사번, INT, PRIMARY KEY, AUTO\_INCREMENT)

EmpName (사원명, VARCHAR(30), NOT NULL)

Dept (부서명, VARCHAR(20), NOT NULL)

HireDate (입사일, DATE, NOT NULL)

Salary (급여, INT, CHECK(Salary >= 2000000))

제약조건을 설정하시오.

EmpName은 중복될 수 없다.

급여(Salary)는 2,000,000원 이상만 입력 가능하다.

1. 테이블을 생성하시오. (제약조건 포함)

2. SQL 문을 작성하시오.

(1) **사원 등록:** Employee 테이블에 다음 데이터를 삽입하시오.

(홍길동, 영업부, 2020-03-01, 2500000)

(이순신, 인사부, 2019-07-15, 3200000)

(강감찬, 개발부, 2021-01-10, 2800000)

(2) **조회:**

- 부서가 '개발부'인 사원의 사번, 이름, 급여를 조회하시오.

- 급여가 3,000,000원 이상인 사원의 이름과 부서를 조회하시오.

(3) **수정:**

- 이순신의 급여를 3,500,000원으로 수정하시오.

(4) **삭제:**

사번이 1번인 사원의 정보를 삭제하시오.

# 홈쇼핑 회원관리ShopMember

<b>ShopMember (회원관리) 테이블</b> 컬럼명과 제약조건: CustNo (회원번호, INT, PRIMARY KEY, AUTO_INCREMENT) CustName (회원성명, VARCHAR(30), NOT NULL) Phone (전화번호, VARCHAR(13), UNIQUE) Address (주소, VARCHAR(50)) JoinDate (가입일자, DATE, NOT NULL) Grade (고객등급, CHAR(1), CHECK(Grade IN ('A','B','C'))) City (도시코드, CHAR(2))	<b>Sale (판매) 테이블</b> 컬럼명과 제약조건: SaleNo (판매번호, INT, PRIMARY KEY, AUTO_INCREMENT) CustNo (회원번호, INT, 외래키: ShopMember.CustNo 참조) PCost (단가, INT) Amount (수량, INT) Price (금액, INT) PCode (상품코드, CHAR(3))
--	---

1. 테이블을 생성하시오. (제약조건 포함)

2. SQL 문을 작성하시오.

(1) **회원 등록**: ShopMember 테이블에 다음 데이터를 삽입하시오.  
( '홍길동', '010-1234-5678', '서울시 강남구', '2020-01-01', 'A', '01' )  
( '이순신', '010-2222-3333', '부산시 해운대구', '2021-03-15', 'B', '02' )  
( '강감찬', '010-7777-8888', '대구시 달서구', '2019-05-20', 'C', '03' )

(2) **회원 조회**

- 고객등급이 A등급인 회원의 이름, 전화번호, 가입일자를 조회.
- 가입일자가 2020년 이후인 회원을 조회하시오.

(3) **판매 등록**: Sale 테이블에 삽입하시오.

- (1, 1000, 10, 10000, 'P01')
- (2, 2000, 5, 10000, 'P02')
- (3, 1500, 7, 10500, 'P03')

(4) **판매 조회**

- 회원별 총 구매금액을 구하시오.  
(출력: 회원번호, 회원성명, 총금액)
- 가장 구매금액이 높은 회원의 이름과 금액을 조회하시오.

(5) **데이터 수정 / 삭제**

- '이순신' 회원의 등급을 A로 수정하시오.
- CustNo = 3 인 회원을 삭제하시오.

# 아티스트(Artist) 관리

Artist (아티스트) 테이블

컬럼명과 제약조건:

ArtistNo

(아티스트번호, INT, PRIMARY KEY, AUTO\_INCREMENT)

ArtistName (아티스트명, VARCHAR(30), NOT NULL, UNIQUE)

DebutDate (데뷔일자, DATE, NOT NULL)

Genre (장르, VARCHAR(20), NOT NULL)

Agency (소속사, VARCHAR(30))

Album (앨범) 테이블

컬럼명과 제약조건:

AlbumNo (앨범번호, INT, PRIMARY KEY, AUTO\_INCREMENT)

ArtistNo (아티스트번호, INT, 외래키: Artist.ArtistNo 참조)

AlbumTitle (앨범명, VARCHAR(50), NOT NULL)

ReleaseDate (발매일자, DATE, NOT NULL)

Sales (판매량, INT, CHECK(Sales >= 0))

1. 테이블을 생성하시오. (제약조건 포함)

2. SQL 문을 작성하시오.

(1) **아티스트 등록:** Artist 테이블에 다음 데이터를 삽입하시오.

- ('아이유', '2008-09-18', '발라드', 'EDAM엔터테인먼트')
- ('BTS', '2013-06-13', 'K-POP', '하이브')
- ('블랙핑크', '2016-08-08', 'K-POP', 'YG엔터테인먼트')

(2) **앨범 등록:**

- (1, '좋은 날', '2010-12-09', 500000)
- (2, 'MAP OF THE SOUL: 7', '2020-02-21', 4300000)
- (3, 'THE ALBUM', '2020-10-02', 1300000)

(3) **조회:**

- 장르가 'K-POP'인 아티스트의 이름과 소속사를 조회.
- 판매량이 1,000,000 이상인 앨범의 제목과 판매량을 조회.
- 아티스트별 총 판매량을 구하시오.  
(출력: 아티스트명, 총 판매량)

(4) **수정:**

- '아이유'의 소속사를 '카카오엔터테인먼트'로 수정하시오.

(5) **삭제:**

- AlbumNo = 1 인 앨범을 삭제하시오.

**확장 문제 (심화)**

- "소속사별 매출현황"  
(소속사, 아티스트 수, 총 판매량)을 조회.

# 골프연습장 관리

<b>GolfMember (회원) 테이블</b> 컬럼명과 제약조건: MNo (회원번호, INT, PRIMARY KEY, AUTO_INCREMENT) MName (회원명, VARCHAR(30), NOT NULL) Phone (전화번호, VARCHAR(13), UNIQUE) JoinDate (가입일자, DATE, NOT NULL) Grade (회원등급, CHAR(1), CHECK(Grade IN ('A','B','C')))	<b>Lesson (강습) 테이블</b> 컬럼명과 제약조건: LNo (강습번호, INT, PRIMARY KEY, AUTO_INCREMENT) MNo (회원번호, INT, 외래키: GolfMember.MNo 참조) Coach (강사명, VARCHAR(30), NOT NULL) StartDate (강습시작일, DATE, NOT NULL) Fee (강습비, INT, CHECK(Fee >= 0))	<b>Usage (연습장 이용내역) 테이블</b> 컬럼명과 제약조건: UNo (이용번호, INT, PRIMARY KEY, AUTO_INCREMENT) MNo (회원번호, INT, 외래키: GolfMember.MNo 참조) UDate (이용일자, DATE, NOT NULL) Time (이용시간(분), INT, CHECK(Time > 0)) Cost (이용요금, INT, CHECK(Cost >= 0))
--	---	--

<p>1. 테이블을 생성하시오. (제약조건 포함)</p> <p>2. SQL 문을 작성하시오.</p> <p><b>(1) 회원 등록:</b> GolfMember 테이블에 다음 데이터를 삽입.</p> <ul style="list-style-type: none"><li>- ('홍길동', '010-1111-2222', '2020-01-01', 'A')</li><li>- ('이순신', '010-3333-4444', '2021-05-10', 'B')</li><li>- ('강감찬', '010-5555-6666', '2022-03-15', 'C')</li></ul> <p><b>(2) 강습 등록:</b> Lesson 테이블에 삽입</p> <ul style="list-style-type: none"><li>- (1, '김프로', '2020-02-01', 300000)</li><li>- (2, '박프로', '2021-06-01', 250000)</li><li>- (3, '이프로', '2022-04-01', 200000)</li></ul> <p><b>(3) 이용내역 등록:</b> Usage 테이블에 삽입</p> <ul style="list-style-type: none"><li>- (1, '2020-02-10', 120, 24000)</li><li>- (2, '2021-06-15', 90, 18000)</li><li>- (3, '2022-04-20', 150, 30000)</li></ul>	<p><b>(4) 조회:</b></p> <ul style="list-style-type: none"><li>- A등급 회원의 이름, 전화번호, 가입일자를 조회.</li><li>- 강습비가 250000 이상인 강습 내역을 조회.</li><li>- 회원별 총 이용요금을 집계. (출력: 회원명, 총요금)</li></ul> <p><b>(5) 수정:</b></p> <ul style="list-style-type: none"><li>- 이순신 회원의 등급을 'A'로 수정</li></ul> <p><b>(5) 삭제:</b></p> <ul style="list-style-type: none"><li>- MNo = 3 인 회원을 삭제</li></ul> <p><b>확장 문제 (심화)</b></p> <ul style="list-style-type: none"><li>- "등급별 통계" (등급, 회원 수, 평균 강습비, 총 이용요금)을 조회</li></ul>
---	---