

# 따릉이 운영을 위한 데이터베이스 설계

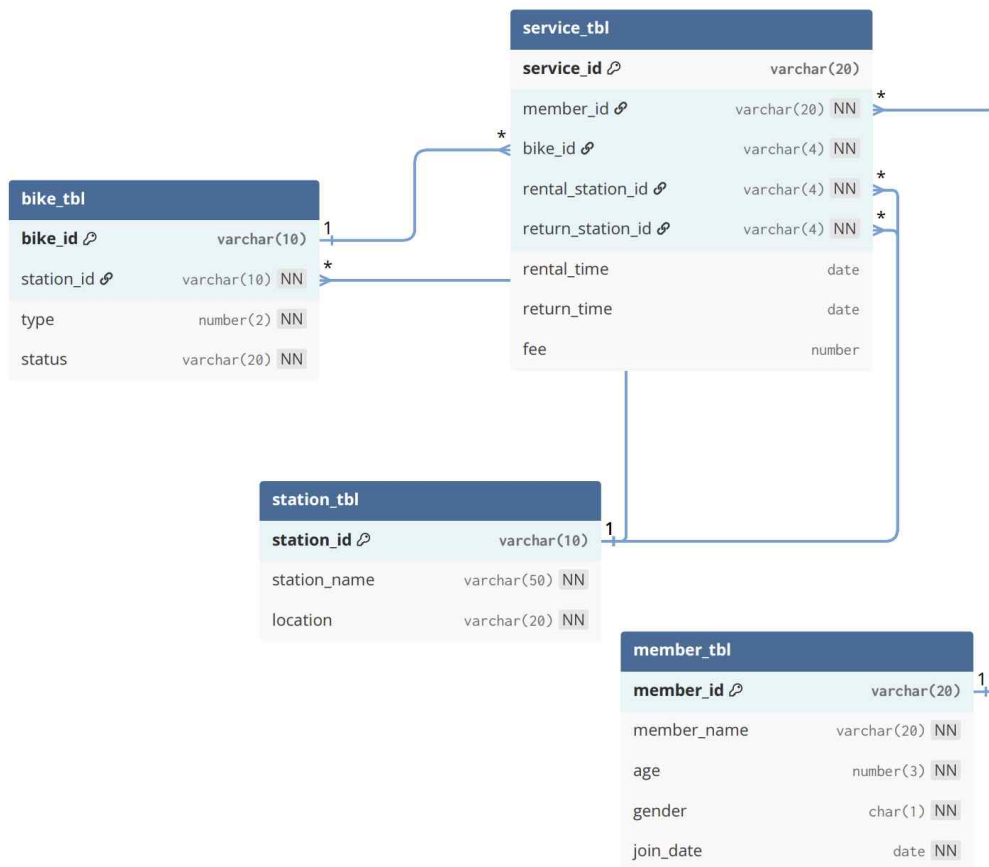
윤태민, 이성빈

## 1. 프로젝트 목표

- 따릉이 운영에 필요한 4개의 테이블 설계
- SQL DDL을 이용한 테이블 설계
- 다양한 분석 시나리오를 활용한 결과 도출

## 2. 데이터베이스 설계

- 개체-관계 다이어그램 (ERD)



- 전체 테이블 종류

테이블 이름	station_tbl	member_tbl	bike_tbl	service_tbl
테이블 설명	대여소 정보	회원 정보	자전거 정보	이용내역

- 테이블 명세서

**station\_tbl**

컬럼명	컬럼ID	타입/길이	N/N	PK	FK	제약조건
대여소번호	station_id	VARCHAR2(10)	O	O		
대여소이름	station_name	VARCHAR2(50)	O			
지역	location	VARCHAR2(20)	O			

**member\_tbl**

컬럼명	컬럼ID	타입/길이	N/N	PK	FK	제약조건
회원ID	member_id	VARCHAR2(20)	O	O		
이름	member_name	VARCHAR2(20)	O			
나이	age	NUMBER(3)	O			
성별	gender	CHAR(1)	O			CHECK (gender IN ('M', 'W'))
가입일	join_date	DATE	O			Default sysdate

**bike\_tbl**

컬럼명	컬럼ID	타입/길이	N/N	PK	FK	제약조건
자전거번호	bike_id	VARCHAR2(10)	O	O		
대여소번호	station_id	VARCHAR2(10)	O		O	station 테이블의 station_id 참조
구분	type	NUMBER(2)	O			CHECK (type IN (10, 20))
자전거상태	status	VARCHAR2(20)	O			CHECK (status IN ('이용가능', '수리중', '고장'))

**service\_tbl**

컬럼명	컬럼ID	타입/길이	N/N	PK	FK	제약조건
이용번호	service_id	VARCHAR2(20)	O	O		
회원ID	member_id	VARCHAR2(20)	O		O	members테이블의 member_id 참조
자전거번호	bicycle_id	VARCHAR2(4)	O		O	bike 테이블의 bike_id 참조
대여대여소	rental_station_id	VARCHAR2(4)	O		O	station 테이블의 station_id 참조
반납대여소	return_station_id	VARCHAR2(4)	O		O	station 테이블의 station_id 참조
대여시간	rental_time	DATE	O			
반납시간	return_time	DATE	O			
회원권	fee	NUMBER	O			일반=1000, 프리미엄=2000

### 3. 데이터베이스 구축

#### 1) station\_tbl

- 테이블생성 (DDL)

```
CREATE TABLE station_tbl (  
    station_id VARCHAR2(10) PRIMARY KEY,  
    station_name VARCHAR2(50) NOT NULL,  
    location VARCHAR2(20) NOT NULL  
);
```

- 데이터입력 (DML)

```
INSERT INTO station_tbl VALUES ( 'S101', '국회의사당 본관', '영등포구' );  
INSERT INTO station_tbl VALUES ( 'S102', '국민일보 앞', '영등포구' );  
INSERT INTO station_tbl VALUES ( 'S103', '여의도역 2번출구 앞', '영등포구' );  
INSERT INTO station_tbl VALUES ( 'S104', '63스퀘어', '영등포구' );  
INSERT INTO station_tbl VALUES ( 'S201', '마포구청역', '마포구' );  
INSERT INTO station_tbl VALUES ( 'S202', '홍대입구역사거리', '마포구' );  
INSERT INTO station_tbl VALUES ( 'S203', '연남동주민센터앞', '마포구' );  
INSERT INTO station_tbl VALUES ( 'S301', '롯데월드타워', '송파구' );  
INSERT INTO station_tbl VALUES ( 'S302', '방이삼거리', '송파구' );  
INSERT INTO station_tbl VALUES ( 'S303', '홈플러스 잠실점', '송파구' );  
COMMIT;
```

- 생성된 테이블

STATION_ID	STATION_NAME	LOCATION
S101	국회의사당 본관	영등포구
S102	국민일보 앞	영등포구
S103	여의도역 2번출구 앞	영등포구
S104	63스퀘어	영등포구
S201	마포구청역	마포구
S202	홍대입구역사거리	마포구
S203	연남동주민센터앞	마포구
S301	롯데월드타워	송파구
S302	방이삼거리	송파구
S303	홈플러스 잠실점	송파구

## 2) member\_tbl

- 테이블생성 (DDL)

```
CREATE TABLE member_tbl (  
    member_id VARCHAR2(20) PRIMARY KEY,  
    member_name VARCHAR2(20) NOT NULL,  
    age NUMBER(3) NOT NULL,  
    gender CHAR(1) NOT NULL CHECK (gender IN ('M', 'F')),  
    join_date DATE DEFAULT SYSDATE NOT NULL  
);
```

- 데이터입력 (DML)

```
INSERT INTO member_tbl VALUES ( 'M1001', '진희', '29', 'F', '2020-05-13' );  
INSERT INTO member_tbl VALUES ( 'M1002', '도영', '21', 'M', '2020-05-28' );  
INSERT INTO member_tbl VALUES ( 'M1003', '이안', '17', 'M', '2021-01-15' );  
.  
.  
INSERT INTO member_tbl VALUES ( 'M1014', '현진', '42', 'M', '2025-09-26' );  
INSERT INTO member_tbl VALUES ( 'M1015', '우주', '25', 'M', '2025-10-24' );  
COMMIT;
```

- 생성된 테이블

MEMBER_ID	MEMBER_NAME	AGE	GENDER	JOIN_DATE
M1001	진희	29	F	2020-05-13
M1002	도영	21	M	2020-05-28
M1003	이안	17	M	2021-01-15
M1004	홍민	44	M	2021-09-29
M1005	서준	33	M	2022-06-26
M1006	윤서	35	F	2023-03-19
M1007	동원	38	M	2023-04-03
M1008	승민	31	M	2023-07-10
M1009	지민	42	M	2023-11-16
M1010	이서	19	F	2024-04-18
M1011	윤아	26	F	2024-07-22
M1012	소민	30	F	2024-08-30
M1013	지현	33	F	2025-02-09
M1014	현진	42	M	2025-09-26
M1015	우주	25	M	2025-10-24

### 3) bike\_tbl

- 테이블생성 (DDL)

```
CREATE TABLE bike_tbl (  
    bike_id VARCHAR2(10) PRIMARY KEY,  
    station_id VARCHAR2(10) NOT NULL REFERENCES station_tbl(station_id),  
    type NUMBER(2) NOT NULL CHECK (type IN (10, 20)),  
    status VARCHAR2(20) NOT NULL CHECK (status IN ('이용가능', '수리중', '고장'))  
);
```

- 데이터입력 (DML)

```
INSERT INTO bike_tbl VALUES ( 'B01', 'S101', '10', '이용가능' );  
INSERT INTO bike_tbl VALUES ( 'B02', 'S101', '20', '고장' );  
.  
.  
INSERT INTO bike_tbl VALUES ( 'B27', 'S104', '20', '수리중' );  
INSERT INTO bike_tbl VALUES ( 'B28', 'S303', '20', '이용가능' );  
INSERT INTO bike_tbl VALUES ( 'B29', 'S301', '10', '이용가능' );  
INSERT INTO bike_tbl VALUES ( 'B30', 'S103', '20', '이용가능' );  
COMMIT;
```

- 생성된 테이블

BIKE_ID	STATION_ID	TYPE	STATUS
B01	S101	10	이용가능
B02	S101	20	고장
B03	S103	20	이용가능
B04	S301	20	수리중
B05	S301	10	이용가능
B06	S102	20	이용가능
B07	S201	20	이용가능
B08	S104	20	고장
B09	S101	20	이용가능
B10	S103	20	이용가능
B11	S201	20	이용가능
B12	S101	10	이용가능
B13	S104	20	이용가능
B14	S202	20	이용가능
B15	S102	20	이용가능
B16	S301	20	이용가능
B17	S101	20	이용가능
B18	S104	20	수리중

B19	S104	20	이용가능
B20	S201	10	이용가능
B21	S203	20	이용가능
B22	S101	20	이용가능
B23	S101	20	이용가능
B24	S301	20	이용가능
B25	S104	20	이용가능
B26	S303	20	이용가능
B27	S104	20	수리중
B28	S303	20	이용가능
B29	S301	10	이용가능
B30	S103	20	이용가능

#### 4) service\_tbl

- 테이블생성 (DDL)

```
CREATE TABLE service_tbl (
    service_id VARCHAR2(20) PRIMARY KEY,
    member_id VARCHAR2(20) NOT NULL REFERENCES member_tbl(member_id),
    bike VARCHAR2(4) NOT NULL REFERENCES bike_tbl(bike_id),
    rental_station_id VARCHAR2(4) NOT NULL REFERENCES station_tbl(station_id),
    return_station_id VARCHAR2(4) NOT NULL REFERENCES station_tbl(station_id),
    rental_time VARCHAR2(5),
    return_time VARCHAR2(5),
    fee number(5)
);
```

- 데이터입력 (DML)

```
INSERT INTO service_tbl VALUES
( 'SV250929-0001', 'M1004', 'B06' , 'S102', 'S101', '06:38', '07:15', 1000 );
INSERT INTO service_tbl VALUES
( 'SV250929-0002', 'M1014', 'B15' , 'S102', 'S201', '07:02', '07:40', 2000 );
.
.
( 'SV250929-0019', 'M1005', 'B10' , 'S103', 'S102', '18:19' , '18:34', 1000 );
INSERT INTO service_tbl VALUES
( 'SV250929-0020', 'M1013', 'B07' , 'S201', 'S301', '19:30' , '21:02', 2000 );
COMMIT;
```

- 생성된 테이블

SERVICE_ID	MEMBER_ID	BIKE_ID	RENTAL_STATION_ID	RETURN_STATION_ID	RENTAL_TIME	RETURN_TIME	FEE
SV250929-0001	M1004	B06	S102	S101	6:38	7:15	1000
SV250929-0002	M1014	B15	S102	S201	7:02	7:40	2000
SV250929-0003	M1006	B21	S203	S101	7:11	7:32	1000
SV250929-0004	M1012	B26	S303	S101	7:15	7:55	1000
SV250929-0005	M1005	B01	S101	S103	7:28	7:45	1000
SV250929-0006	M1011	B30	S103	S202	7:29	8:08	1000
SV250929-0007	M1010	B14	S202	S102	8:32	8:58	1000
SV250929-0008	M1007	B28	S303	S201	8:47	9:15	1000
SV250929-0009	M1015	B01	S103	S303	10:32	12:35	2000
SV250929-0010	M1009	B25	S104	S202	12:04	12:30	1000
SV250929-0011	M1002	B29	S301	S303	12:36	13:02	1000
SV250929-0012	M1002	B29	S303	S302	13:12	13:53	1000
SV250929-0013	M1008	B24	S301	S202	17:49	19:05	1000
SV250929-0014	M1015	B19	S104	S201	17:55	19:30	2000
SV250929-0015	M1001	B30	S202	S303	17:59	18:46	1000
SV250929-0016	M1003	B16	S301	S203	18:03	19:53	1000
SV250929-0017	M1007	B05	S301	S103	18:07	20:19	2000
SV250929-0018	M1010	B29	S302	S202	18:18	19:22	1000
SV250929-0019	M1005	B10	S103	S102	18:19	18:34	1000
SV250929-0020	M1013	B07	S201	S301	19:30	21:02	2000

## 4 데이터활용 및 분석

### 1. 출퇴근 시간에 따릉이를 모두 이용한 사람 조회하기

- 데이터조회 (DML)

-- 1. 출퇴근 시간에 이용하는 회원 정보 조회하기 (이용내역 join 회원정보)

```
select *  
from service_tbl s  
join member_tbl m  
on s.member_id = m.member_id;
```

-- 2. 출근(07시 ~ 09시), 퇴근(18시 ~ 19시) 시간에 이용하는 회원 정보 조회하기

```
select s.member_id, m.member_name, s.rental_time, s.return_time  
from service_tbl s  
join member_tbl m  
on s.member_id = m.member_id  
where s.rental_time between '07:00' and '09:00' or  
       s.rental_time between '18:00' and '19:00'  
order by rental_time;
```

-- 3. 출근(07시 ~ 09시), 퇴근(18시 ~ 19시) 시간 모두 이용하는 회원 정보 조회하기

```
select s.member_id, m.member_name  
from service_tbl s  
join member_tbl m  
on s.member_id = m.member_id  
where s.rental_time between '07:00' and '09:00'
```

intersect

```
select s.member_id, m.member_name  
from service_tbl s  
join member_tbl m  
on s.member_id = m.member_id  
where s.rental_time between '18:00' and '19:00';
```

- 결과

MEMBER_ID	MEMBER_NAME	RENTAL_TIME	RETURN_TIME
M1014	현진	07:02	07:40
M1006	윤서	07:11	07:32
M1012	소민	07:15	07:55
M1005	서준	07:28	07:45
M1011	윤아	07:29	08:08



M1010	이서	08:32	08:58
M1007	동원	08:47	09:15
M1003	이안	18:03	19:53
M1007	동원	18:07	20:19
M1010	이서	18:18	19:22
M1005	서준	18:19	18:34

MEMBER_ID	MEMBER_NAME
M1005	서준
M1007	동원
M1010	이서

## 2. 성별&연령대 별 따릉이 이용자 수 조회하기

- 데이터조회 (DML)

-- 연령대와 성별에 따른 이용자 수 조회하기 (이용내역 join 회원)

```
select *
from service_tbl s
join member_tbl m
on s.member_id = m.member_id;

select trunc(m.age/10)*10 || '대' "Age_group", m.gender "Gender" , count(*)"Count"
from service_tbl s
join member_tbl m
on s.member_id = m.member_id
group by trunc(m.age/10)*10, m.gender
order by 1;
```

- 결과

Age_group	Gender	Count
10대	F	2
10대	M	1
20대	F	2
20대	M	4
30대	M	5
30대	F	3
40대	M	3

### 3. 고객별 따름이 이용시간 조회하기, 따름이 이용시간에 따른 최종요금 출력하기

- 데이터조회 (DML)

--멤버와 이용내역 테이블 조회하기

```
select *  
from service_tbl se  
join member_tbl m  
on se.member_id = m.member_id;
```

--서브쿼리

```
select member_id, rental_time, return_time,  
round((TO_DATE(return_time, 'HH24:MI') - TO_DATE(rental_time, 'HH24:MI')) * 24 *  
60,1) usetime  
from service_tbl;
```

--서브쿼리와 합치기 (멤버별 이용시간 출력하기)

```
select m.member_name, se.usetime || '분'  
from (select member_id, rental_time, return_time,  
round((TO_DATE(return_time, 'HH24:MI') - TO_DATE(rental_time, 'HH24:MI')) * 24 *  
60,1) as usetime  
from service_tbl) se  
join member_tbl m  
on se.member_id = m.member_id;
```

--이용시간에 따른 최종요금 출력하기 (초과시 +1000원부가)

```
select m.member_name, se.usetime || '분' , se.fee || '원' as 기본요금,  
case when se.fee = 1000 AND se.usetime > 60 then se.fee + 1000  
      when se.fee = 2000 AND se.usetime > 120 then se.fee + 1000  
      else se.fee  
      end || '원' as 최종요금  
from (select member_id, rental_time, return_time, fee,  
round((TO_DATE(return_time, 'HH24:MI') - TO_DATE(rental_time, 'HH24:MI')) * 24 *  
60,1) as usetime  
from service_tbl) se  
join member_tbl m  
on se.member_id = m.member_id;
```

- 결과

MEMBER_NAME	USE_TIME
홍민	37분
현진	38분

윤서	21분
소민	40분
서준	1분
윤아	39분
이서	26분
동원	28분
우주	123분
지민	26분
도영	26분
도영	41분
승민	76분
우주	95분
진희	47분
이안	110분
동원	132분
이서	64분
서준	15분
지현	92분

MEMBER_NAME	USE_TIME	기본요금	최종요금
홍민	37분	1000원	1000원
현진	38분	2000원	2000원
윤서	21분	1000원	1000원
소민	40분	1000원	1000원
서준	17분	1000원	1000원
윤아	39분	1000원	1000원
이서	26분	1000원	1000원
동원	28분	1000원	1000원
우주	123분	2000원	3000원
지민	26분	1000원	1000원
도영	26분	1000원	1000원
도영	41분	1000원	1000원
승민	76분	1000원	2000원
우주	95분	2000원	2000원
진희	47분	1000원	1000원
이안	110분	1000원	2000원
동원	132분	2000원	3000원
이서	64분	1000원	2000원
서준	15분	1000원	1000원
지현	92분	2000원	2000원

#### 4. 대여소별 이용 가능한 따릉이 수 조회하기 (순위출력)

- 데이터조회 (DML)

--자전거 테이블에서 이용가능한 자전거만 조회하기

```
select *  
from station_tbl st  
join bike_tbl b  
on st.station_id = b.station_id and b.status = '이용가능';
```

--대여소별 이용가능한 자전거의 개수 출력하기

```
select station_name, count(*)  
from station_tbl st  
join bike_tbl b  
on st.station_id = b.station_id and b.status = '이용가능'  
group by station_name;
```

--대여소별 이용가능한 자전거가 많은 순서대로 순위매기기

```
select st.station_id, st.station_name, count(*) as bike_count,  
       dense_rank() over (order by count(*) desc) as rnk  
from bike_tbl b  
join station_tbl st  
on b.station_id = st.station_id  
where b.status = '이용가능'  
group by st.station_id, st.station_name;
```

- 결과

STATION_ID	STATION_NAME	BIKE_COUNT	RNK
S101	국회의사당 본관	6	1
S301	롯데월드타워	4	2
S104	63스퀘어	3	3
S201	마포구청역	3	3
S103	여의도역 2번출구 앞	3	3
S102	국민일보 앞	2	4
S303	홈플러스 잠실점	2	4
S202	홍대입구역사거리	1	5
S203	연남동주민센터앞	1	5