Like 문 이용해서 출력하기

```
SELECT ANIMAL_ID,NAME
FROM ANIMAL_INS
WHERE   INTAKE_CONDITION NOT LIKE "Aged"
ORDER BY ANIMAL_ID;
```

아이디와 이름만 출력하는 부분

```
SELECT ANIMAL_ID,NAME
FROM ANIMAL_INS
ORDER BY  ANIMAL_ID;
```

여러 기준으로 정렬하기

```
SELECT ANIMAL_ID    ,NAME    ,DATETIME
FROM ANIMAL_INS
ORDER BY NAME,  DATETIME DESC;
```

상위 n개의 레코드

```
SELECT NAME
FROM ANIMAL_INS
ORDER BY DATETIME
LIMIT 1;
```

최댓값 구하기

```
SELECT DATETIME AS 시간
FROM ANIMAL_INS
ORDER BY DATETIME DESC
LIMIT 1;
```

최솟값 구하기

```
SELECT DATETIME AS 시간
FROM ANIMAL_INS
ORDER BY DATETIME
LIMIT 1;
```

동물 수 구하기

```
SELECT count(*)
from ANIMAL_INS;
```

중복 제거하기

```
SELECT count(DISTINCT NAME)
FROM ANIMAL_INS;
```

개와 고양이는 몇마리 있을까

```
select ANIMAL_TYPE, count(*) as "count"
from ANIMAL_INS
group by ANIMAL_TYPE
order by ANIMAL_TYPE;
```

동명 동물 수 찾기

```
select NAME,count(*)    as "COUNT"
from ANIMAL_INS
group by NAME
having count(*) >= 2 and name is  not null
order by NAME
```

입양 시각 구하기(1)

```
SELECT hour(DATETIME) as HOUR, COUNT(*) as COUNT
FROM ANIMAL_OUTS
where hour(DATETIME) between 9 and 19
GROUP BY hour(DATETIME)
ORDER BY hour(DATETIME);
```

입양 시각 구하기(2)

```
SET @hour = -1;
SELECT (@hour := @hour + 1) AS 'HOUR',
( SELECT COUNT(DATETIME)  AS 'COUNT'
 FROM ANIMAL_OUTS
 WHERE HOUR(DATETIME) = @hour )
 FROM ANIMAL_OUTS
 WHERE @hour < 23
```

이름이 없는 동물의 아이디

```
SET @hour = -1;
SELECT (@hour := @hour + 1) AS 'HOUR',
( SELECT COUNT(DATETIME)  AS 'COUNT'
 FROM ANIMAL_OUTS
 WHERE HOUR(DATETIME) = @hour )
 FROM ANIMAL_OUTS
 WHERE @hour < 23
```

이름이 있는 동물의 아이디

```
SELECT ANIMAL_ID
FROM ANIMAL_INS
WHERE NAME IS NOT NULL
ORDER BY ANIMAL_ID;
```

NULL 처리하기

```
SELECT ANIMAL_TYPE  ,IFNULL(NAME, "No name")    ,SEX_UPON_INTAKE
FROM ANIMAL_INS
ORDER BY ANIMAL_ID;
```

없어진 기록 찾기

```
select B.ANIMAL_ID, B.NAME
from ANIMAL_INS A RIGHT join ANIMAL_OUTS B
on A.ANIMAL_ID = B.ANIMAL_ID
WHERE A.ANIMAL_ID IS NULL
```

있었는데요 없었습니다

```
SELECT INS.ANIMAL_ID    ,INS.NAME
FROM ANIMAL_INS INS INNER JOIN ANIMAL_OUTS OUTS
ON INS.ANIMAL_ID = OUTS.ANIMAL_ID
WHERE INS.DATETIME > OUTS.DATETIME
ORDER BY INS.DATETIME
```

오랜 기간 보호한 동물(1)

```
SELECT INS.NAME ,INS.DATETIME
FROM ANIMAL_INS INS LEFT JOIN ANIMAL_OUTS OUTS
ON INS.ANIMAL_ID = OUTS.ANIMAL_ID
WHERE OUTS.DATETIME IS NULL
ORDER BY INS.DATETIME
LIMIT 3
```

보호소에서 중성화한 동물

```
SELECT ins.ANIMAL_ID    ,ins.ANIMAL_TYPE,   ins.NAME
from ANIMAL_INS ins inner join ANIMAL_outs outs
on ins.animal_id = outs.animal_id
where ins.SEX_UPON_INTAKE like "intact%"
and (outs.SEX_UPON_OUTCOME like "Neutered%"
    OR outs.SEX_UPON_OUTCOME like "Spayed%"
    )
order by ins.animal_id
```

루시와 엘라 찾기

```
select ANIMAL_ID,   NAME,   SEX_UPON_INTAKE
from ANIMAL_INS
where name in ("Lucy", "Ella", "Pickle", "Rogan", "Sabrina", "Mitty")
order by animal_id
```

이름에 el이 들어가는 동물 찾기

```
select ANIMAL_ID    ,NAME
from ANIMAL_INS
where name like "%el%"
and animal_type = "Dog"
order by name
```

중성화 여부 파악하기

```
select ANIMAL_ID,   NAME,
case when (SEX_UPON_INTAKE like "Intact%") then "X" else "O" end
as 중성화
from ANIMAL_INS
```

오랜 기간 보호한 동물(2)

```
select ins.ANIMAL_ID,   ins.NAME
from ANIMAL_INS ins inner join ANIMAL_OUTS outs
on ins.animal_id = outs.animal_id
order by (outs.datetime-ins.datetime) desc
limit 2
```

DATETIME에서 DATE로 형 변환

```
select ANIMAL_ID,   NAME,   DATE_FORMAT(datetime,"%Y-%m-%d") as 날짜
from ANIMAL_INS
order by animal_id
```