



가독성 있는 SQL 코드 컨벤션



자료배경

<https://velog.io/@datarian/good-sql-code> 을 다수 참고하였음을 밝힙니다.



쿼리 선호도 조사 (SELECT)

SELECT

1.

```
SELECT  
    column1,  
    column2
```

2.

```
SELECT  
    column1  
    , column2
```

3.

```
SELECT column1, column2
```

4.

```
SELECT column1,  
    column2
```



쿼리 선호도 조사 (WHERE)

WHERE

1.

WHERE

a = true AND b = true

2.

WHERE a = true

AND b = true



쿼리 선호도 조사 (INDENT)

들여쓰기(indent)

```
1.
SELECT first_name AS fn
FROM staff AS s1
JOIN students AS s2
  ON s2.mentor_id = s1.staff_num

2.
SELECT first_name AS fn
FROM staff AS s1
JOIN students AS s2
  ON s2.mentor_id = s1.staff_num

3.
SELECT first_name AS fn
FROM   staff AS s1
JOIN   students AS s2
  ON   s2.mentor_id = s1.staff_num
```

JOIN



쿼리 선호도 조사 (JOIN)

```
1.
SELECT column1
FROM TABLE_A
LEFT JOIN TABLE_B ON TABLE_A.id = TABLE_B.id AND TABLE_A.id2 = TABLE_B.id2

2.
SELECT column1
FROM TABLE_A
LEFT JOIN TABLE_B ON TABLE_A.id = TABLE_B.id
    AND TABLE_A.id2 = TABLE_B.id2

3.
SELECT column1
FROM TABLE_A
LEFT JOIN TABLE_B
ON TABLE_A.id = TABLE_B.id
AND TABLE_A.id2 = TABLE_B.id2

4.
SELECT column1
FROM TABLE_A
LEFT JOIN TABLE_B
ON TABLE_A.id = TABLE_B.id
    AND TABLE_A.id2 = TABLE_B.id2

5.
SELECT column1
FROM TABLE_A
LEFT JOIN TABLE_B
ON
    TABLE_A.id = TABLE_B.id
    AND TABLE_A.id2 = TABLE_B.id2
```



코드 컨벤션이란?

코딩 컨벤션이란? 코딩 컨벤션은 읽고, 관리하기 쉬운 코드를 작성하기 위한 코딩 스타일 규약이다.

들여쓰기 (indent)

띄어쓰기 (, 이후 한 칸 / 중괄호, 소괄호 이후 한 칸 등)

변수 작성

주석 규칙





코드 컨벤션을 잘 지켜야 하는 이유는 무엇일까?

가독성.

각 언어마다 컨벤션이 있음.(파이썬의 경우 [PEP8](#))

다른 언어에 비해 SQL은 각 DB별로 성격도 다르고 배우는 곳에 따라 다른 스타일을 가진다.
그럼에도 어느정도 정해진 규칙은 있고 최대한 그것을 지켜나가면 좋다.

반드시 지킬 필요는 없다. 마치 젓가락질 못해도 밥은 먹는 것처럼.

하지만 앞으로 혼자서 계속 일할 것이 아니라면 습관을 잘 들이는 것이 좋다.



1. 예약어는 대문자로

SELECT, FROM, GROUP BY, HAVING, AS, AND 와 같은 SQL 예약어들을 대문자로 쓰는 습관

```
SELECT tip, total_bill  
FROM tips
```

```
select tip, total_bill  
from tips
```



2. 행갈이를 자주 하자

SELECT, FROM, GROUP BY 와 같이 다른 절이 한 라인에 있는 코드를 작성하지 않기.

각 라인의 첫 번째 예약어들만 눈으로 쓱 훑어도 작성자의 의도와 코드의 구조를 파악할 수 있다.

```
SELECT day
      , time
      , SUM(total_bill)
FROM tips
GROUP BY day, time
```

```
SELECT day, time, SUM(total_bill) FROM tips GROUP BY day, time
```



3. 행갈이를 더 자주 하자

```
SELECT day
      , time
      , SUM(total_bill)
FROM tips
WHERE sex = 'Female'
AND smoker = 'Yes'
GROUP BY day
      , time
```

```
SELECT day, time, SUM(total_bill)
FROM tips
WHERE sex = 'Female' AND smoker = 'Yes'
GROUP BY day, time
```



4. 주석을 쓰자

```
SELECT AVG(sales) avg_sales
FROM (
  -- 요일별 매출액 합계 계산
  SELECT day -- 일별 X 요일별 O
         , SUM(total_bill) sales
  FROM tips
  WHERE sex = 'Female' -- 결제자의 성별이 여성인 경우
  GROUP BY day
) daily_sales
```



5. Alias를 잘 쓰자

CTE(Common Table Expression) 및 서브쿼리에서 a,b,t1,t2 등으로 표현을 최대한 덜하기.

지금 당장은 편할 수 있지만, 6개월 뒤 다시 쿼리를 봐야한다면 조금 고통스러울 수 있음.

테이블 Alias는 단어별 앞글자를 따서 작성하는편

네이밍 컨벤션

camelCase(카멜 케이스)

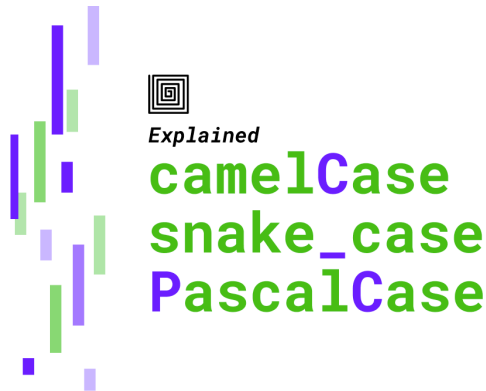
: 문자열의 첫 문자를 제외하고 단어의 첫 글자마다 대문자로 표현하는 방식

snake_case(스네이크 케이스)

: 단어와 단어 사이를 언더스코어(_)를 이용하여 구분

PascalCase(파스칼 케이스)

: 카멜 케이스와 유사하지만 첫 문자도 대문자로 표현



끝

즐거운 SQL 사용합시다

