

쿼리 안의 또 다른 쿼리  
: 서브쿼리

# 1. 서브쿼리 살펴보기

## (1) 서브쿼리란

- 하나의 쿼리 안에 있는 또 다른 SELECT 문
- 괄호로 둘러 싸여 있음
- 보조쿼리 또는 하위 쿼리라고도 함

# 1. 서브쿼리 살펴보기

## (2) 서브쿼리는 왜 필요한가

- 연도별 1위 영화

	YEAR(a.release_date)	movie_name	sale_amt
▶	2004	태극기 휘날리며	15687180500
	2005	웰컴 투 동막골	40328508500
	2006	괴물	66715713300
	2007	디워	49339934700
	2008	좋은 놈, 나쁜 놈, 이상한 놈	43747552000
	2009	해운대	81025004000
	2009	아바타	81455728000
	2011	트랜스포머 3	74840681500
	2012	도둑들	93664808500
	2013	7번방의 선물	91431914670
	2014	명량	135748398910
	2015	베테랑	105168155250
	2016	부산행	93178283048
	2017	택시운전사	95853645649
	2018	신과함께: 인과 연	102666146909
	2019	극한직업	139651845516

- 연도별 1위 영화들의 평균 매출액보다 매출액이 큰 영화만 조회하려면?

```
SELECT YEAR(a.release_date), a.movie_name,  
       a.sale_amt, AVG(a.sale_amt)
```

```
FROM box_office a
```

```
WHERE a.ranks = 1
```

```
GROUP BY 1, 2
```

```
ORDER BY 1;
```

	YEAR(a.release_date)	movie_name	sale_amt	AVG(a.sale_amt)
▶	2004	태극기 휘날리며	15687180500	15687180500
	2005	웰컴 투 동막골	40328508500	40328508500
	2006	괴물	66715713300	66715713300
	2007	디워	49339934700	49339934700
	2008	좋은 놈, 나쁜 놈, 이상한 놈	43747552000	43747552000
	2009	아바타	81455728000	81455728000
	2009	해운대	81025004000	81025004000
	2011	트랜스포머 3	74840681500	74840681500
	2012	도둑들	93664808500	93664808500
	2013	7번방의 선물	91431914670	91431914670
	2014	명량	135748398910	135748398910
	2015	베테랑	105168155250	105168155250
	2016	부산행	93178283048	93178283048
	2017	택시운전사	95853645649	95853645649
	2018	신과함께: 인과 연	102666146909	102666146909
	2019	극한직업	139651845516	139651845516

# 1. 서브쿼리 살펴보기

## (3) 서브쿼리의 유형

- 스칼라(scalar) 서브쿼리 : SELECT 절에 있는 서브쿼리
- 파생(derived) 테이블 : FROM 절에 있는 서브쿼리
- WHERE 절의 서브쿼리

## 2. SELECT와 FROM 절의 서브쿼리

### (1) 스칼라 서브쿼리

- 메인쿼리의 SELECT 절에 오는 서브쿼리
- 하나의 칼럼이나 표현식 역할
- 단일 로우, 단일 값 반환
- 메인쿼리와 조인 하는 것이 일반적임

```
SELECT 칼럼1, 칼럼2, ...
```

```
    (SELECT ...
```

```
        FROM ...
```

```
        WHERE ...
```

```
    ) [AS] 별칭1
```

```
FROM ...
```

```
WHERE ...;
```

## 2. SELECT와 FROM 절의 서브쿼리

### (2) 파생 테이블

- 메인쿼리의 FROM 절에 오는 서브쿼리
- 하나의 테이블 역할, 별칭 반드시 사용
- 여러 로우, 여러 값 반환 가능
- 메인쿼리와 조인 → 조인 조건은 메인쿼리에서만 가능

```
SELECT 칼럼1, 칼럼2, ...  
  FROM 테이블1 [AS] 별칭1,  
        (SELECT ...  
         FROM ...  
         WHERE ...  
        ) [AS] 별칭2,  
        ...  
WHERE ...;
```

### 3. WHERE 절의 서브쿼리

#### (1) 조건 서브쿼리

- WHERE 절의 조건 비교 값으로 서브쿼리 사용
- 서브쿼리 반환 값이 조건 비교
- 서브쿼리 반환 값은 단일 또는 여러 값 반환 가능

### 3. WHERE 절의 서브쿼리

#### (2) ANY, SOME, ALL 연산자를 사용한 조건 서브쿼리

- ANY(SOME)연산자
  - 비교하려는 칼럼 값이 조건 서브쿼리가 반환하는 여러 값(비교 값) 중 **어느 하나와** 조건을 만족
- ALL 연산자
  - 비교하려는 칼럼 값이 조건 서브쿼리가 반환하는 **모든 값**에 대해 조건을 만족



### 3. WHERE 절의 서브쿼리

#### (3) IN과 EXISTS 연산자를 사용한 조건 서브쿼리

- IN 연산자
  - 비교할 칼럼 값에 비교 대상 값 중 하나라도 포함되어 있다면 모두 조회
- EXISTS 연산자
  - IN 연산자와 유사
  - 비교할 칼럼 값이 비교 대상 값 중에 존재하는지 확인