

DB

6 조인의 개념 및 종류

주요 질문

6.

조인의 개념 및
종류

- ✓ DBMS에서 의미하는 Join의 개념과 종류에 대해서 설명해 보세요.
- ✓ Nested Loop Join 의 Driving Table 이 왜 중요할까요?
- ✓ ANSI Join 에 대해서 설명해 보세요.

조인(Join) 개념

6.

조인의 개념 및 종류

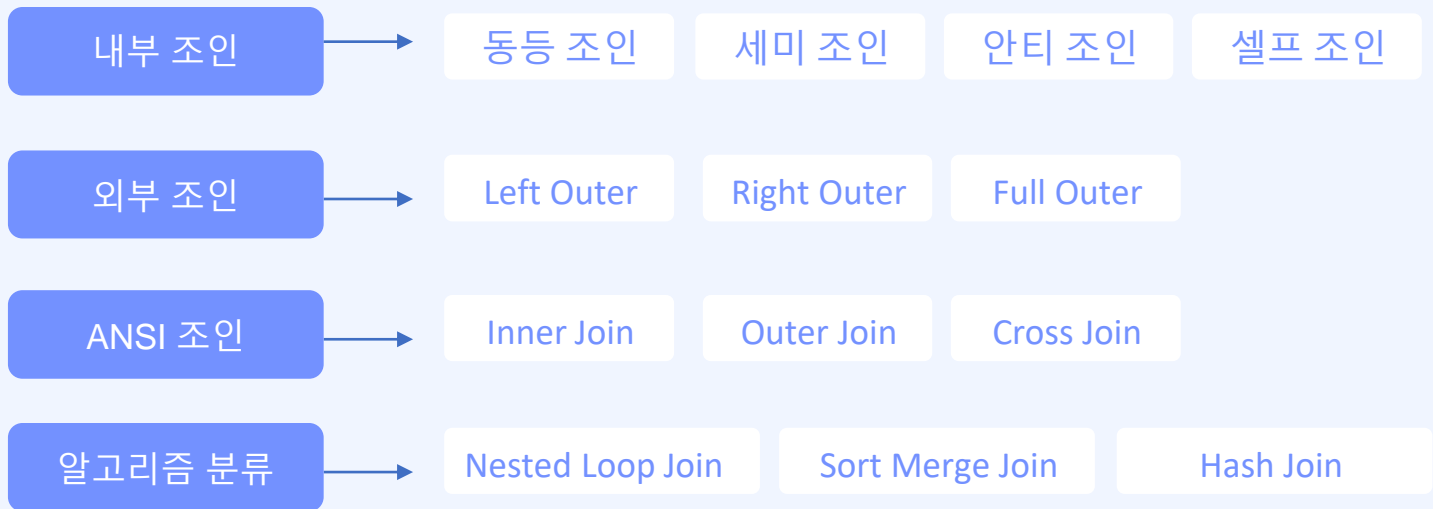
- 서로 관계 있는 데이터가 저장된 다중 테이블의 데이터를 효율적으로 검색하기 위해 서로 연관된 데이터를 비교하여 조회하는 기법
- 두 릴레이션으로 부터 관련된 튜플들을 결합하여 하나의 튜플로 만드는 대표적인 데이터 연결 방법

조인의 종류

6.

조인의 개념 및
종류

대표적인 Join 의 분류



내부 조인 종류

6.

조인의 개념 및
종류

내부 조인 종류

| 조인 종류 | 기본 설명 | 키워드 |
|----------------------|---|-----------------------|
| 동등 조인 (Equi-Join) | Select * from user a, group b where a.group_id=b.group_id | “=” |
| 세미 조인 (Semi-Join) | Select b.* from group b where b.group_id IN (select a.group_id from user a where a.salary>60,000) | EXISTS, IN |
| 안티 조인 (Anti-Join) | Select a.* from user a where NOT EXISTS (select 1 from group b where a.group_id=b.group_id and b.leader_id is null) | NOT EXISTS, NOT IN |
| 셀프 조인 (Self-Join) | Select * from user a, user b where a.user_id < b.user_id and a.group_id=b.group_id and group_id='5032' | |

외부 조인 종류

6.

조인의 개념 및
종류

외부 조인 종류

| 조인 명칭 | 설명 |
|--|---|
| Left Outer Join | Select * from user a, group b where a.group_id = b.group_id (+) |
| Right Outer Join | Select * from user a, group b where a.group_id (+) = b.group_id |
| Full Outer Join (Cartesian Product) | Select * from user a, group b; |

ANSI 조인 종류

6.

조인의 개념 및
종류

ANSI 조인 종류

| 조인 명칭 | 설명 |
|------------------|--|
| Inner Join | Select * From user a INNER JOIN group b ON (a.group_id=b.group_id) Where a.salary>60,000; |
| Left Outer Join | Select * From user a LEFT OUTER JOIN group b ON (a.group_id=b.group_id) Where a.salary>60,000; |
| Right Outer Join | Select * From user a RIGHT OUTER JOIN group b ON (a.group_id=b.group_id) Where a.salary>60,000; |
| Cross Join | Select * From user a CROSS JOIN group b; |

ANSI 조인 결과 (예시)

6.

조인의 개념 및
종류

user

| 123 user_id | ABC name | 123 group_id |
|-------------|----------|--------------|
| 100 | 홍길동 | 400 |
| 101 | 김이동 | 400 |
| 103 | 박진구 | [NULL] |

group

| 123 group_id | ABC group_name |
|--------------|----------------|
| 400 | 기획팀 |
| 402 | 인사팀 |

INNER JOIN

select * from user a inner
join group b on
(a.group_id =b.group_id)

| 123 u | ABC n | 123 g | 123 g | ABC gro |
|-------|-------|-------|-------|---------|
| 100 | 홍길동 | 400 | 400 | 기획팀 |
| 101 | 김이동 | 400 | 400 | 기획팀 |

LEFT OUTER JOIN

select * from user a left outer join
group b on (a.group_id=b.group_id)

| 123 user | ABC name | 123 group | 123 group | ABC group |
|----------|----------|-----------|-----------|-----------|
| 100 | 홍길동 | 400 | 400 | 기획팀 |
| 101 | 김이동 | 400 | 400 | 기획팀 |
| 103 | 박진구 | [NULL] | [NULL] | [NULL] |

CROSS JOIN

select * from user a cross join
group b

| 123 user_id | ABC name | 123 group_id | 123 group_id | ABC group_name |
|-------------|----------|--------------|--------------|----------------|
| 100 | 홍길동 | 400 | 400 | 기획팀 |
| 100 | 홍길동 | 400 | 402 | 인사팀 |
| 101 | 김이동 | 400 | 400 | 기획팀 |
| 101 | 김이동 | 400 | 402 | 인사팀 |
| 103 | 박진구 | [NULL] | 400 | 기획팀 |
| 103 | 박진구 | [NULL] | 402 | 인사팀 |

알고리즘 분류

6.

조인의 개념 및
종류

| 분류 | 설명 |
|------------------|--|
| Nested Loop Join | <ul style="list-style-type: none"> 먼저 선행테이블의 처리범위를 하나씩 액세스하면서 그 추출된 값으로 연결할 테이블을 조인하는 방식 |
| Sort Merge Join | <ul style="list-style-type: none"> 양쪽 테이블의 처리범위를 각자 액세스하여 정렬한 결과를 차례로 scan하면서 연결고리의 조건으로 merge해 가는 방식 |
| Hash Join | <ul style="list-style-type: none"> 해시값을 이용하여 테이블을 조인하는 방식 |

Nested Loop Join

6.

조인의 개념 및
종류

```
select /*+ ordered use_nl(e)*/
```

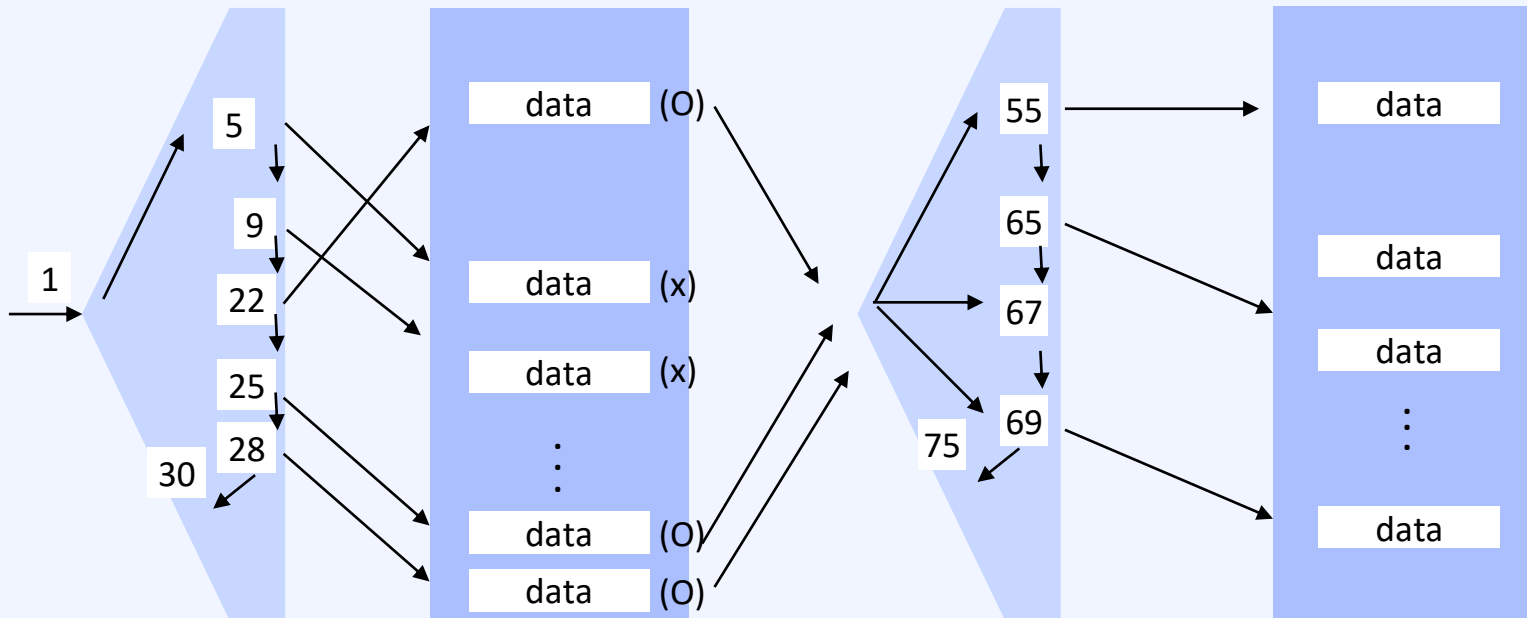
```
* from user a inner join group b on ( a.group_id = b.group_id) where b.group_id like '2%'
```

group_id_index

group

group_id_index

user



Sort Merge Join

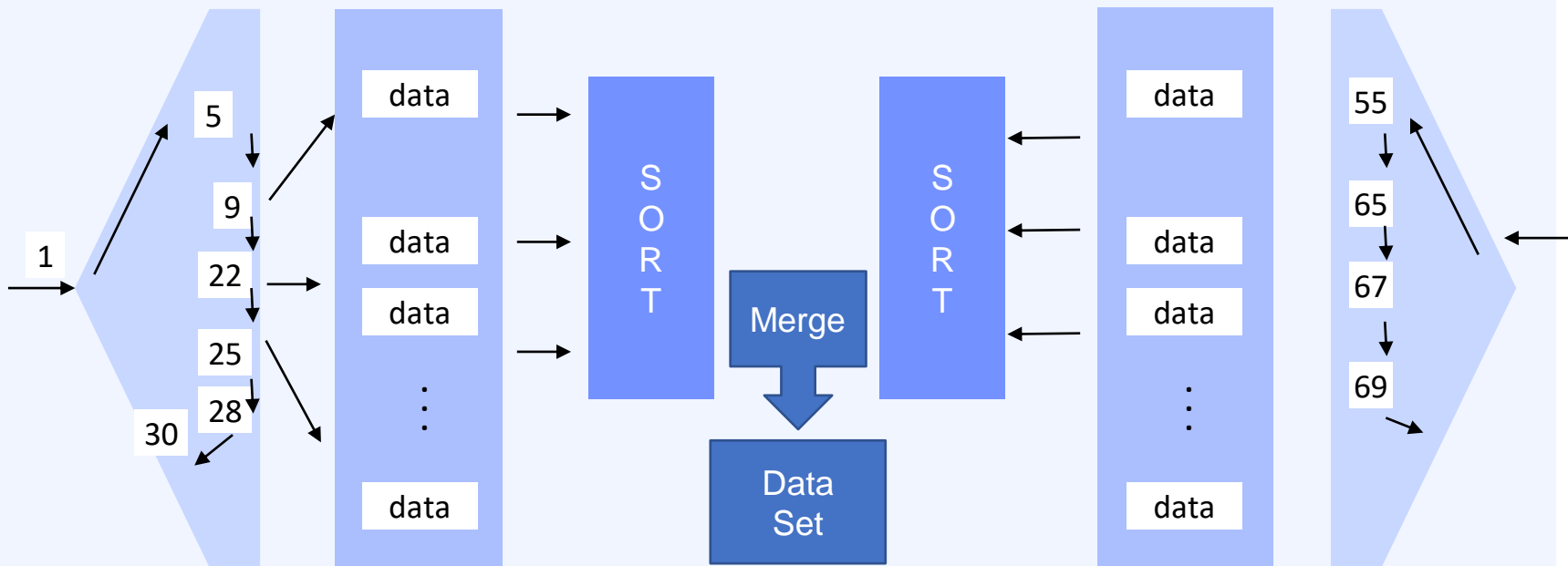
6.

조인의 개념 및
종류

```
select /*+ ordered full(d) use_merge(e)*/
```

```
* from user a inner join group b on ( a.group_name = b.group_name) where b.level > 2 and a.location = '서울'
```

(조인 컬럼에 적당한 인덱스가 없고, 대용량 자료 조인)



Hash Join

6.

조인의 개념 및
종류

```
select /*+ ordered use_hash(e)*/
```

```
* from user a inner join group b on ( a.group_name =b.group_name) where b.group_code = '30';
```

(조인 컬럼에 적당한 인덱스가 없음. CBO만 가능)

