JOIN 개념과 종류

- description : JOIN 개념과 종류
- author : 연목구어
- email : yspark@repia.com
- lastupdate : 2020-03-17

참고 사이트

- https://www.codeproject.com/Articles/33052/Visual-Representation-of-SQL-Joins 사이트 번역본

설명

- 이 문서에서는 시각적 방식으로 SQL 조인과 시각화 된 조인을 작성하는 가장 효율적인 방법에 대해 설명합니다.
- 이것은 SQL JOIN을 시각적으로 설명하는 간단한 기사입니다. 이 기사에서는 두 관계형 테이블에서 데이터를 반환 할 수있는 7 가지 방법에 대해 설명하겠습니다.
- 내가 논의 할 7 개의 조인은 내부 조인, 왼쪽 조인, 오른쪽 조인, 외부 조인, 왼쪽 제외 조인, 오른쪽 제외 조인, 외부 조인 제외, 각각의 예를 제공합니다.

배경

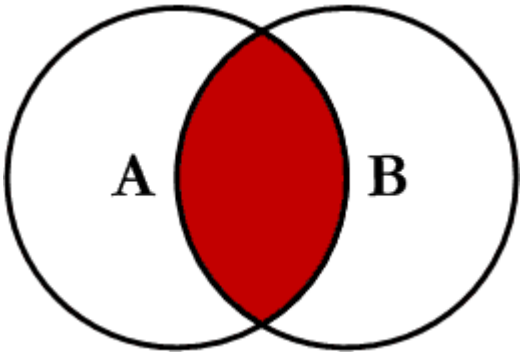
- 나는 꽤 시각적 인 사람입니다. 상황이 그림으로 이해되는 것 같습니다. 인터넷을 통해 SQL JOIN의 그래픽을 잘 보여 주었지만 원하는대로 찾을 수 없었습니다.
- 일부는 좋은 다이어그램을 가지고 있지만 완전성이 부족했습니다 (가능한 모든 JOIN이 없었습니다).
- 그래서 나는 내 자신을 만들고 기사를 작성하기로 결정했습니다.

Using the code

- 두 관계형 테이블에서 데이터를 반환 할 수있는 7 가지 방법에 대해 설명하겠습니다. 교차 조인 및 자체 참조 조인은 제외합니다. 내가 논의 할 7 가지 조인은 다음과 같습니다.
- 1.INNER JOIN
- 2.LEFT JOIN
- 3.RIGHT JOIN
- 4.OUTER JOIN
- 5.LEFT JOIN EXCLUDING INNER JOIN
- 6.RIGHT JOIN EXCLUDING INNER JOIN
- 7.OUTER JOIN EXCLUDING INNER JOIN
- 이 문서의 위해, 나는대로 5, 6, 7을 참조 것 LEFT EXCLUDING JOIN, RIGHT EXCLUDING JOIN그리고 OUTER EXCLUDING JOIN각각. 일부는 5, 6 및 7이 실제로 두 테이블을 조인하지 않는다고 주장할 수 있지만 간결성을 위해 각 쿼리에서 SQL 조인을 사용하기 때문에 여전히 조인이라고 부릅니다 (그러나 WHERE절로 일부 레코드는 제외) .

INNER JOIN

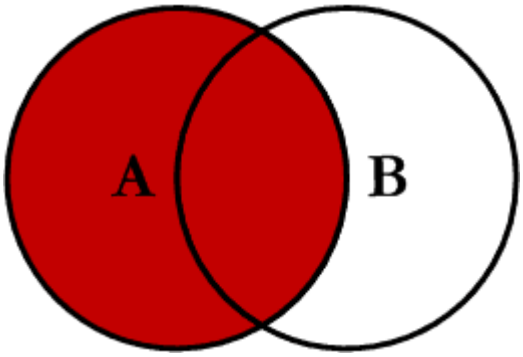
- 가장 단순하고 이해하기 쉬운 조인이며 가장 일반적입니다. 이 쿼리는 오른쪽 테이블 (테이블 B)에 일치하는 레코드가있는 왼쪽 테이블 (테이블 A)의 모든 레코드를 반환합니다. 이 조인은 다음과 같이 작성됩니다.




```
SELECT <select_list>
FROM Table_A A
INNER JOIN Table_B B
ON A.Key = B.Key
```

Left JOIN

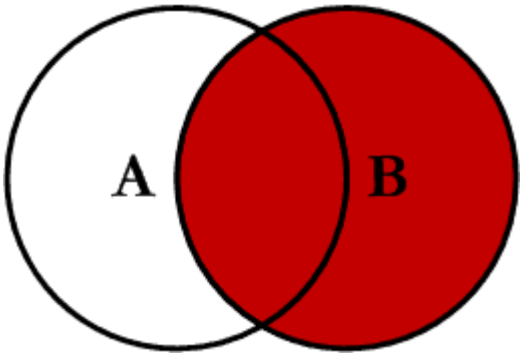
- 이 쿼리는 오른쪽 테이블 (테이블 B)과 일치하는 레코드가 있는지 여부에 관계없이 왼쪽 테이블 (테이블 A)의 모든 레코드를 반환합니다. 또한 오른쪽 테이블에서 일치하는 레코드를 반환합니다




```
SELECT <select_list>
FROM Table_A A
LEFT JOIN Table_B B
ON A.Key = B.Key
```

Right JOIN

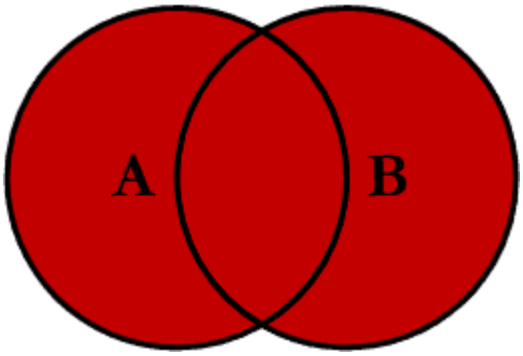
- 이 쿼리는 왼쪽 테이블 (테이블 A)과 일치하는 레코드가 있는지에 관계없이 오른쪽 테이블 (테이블 B)의 모든 레코드를 반환합니다. 또한 왼쪽 테이블에서 일치하는 레코드를 반환합니다. 이 조인은 다음과 같이 작성




```
SELECT <select_list>
FROM Table_A A
RIGHT JOIN Table_B B
ON A.Key = B.Key
```

Outer JOIN

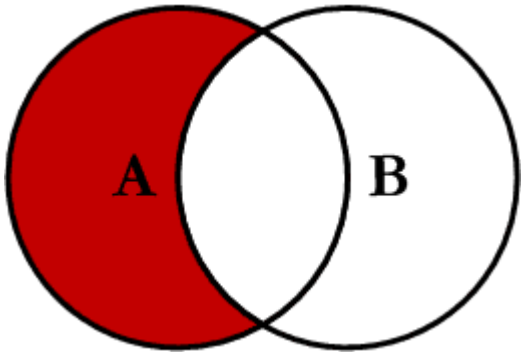
- 이 조인은 a FULL OUTER JOIN또는 a 라고도 합니다 FULL JOIN. 이 쿼리는 두 테이블의 모든 레코드를 반환하고 오른쪽 테이블 (테이블 B)의 레코드와 일치하는 왼쪽 테이블 (테이블 A)의 레코드를 조인




```
SELECT <select_list>
FROM Table_A A
FULL OUTER JOIN Table_B B
ON A.Key = B.Key
```

Left Excluding JOIN

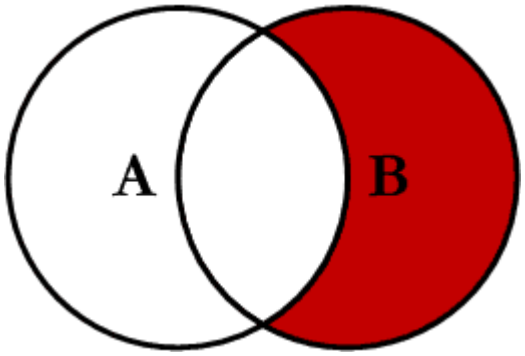
- 이 쿼리는 오른쪽 테이블 (테이블 B)의 레코드와 일치하지 않는 왼쪽 테이블 (테이블 A)의 모든 레코드를 반환합니다. 이 조인은 다음과 같이 작성




```
SELECT <select_list>
FROM Table_A A
LEFT JOIN Table_B B
ON A.Key = B.Key
WHERE B.Key IS NULL
```

Right Excluding JOIN

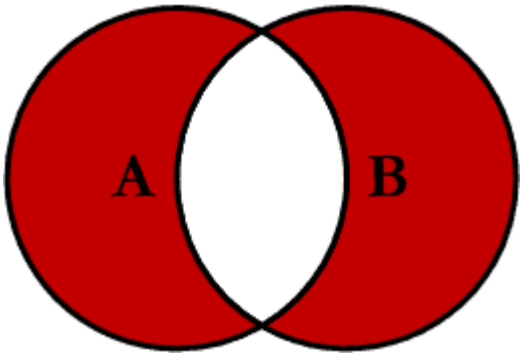
- 이 쿼리는 왼쪽 테이블 (테이블 A)의 레코드와 일치하지 않는 오른쪽 테이블 (테이블 B)의 모든 레코드를 반환합니다. 이 조인은 다음과 같이 작성




```
SELECT <select_list>
FROM Table_A A
RIGHT JOIN Table_B B
ON A.Key = B.Key
WHERE A.Key IS NULL
```

Outer Excluding JOIN

- 이 쿼리는 왼쪽 테이블 (테이블 A)의 모든 레코드와 오른쪽 테이블 (테이블 B)의 모든 레코드가 일치하지 않는 것을 반환합니다. 아직이 유형의 Join을 사용해야 할 필요는 없지만 다른 모든 유형은 매우 자주 사용합니다. 이 조인은 다음과 같이 작성




```
SELECT <select_list>
FROM Table_A A
FULL OUTER JOIN Table_B B
ON A.Key = B.Key
WHERE A.Key IS NULL OR B.Key IS NULL
```

Ref

- 원본사이트/SQL 조인의 시각적 표현

From:
http://125.132.25.164/dokuwiki/ - **문제를 잘 정의하는 것은 문제를 절반 해결한 것이다. - 2023.12**

Permanent link:
http://125.132.25.164/dokuwiki/doku.php?id=wiki:database:common&rev=1584447321

Last update: **2022/03/10 19:52**