**MOAZIO**

**DB Query 튜닝 세부 내역서**

1. **개발사 직접 적용 필요 Query**

* 게임 서버에서 직접 만들어서 호출하는 쿼리로 개발사에서 반영 작업 필요

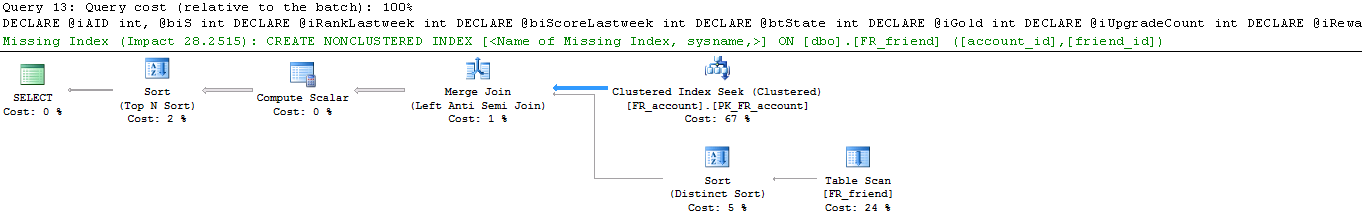
--문제 쿼리

SELECT TOP 20 account\_id, nickname, win\_point, [level], license FROM FR\_account WHERE account\_id != @iAccountID AND account\_id > 1000

AND account\_id NOT IN (SELECT friend\_id FROM FR\_friend WHERE account\_id = @iAccountID) ORDER BY NEWID()

--증상

대량 읽기가 발생하고 있습니다.



--원인

ORDER BY NEWID()

ORDER BY NEWID() 를 사용하게 되면 DBMS는 순서를 알 수 없기 때문에

TOP을 사용한다고 해도 테이블의 모든 데이터를 읽어서 임시테이블에서 정렬 후

필요한 만큼만을 추출하게 됩니다.

이때 모든 데이터를 읽게 된다는 점

대량의 데이터를 정렬해야 한다는 점

이 두 가지 때문에 성능저하가 일어나게 됩니다.

또한 NOT IN 쿼리의 경우 NOT EXISTS로 변경 하게 되면 옵티마이저가 최적화 해 줄 가능성이 더 높습니다.

--방안

쿼리 수정

SELECT TOP 20 a.account\_id, a.nickname, a.win\_point, a.[level], a.license

FROM

(

SELECT TOP 100 a.account\_id, a.nickname, a.win\_point, a.[level], a.license

FROM dbo.FR\_account AS a WITH(NOLOCK)

WHERE a.account\_id != @iAccountID

AND a.account\_id > 1000

AND NOT EXISTS(SELECT TOP 1 '' FROM dbo.FR\_friend AS b WITH(NOLOCK) WHERE a.account\_id = @iAccountID

AND a.account\_id = b.friend\_id)

AND (ABS(CAST((BINARY\_CHECKSUM(a.account\_id) \* RAND()) as int)) % 100) < 10

) AS a

ORDER BY NEWID()

(ABS(CAST((BINARY\_CHECKSUM(a.account\_id) \* RAND()) as int)) % 100) < 10

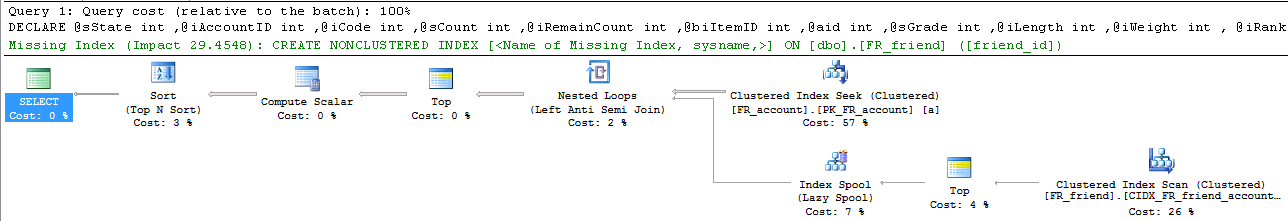
를 사용하여 대상을 선정 후

소량의 데이터에 대하여 임의정렬을 시도합니다.

참고 자료

<https://msdn.microsoft.com/en-us/library/cc441928.aspx?f=255&MSPPError=-2147217396>

--튜닝 후



Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

**추후 적용이 필요합니다.**

--문제 쿼리

DECLARE DemoCursor CURSOR FOR

SELECT account\_id, score FROM FR\_stage\_rank\_8

OPEN DemoCursor

FETCH NEXT FROM DemoCursor INTO @iAID, @biS

WHILE @@FETCH\_STATUS = 0

BEGIN

SELECT @iRank = (count(\*)+1) FROM FR\_stage\_rank\_8 WHERE score > @biS

UPDATE FR\_stage\_rank\_8 SET [rank] = @iRank WHERE account\_id = @iAID

FETCH NEXT FROM DemoCursor INTO @iAID, @biS

END

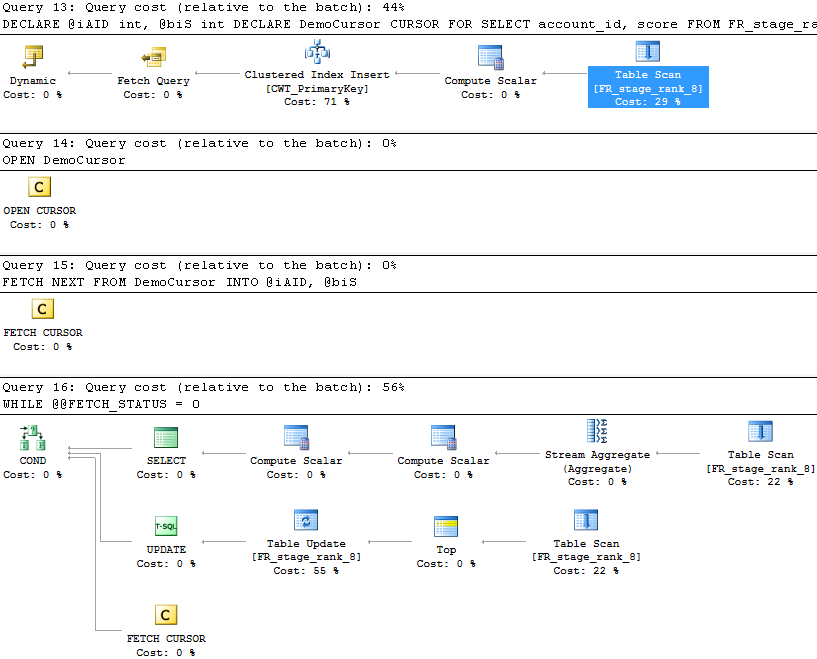
CLOSE DemoCursor

--증상

잦은 트렌젝션

(count(\*)+1) 을 호출 하게 됨으로 매 루프마다 FR\_stage\_rank\_8 테이블에 대한 대량 읽기 발생

인덱스가 없음으로 FR\_stage\_rank\_8 테이블을 읽는 모든 쿼리가 FR\_stage\_rank\_8 의 모든 데이터를 읽음



--원인

해당 쿼리는 커서를 사용 하고 있습니다.

커서는 잦은 트렌젝션과 테이블에 대한 잠금을 유발하여

해당 쿼리의 성능 뿐 아니라 DB전반에 걸쳐 해당 테이블을 사용하는 모든 쿼리에 성능 저하를 일으 킬 수 있습니다.

유저 수가 증가 할 수록 성능은 기하급수 적으로 낮아 집니다.

--방안

FR\_stage\_rank\_8 에 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_stage\_rank\_8\_account\_id ON dbo.FR\_stage\_rank\_8(account\_id);

CREATE NONCLUSTERED INDEX IDX\_FR\_stage\_rank\_8\_score ON dbo.FR\_stage\_rank\_8(score);

쿼리 수정

UPDATE b

SET b.rank = a.NewRank

FROM

(

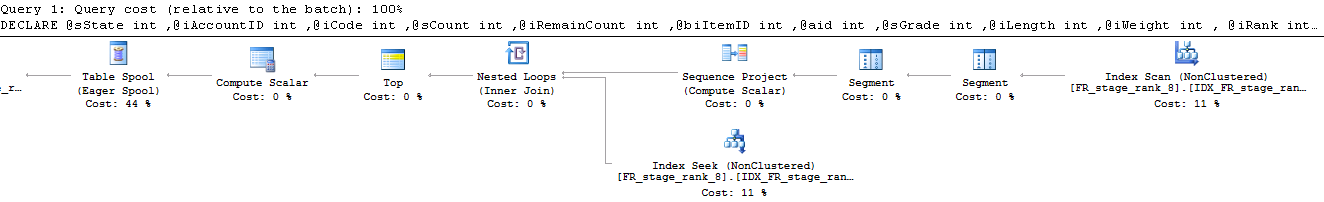
SELECT account\_id, score, RANK() OVER(ORDER BY score DESC) AS NewRank

FROM dbo.FR\_stage\_rank\_8 WITH(NOLOCK)

) AS a

INNER JOIN FR\_stage\_rank\_8 AS b ON a.account\_id = b.account\_id AND a.score = b.score;

--튜닝 후



변경 할 경우 위 처럼 효과적인 실행 계획으로 변하나 서버 프로그램에 하드코딩 되어 있는 관계로 수정 할 수 없었습니다. 추 후 **적용이 필요합니다**.

1. **Query 튜닝 세부 내역**

* Test DB 에만 반영 되어 있음.
* “9-1. 인덱스 생성 쿼리.sql” 를 Live DB 에 수행하면 적용 됨

--문제 쿼리

UPDATE FR\_achievement SET

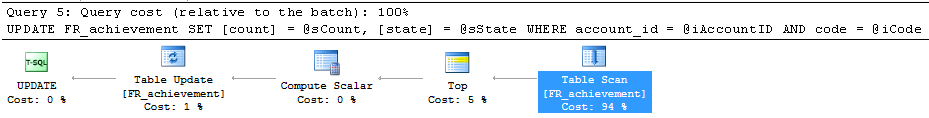
[count] = @sCount, [state] = @sState

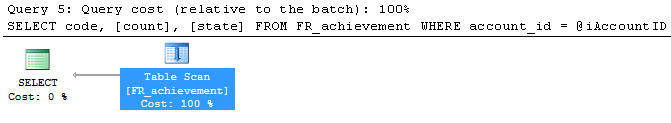
WHERE account\_id = @iAccountID AND code = @iCode

SELECT code, [count], [state] FROM FR\_achievement WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생하고 있습니다.





--원인

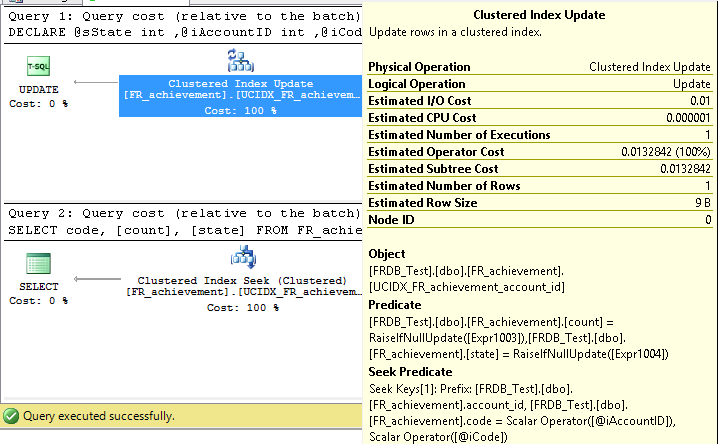
적절한 인덱스가 없습니다.

--방안

account\_id, code 복합 유니크 클러스터드 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_achievement\_account\_id ON dbo.FR\_achievement(account\_id, code);

--튜닝 후



Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

UPDATE FR\_wasting\_item SET

code = @iCode, remain\_count = @iRemainCount

WHERE account\_id = @iAccountID AND item\_id = @biItemID

--증상

대량 읽기가 발생하고 있습니다.



--원인

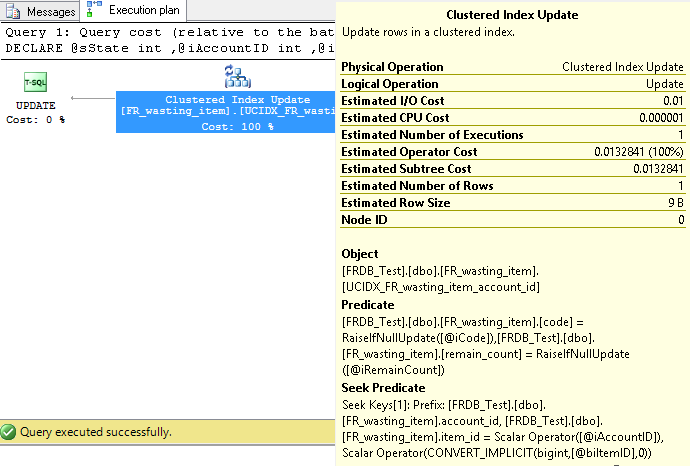
적절한 인덱스가 없습니다.

--방안

account\_id, item\_id 복합 유니크 클러스터드 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_wasting\_item\_account\_id ON dbo.FR\_wasting\_item(account\_id, item\_id);

--튜닝 후



Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

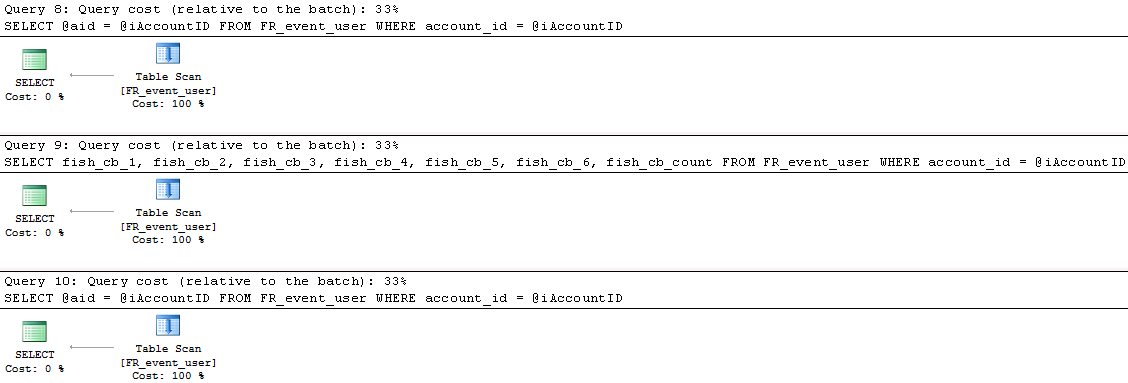
SELECT @aid = @iAccountID FROM FR\_event\_user WHERE account\_id = @iAccountID

SELECT fish\_cb\_1, fish\_cb\_2, fish\_cb\_3, fish\_cb\_4, fish\_cb\_5, fish\_cb\_6, fish\_cb\_count FROM FR\_event\_user WHERE account\_id = @iAccountID

SELECT @aid = @iAccountID FROM FR\_event\_user WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생하고 있습니다.



--원인

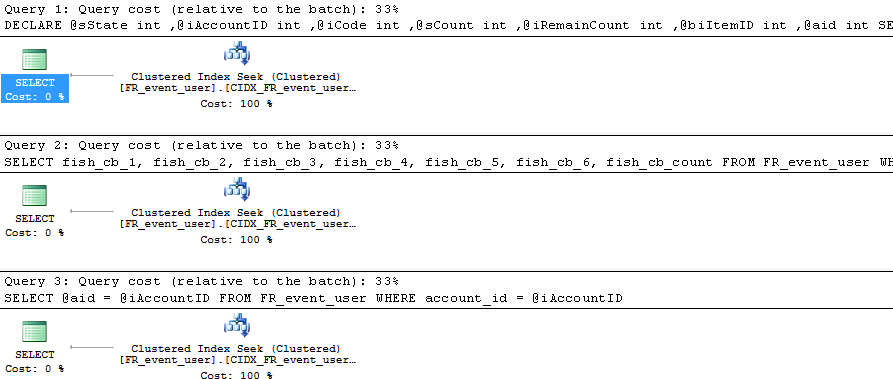
적절한 인덱스가 없습니다.

--방안

account\_id 유니크 클러스터드 인덱스 생성

CREATE CLUSTERED INDEX CIDX\_FR\_event\_user\_account\_id ON dbo.FR\_event\_user(account\_id);

--튜닝 후



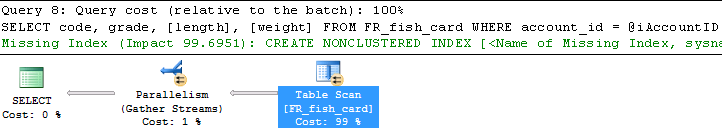
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT code, grade, [length], [weight] FROM FR\_fish\_card WHERE account\_id = @iAccountID

--증상

대량 읽기 가 발생하고 있습니다.



--원인

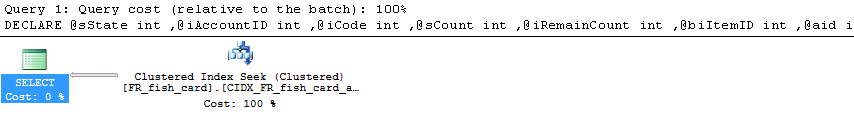
적절한 인덱스가 없습니다.

--방안

FR\_fish\_card account\_id 클러스터드 인덱스 생성

CREATE CLUSTERED INDEX CIDX\_FR\_fish\_card\_account\_id ON dbo.FR\_fish\_card(account\_id);

--튜닝 후



Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT c.account\_id

, c.nickname

, b.grade

, b.[length]

, b.[weight]

, DATEPART(YEAR, b.[date]), DATEPART(MONTH, b.[date]), DATEPART(DAY, b.[date])

FROM (

SELECT TOP 3 p.account\_id AS account\_id, p.nickname AS nickname

FROM FR\_friend AS a

INNER JOIN FR\_account AS p ON p.account\_id = a.friend\_id

WHERE a.account\_id = @iAccountID AND a.[state] = 1

) AS c

inner join FR\_fish\_card AS b ON b.account\_id = c.account\_id

WHERE b.code = @iCode ORDER BY b.[length] DESC, b.weight DESC, b.date ASC

UPDATE FR\_fish\_card SET

grade = @sGrade, [length] = @iLength, [weight] = @iWeight, [date] = GETUTCDATE()

WHERE account\_id = @iAccountID AND code = @iCode

SELECT TOP 3 a.account\_id, p.nickname, grade, [length], [weight], DATEPART(YEAR, a.[date]), DATEPART(MONTH, a.[date]), DATEPART(DAY, a.[date])

FROM FR\_fish\_card AS a

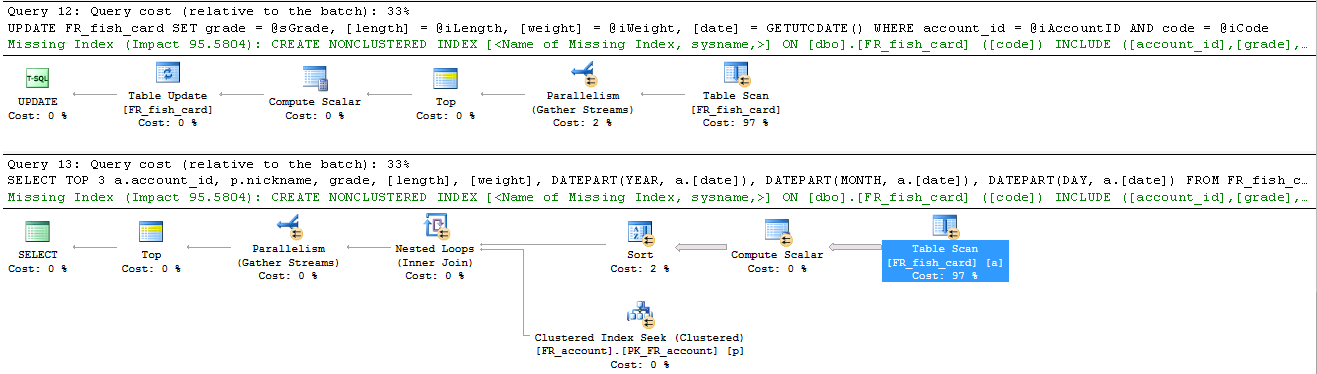
INNER JOIN FR\_account AS p ON p.account\_id = a.account\_id

WHERE code = @iCode

ORDER BY a.[length] DESC, a.[weight] DESC, a.[date] ASC

--증상

대량 읽기 가 발생하고 있습니다.



--원인

적절한 인덱스가 없습니다.

--방안

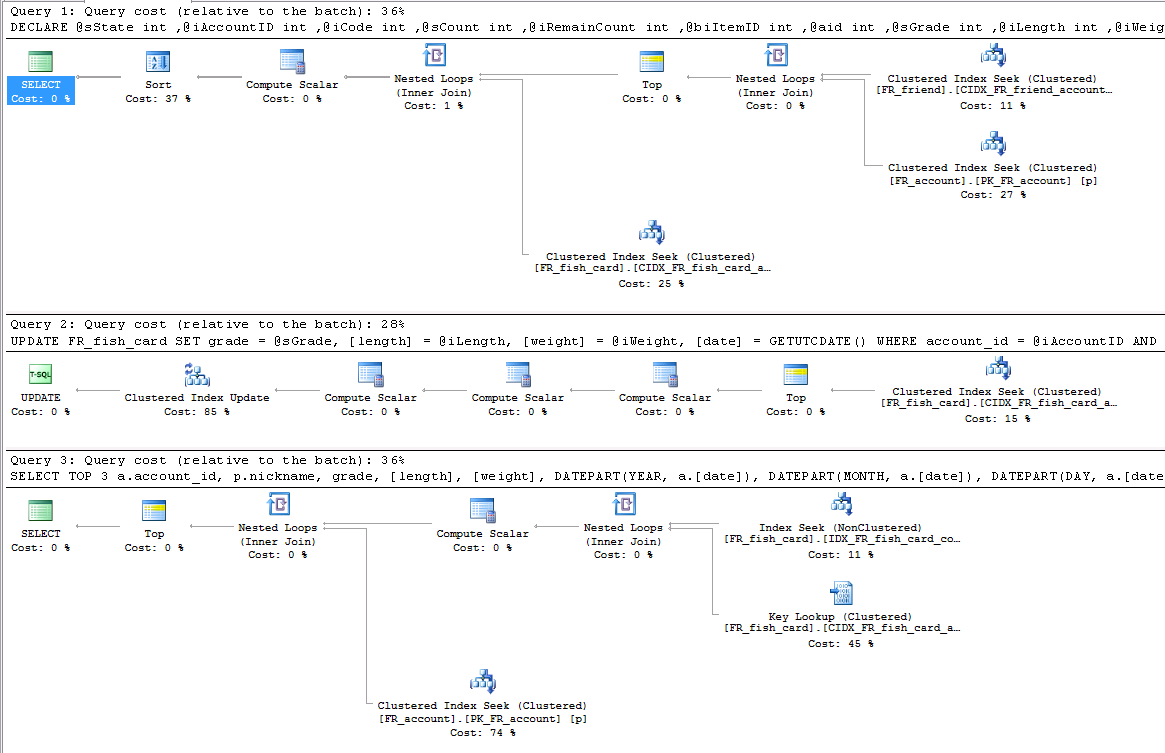
FR\_friend account\_id, state 인덱스 생성

CREATE CLUSTERED INDEX CIDX\_FR\_friend\_account\_id ON dbo.FR\_friend(account\_id, state);

FR\_fish\_card account\_id, code, length, weight, date DESC 넌클러스터드 인덱스 생성

CREATE NONCLUSTERED INDEX IDX\_FR\_fish\_card\_code ON dbo.FR\_fish\_card(code, length, weight, date DESC);

--튜닝 후



Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 대상

FR\_account 테이블

--증상

불필요한 인덱스의 유지비용이 소모되고 있습니다.

--원인

인덱스 중복 생성



--방안

IX\_FR\_account 인덱스 제거 (PK\_FR\_account 와 중복된 인덱스로 유지 관리 비용 추가 발생)

--튜닝 후



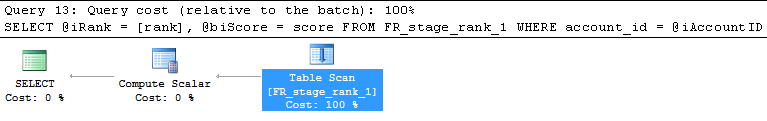
공통 Column 에 대한 인덱스가 삭제되었습니다.

--문제 쿼리

SELECT @iRank = [rank], @biScore = score FROM FR\_stage\_rank\_1 WHERE account\_id = @iAccountID

--증상

대량 읽기 발생 가능성이 있습니다.



--원인

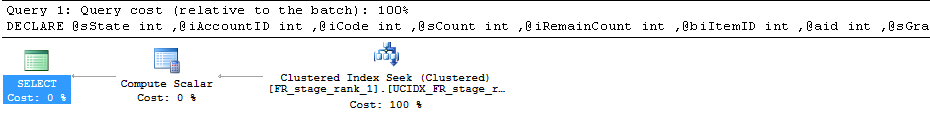
적절한 인덱스가 없습니다.

--방안

FR\_stage\_rank\_1의 account\_id 에 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_stage\_rank\_1\_account\_id ON dbo.FR\_stage\_rank\_1(account\_id);

--튜닝 후



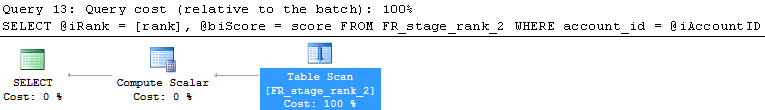
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRank = [rank], @biScore = score FROM FR\_stage\_rank\_2 WHERE account\_id = @iAccountID

--증상

대량 읽기 발생 가능성이 있습니다.



--원인

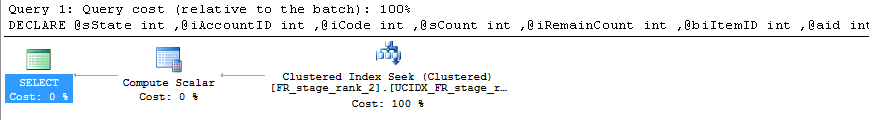
적절한 인덱스가 없습니다.

--방안

FR\_stage\_rank\_2의 account\_id 에 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_stage\_rank\_2\_account\_id ON dbo.FR\_stage\_rank\_2(account\_id);

--튜닝 후



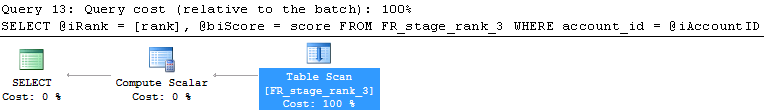
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRank = [rank], @biScore = score FROM FR\_stage\_rank\_3 WHERE account\_id = @iAccountID

--증상

대량 읽기 발생 가능성이 있습니다.



--원인

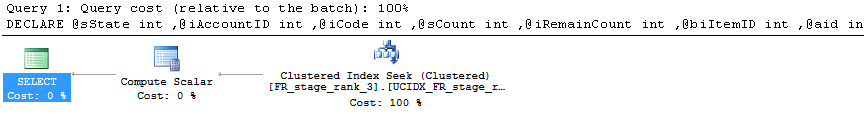
적절한 인덱스가 없습니다.

--방안

FR\_stage\_rank\_3의 account\_id 에 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_stage\_rank\_3\_account\_id ON dbo.FR\_stage\_rank\_3(account\_id);

--튜닝 후



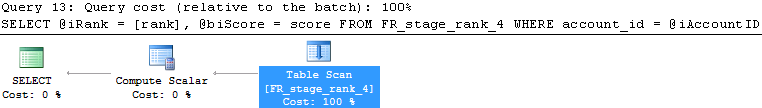
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRank = [rank], @biScore = score FROM FR\_stage\_rank\_4 WHERE account\_id = @iAccountID

--증상

대량 읽기 발생 가능성이 있습니다.



--원인

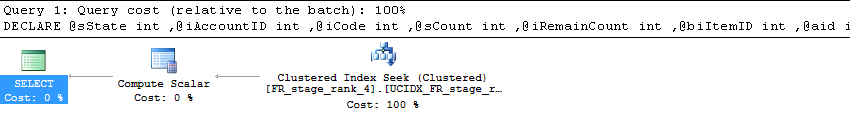
적절한 인덱스가 없습니다.

--방안

FR\_stage\_rank\_4의 account\_id 에 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_stage\_rank\_4\_account\_id ON dbo.FR\_stage\_rank\_4(account\_id);

--튜닝 후



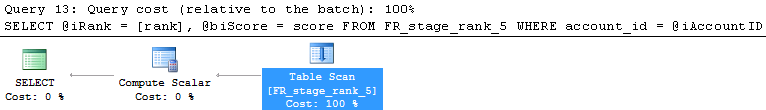
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRank = [rank], @biScore = score FROM FR\_stage\_rank\_5 WHERE account\_id = @iAccountID

--증상

대량 읽기 발생 가능성이 있습니다.



--원인

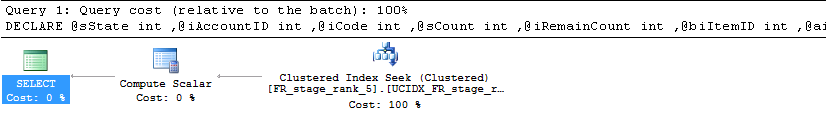
적절한 인덱스가 없습니다.

--방안

FR\_stage\_rank\_5의 account\_id 에 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_stage\_rank\_5\_account\_id ON dbo.FR\_stage\_rank\_5(account\_id);

--튜닝 후



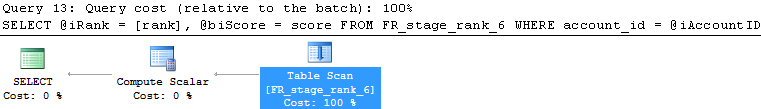
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRank = [rank], @biScore = score FROM FR\_stage\_rank\_6 WHERE account\_id = @iAccountID

--증상

대량 읽기 발생 가능성이 있습니다.



--원인

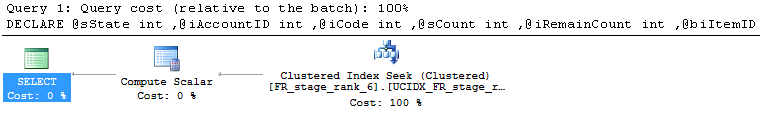
적절한 인덱스가 없습니다.

--방안

FR\_stage\_rank\_6의 account\_id 에 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_stage\_rank\_6\_account\_id ON dbo.FR\_stage\_rank\_6(account\_id);

--튜닝 후



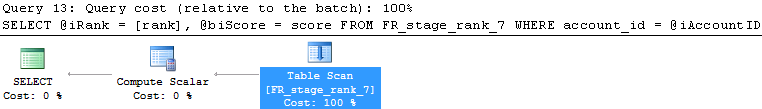
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRank = [rank], @biScore = score FROM FR\_stage\_rank\_7 WHERE account\_id = @iAccountID

--증상

대량 읽기 발생 가능성이 있습니다.



--원인

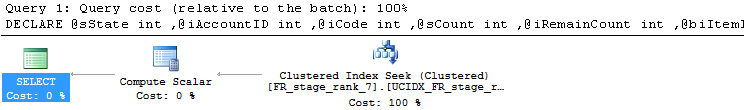
적절한 인덱스가 없습니다.

--방안

FR\_stage\_rank\_7의 account\_id 에 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_stage\_rank\_7\_account\_id ON dbo.FR\_stage\_rank\_7(account\_id);

--튜닝 후



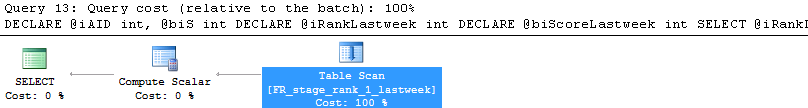
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRankLastweek = [rank], @biScoreLastweek = score FROM FR\_stage\_rank\_1\_lastweek WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생하고 있습니다.



--원인

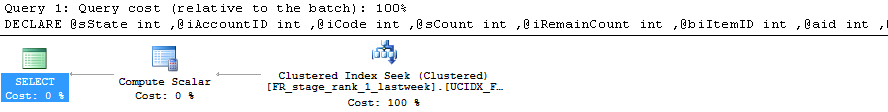
적절한 인덱스가 없습니다.

--방안

account\_id 에 클러스터드 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_stage\_rank\_1\_lastweek\_account\_id ON dbo.FR\_stage\_rank\_1\_lastweek(account\_id);

--튜닝 후



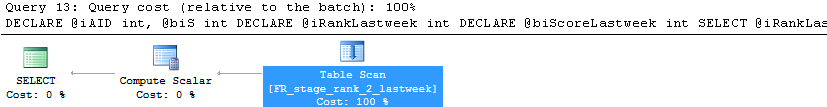
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRankLastweek = [rank], @biScoreLastweek = score FROM FR\_stage\_rank\_2\_lastweek WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생하고 있습니다.



--원인

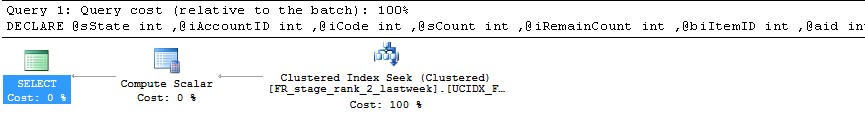
적절한 인덱스가 없습니다.

--방안

account\_id 에 클러스터드 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_stage\_rank\_2\_lastweek\_account\_id ON dbo.FR\_stage\_rank\_2\_lastweek(account\_id);

--튜닝 후



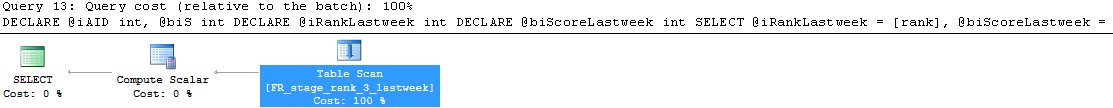
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRankLastweek = [rank], @biScoreLastweek = score FROM FR\_stage\_rank\_3\_lastweek WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생하고 있습니다.



--원인

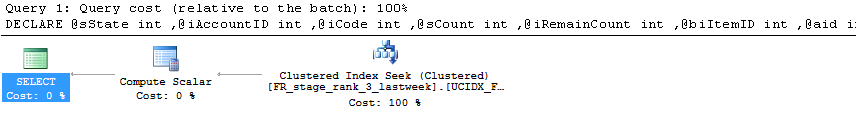
적절한 인덱스가 없습니다.

--방안

account\_id 에 클러스터드 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_stage\_rank\_3\_lastweek\_account\_id ON dbo.FR\_stage\_rank\_3\_lastweek(account\_id);

--튜닝 후



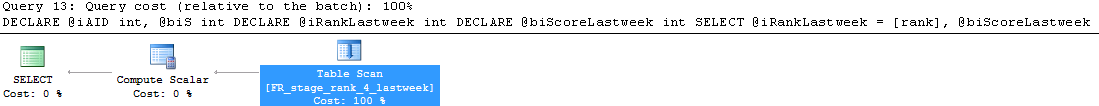
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRankLastweek = [rank], @biScoreLastweek = score FROM FR\_stage\_rank\_4\_lastweek WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생하고 있습니다.



--원인

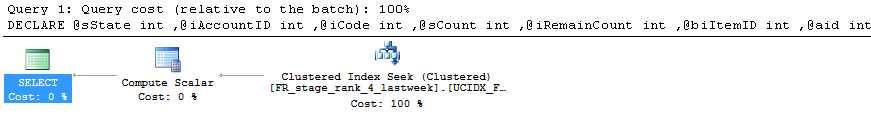
적절한 인덱스가 없습니다.

--방안

account\_id 에 클러스터드 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_stage\_rank\_4\_lastweek\_account\_id ON dbo.FR\_stage\_rank\_4\_lastweek(account\_id);

--튜닝 후



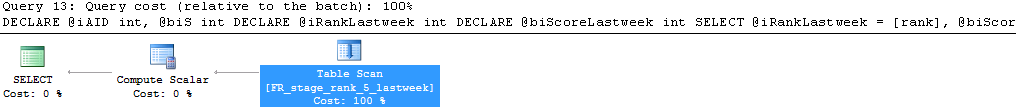
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRankLastweek = [rank], @biScoreLastweek = score FROM FR\_stage\_rank\_5\_lastweek WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생하고 있습니다.



--원인

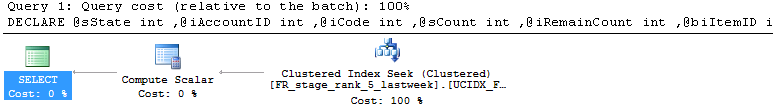
적절한 인덱스가 없습니다.

--방안

account\_id 에 클러스터드 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_stage\_rank\_5\_lastweek\_account\_id ON dbo.FR\_stage\_rank\_5\_lastweek(account\_id);

--튜닝 후



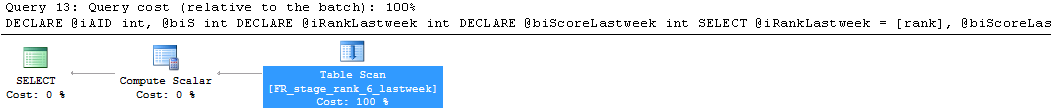
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRankLastweek = [rank], @biScoreLastweek = score FROM FR\_stage\_rank\_6\_lastweek WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생하고 있습니다.



--원인

적절한 인덱스가 없습니다.

--방안

account\_id 에 클러스터드 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_stage\_rank\_6\_lastweek\_account\_id ON dbo.FR\_stage\_rank\_6\_lastweek(account\_id);

--튜닝 후



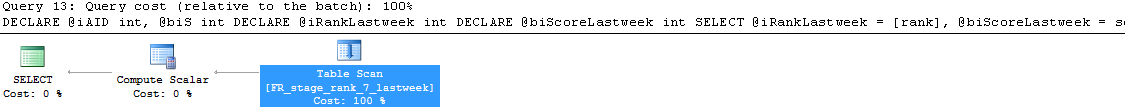
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRankLastweek = [rank], @biScoreLastweek = score FROM FR\_stage\_rank\_7\_lastweek WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생하고 있습니다.



--원인

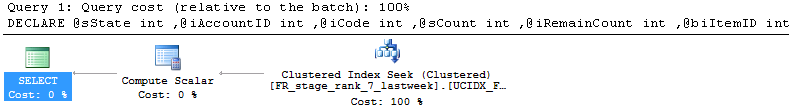
적절한 인덱스가 없습니다.

--방안

account\_id 에 클러스터드 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_stage\_rank\_7\_lastweek\_account\_id ON dbo.FR\_stage\_rank\_7\_lastweek(account\_id);

--튜닝 후



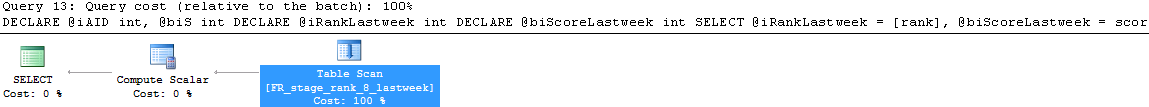
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRankLastweek = [rank], @biScoreLastweek = score FROM FR\_stage\_rank\_8\_lastweek WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생하고 있습니다.



--원인

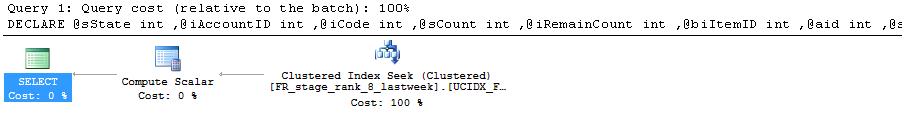
적절한 인덱스가 없습니다.

--방안

account\_id 에 클러스터드 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_stage\_rank\_8\_lastweek\_account\_id ON dbo.FR\_stage\_rank\_8\_lastweek(account\_id);

--튜닝 후



Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

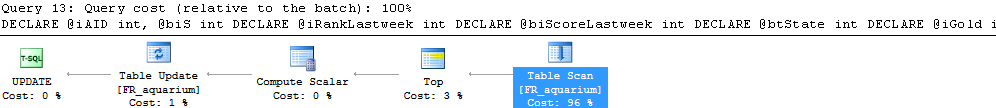
UPDATE FR\_aquarium SET

state = @btState, code = @iCode, gold = @iGold, upgrade\_count = @iUpgradeCount, reward\_level = @iRewardLevel, last\_gold\_time = @biLastGoldTime

WHERE account\_id = @iAccountID AND aquarium\_id = @iAquariumID

--증상

대량 읽기가 발생하고 있습니다.



--원인

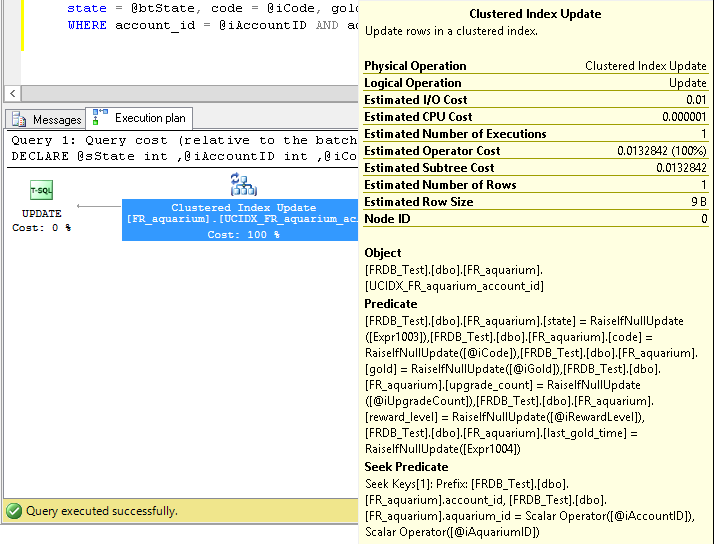
적절한 인덱스가 없습니다.

--방안

account\_id, aquarium\_id 유니크 클러스터드 복합 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_aquarium\_account\_id ON dbo.FR\_aquarium(account\_id, aquarium\_id);

--튜닝 후



Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT message\_id

, sender\_id

, b.nickname

, title

, [text]

, item\_code

, item\_count

, [read]

, DATEPART(YEAR, a.date\_create)

, DATEPART(MONTH, a.date\_create)

, DATEPART(DAY, a.date\_create)

FROM (SELECT row\_number() OVER(ORDER BY date\_create DESC) AS \_idx, \*

FROM FR\_message

WHERE account\_id = @iAccountID AND [delete] = 0

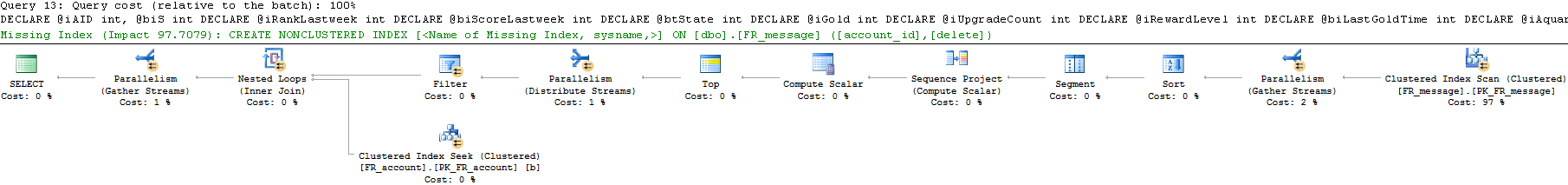
) AS a

inner join FR\_account AS b ON b.account\_id = a.sender\_id

WHERE a.\_idx >= @iOffset and a.\_idx < (@iOffset + @iLen) ORDER BY a.\_idx ASC

--증상

대량 읽기가 발생하고 있습니다.



--원인

적절한 인덱스가 없습니다.

--방안

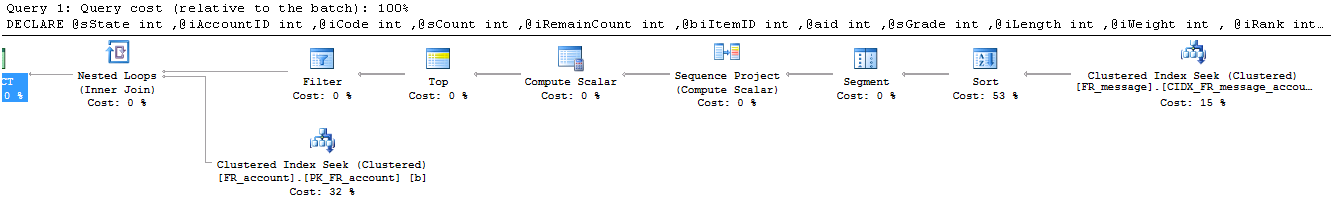
FR\_message 의 account\_id 에 인덱스 변경 및 생성

ALTER TABLE dbo.FR\_message DROP CONSTRAINT PK\_FR\_message;

ALTER TABLE dbo.FR\_message ADD CONSTRAINT PK\_FR\_message PRIMARY KEY NONCLUSTERED (message\_id);

CREATE CLUSTERED INDEX CIDX\_FR\_message\_account\_id ON dbo.FR\_message(account\_id, [delete]);

--튜닝 후

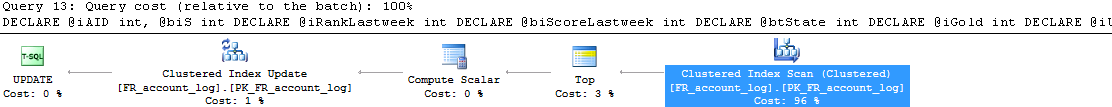
Table Scan 이 아닌 Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

UPDATE FR\_account\_log SET status = 0, date\_logout = GETUTCDATE() WHERE status = 1 and account\_id = @dwAccountID

--증상

대량 읽기가 발생하고 있습니다.



--원인

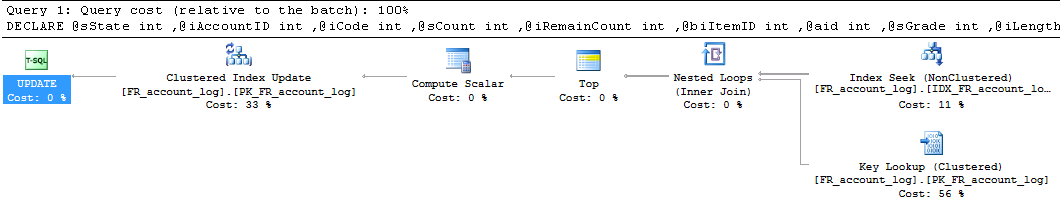
적절한 인덱스가 없습니다.

--방안

account\_id 에 인덱스 생성

CREATE INDEX IDX\_FR\_account\_log\_account\_id ON dbo.FR\_account\_log(account\_id);

--튜닝 후



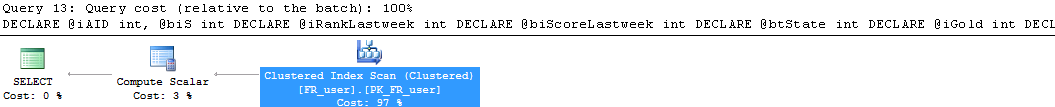
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @szUserNo = user\_no FROM FR\_user WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생하고 있습니다.



--원인

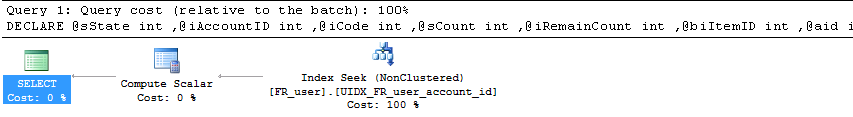
적절한 인덱스가 없습니다.

--방안

FR\_user의 user\_no와 account\_id 가 1:1의 관계라면 유니크 인덱스를 생성하고 아니라면 일반 인덱스 생성

CREATE UNIQUE INDEX UIDX\_FR\_user\_account\_id ON dbo.FR\_user(account\_id);

--튜닝 후



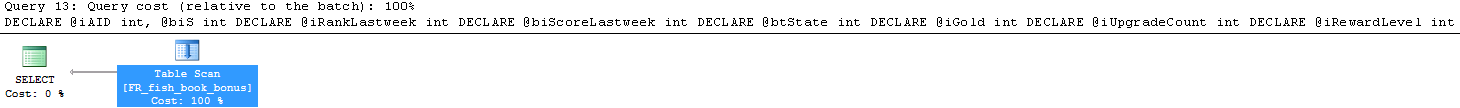
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT code, check\_0, check\_1, check\_2, check\_3, check\_4 FROM FR\_fish\_book\_bonus WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생하고 있습니다.



--원인

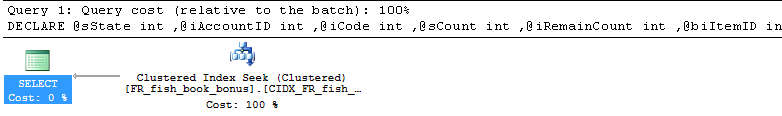
적절한 인덱스가 없습니다.

--방안

FR\_fish\_book\_bonus의 account\_id 에 클러스터드 인덱스 생성

CREATE CLUSTERED INDEX CIDX\_FR\_fish\_book\_bonus\_account\_id ON dbo.FR\_fish\_book\_bonus(account\_id);

--튜닝 후



Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

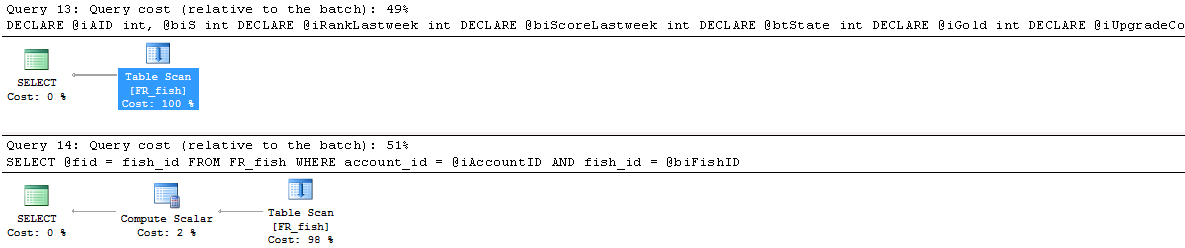
--문제 쿼리

SELECT fish\_id, aquarium\_id, code, grade, length, weight, is\_upgrade, feeding\_count, last\_feeding\_time FROM FR\_fish WHERE account\_id = @iAccountID

SELECT @fid = fish\_id FROM FR\_fish WHERE account\_id = @iAccountID AND fish\_id = @biFishID

--증상

대량 읽기가 발생하고 있습니다.



--원인

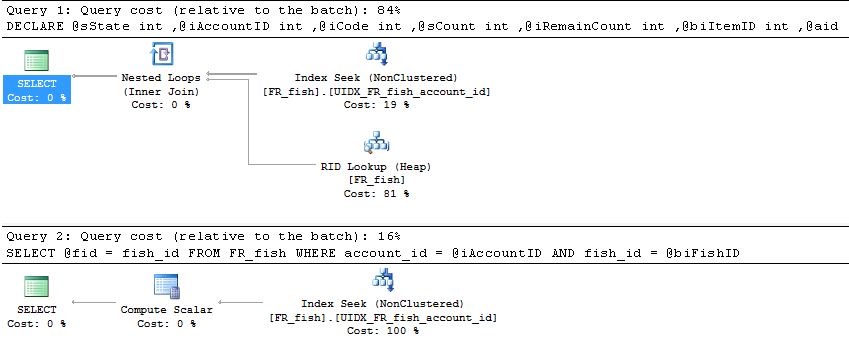
적절한 인덱스가 없습니다.

--방안

FR\_fish의 account\_id 와 fish\_id 에 인덱스를 생성

CREATE UNIQUE INDEX UIDX\_FR\_fish\_account\_id ON dbo.FR\_fish(account\_id, fish\_id);

--튜닝 후



Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @aid = account\_id FROM FR\_lotto WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생하고 있습니다.



--원인

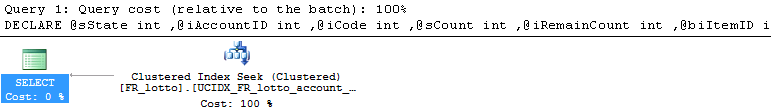
적절한 인덱스가 없습니다.

--방안

FR\_lotto 의 account\_id 에 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_lotto\_account\_id ON dbo.FR\_lotto(account\_id);

--튜닝 후



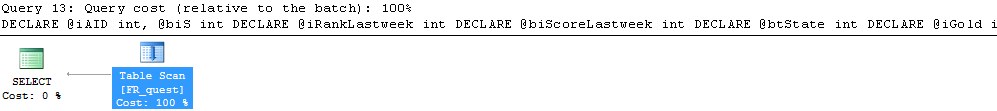
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT code, [count], [state] FROM FR\_quest WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생하고 있습니다.



--원인

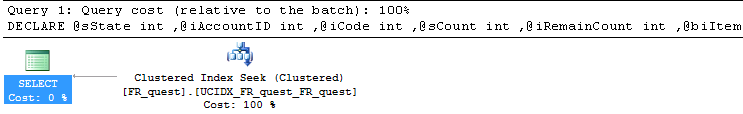
적절한 인덱스가 없습니다.

--방안

FR\_quest 의 account\_id 에 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_quest\_FR\_quest ON dbo.FR\_quest(account\_id);

--튜닝 후



Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

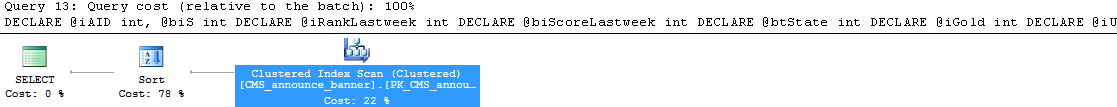
--문제 쿼리

SELECT announce\_banner\_id, image\_url, goto\_url FROM CMS\_announce\_banner WHERE sort = @iSort AND date\_start < GETUTCDATE()

AND date\_end > GETUTCDATE() ORDER BY date\_create ASC

--증상

대량 읽기가 발생하고 있습니다.



--원인

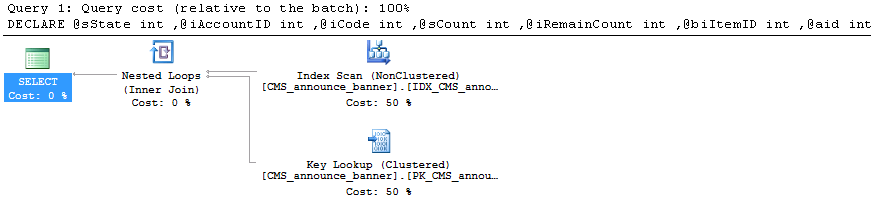
적절한 인덱스가 없습니다.

--방안

CMS\_announce\_banner 의 date\_start, date\_end, sort 에 복합 인덱스 생성

CREATE INDEX IDX\_CMS\_announce\_banner\_date\_create ON dbo.CMS\_announce\_banner(date\_create, date\_start, date\_end, sort);

--튜닝 후



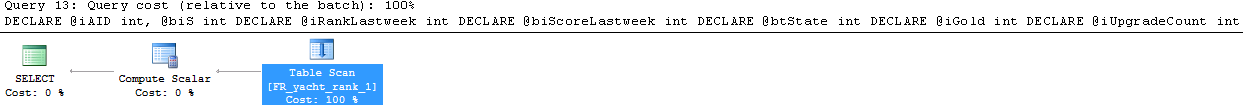
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRank = [rank], @biScore = score FROM FR\_yacht\_rank\_1 WHERE account\_id = @iAccountID

--증상

대량 읽기 발생 가능성이 있습니다.



--원인

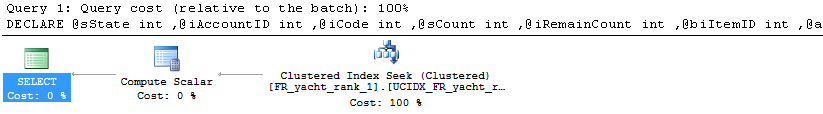
적절한 인덱스가 없습니다.

--방안

FR\_yacht\_rank\_1 의 account\_id 에 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_yacht\_rank\_1\_account\_id ON dbo.FR\_yacht\_rank\_1(account\_id);

--튜닝 후



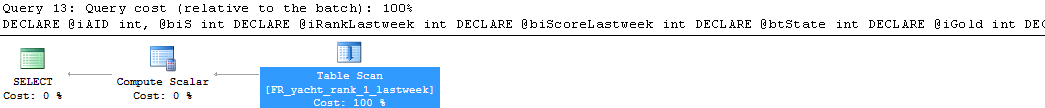
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRankLastweek = [rank], @biScoreLastweek = score FROM FR\_yacht\_rank\_1\_lastweek WHERE account\_id = @iAccountID

--증상

대량 읽기 발생 가능성이 있습니다.



--원인

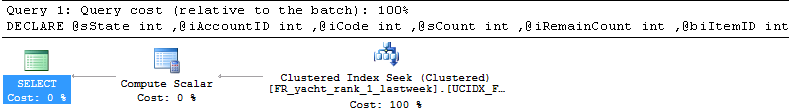
적절한 인덱스가 없습니다.

--방안

FR\_yacht\_rank\_1\_lastweek 의 account\_id 에 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_yacht\_rank\_1\_lastweek\_account\_id ON dbo.FR\_yacht\_rank\_1\_lastweek(account\_id);

--튜닝 후



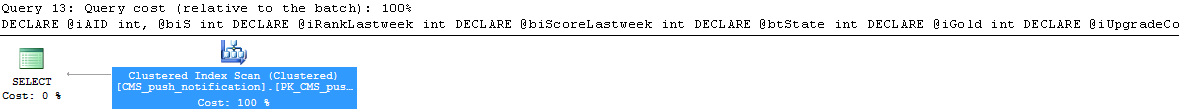
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT push\_notification\_id, title, [message] FROM CMS\_push\_notification WHERE [read] = 0 AND date\_send < GETUTCDATE()

--증상

대량 읽기가 발생하고 있습니다.



--원인

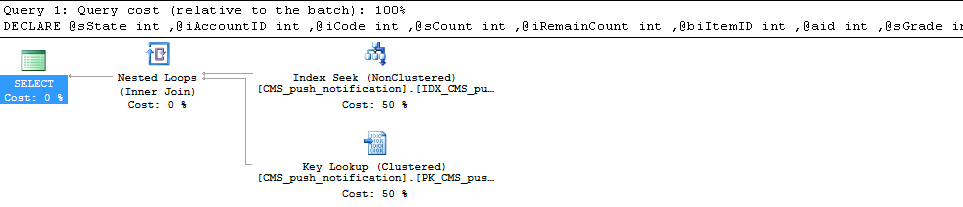
적절한 인덱스가 없습니다.

--방안

CMS\_push\_notification 의 [read], date\_send 에 인덱스 생성

CREATE INDEX IDX\_CMS\_push\_notification\_read ON dbo.CMS\_push\_notification([read], date\_send);

--튜닝 후



Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iS = stage FROM FR\_jackpot WHERE stage = @iStage

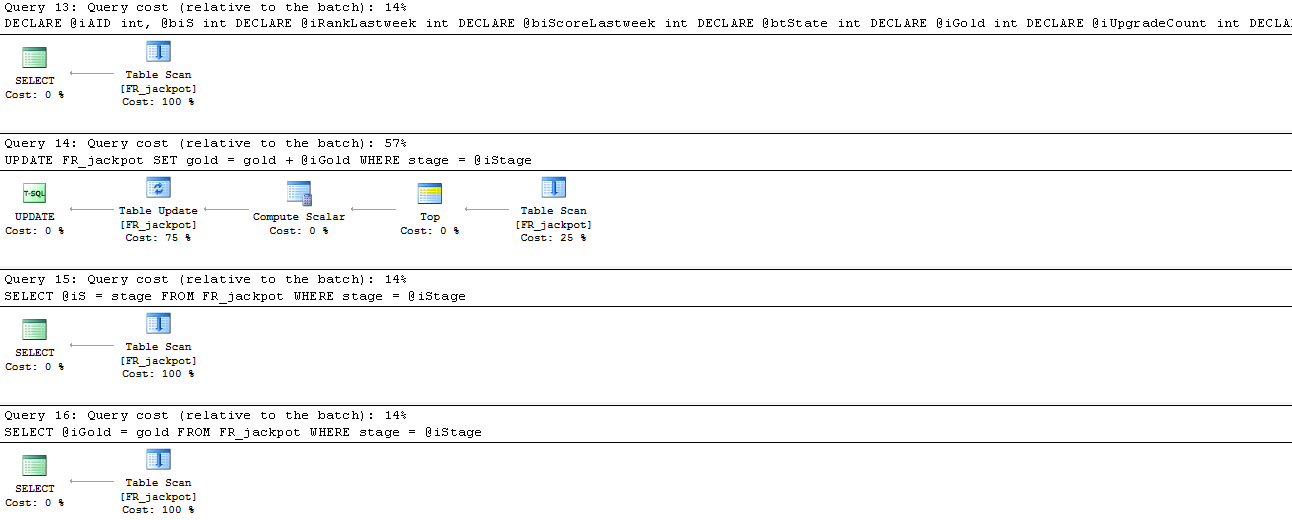
UPDATE FR\_jackpot SET gold = gold + @iGold WHERE stage = @iStage

SELECT @iS = stage FROM FR\_jackpot WHERE stage = @iStage

SELECT @iGold = gold FROM FR\_jackpot WHERE stage = @iStage

--증상

대량 읽기가 발생 할 우려가 있습니다.



--원인

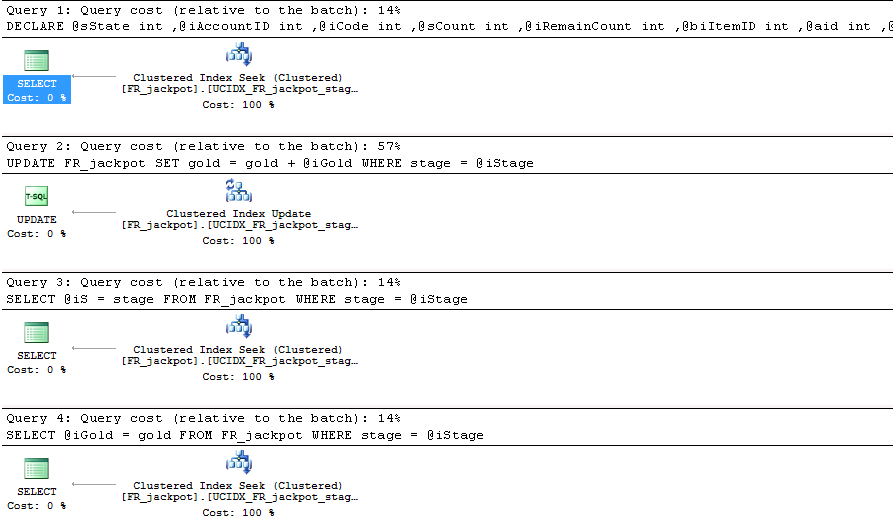
적절한 인덱스가 없습니다.

--방안

FR\_jackpot 의 stage 에 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_jackpot\_stage ON dbo.FR\_jackpot(stage);

--튜닝 후



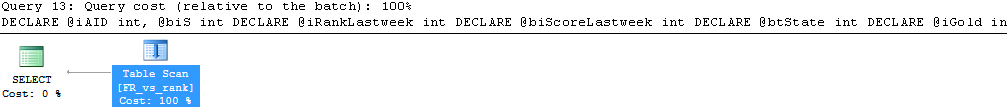
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRank = [rank], @iScore = score FROM FR\_vs\_rank WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생 할 우려가 있습니다.



--원인

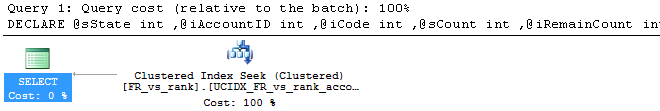
적절한 인덱스가 없습니다.

--방안

FR\_vs\_rank 의 account\_id 에 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_vs\_rank\_account\_id ON dbo.FR\_vs\_rank(account\_id);

--튜닝 후



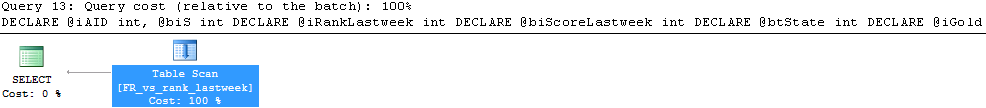
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRankLastweek = [rank], @iScoreLastweek = score FROM FR\_vs\_rank\_lastweek WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생 할 우려가 있습니다.



--원인

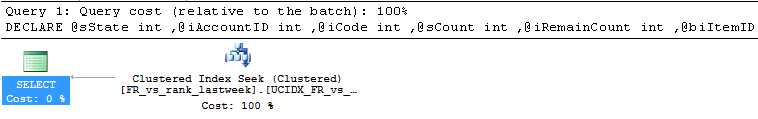
적절한 인덱스가 없습니다.

--방안

FR\_vs\_rank\_lastweek 의 account\_id 에 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_vs\_rank\_lastweek\_account\_id ON dbo.FR\_vs\_rank\_lastweek(account\_id);

--튜닝 후



Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

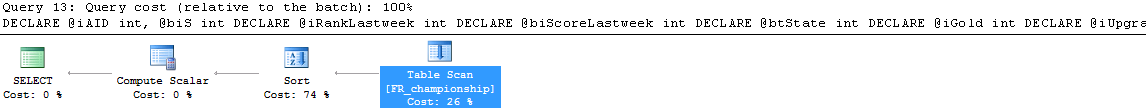
--문제 쿼리

SELECT championship\_id, title, DATEPART(YEAR, [date]), DATEPART(MONTH, [date]), DATEPART(DAY, [date]), DATEPART(HOUR, [date])

, DATEPART(MINUTE, [date]), stage\_code FROM FR\_championship WHERE [read] = 0 ORDER BY [date] ASC

--증상

대량 읽기가 발생 할 우려가 있습니다.



--원인

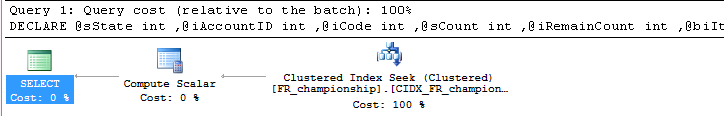
적절한 인덱스가 없습니다.

--방안

FR\_championship 의 read, date 에 복합 인덱스 생성

CREATE CLUSTERED INDEX CIDX\_FR\_championship\_read ON dbo.FR\_championship([read], [date]);

--튜닝 후



Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT game\_id, game\_time, rest\_time, target\_fish\_code, [rule], fee, min\_level, max\_level, reward\_item\_code\_a\_1

, reward\_item\_value\_a\_1, reward\_item\_code\_b\_1, reward\_item\_value\_b\_1, reward\_count\_1, reward\_item\_code\_a\_2

, reward\_item\_value\_a\_2, reward\_item\_code\_b\_2, reward\_item\_value\_b\_2, reward\_count\_2, reward\_item\_code\_a\_3

, reward\_item\_value\_a\_3, reward\_item\_code\_b\_3, reward\_item\_value\_b\_3, reward\_count\_3, reward\_item\_code\_a\_4

, reward\_item\_value\_a\_4, reward\_item\_code\_b\_4, reward\_item\_value\_b\_4, reward\_count\_4, reward\_item\_code\_a\_5

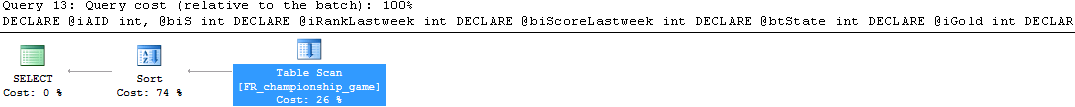
, reward\_item\_value\_a\_5, reward\_item\_code\_b\_5, reward\_item\_value\_b\_5, reward\_count\_5, reward\_item\_code\_a\_6

, reward\_item\_value\_a\_6, reward\_item\_code\_b\_6, reward\_item\_value\_b\_6, reward\_count\_6

FROM FR\_championship\_game WHERE championship\_id = @iChampionshipID ORDER BY game\_id ASC

--증상

대량 읽기가 발생 할 우려가 있습니다.



--원인

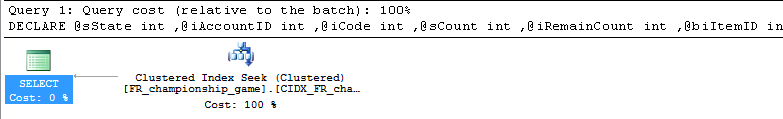
적절한 인덱스가 없습니다.

--방안

FR\_championship\_game 의 championship\_id, game\_id 에 복합 인덱스 생성

CREATE CLUSTERED INDEX CIDX\_FR\_championship\_game\_championship\_id ON dbo.FR\_championship\_game(championship\_id, game\_id);

--튜닝 후



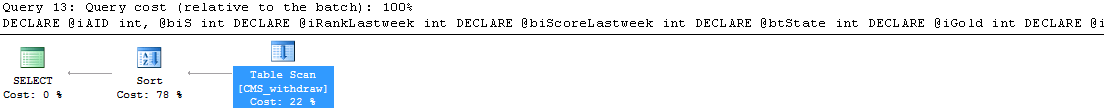
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT withdraw\_id, item\_code, item\_count FROM CMS\_withdraw WHERE account\_id = @iAccountID AND [read] = 0 ORDER BY date\_create ASC

--증상

대량 읽기가 발생 할 우려가 있습니다.



--원인

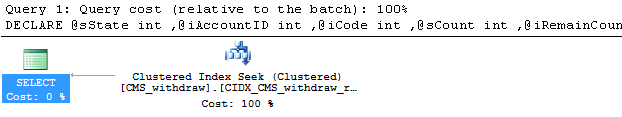
적절한 인덱스가 없습니다.

--방안

CMS\_withdraw 의 read, date\_create 에 인덱스 생성하고

CREATE CLUSTERED INDEX CIDX\_CMS\_withdraw\_read ON dbo.CMS\_withdraw([read], date\_create);

--튜닝 후



Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

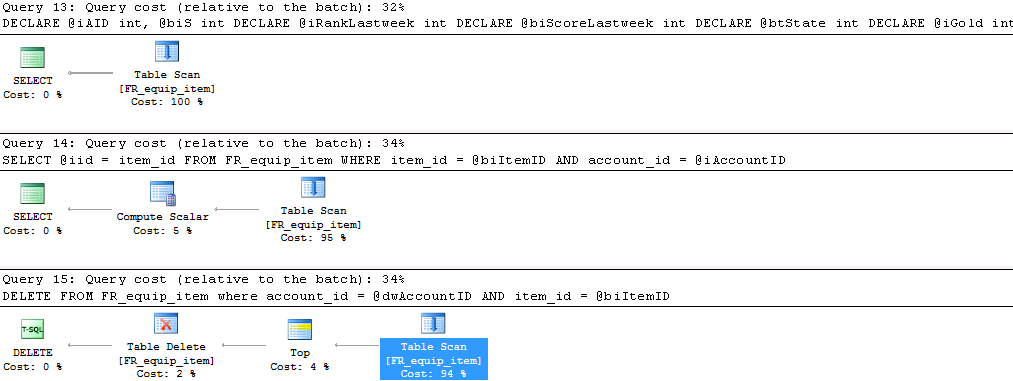
SELECT item\_id, code, enchant\_count FROM FR\_equip\_item WHERE account\_id = @iAccountID

SELECT @iid = item\_id FROM FR\_equip\_item WHERE item\_id = @biItemID AND account\_id = @iAccountID

DELETE FROM FR\_equip\_item where account\_id = @dwAccountID AND item\_id = @biItemID

--증상

대량 읽기가 발생 할 우려가 있습니다.



--원인

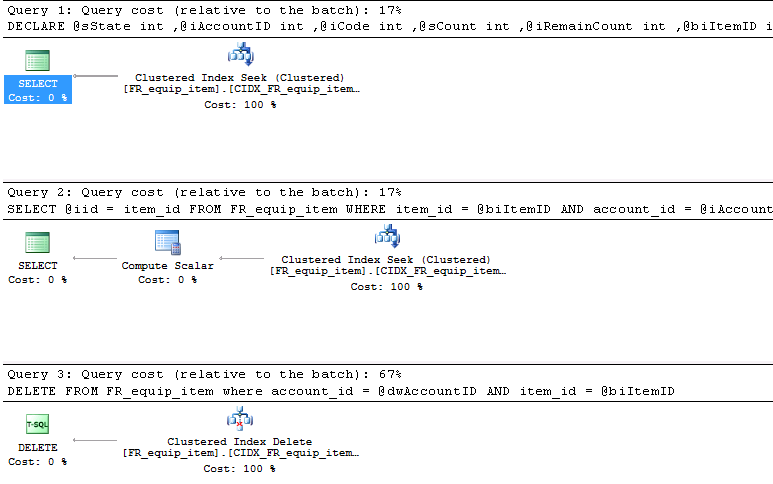
적절한 인덱스가 없습니다.

--방안

FR\_equip\_item 의 account\_id, item\_id 에 인덱스 생성

CREATE CLUSTERED INDEX CIDX\_FR\_equip\_item\_account\_id ON dbo.FR\_equip\_item(account\_id, item\_id);

--튜닝 후



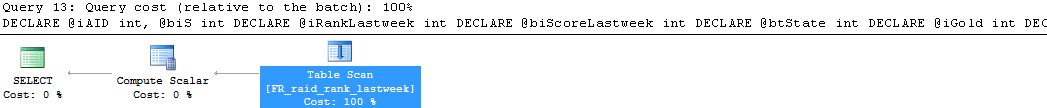
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRankLastweek = [rank], @iScoreLastweek = score FROM FR\_raid\_rank\_lastweek WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생 할 우려가 있습니다.



--원인

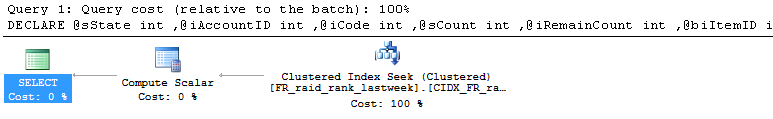
적절한 인덱스가 없습니다.

--방안

FR\_raid\_rank\_lastweek 의 account\_id 에 인덱스 생성

CREATE CLUSTERED INDEX CIDX\_FR\_raid\_rank\_lastweek\_account\_id ON dbo.FR\_raid\_rank\_lastweek(account\_id);

--튜닝 후



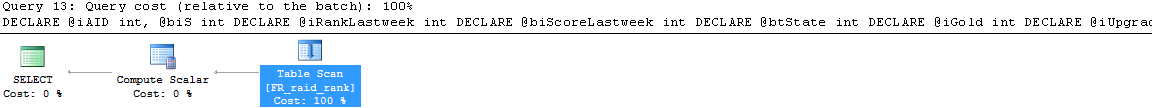
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @iRank = [rank], @iScore = score FROM FR\_raid\_rank WHERE account\_id = @iAccountID

--증상

대량 읽기가 발생 할 우려가 있습니다.



--원인

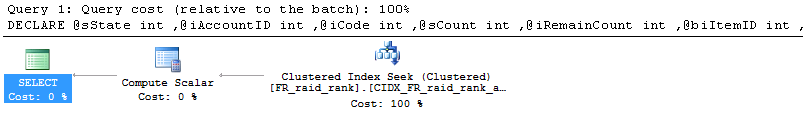
적절한 인덱스가 없습니다.

--방안

FR\_raid\_rank 의 account\_id 에 인덱스 생성

CREATE CLUSTERED INDEX CIDX\_FR\_raid\_rank\_account\_id ON dbo.FR\_raid\_rank(account\_id);

--튜닝 후



Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT TOP 5 a.account\_id, p.nickname, a.gold, DATEPART(YEAR, a.date), DATEPART(MONTH, a.date)

, DATEPART(DAY, a.date), DATEPART(HOUR, a.date), DATEPART(MINUTE, a.date)

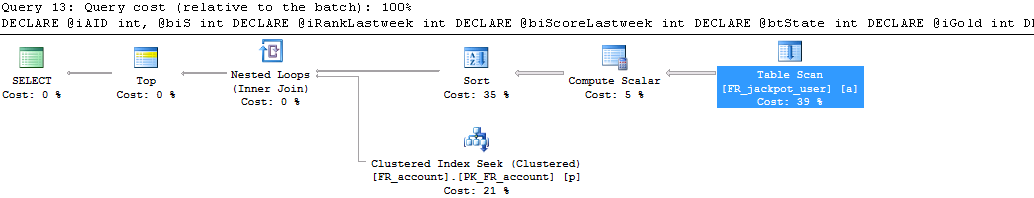
FROM FR\_jackpot\_user AS a

INNER JOIN FR\_account AS p ON p.account\_id = a.account\_id

WHERE a.stage = @iStage ORDER BY a.date DESC

--증상

대량 읽기가 발생 할 우려가 있습니다.



--원인

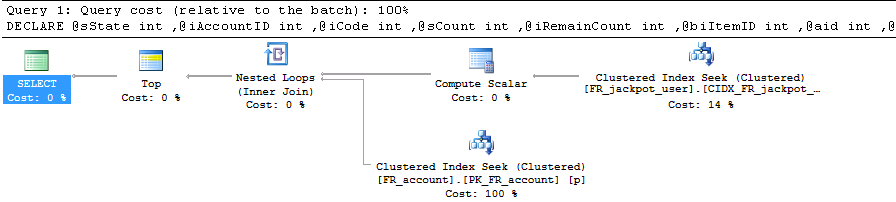
적절한 인덱스가 없습니다.

--방안

FR\_jackpot\_user 의 stage, date 에 인덱스 생성

CREATE CLUSTERED INDEX CIDX\_FR\_jackpot\_user\_stage ON dbo.FR\_jackpot\_user(stage, [date]) WITH(ONLINE = ON);

--튜닝 후



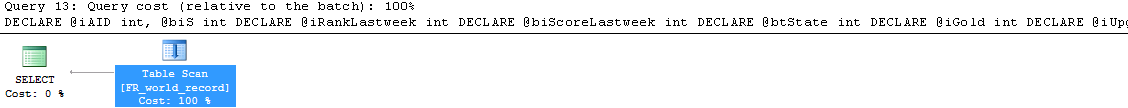
Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.

--문제 쿼리

SELECT @before\_length = [length], @before\_weight = [weight] FROM FR\_world\_record WHERE code = @iCode

--증상

대량 읽기가 발생 할 우려가 있습니다.



--원인

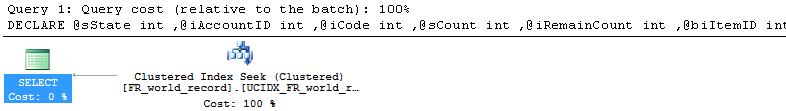
적절한 인덱스가 없습니다.

--방안

FR\_world\_record 의 code 에 인덱스 생성

CREATE UNIQUE CLUSTERED INDEX UCIDX\_FR\_world\_record\_code ON dbo.FR\_world\_record(code);

--튜닝 후



Index 사용으로 Query가 실행 될 때 Table 전체에 대한 Lock이 발행하지 않고 적은 읽기 수로 대상을 찾을 수 있습니다.