**Question 1:**

USE springboardopt;

-- -------------------------------------

SET @v1 = 1612521;

SET @v2 = 1145072;

SET @v3 = 1828467;

SET @v4 = 'MGT382';

SET @v5 = 'Amber Hill';

SET @v6 = 'MGT';

SET @v7 = 'EE';

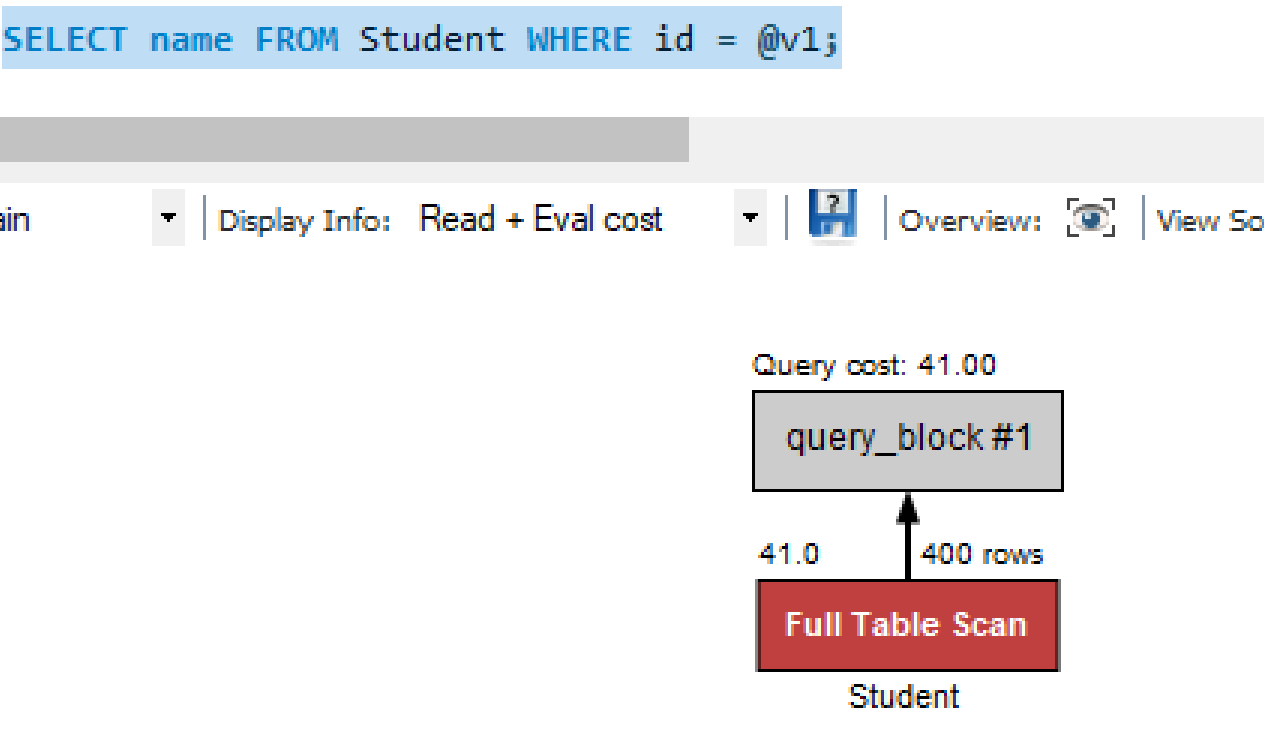
SET @v8 = 'MAT';

-- 1. List the name of the student with id equal to v1 (id).

SELECT name FROM Student WHERE id = @v1;

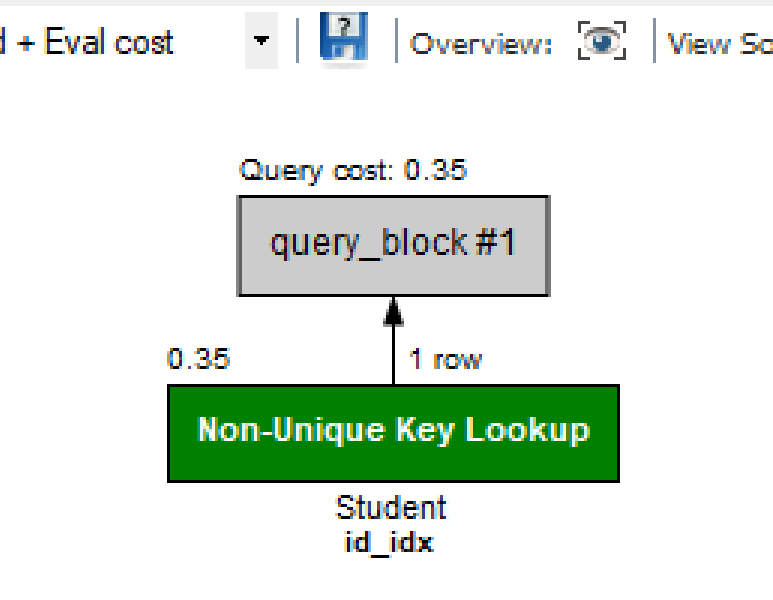
**Answer 1:**

Issue with the Query is full table scan:



**Index is missing from ID column.**

**CREATE INDEX id\_idx ON Student (id);**



**Question 2:**

USE springboardopt;

-- -------------------------------------

SET @v1 = 1612521;

SET @v2 = 1145072;

SET @v3 = 1828467;

SET @v4 = 'MGT382';

SET @v5 = 'Amber Hill';

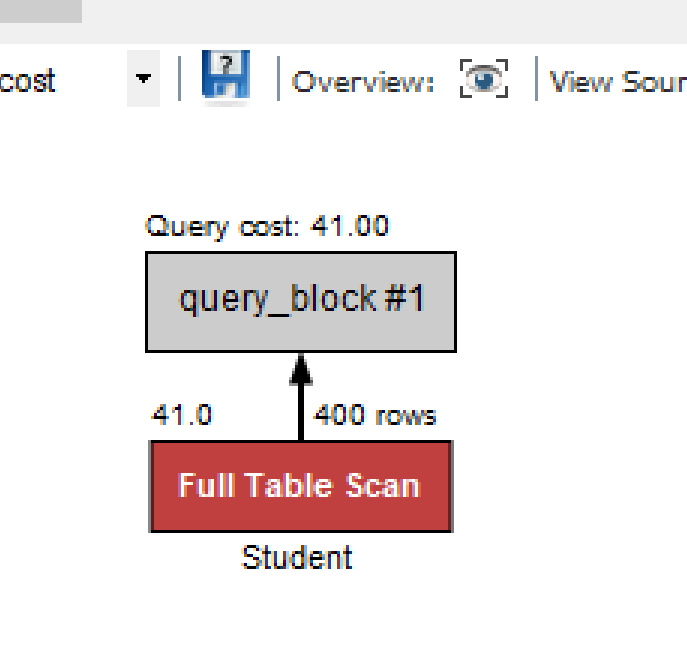
SET @v6 = 'MGT';

SET @v7 = 'EE';

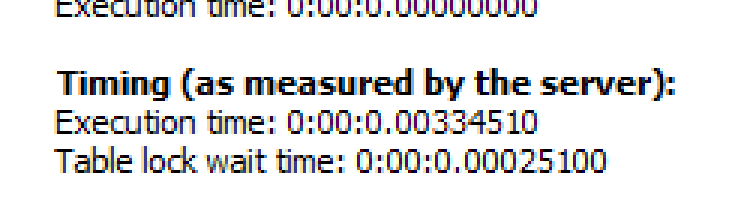
SET @v8 = 'MAT';

-- 2. List the names of students with id in the range of v2 (id) to v3 (inclusive).

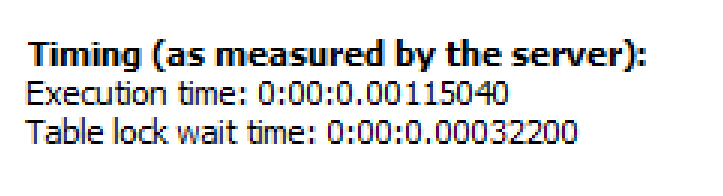
SELECT name FROM Student WHERE id BETWEEN @v2 AND @v3;

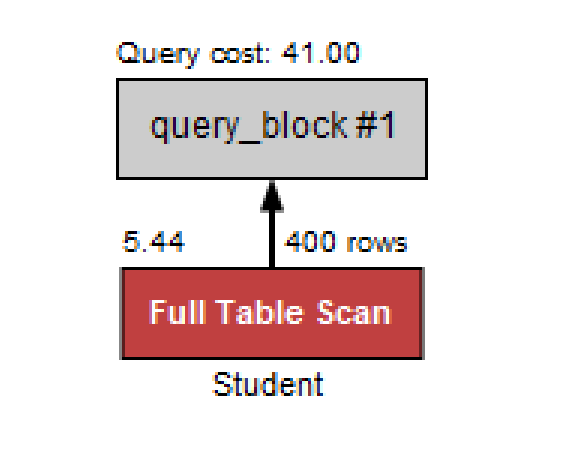


Time with index on id column



Time without index on id column





Total rows = 400

Rows as per where clause = 278

More than 50% of data is extracted hence better to use full table scan

**Question 3:**

USE springboardopt;

-- -------------------------------------

SET @v1 = 1612521;

SET @v2 = 1145072;

SET @v3 = 1828467;

SET @v4 = 'MGT382';

SET @v5 = 'Amber Hill';

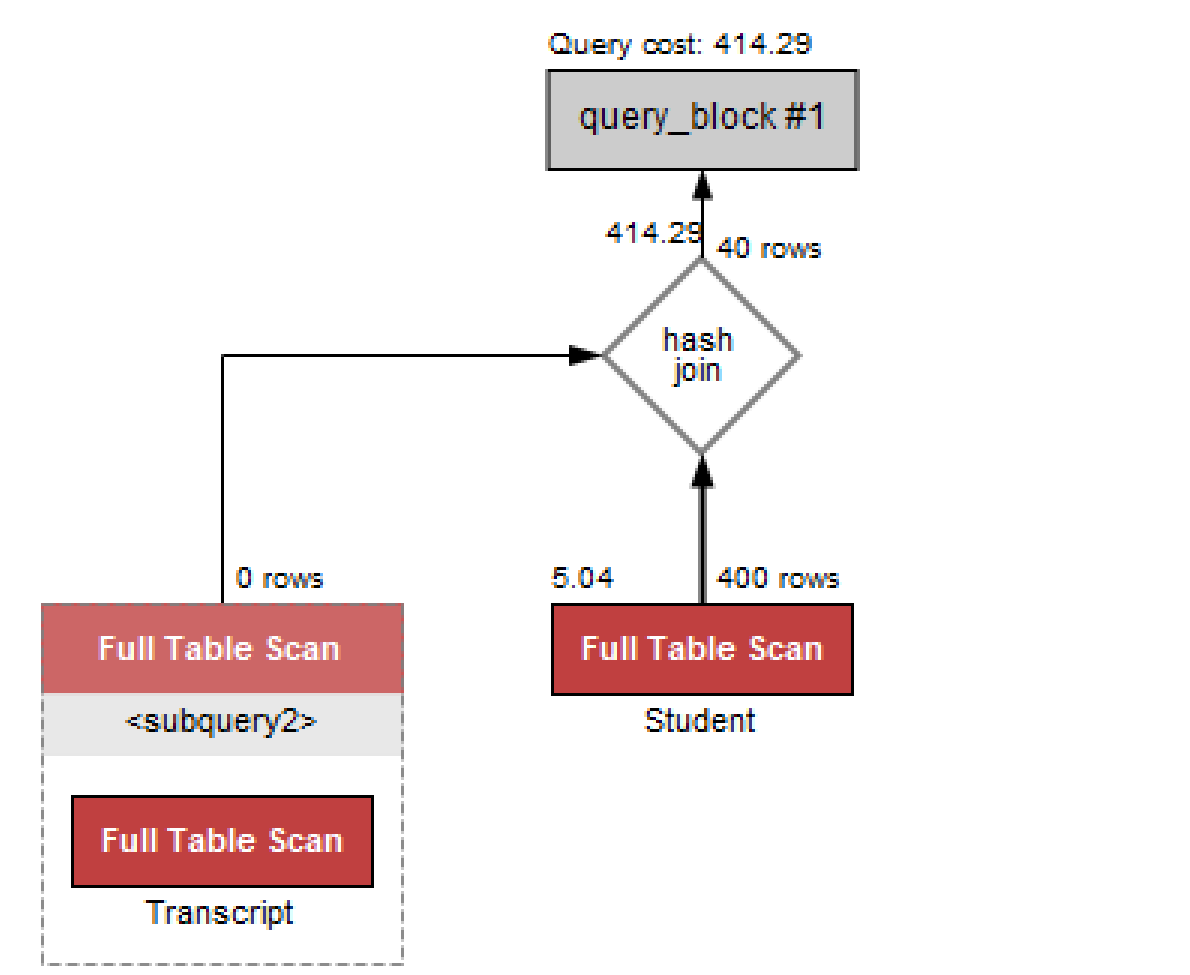
SET @v6 = 'MGT';

SET @v7 = 'EE';

SET @v8 = 'MAT';

-- 3. List the names of students who have taken course v4 (crsCode).

SELECT name FROM Student WHERE id IN (SELECT studId FROM Transcript WHERE crsCode = @v4);



**After creating indexes and re-writing the query:**

create index id\_idx on student (id);

create index crsCode\_idx on Transcript (crsCode);

create index studId\_idx on Transcript (studId);

select std.name

from Student std left outer join Transcript ts

on std.id=ts.studId

where ts.crsCode= @v4;

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Question 4:**  USE springboardopt;  -- -------------------------------------  SET @v1 = 1612521;  SET @v2 = 1145072;  SET @v3 = 1828467;  SET @v4 = 'MGT382';  SET @v5 = 'Amber Hill';  SET @v6 = 'MGT';  SET @v7 = 'EE';  SET @v8 = 'MAT';  -- 4. List the names of students who have taken a course taught by professor v5 (name).  SELECT name FROM Student,  (SELECT studId FROM Transcript,  (SELECT crsCode, semester FROM Professor  JOIN Teaching  WHERE Professor.name = @v5 AND Professor.id = Teaching.profId) as alias1  WHERE Transcript.crsCode = alias1.crsCode AND Transcript.semester = alias1.semester) as alias2  WHERE Student.id = alias2.studId;    create index name\_idx on professor(name);  create index id\_idx on professor(id);  create index profId\_idx on teaching(profId);  create index semester\_idx on transcript(semseter);  WITH Stud as (select tp.studId  from teaching t , professor p , transcript tp  where p.id = t.profId  and t.semester = tp.semester  and t.crsCode = tp.crsCode  and p.name = @v5)  select name from Stud inner join student on stud.studId = student.id;      **Question 5:**  USE springboardopt;  -- -------------------------------------  SET @v1 = 1612521;  SET @v2 = 1145072;  SET @v3 = 1828467;  SET @v4 = 'MGT382';  SET @v5 = 'Amber Hill';  SET @v6 = 'MGT';  SET @v7 = 'EE';  SET @v8 = 'MAT';  -- 5. List the names of students who have taken a course from department v6 (deptId), but not v7.  with alias as (SELECT studId FROM Transcript, Course WHERE deptId = @v6 AND Course.crsCode = Transcript.crsCode  AND studId NOT IN  (SELECT studId FROM Transcript, Course WHERE deptId = @v7 AND Course.crsCode = Transcript.crsCode))  select \* from student , alias  where Student.id = alias.studId;    create index name\_idx on professor(name);  create index id\_idx on professor(id);  create index profId\_idx on teaching(profId);  create index semester\_idx on transcript(semseter);  create index dept\_idx on course(deptId); |  |  |  |  |  |

**Question 6**

USE springboardopt;

-- -------------------------------------

SET @v1 = 1612521;

SET @v2 = 1145072;

SET @v3 = 1828467;

SET @v4 = 'MGT382';

SET @v5 = 'Amber Hill';

SET @v6 = 'MGT';

SET @v7 = 'EE';

SET @v8 = 'MAT';

-- 6. List the names of students who have taken all courses offered by department v8 (deptId).

SELECT name FROM Student,

(SELECT studId

FROM Transcript

WHERE crsCode IN

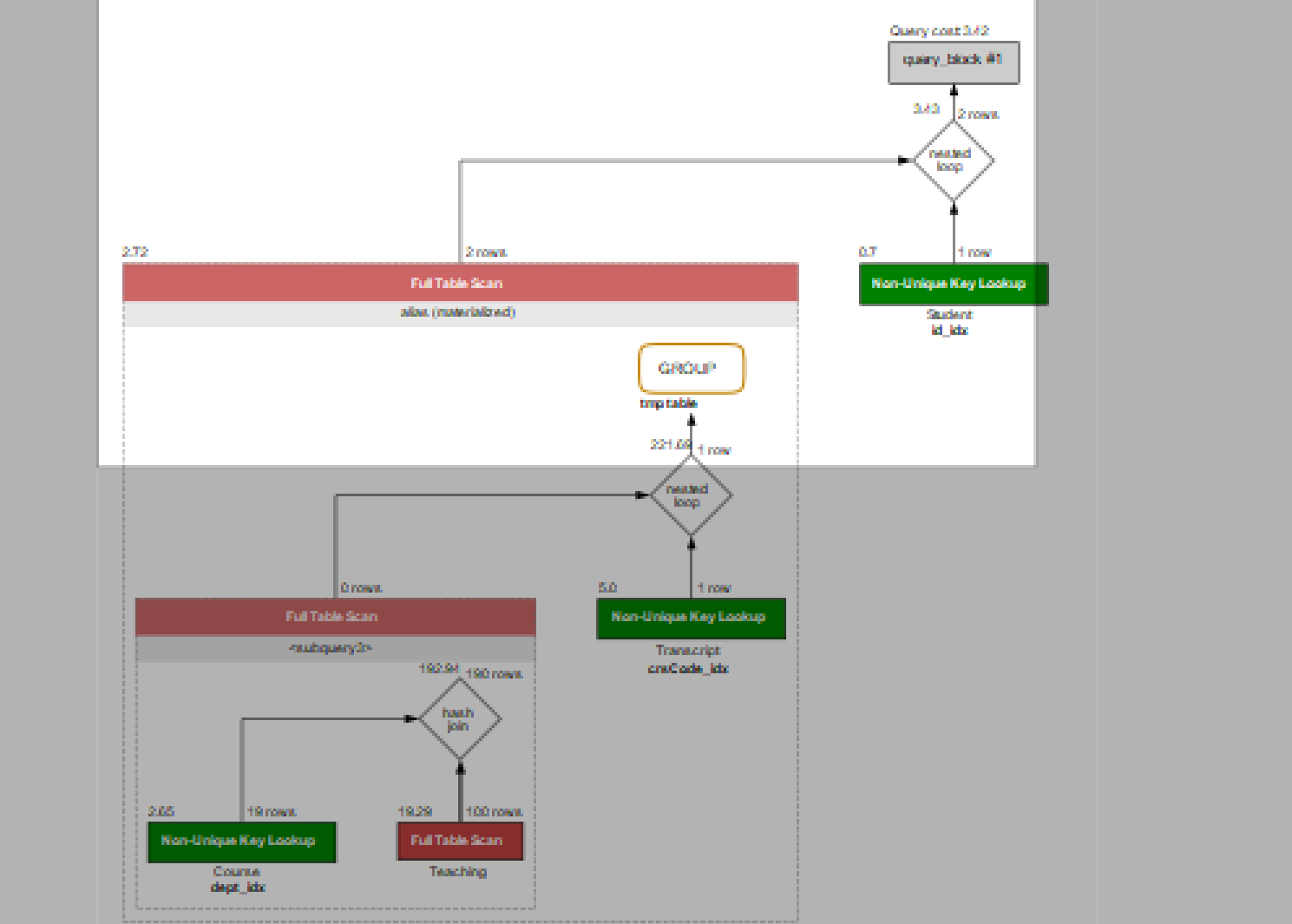
(SELECT crsCode FROM Course WHERE deptId = @v8 AND crsCode IN (SELECT crsCode FROM Teaching))

GROUP BY studId

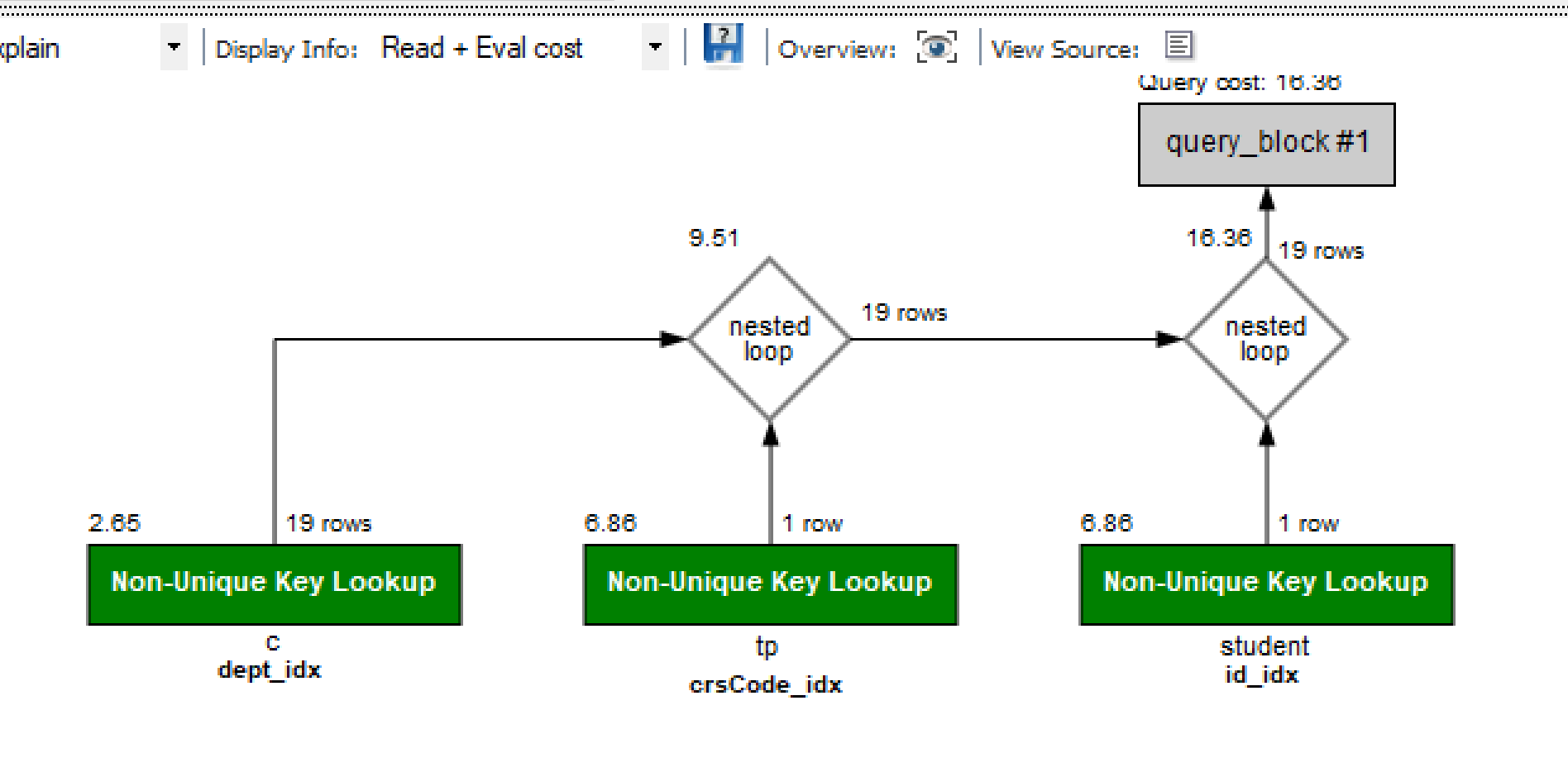
HAVING COUNT(\*) =

(SELECT COUNT(\*) FROM Course WHERE deptId = @v8 AND crsCode IN (SELECT crsCode FROM Teaching))) as alias

WHERE id = alias.studId;



**New improved query**



with STUD\_CTE as (select tp.studId as id from

course c , transcript tp

where c.crscode = tp.crscode

and deptId = 'MAT')

select name from student inner join STUD\_CTE

on student.id = STUD\_CTE.id;