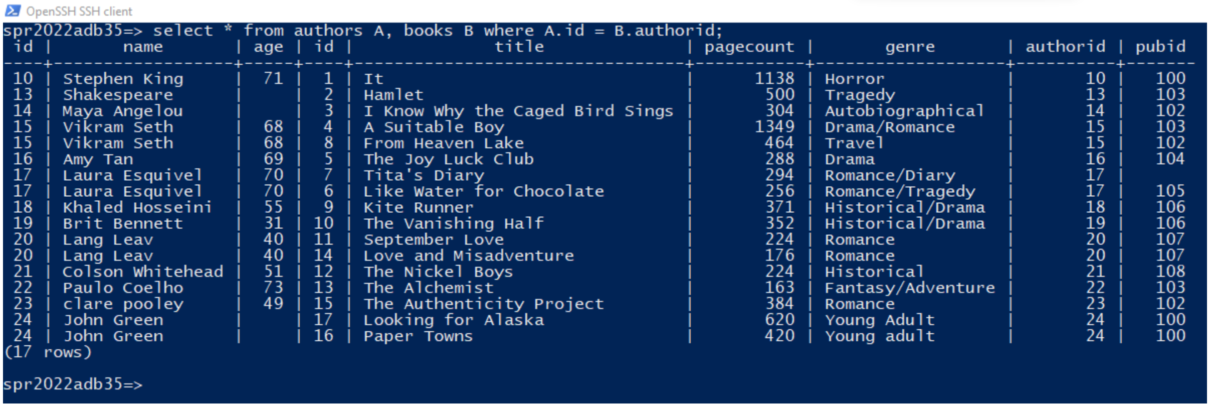
Video Activities- Week-9

Submitted by: - Parth Parashar

Video 15.1: -

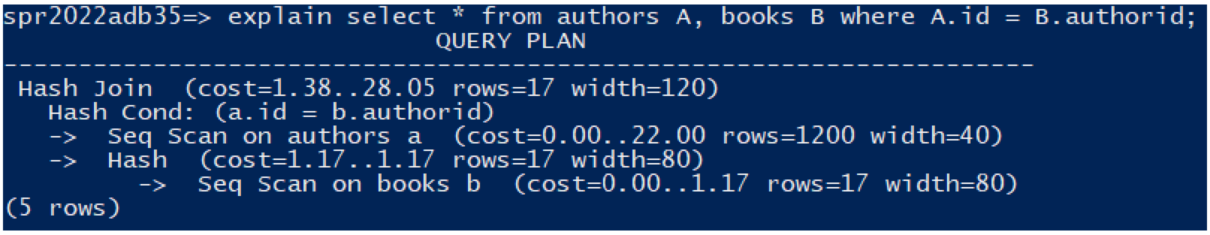
select * from authors A, books B where A.id = B.authorid;



```
spr2022adb35=> select * from authors A, books B where A.id = B.authorid;
 id |       name       | age | id |              title              | pagecount |       genre       | authorid | pubid
----+------------------+-----+----+---------------------------------+-----------+-------------------+----------+-------
 10 | Stephen King     |  71 |  1 | It                              |      1138 | Horror            |       10 |   100
 13 | Shakespeare      |     |  2 | Hamlet                          |       500 | Tragedy           |       13 |   103
 14 | Maya Angelou     |     |  3 | I Know Why the Caged Bird Sings |       304 | Autobiographical  |       14 |   102
 15 | Vikram Seth      |  68 |  4 | A Suitable Boy                  |      1349 | Drama/Romance     |       15 |   103
 15 | Vikram Seth      |  68 |  8 | From Heaven Lake                |       464 | Travel            |       15 |   102
 16 | Amy Tan          |  69 |  5 | The Joy Luck Club               |       288 | Drama             |       16 |   104
 17 | Laura Esquivel   |  70 |  7 | Tita's Diary                    |       294 | Romance/Diary     |       17 |
 17 | Laura Esquivel   |  70 |  6 | Like Water for Chocolate        |       256 | Romance/Tragedy   |       17 |   105
 18 | Khaled Hosseini  |  55 |  9 | Kite Runner                     |       371 | Historical/Drama  |       18 |   106
 19 | Brit Bennett     |  31 | 10 | The Vanishing Half              |       352 | Historical/Drama  |       19 |   106
 20 | Lang Leav        |  40 | 11 | September Love                  |       224 | Romance           |       20 |   107
 20 | Lang Leav        |  40 | 14 | Love and Misadventure           |       176 | Romance           |       20 |   107
 21 | Colson Whitehead |  51 | 12 | The Nickel Boys                 |       224 | Historical        |       21 |   108
 22 | Paulo Coelho     |  73 | 13 | The Alchemist                   |       163 | Fantasy/Adventure |       22 |   103
 23 | clare pooley     |  49 | 15 | The Authenticity Project        |       384 | Romance           |       23 |   102
 24 | John Green       |     | 17 | Looking for Alaska              |       620 | Young Adult       |       24 |   100
 24 | John Green       |     | 16 | Paper Towns                     |       420 | Young adult       |       24 |   100
(17 rows)

spr2022adb35=>
```

For finding out the cost and hash value, we can run the select query as given below: -



```
spr2022adb35=> explain select * from authors A, books B where A.id = B.authorid;
                                QUERY PLAN
---------------------------------------------------------------------
 Hash Join  (cost=1.38..28.05 rows=17 width=120)
   Hash Cond: (a.id = b.authorid)
   ->  Seq Scan on authors a  (cost=0.00..22.00 rows=1200 width=40)
   ->  Hash  (cost=1.17..1.17 rows=17 width=80)
         ->  Seq Scan on books b  (cost=0.00..1.17 rows=17 width=80)
(5 rows)
```

The steps followed for building a hash table are: -

1) Take the join attribute
2) Hash value = hash function(join attribute)
3) Insert the tuple into hash table based on the hash value.

Hashing works for the equality comparisons.

Simple case: - Cost: M+ N

Harder case: - Cost: 3(M+N)

Video 15.2: -

Join Activity

Select * from authors A, books B where A.id = B.authorid; (index on Authors.id)

In this query, I will use the INL or Hash or Sort Merge. For hash, Hash table on authors.

Select * from authors A, books B where A.id = B.authorid; (No index)

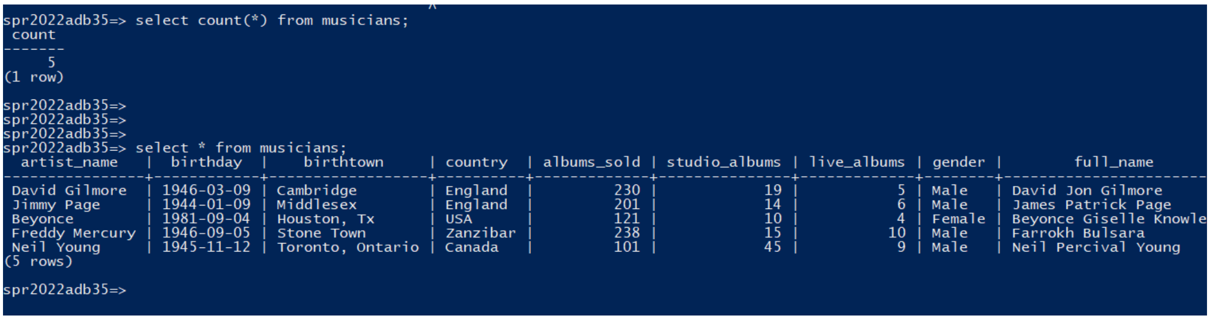
For this query, I will use Hash or Sort Merge. For hash, Hash Table on authors.

Block nested loops is super efficient and works with non-equi joins as well. Hash Join is efficient if one relation fits in memory and works only with equi-join.

Video 15.3: -

Aggregate Hash

Select count(*) from musicians group by instruments;



```
spr2022adb35=> select count(*) from musicians;
 count
-------
     5
(1 row)

spr2022adb35=>
spr2022adb35=>
spr2022adb35=>
spr2022adb35=> select * from musicians;
  artist_name   |  birthday  |    birthtown     | country  | albums_sold | studio_albums | live_albums | gender |       full_name
----------------+------------+------------------+----------+-------------+---------------+-------------+--------+----------------------
 David Gilmore  | 1946-03-09 | Cambridge        | England  |         230 |            19 |           5 | Male   | David Jon Gilmore
 Jimmy Page     | 1944-01-09 | Middlesex        | England  |         201 |            14 |           6 | Male   | James Patrick Page
 Beyonce        | 1981-09-04 | Houston, Tx      | USA      |         121 |            10 |           4 | Female | Beyonce Giselle Knowle
 Freddy Mercury | 1946-09-05 | Stone Town       | Zanzibar |         238 |            15 |          10 | Male   | Farrokh Bulsara
 Neil Young     | 1945-11-12 | Toronto, Ontario | Canada   |         101 |            45 |           9 | Male   | Neil Percival Young
(5 rows)

spr2022adb35=>
```

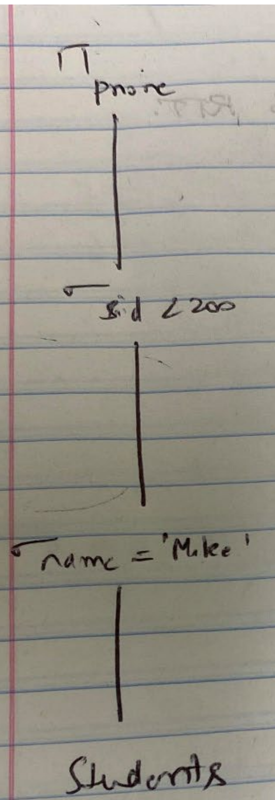
Result is: - (instrument, count)
(drums, 1)
(guitar, 1)
(vocals, 2)

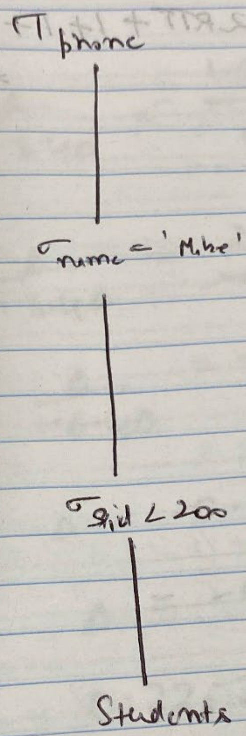
Activity 16.1: -

Q-1: - select name, phone from students where sid = 1236;

Q-2: - select phone from students where name = ‘Mike’ and sid < 200;

Q-3: - select name , cid from students S JOIN isTaking I on S.sid = I.sid where name = ‘Xayne’;





$\sigma_{c_1}(\sigma_{c_2}(A)) = \sigma_{c_2}(\sigma_{c_1}($

Select commute

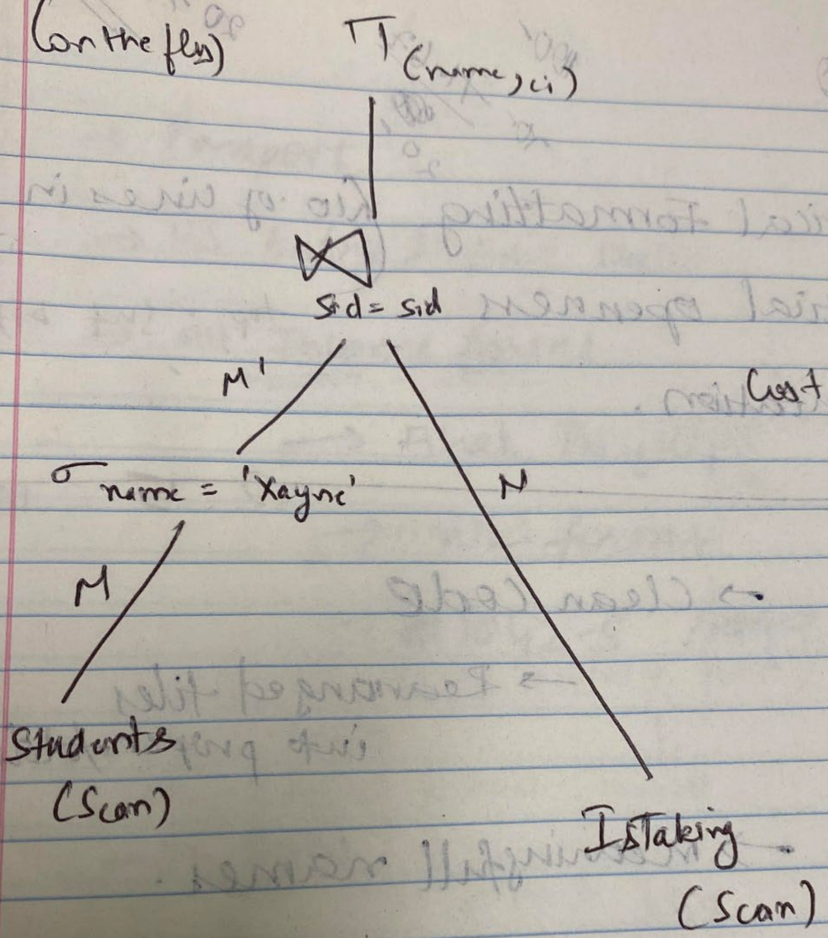
↓

RA Equivalen

to transform

on left to one

Right.


(on the fly)
$\Pi_{(name, ci)}$
$\bowtie_{sid = sid}$
M'
N
$\sigma_{name = 'Xayne'}$
M
Students
(Scan)
IsTaking
(Scan)

Cost = 3*(M+N)

Activity 16.2: -

1) How many rows does this query return?
   a) 1 or 0
   b) Assuming sid is the primary key or sid is unique.
   c) Look at constraints.
   d) Or look at # distinct values in sid column compared to the # of tuples
2) Select * from students where sid < 1236;
   a) Probably a lot
   b) Need to know how many sids exist / # of students.
   c) Are they sequentially assigned or removed
   d) Things in catalogue that might help
      - Min and max sid
      - # distinct values in sid column
      - Histogram of the distribution of the sid values