Homework # 5

Ruchi Panwar (rpanwar@kent.edu)

Advance Database

Fall 2015

Professor: Dr. Gregory S DeLozier

Oct 20, 2015

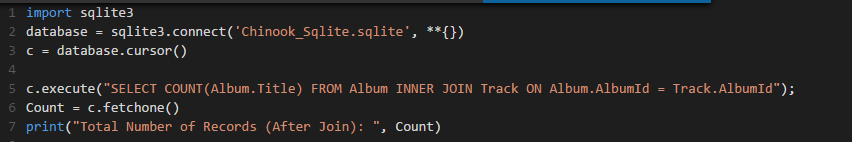
**Chinook Database Homework**

Code can be found on below Github Link - <https://github.com/rpanwar01/Database/blob/master/Lecture%207%20%26%208%20%26%20Chinook%20Database%20homework/Chinook_Datbase_Homework/Homework.py>

I have used Tables **Track** and **Album** for this assignment. I have created the query to get the Album name based on Track.Name and Track.Composer.

**Step 1: Join**

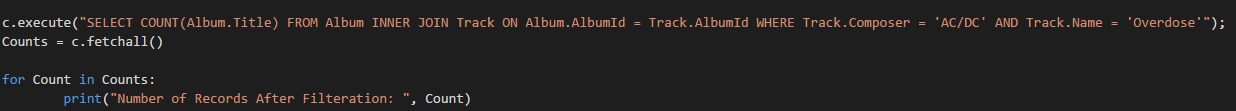
In first step I've used inner join for tables Album and tack based on AlbumId and the number of records returned are 3503.





**Step 2: Filtration**

In the next step, I've used the filter where Track.Name ='Overdose' and Track.Composer = 'AC/DC' that returns only 1 record.

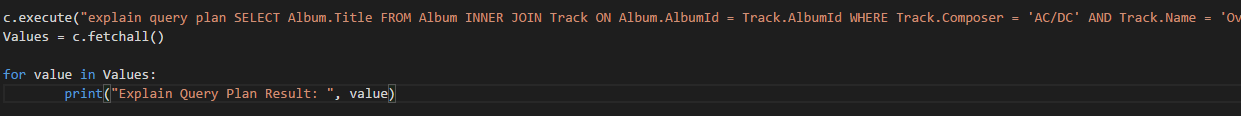




**Step 3: Explain Query Plan**

If you can see the result below, query is scanning the table Track that has 3503 records and then searching the corresponding value in table Album based on primary key.

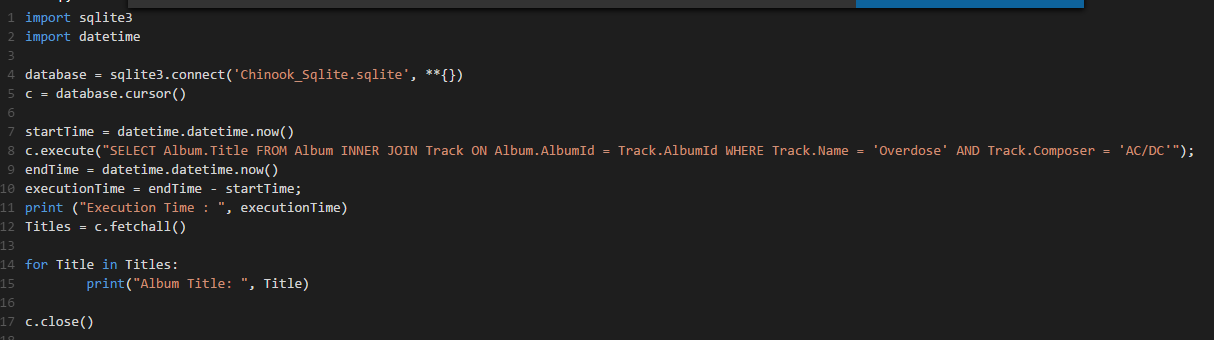
No Indexed value has been used.



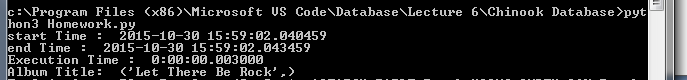


**Step 4: Execution Time**

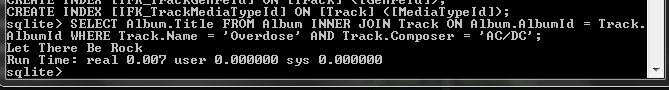
To capture the execution time, I'm using datetime function in python and .Time ON function in Sqlite3.



**Using Python code :** Execution time = 0.003000 seconds

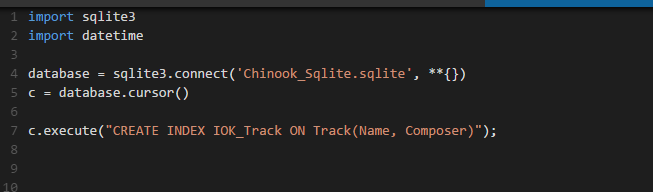


**Using SqLite3 Prompt :** Execution time = .007 seconds

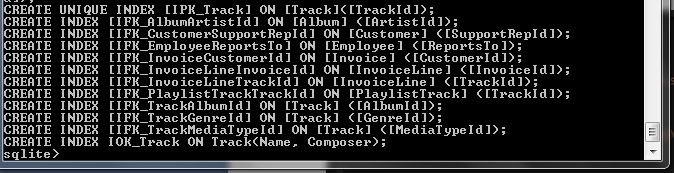


**Step 5: Query Optimization**

To optimize the search, I've created the index for table Track based on Track.Name and Track.Composer.

****

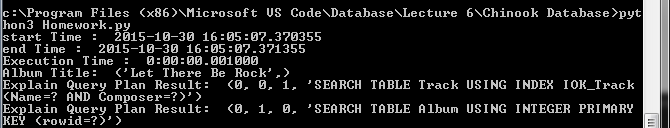
Entry got added to Chinook Database Schema.

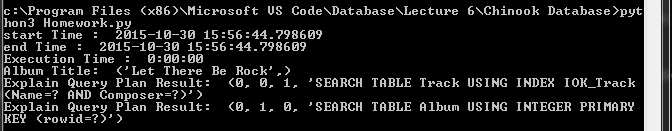
****

**Step 6: Optimization Query Execution Time**

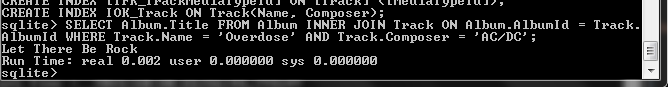
**Using Python code**

After the optimization the execution time is showing as 0.001000 seconds or sometime showing as 0 as well.



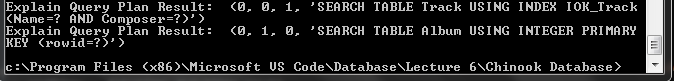


**Using SqLite3 Prompt** Execution time = .002 seconds



**Step 7: Explain Query Plan after Optimization**

As you can see in the below screenshot after optimization (adding the multi-column index), the query is searching the records based on index IOK\_Track rather than scanning the whole Track table.



.**Step 7: Final Result**

As you can see in above screenshots that after adding the index the execution time has come down to one-third (and less) of the actual time.

**Step 8: Time Vs Standard Deviation Plot**

I have used the execution time of the query before and after the creating the Index to plot the graph.