Problem Set B Submission Form

Overview

Your Name	Hendi Kushta
Your SU Email	hkushta@syr.edu

Instructions

Put your name and SU email at the top. Answer these questions all from the lab. When asked to include screenshots, please follow the screen shot guidelines from the first homework.

Remember as you complete the homework it is not only about getting it right / correct. We will discuss the answers in class so it's important to articulate anything you would like to contribute to the discussion in your answer:

- If you feel the question is vague, include any assumptions you've made.
- If you feel the answer requires interpretation or justification provide it.
- If you do not know the answer to the question, articulate what you tried and how you are stuck.
- Highlight any doubts or questions you would like me to review.

This how you receive credit for answering questions which might not be correct. In addition, you must complete the reflection portion of the homework assignment for full credit. Since most answers will be similar this is an important part of your individual submission.

Complete Part II of this document first, then go back and complete the Reflection in Part I.

Part I - Reflection

Use this section to reflect on your learning. To achieve the highest grade on the assignment you must be as descriptive and personal as possible with your reflection.

1. As you completed this assignment, identify what have you learned?

2. What barriers or challenges did you encounter while completing this assignment?

   I tried to understand the query plans, but I face difficulties in reading them.

3. How prepared were you to complete this assignment? What can you do to be better prepared?
   More examples on reading and understanding the query plans.

4. Rate your comfort level with this week’s material. Use the rubric provided. Highlight the number representing your ability.

4 ==> I understand this material and can explain it to others.
3 ==> I understand this material.
2 ==> I somewhat understand the material but sometimes need guidance from others.
1 ==> I understand very little of this material and need extra help.

Part II – Questions

Paste your answers to the Exercises found in the lab document. Make sure to include your netid in any screenshots you provide (no screenshots needed for this first lab). If the question asks for commands, only include those commands which are necessary to complete the tasks. Number each answer.

1. List the Product ID, Category ID, product name and product unit price for products that are not discontinued.

```
SELECT ProductID, CategoryID, ProductName, UnitPrice
    FROM Products
    WHERE Discontinued = 0
```

2. Join the categories table so that you display the category name in the first query.

```
SELECT P.ProductID, P.CategoryID, C.CategoryName, P.ProductName, P.UnitPrice
    FROM Products P
    JOIN Categories C ON P.CategoryID = C.CategoryID
    WHERE P.Discontinued = 0;
```

3. Produce a query displaying the category ID, category name, and average product unit price.

```
SELECT C.CategoryID, C.CategoryName, AVG(P.UnitPrice) AS Avg_Product_Unit_Price
    FROM Products P
    JOIN Categories C ON P.CategoryID = C.CategoryID
    GROUP BY C.CategoryID, C.CategoryName;
```

4. Next, combine queries 2 and 3 with a join. Join on category ID so you can display product id, category id, product name, product unit price, and average product unit price, then subtract the unit price from the average.

   HINT: Use a common table expression WITH clause to create two named queries then use them as tables to join for the last query.

```
WITH table1 AS (
    SELECT P.ProductID, P.CategoryID, C.CategoryName, P.ProductName, P.UnitPrice
    FROM Products P
    JOIN Categories C ON P.CategoryID = C.CategoryID
    WHERE P.Discontinued = 0
```

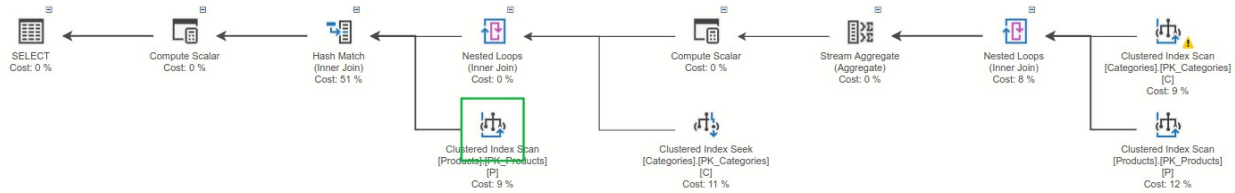
```
),
table2 AS (
        SELECT C.CategoryID, C.CategoryName, AVG(P.UnitPrice) AS Avg_Product_Unit_Price
        FROM Products P
        JOIN Categories C ON P.CategoryID = C.CategoryID
        WHERE P.Discontinued = 0
        GROUP BY C.CategoryID, CategoryName
)
SELECT t1.ProductID, t1.CategoryID, t1.ProductName, t1.UnitPrice AS Product_Unit_Price,
        t2.Avg_Product_Unit_Price,
        t1.UnitPrice - t2.Avg_Product_Unit_Price AS Diff_Betw_Unit_Price_and_Avg_Product_Unit_Price
FROM table1 t1
JOIN table2 t2 ON t1.CategoryID = t2.CategoryID;
```

5. Re-write the query in 4 to use a window function instead of 3 queries.

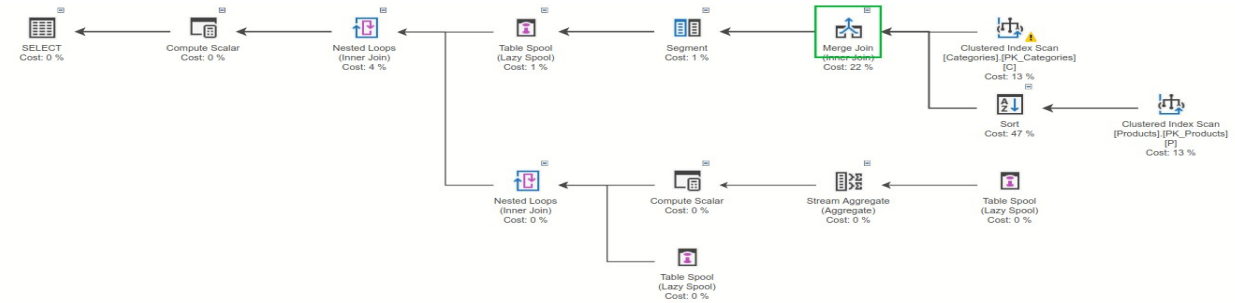
```
SELECT
        ProductID,
        C.CategoryID,
        ProductName,
        UnitPrice AS Product_Unit_Price,
        AVG(UnitPrice) OVER (PARTITION BY C.CategoryID) AS Avg_Product_Unit_Price,
        UnitPrice - AVG(UnitPrice) OVER (PARTITION BY C.CategoryID) AS
Diff_Betw_Unit_Price_and_Avg_Product_Unit_Price
        FROM Products P
        JOIN Categories C ON P.CategoryID = C.CategoryID
        WHERE Discontinued = 0;
```

6. Compare the query plans of 4 and 5. Are they the same?

Query plan for exercise 4



Query plan for exercise 5



The query plans are different. I need help on understanding how to read the plans.

7. What is a good index candidate for query 4 or 5? How would it improve the performance of the query?

   For both Query 4 and Query 5, a good index candidate to consider would be a composite index on the columns used in the join and filtering conditions. Specifically, an index on the columns involved in the join and the Discontinued column could be beneficial.
   This index includes the CategoryID column used in the join conditions and the Discontinued column used in the filtering conditions. Having an index on these columns can improve the performance of the queries by allowing the database engine to quickly locate and retrieve the relevant rows during the join and filtering processes.