

Zadanie - wykład 3

Mikołaj Kubś, 272662

23 marca 2025

PODSUMOWANIE SQL - ZADANIA

BAZA: ADVENTUREWORKS

1. Jaka była łączna suma transakcji (SalesOrderHeader.SubTotal) w poszczególnych latach dla kolejnych dni tygodnia?
2. Zaproponuj podział klientów na 3 rozłączne grupy wiekowe. Ilu różnych klientów dokonało zakupów w kolejnych miesiącach roku w każdej z grup? Ilu klientów w poszczególnych grupach wykonało zakup dokładnie jeden raz?
3. Przygotuj zestawienie produktów, których sprzedaje się miesięcznie min. 20 sztuk. Dla każdego produktu podaj jego kategorię.

Przeanalizuj uzyskane wyniki.

Jeśli w zapytaniu warto użyć CTE, to porównaj efektywność swojego rozwiązania z wersją bez CTE.

Rysunek 1: Opis zadań 1-3

PODSUMOWANIE SQL - ZADANIA

4. Przygotuj zestawienie, w którym przeanalizujesz, ile jest różnych klientów dla każdej płci w kolejnych miesiącach (05.2011 – 06.2024)? Jak procentowo rozkłada się ich udział w całkowitej wartości sprzedaży (Sales.SalesOrderHeader.TotalDue)?
5. Przeanalizuj udział sprzedanych produktów w poszczególnych podkategoriach w stosunku do całych kategorii (zarówno pod względem liczbowym jak i wartościowym).
6. Przygotuj zestawienie, w którym możliwa będzie analiza regionalna z uwzględnieniem lokalnej waluty (kwoty sprzedaży w zależności od waluty i regionu).

Przeanalizuj uzyskane wyniki.

Rysunek 2: Opis zadań 4-6

1 Kod kwerend 1-3

1.1 Kwerenda 1

"Jaka była łączna suma transakcji (SalesOrderHeader.SubTotal) w poszczególnych latach dla kolejnych dni tygodnia?"

```
1 SET DATEFIRST 1;
2 SET LANGUAGE Polish;
3
4 SELECT
5     SUM(SubTotal) AS "Suma",
6     DATENAME(DW, OrderDate) AS "Dzień tygodnia",
7     DATEPART(YEAR, OrderDate) AS "Rok"
8 FROM Sales.SalesOrderHeader
9 GROUP BY
10    DATENAME(DW, OrderDate),
11    DATEPART(YEAR, OrderDate),
12    DATEPART(DW, OrderDate)
13 ORDER BY DATEPART(YEAR, OrderDate), DATEPART(DW, OrderDate)
```

	Suma	Dzień tygodnia	Rok
1	3443079,9645	poniedziałek	2011
2	1080780,5945	wtorek	2011
3	1372341,7495	środa	2011
4	1245302,0837	czwartek	2011
5	2068459,2198	piątek	2011
6	2889264,3005	sobota	2011
7	519444,0004	niedziela	2011
8	4303006,4943	poniedziałek	2012
9	5999143,0718	wtorek	2012
10	4161627,6385	środa	2012
11	2692535,6542	czwartek	2012
12	4570299,305	piątek	2012
13	4530528,4594	sobota	2012
14	7266760,7028	niedziela	2012
15	6653314,1953	poniedziałek	2013
16	6228219,3549	wtorek	2013
17	9039178,7091	środa	2013
18	6046847,6099	czwartek	2013
19	3786578,9276	piątek	2013
20	6055191,4772	sobota	2013
21	5803147,7797	niedziela	2013
22	4564125,5899	poniedziałek	2014
23	1176647,534	wtorek	2014
24	4001988,8004	środa	2014
25	4628959,4393	czwartek	2014
26	1114231,4756	piątek	2014
27	3307937,8877	sobota	2014

Query executed successfully.

DESKTOP-2GLE46P (15.0 RTM) | DESKTOP-2GLE46P\jantar... | AdventureWorks2014 | 00:00:00 | 28 rows

Rysunek 3: Wyniki 1. kwerendy

W tym podpunkcie użycie CTE nie przyniosłoby korzyści.

Rok	Dw	Suma z Suma
2011	czwartek	12,483,020,837
	niedziela	5,394,440,004
	piątek	20,684,592,198
	poniedziałek	34,430,799,645
	sobota	28,892,643,005
	środa	13,723,417,495
	wtorek	10,807,808,945
2011 Suma		126,416,722,129
2012	czwartek	26,925,356,542
	niedziela	72,667,607,028
	piątek	4,570,299,305
	poniedziałek	43,030,064,943
	sobota	45,309,284,594
	środa	41,616,276,385
	wtorek	59,991,430,718
2012 Suma		294,110,319,515
2013	czwartek	60,468,476,099
	niedziela	58,031,477,797
	piątek	37,865,789,276
	poniedziałek	66,533,141,953
	sobota	60,651,914,772
	środa	90,391,797,091
	wtorek	62,282,193,549
2013 Suma		436,224,790,537
2014	czwartek	46,288,594,293
	niedziela	12,641,380,944
	piątek	11,142,314,756
	poniedziałek	45,641,255,899
	sobota	33,079,378,877
	środa	40,019,888,004
	wtorek	1,176,647,534
2014 Suma		189,989,460,307

Rysunek 4: Tabela przestawna dla 1. kwerendy

Dla łatwiejszej analizy dodano tabelę przestawną.

W 2012 i 2013 roku suma roczna szybko rosła. 2014 rok się jeszcze nie skończył, co częściowo wyjaśnia niższą sumę dla 2014 roku.

W 2011 roku przychody w niedzielę były 2 razy niższe niż w drugim najgorszym dniu tygodnia tego roku. Poniedziałek był zdecydowanie najlepszy, a sobota druga.

W 2012 roku piątek był dniem bardzo niskich przychodów - suma była ponad 16 razy mniejsza niż dla najlepszego dnia, niedzieli.

W 2013 roku przychody były bardziej wyrównane, gdzie piątek (najsłabszy dzień) był ponad 2 razy mniej dochodowy niż najlepszy (środa).

W 2014 roku wtorek osiągnął bardzo niski wynik, prawie 40 razy niższy od najlepszego (czwartku).

Wnioskiem jest to, że różne dni w różnych latach odbiegają od normy dla danego roku. Jednak nie w każdym roku różnice były aż tak widoczne.

1.2 Kwerenda 2

"Zaproponuj podział klientów na 3 rozłączne grupy wiekowe. Ilu różnych klientów dokonało zakupów w kolejnych miesiącach roku w każdej z grup? Ilu klientów w poszczególnych grupach wykonało zakup dokładnie jeden raz?"

Zdecydowano się na podział klientów na 3 rozłączne grupy wiekowe o możliwie najrówniejszej liczbie członków. Do zrealizowania tego użyto funkcji "NTILE". Grupa 1 to najmłodsi, a 3 to najstarsi.

Zadanie dotyczy tak naprawdę dwóch kwerend.

1.2.1 Ilu różnych klientów dokonało zakupów w kolejnych miesiącach roku w każdej z grup?

Wersja z CTE:

```
1 WITH CustomerOrders AS (  
2     SELECT  
3         Customer.CustomerID,  
4         NTILE(3) OVER (ORDER BY YEAR(GETDATE())) - YEAR(  
5             Person.Demographics.value(  
6                 'declare default element namespace  
7                 "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";  
8                 (//BirthDate)[1]',  
9                 'DATE'  
10            )  
11        )) AS AgeGroup,  
12        YEAR(SalesOrderHeader.OrderDate) AS OrderYear
```

```

13     MONTH(SalesOrderHeader.OrderDate) AS OrderMonth,
14 FROM Sales.Customer AS Customer
15 JOIN Person.Person AS Person ON Person.BusinessEntityID = Customer.PersonID
16 JOIN
17     Sales.SalesOrderHeader AS SalesOrderHeader ON
18     SalesOrderHeader.CustomerID = Customer.CustomerID
19 WHERE
20     Person.Demographics.exist(
21         'declare default element namespace
22         "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
23         (//BirthDate)[1]' ) = 1
24 )
25 SELECT
26     OrderYear AS "Rok",
27     OrderMonth AS "Miesiąc",
28     AgeGroup AS "Grupa wiekowa",
29     COUNT(DISTINCT CustomerID) AS "Liczba unikalnych klientów"
30 FROM CustomerOrders
31 GROUP BY AgeGroup, OrderYear, OrderMonth
32 ORDER BY OrderYear, OrderMonth, AgeGroup

```

Wersja bez CTE:

```

1 SELECT
2     AgeGroup AS "Grupa wiekowa",
3     OrderYear AS "Rok",
4     OrderMonth AS "Miesiąc",
5     COUNT(DISTINCT CustomerID) AS "Liczba unikalnych klientów"
6 FROM (
7     SELECT
8         Customer.CustomerID,
9         NTILE(3) OVER (ORDER BY YEAR(GETDATE()) - YEAR(
10             Person.Demographics.value(
11                 'declare default element namespace
12                 "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
13                 (//BirthDate)[1]',
14                 'DATE'
15             )
16         )) AS AgeGroup,
17         YEAR(SalesOrderHeader.OrderDate) AS OrderYear,
18         MONTH(SalesOrderHeader.OrderDate) AS OrderMonth
19 FROM Sales.Customer AS Customer
20 INNER JOIN Person.Person AS Person
21     ON Customer.PersonID = Person.BusinessEntityID

```

```

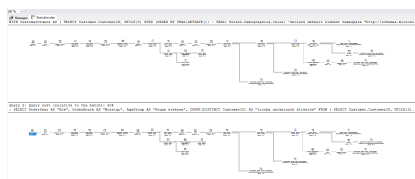
22     INNER JOIN Sales.SalesOrderHeader AS SalesOrderHeader
23         ON Customer.CustomerID = SalesOrderHeader.CustomerID
24     WHERE Person.Demographics.exist(
25         'declare default element namespace
26         "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
27         (//BirthDate)[1]' ) = 1
28 ) AS CustomerOrderData
29 GROUP BY AgeGroup, OrderYear, OrderMonth
30 ORDER BY OrderYear, OrderMonth, AgeGroup

```

Rysunek 5: Wyniki 2. kwerendy

Wersja "bez" CTE i z nim są bardzo podobne. Nie da się przenieść bezpośrednio NTILE'a i reszty logiki do jednego zapytania bez wcześniejszego wyliczenia grup wiekowych, ponieważ nie da się pogrupować po funkcji okienkowej. Nielogiczne byłoby naraz grupowanie i wybieranie. Jednym ze sposobów rozwiązania tego problemu mogłoby być wyliczenie wcześniej górnej granicy wieku dla 1. i 2. grupy, a następnie przypisanie klientów i grupowanie w jednym zapytaniu - nadal trzeba jednak coś wyliczyć wcześniej.

W większości miesięcy liczba unikalnych klientów jest względnie podobna do siebie dla każdej grupy. Zdarza się, że najliczniejsza grupa jest 2-3 razy liczniejsza od tej najmniej licznej w danym miesiącu. Często się potem w tym samym roku zdarza odwrotna sytuacja. W 2012 roku najmłodsza grupa jest najliczniejsza, a w 2014 najmniej liczna o dość podobną liczbę osób.



Rysunek 6: Porównanie execution plan

Obie kwerendy działają tak samo z wcześniej opisanych powodów.

1.2.2 Ilu klientów w poszczególnych grupach wykonało zakup dokładnie jeden raz?

Wersja z CTE:

```
1 WITH OrdersWithCustomerDetails AS (  
2     SELECT  
3         Customer.CustomerID,  
4         NTILE(3) OVER (ORDER BY YEAR(GETDATE()) - YEAR(  
5             Person.Demographics.value(  
6                 'declare default element namespace  
7 "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";  
8                 (//BirthDate)[1]',  
9                 'DATE'  
10            )  
11        )) AS AgeGroup,  
12        SalesOrderHeader.SalesOrderID,  
13        YEAR(SalesOrderHeader.OrderDate) AS OrderYear,  
14        MONTH(SalesOrderHeader.OrderDate) AS OrderMonth  
15    FROM Sales.Customer AS Customer  
16    INNER JOIN Person.Person AS Person  
17        ON Customer.PersonID = Person.BusinessEntityID  
18    INNER JOIN Sales.SalesOrderHeader AS SalesOrderHeader  
19        ON Customer.CustomerID = SalesOrderHeader.CustomerID  
20    WHERE Person.Demographics.exist(  
21        'declare default element namespace  
22 "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";  
23        (//BirthDate)[1]' ) = 1  
24 ),  
25 CustomersWithSingleOrder AS (  
26     SELECT  
27         CustomerID,  
28         AgeGroup  
29     FROM OrdersWithCustomerDetails  
30     GROUP BY CustomerID, AgeGroup  
31     HAVING COUNT(SalesOrderID) = 1  
32 )  
33 SELECT  
34     AgeGroup AS "Grupa wiekowa",  
35     COUNT(CustomerID) AS "Liczba klientów z pojedynczym zamówieniem"  
36 FROM CustomersWithSingleOrder  
37 GROUP BY AgeGroup  
38 ORDER BY AgeGroup
```

Wersja bez CTE:

```

1  SELECT
2      CustomerAgeGroup AS "Grupa wiekowa",
3      COUNT(*) AS "Liczba klientów z jednym zamówieniem"
4  FROM (
5      SELECT
6          CustomerID,
7          CustomerAgeGroup
8      FROM (
9          SELECT
10             SalesCustomer.CustomerID,
11             NTILE(3) OVER (ORDER BY YEAR(GETDATE()) - YEAR(
12                 PersonTable.Demographics.value(
13                     'declare default element namespace
14 "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
15                 (//BirthDate)[1]',
16                 'DATE'
17             )
18             )) AS CustomerAgeGroup,
19             SalesOrderHeader.SalesOrderID
20     FROM Sales.Customer AS SalesCustomer
21     INNER JOIN Person.Person AS PersonTable
22         ON SalesCustomer.PersonID = PersonTable.BusinessEntityID
23     INNER JOIN Sales.SalesOrderHeader AS SalesOrderHeader
24         ON SalesCustomer.CustomerID = SalesOrderHeader.CustomerID
25     WHERE PersonTable.Demographics.exist(
26         'declare default element namespace
27 "http://schemas.microsoft.com/sqlserver/2004/07/adventure-works/IndividualSurvey";
28         (//BirthDate)[1]'
29     ) = 1
30     ) AS OrdersPerCustomer
31     GROUP BY CustomerID, CustomerAgeGroup
32     HAVING COUNT(SalesOrderID) = 1
33 ) AS SingleOrderCustomers
34 GROUP BY CustomerAgeGroup
35 ORDER BY CustomerAgeGroup

```

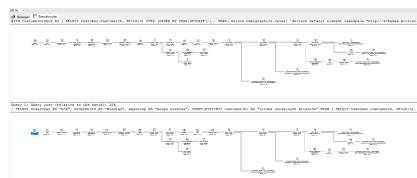

Results		Messages	Execution plan
Grupa wiekowa		Liczba klientów z 1 zamówieniem	
1	1	3585	
2	2	3784	
3	3	4250	

Grupa wiekowa		Liczba klientów z 1 zamówieniem	
1	1	3585	
2	2	3784	
3	3	4250	

Rysunek 7: Wyniki 2. kwerendy, część 2.

Wersja "bez" CTE i z CTE są bardzo podobne, z powodów opisanych wcześniej, dla pierwszej kwerendy w zadaniu 2. Dużym błędem byłoby tutaj wyliczenie najpierw klientów z jedną transakcją, a potem grupowanie ich NTILE'm na podstawie wyniku - NTILE wtedy analizowałby tylko wycinek całej populacji

Liczba klientów z jedną transakcją jest względnie podobna dla każdej grupy klientów. Najwięcej najstarszych klientów kupiło coś tylko raz. Najmniej młodych dokonało zakupu tylko raz.



Rysunek 8: Porównanie execution plan

Obie kwerendy działają tak samo z wcześniej opisanych powodów.

1.3 Kwerenda 3

"Przygotuj zestawienie produktów, których sprzedaje się miesięcznie min. 20 sztuk. Dla każdego produktu podaj jego kategorię"

Kwerenda z CTE:

```

1 WITH ProductCategories AS (
2     SELECT
3         ProductSubcategory.ProductSubcategoryID,
4         ProductCategory.Name AS CategoryName
5     FROM Production.ProductSubcategory
6     JOIN Production.ProductCategory

```

```

7      ON ProductSubcategory.ProductCategoryID = ProductCategory.ProductCategoryID
8  ),
9  MonthlySalesWithMin AS (
10     SELECT
11         Product.ProductID,
12         Product.ProductSubcategoryID,
13         Product.Name AS ProductName,
14         YEAR(SalesOrderHeader.OrderDate) AS SalesYear,
15         MONTH(SalesOrderHeader.OrderDate) AS SalesMonth,
16         SUM(SalesDetail.OrderQty) AS MonthlyQty,
17         MIN(SUM(SalesDetail.OrderQty)) OVER (PARTITION BY Product.ProductID)
18         AS MinMonthlyQty
19     FROM Sales.SalesOrderDetail AS SalesDetail
20     JOIN Production.Product AS Product
21         ON SalesDetail.ProductID = Product.ProductID
22     JOIN Sales.SalesOrderHeader AS SalesOrderHeader
23         ON SalesDetail.SalesOrderID = SalesOrderHeader.SalesOrderID
24     GROUP BY
25         Product.ProductID,
26         Product.ProductSubcategoryID,
27         Product.Name,
28         YEAR(SalesOrderHeader.OrderDate),
29         MONTH(SalesOrderHeader.OrderDate)
30 )
31 SELECT
32     MonthlySalesWithMin.ProductName AS "Nazwa produktu",
33     ProductCategories.CategoryName AS "Nazwa kategorii",
34     MonthlySalesWithMin.SalesYear AS "Rok",
35     MonthlySalesWithMin.SalesMonth AS "Miesiąc",
36     MonthlySalesWithMin.MonthlyQty AS "Liczba zakupionych produktów"
37 FROM MonthlySalesWithMin
38 LEFT JOIN ProductCategories
39     ON MonthlySalesWithMin.ProductSubcategoryID =
40         ProductCategories.ProductSubcategoryID
41 WHERE MonthlySalesWithMin.MinMonthlyQty >= 20
42 ORDER BY
43     MonthlySalesWithMin.ProductName,
44     MonthlySalesWithMin.SalesYear,
45     MonthlySalesWithMin.SalesMonth

```

Kwerenda bez CTE:

```

1 SELECT
2     Sales.ProductName AS "Nazwa produktu",

```

```

3     Sales.CategoryName AS "Nazwa kategorii",
4     Sales.SalesYear AS "Rok",
5     Sales.SalesMonth AS "Miesiąc",
6     Sales.MonthlyQty AS "Liczba zakupionych produktów"
7 FROM (
8     SELECT
9         Product.ProductID,
10        Product.Name AS ProductName,
11        ProductCategory.Name AS CategoryName,
12        YEAR(SalesOrderHeader.OrderDate) AS SalesYear,
13        MONTH(SalesOrderHeader.OrderDate) AS SalesMonth,
14        SUM(SalesDetail.OrderQty) AS MonthlyQty
15 FROM Sales.SalesOrderDetail AS SalesDetail
16 JOIN Production.Product AS Product
17     ON SalesDetail.ProductID = Product.ProductID
18 JOIN Sales.SalesOrderHeader AS SalesOrderHeader
19     ON SalesDetail.SalesOrderID = SalesOrderHeader.SalesOrderID
20 LEFT JOIN Production.ProductSubcategory AS ProductSubcategory
21     ON Product.ProductSubcategoryID = ProductSubcategory.ProductSubcategoryID
22 LEFT JOIN Production.ProductCategory AS ProductCategory
23     ON ProductSubcategory.ProductCategoryID = ProductCategory.ProductCategoryID
24 GROUP BY
25     Product.ProductID,
26     Product.Name,
27     ProductCategory.Name,
28     YEAR(SalesOrderHeader.OrderDate),
29     MONTH(SalesOrderHeader.OrderDate)
30 ) AS Sales
31 INNER JOIN (
32     SELECT ProductID
33 FROM (
34     SELECT
35         Product.ProductID,
36         SUM(SalesDetail.OrderQty) AS MonthlyQty
37 FROM Sales.SalesOrderDetail AS SalesDetail
38 JOIN Production.Product AS Product
39     ON SalesDetail.ProductID = Product.ProductID
40 JOIN Sales.SalesOrderHeader AS SalesOrderHeader
41     ON SalesDetail.SalesOrderID =
42         SalesOrderHeader.SalesOrderID
43 GROUP BY
44     Product.ProductID,
45     YEAR(SalesOrderHeader.OrderDate),

```

```

46     MONTH(SalesOrderHeader.OrderDate)
47 ) AS Monthly
48 GROUP BY ProductID
49 HAVING MIN(MonthlyQty) >= 20
50 ) AS FilteredProducts
51 ON Sales.ProductID = FilteredProducts.ProductID
52 ORDER BY Sales.ProductName, Sales.SalesYear, Sales.SalesMonth;

```

	Nazwa produktu	Nazwa kategorii	Rok	Miesiąc	Liczba zakupionych produktów
1	AWC Logo Cap	Clothing	2011	5	40
2	AWC Logo Cap	Clothing	2011	7	103
3	AWC Logo Cap	Clothing	2011	8	137
4	AWC Logo Cap	Clothing	2011	10	240
5	AWC Logo Cap	Clothing	2011	12	25
6	AWC Logo Cap	Clothing	2012	1	153
7	AWC Logo Cap	Clothing	2012	2	48
8	AWC Logo Cap	Clothing	2012	3	125
9	AWC Logo Cap	Clothing	2012	4	114
10	AWC Logo Cap	Clothing	2012	5	215
11	AWC Logo Cap	Clothing	2012	6	345
12	AWC Logo Cap	Clothing	2012	7	230
13	AWC Logo Cap	Clothing	2012	8	164

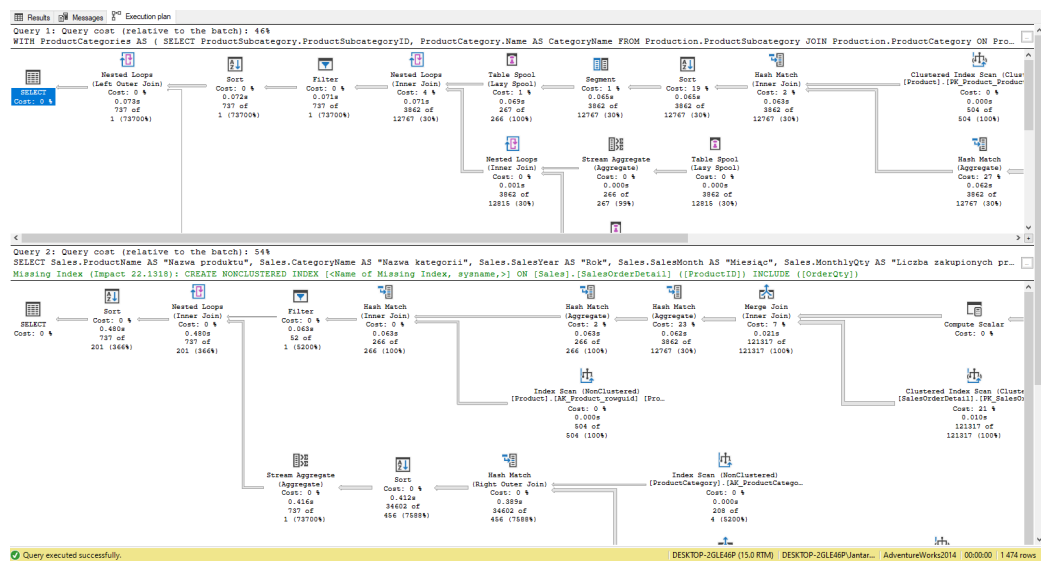
Rysunek 9: Wyniki 3. kwerendy

Nazwa kategorii	Liczba z Nazwa kategorii
Accessories	82
Bikes	294
Clothing	145
Components	216
Suma końcowa	737

Rysunek 10: Tabela przestawna

Są 52 produkty, które konsekwentnie sprzedają się w liczbie co najmniej 20 sztuk miesięcznie. Z podziałem na kategorie, to rowery i komponenty są najbardziej konsekwentnie kupowane, potem ubrania i akcesoria.

Trzeba pamiętać, że kwerenda z tak sztywnymi zasadami mogła wykluczyć wiele dobrze sprzedających się produktów, które mogły sprzedać się w niskiej liczbie w co najmniej jednym miesiącu z wielu.



Rysunek 11: Porównanie exeuction plan

Wersja z CTE jest tylko nieznacznie szybsza - 46% do 54%.

2 Kod kwerend 4-6