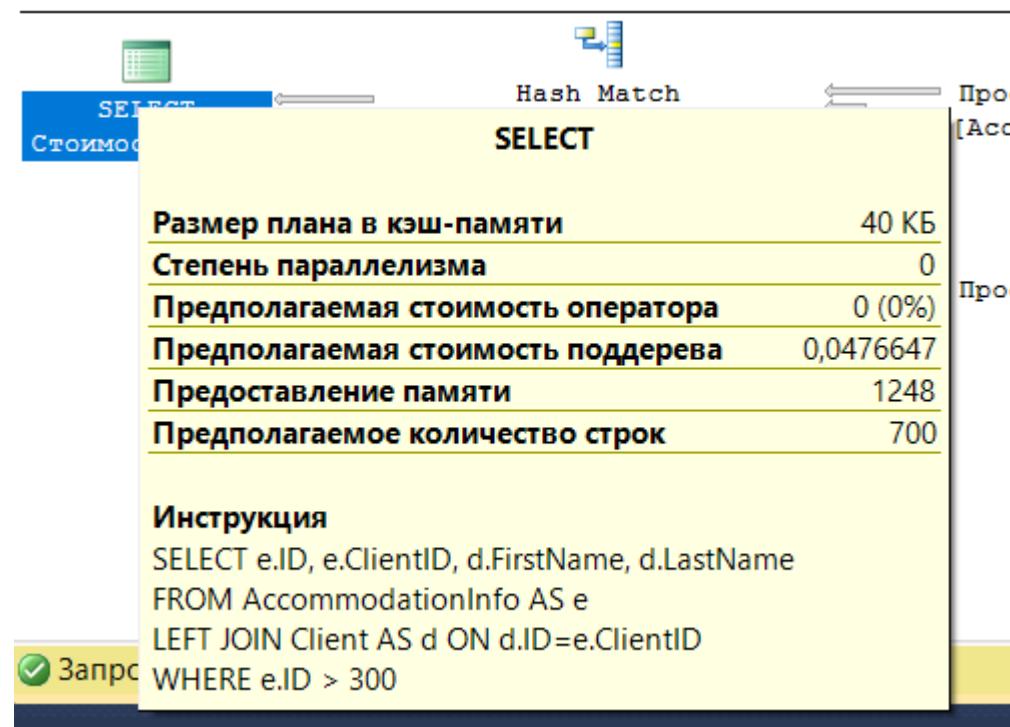


1) Запросы с использованием различных видов соединений таблиц (в том числе самосоединения).

Запрос

```
SELECT e.ID, e.ClientID, d.FirstName, d.LastName  
FROM AccommodationInfo AS e  
LEFT JOIN Client AS d ON d.ID=e.ClientID  
WHERE e.ID > 300
```

Без индекса



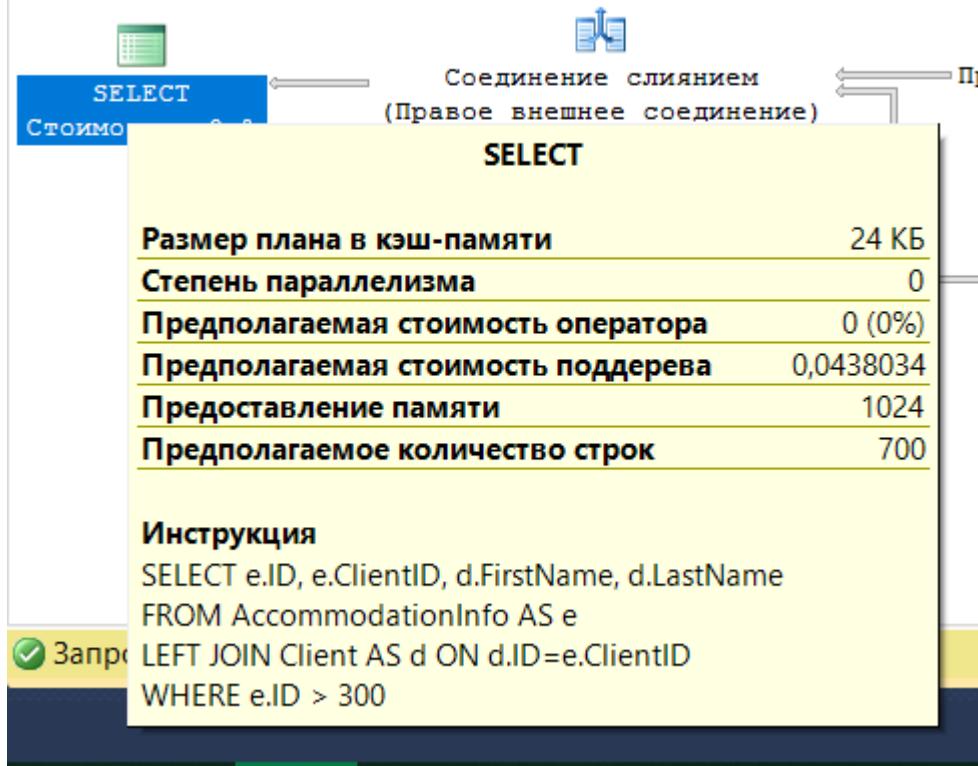
Некластерный:

```
CREATE NONCLUSTERED INDEX NonCL_Name --СОСТАВНОЙ  
ON Client(ID, LastName, FirstName)
```

```
CREATE NONCLUSTERED INDEX NonCL_ClientAcc --СОСТАВНОЙ  
ON AccommodationInfo(ID, ClientID)
```

1) составной

```
SELECT e.ID, e.ClientID, d.FirstName, d.LastName
```



2) покрывающий

```
CREATE NONCLUSTERED INDEX Cov_Name ON Client(LastName)  
INCLUDE (ID, FirstName)
```

```
CREATE NONCLUSTERED INDEX Cov_Client ON AccommodationInfo(ID)  
INCLUDE (ClientID)
```

The screenshot shows a SQL Server Management Studio interface. A tooltip or context menu is open over a query result set. The tooltip title is "SELECT". It contains the following information:

Размер плана в кэш-памяти	40 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0438918
Предоставление памяти	1248
Предполагаемое количество строк	700

Инструкция

```
SELECT e.ID, e.ClientID, d.FirstName, d.LastName  
FROM AccommodationInfo AS e  
LEFT JOIN Client AS d ON d.ID=e.ClientID  
WHERE e.ID > 300
```

Запрос

3) уникальный

```
CREATE UNIQUE NONCLUSTERED INDEX Unique_ID  
ON Client(ID, FirstName, LastName)
```

```
CREATE UNIQUE NONCLUSTERED INDEX Unique_AccClientID  
ON AccommodationInfo(ID, ClientID)
```

```
SELECT e.ID, e.ClientID, d.FirstName, d.LastName
```

Стоимость выполнения запроса

Примечание: План выполнения оптимизирован для производительности.

SELECT

Размер плана в кэш-памяти	24 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0438034
Предоставление памяти	1024
Предполагаемое количество строк	700

Инструкция

```
SELECT e.ID, e.ClientID, d.FirstName, d.LastName
FROM AccommodationInfo AS e
LEFT JOIN Client AS d ON d.ID=e.ClientID
WHERE e.ID > 300
```

Запрос успешно выполнен.

4) индекс с включенными столбцами

```
CREATE NONCLUSTERED INDEX Include_Name ON Client(ID)
INCLUDE (LastName, FirstName)
```

```
CREATE NONCLUSTERED INDEX Include_AccClientID ON AccommodationInfo(ClientID)
INCLUDE (ID, Cost)
```

Стоимость выполнения запроса

Просмотр индексов [Client].[Include_Name]

Просмотр индексов [AccommodationInfo].[Include_AccClientID]

SELECT

Размер плана в кэш-памяти	24 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0236722
Предполагаемое количество строк	700

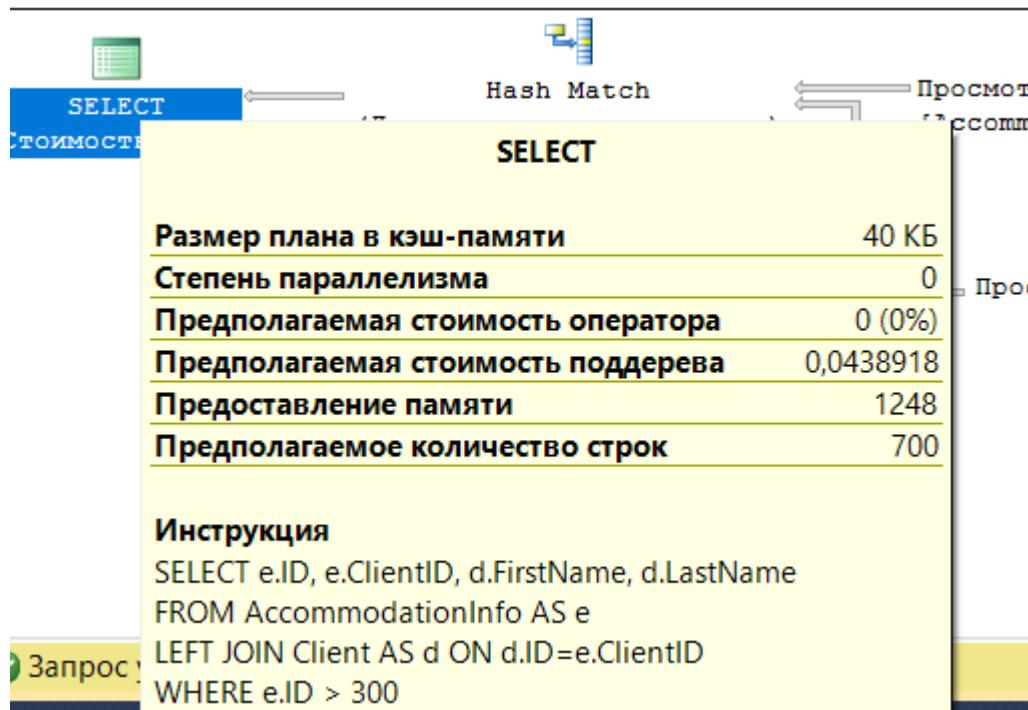
Инструкция

```
SELECT e.ID, e.ClientID, d.FirstName, d.LastName
FROM AccommodationInfo AS e
LEFT JOIN Client AS d ON d.ID=e.ClientID
WHERE e.ID > 300
```

Запрос успешно выполнен.

5) фильтрованный индекс

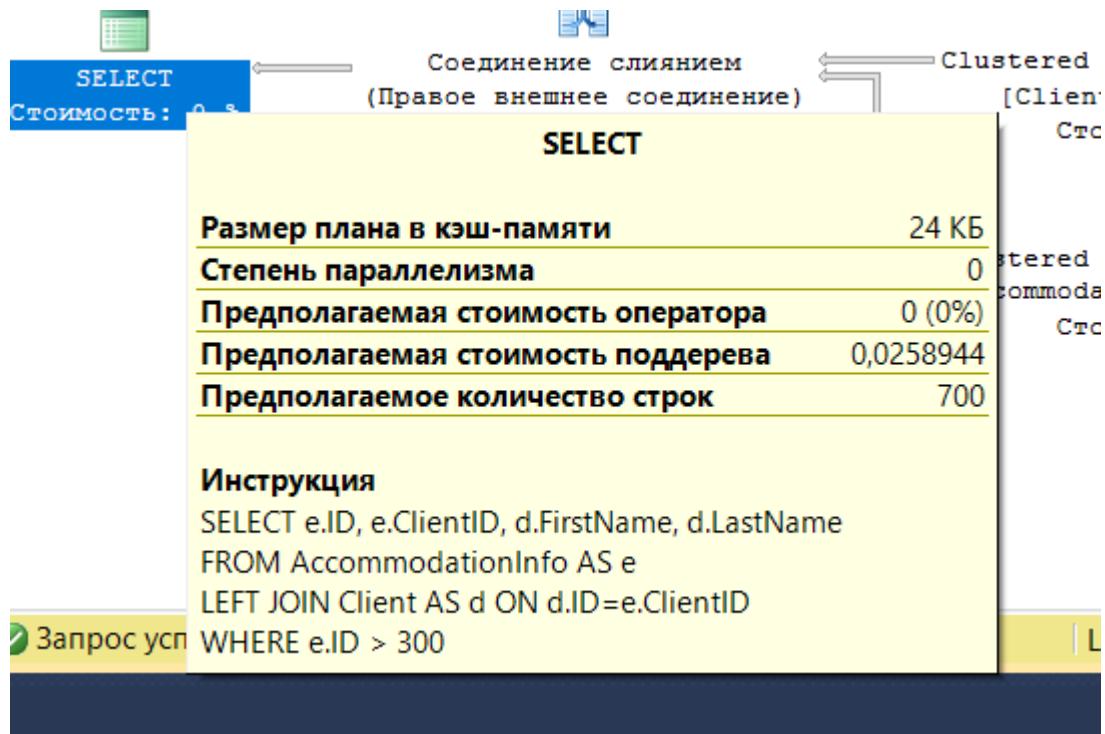
```
CREATE NONCLUSTERED INDEX Filter_Name ON AccommodationInfo(ID, ClientID)
--filter
WHERE ID > 300;
```



Кластерный

```
CREATE CLUSTERED INDEX CL_Name
ON Client(ID, LastName)
```

```
CREATE CLUSTERED INDEX CL_ClientID
ON AccommodationInfo(ClientID, ID)
```



2) Фильтрация данных в запросах с использованием предикатов (EXISTS, IN, ALL, SOME/ANY, BETWEEN, LIKE).

Запрос

```

SELECT ID, ClientID, Cost
FROM AccommodationInfo
WHERE Cost BETWEEN 0 AND 2000 -- номера, цена которых в диапазоне 0-2000
ORDER BY Cost
  
```

Без индекса

The screenshot shows the execution plan for a query without an index. The plan consists of two main steps:

- SELECT**: This step is highlighted in blue and contains the sub-step **Стоимость** (Cost).
- SELECT**: This step is highlighted in black and contains the sub-step **Сортировка** (Sort).

Below the plan, there is a table with the following data:

Параметр	Значение
Размер плана в кэш-памяти	24 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предоставление памяти	1024
Предполагаемая стоимость поддерева	0,0246493
Предполагаемое количество строк	331

Инструкция

```
SELECT [ID],[ClientID],[Cost] FROM [AccommodationInfo]
WHERE [Cost]>=@1 AND [Cost]<=@2 ORDER BY [Cost]
ASC
```

Запрос успешно выполнен.

Некластерный:

1) составной

```
CREATE NONCLUSTERED INDEX NonCL_ClientAcc
ON AccommodationInfo(Cost, ID, ClientID)
```

The screenshot shows the execution plan for a query using a non-clustered index. The plan consists of two main steps:

- SELECT**: This step is highlighted in blue and contains the sub-step **Стоимость** (Cost).
- Поиск в индексе (NonClustered)**: This step is highlighted in black and contains the sub-step **SELECT**.

Below the plan, there is a table with the following data:

Параметр	Значение
Размер плана в кэш-памяти	24 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0043868
Предполагаемое количество строк	331

Инструкция

```
SELECT [ID],[ClientID],[Cost] FROM [AccommodationInfo]
WHERE [Cost]>=@1 AND [Cost]<=@2 ORDER BY [Cost]
ASC
```

2) покрывающий

```
CREATE NONCLUSTERED INDEX Cov_Client ON AccommodationInfo(Cost)
INCLUDE (ID, ClientID)
```

The screenshot shows the execution plan for a query. At the top, there is a summary bar with the text "Поиск в индексе (NonClustered)" and "[AccommodationInfo].[Cov_Cli...". Below this is a large yellow box containing the following information:

SELECT	
Размер плана в кэш-памяти	16 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0043868
Предполагаемое количество строк	331

Below this, under the heading "Инструкция", is the SQL query:

```
SELECT [ID],[ClientID],[Cost] FROM [AccommodationInfo]
WHERE [Cost]>=@1 AND [Cost]<=@2 ORDER BY [Cost]
ASC
```

A green checkmark icon and the text "Запрос устр" are visible at the bottom left of the yellow box.

3) уникальный

```
CREATE UNIQUE NONCLUSTERED INDEX Unique_AccClientID
ON AccommodationInfo(Cost, ClientID, ID)
```

The screenshot shows the execution plan for a query. At the top, there is a summary bar with the text "Поиск в индексе /NonClustered" and "[AccommodationInfo].[Unique_AccClientID...". Below this is a large yellow box containing the following information:

SELECT	
Размер плана в кэш-памяти	24 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0043868
Предполагаемое количество строк	331

Below this, under the heading "Инструкция", is the SQL query:

```
SELECT [ID],[ClientID],[Cost] FROM [AccommodationInfo]
WHERE [Cost]>=@1 AND [Cost]<=@2 ORDER BY [Cost]
ASC
```

4) индекс с включенными столбцами

```
CREATE NONCLUSTERED INDEX Include_AccClientID  
ON AccommodationInfo(ID)  
INCLUDE (Cost, ClientID, RoomID)
```

The screenshot shows the execution plan for a query in SQL Server Management Studio. The plan consists of two main operators: a 'SELECT' operator at the top and a 'SELECT' operator at the bottom. The top 'SELECT' operator has a 'Сортировка' (Sort) node below it. The bottom 'SELECT' operator has an 'Инструкция' (Query) node below it containing the query code.

Сортировка

Просмотр индекса

Стоймост

SELECT

SELECT

Размер плана в кэш-памяти	24 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предоставление памяти	1024
Предполагаемая стоимость поддерева	0,0231678
Предполагаемое количество строк	331

Инструкция

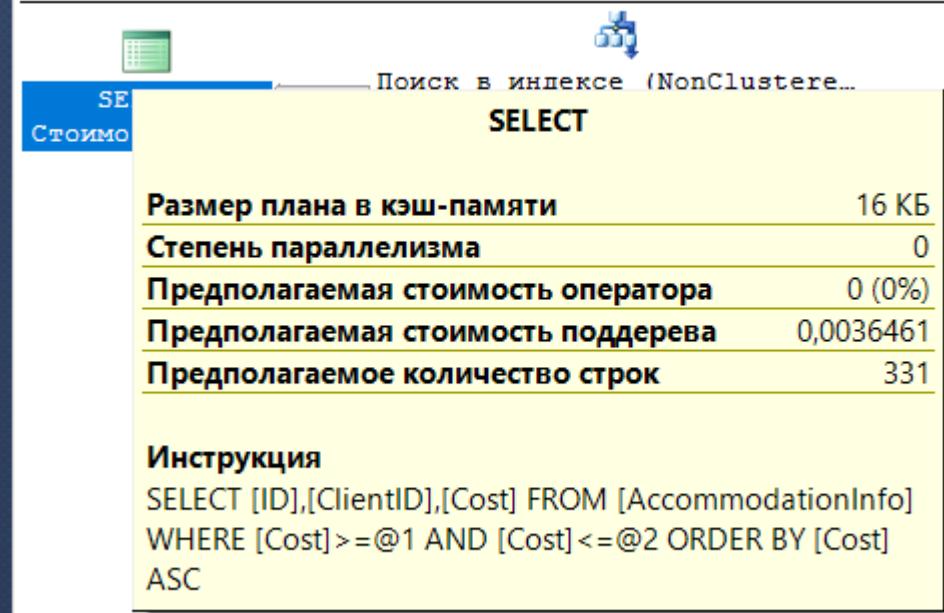
```
SELECT [ID],[ClientID],[Cost] FROM [AccommodationInfo]  
WHERE [Cost]>=@1 AND [Cost]<=@2 ORDER BY [Cost]  
ASC
```

Запрос

5)фильтрованный индекс

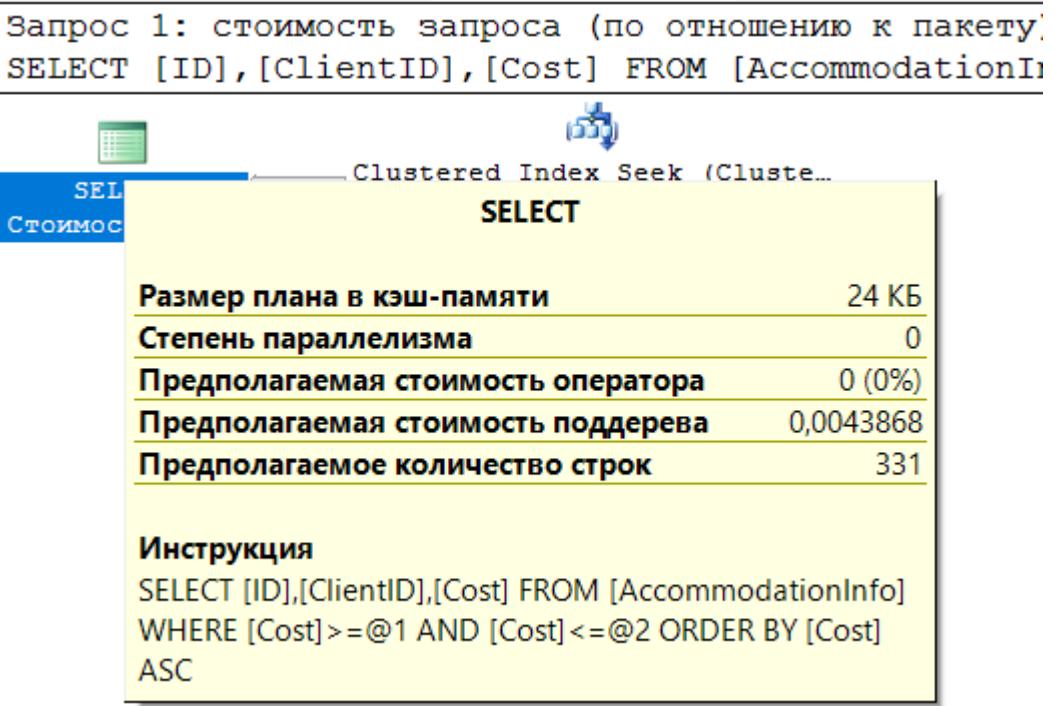
```
CREATE NONCLUSTERED INDEX Filter_Name ON AccommodationInfo(Cost, ID, ClientID)  
WHERE Cost >= 1500;
```

```
SELECT [ID],[ClientID],[Cost] FROM [AccommodationI
```



Кластерный

```
CREATE CLUSTERED INDEX CL_ID
ON AccommodationInfo(Cost, ID, ClientID)
```

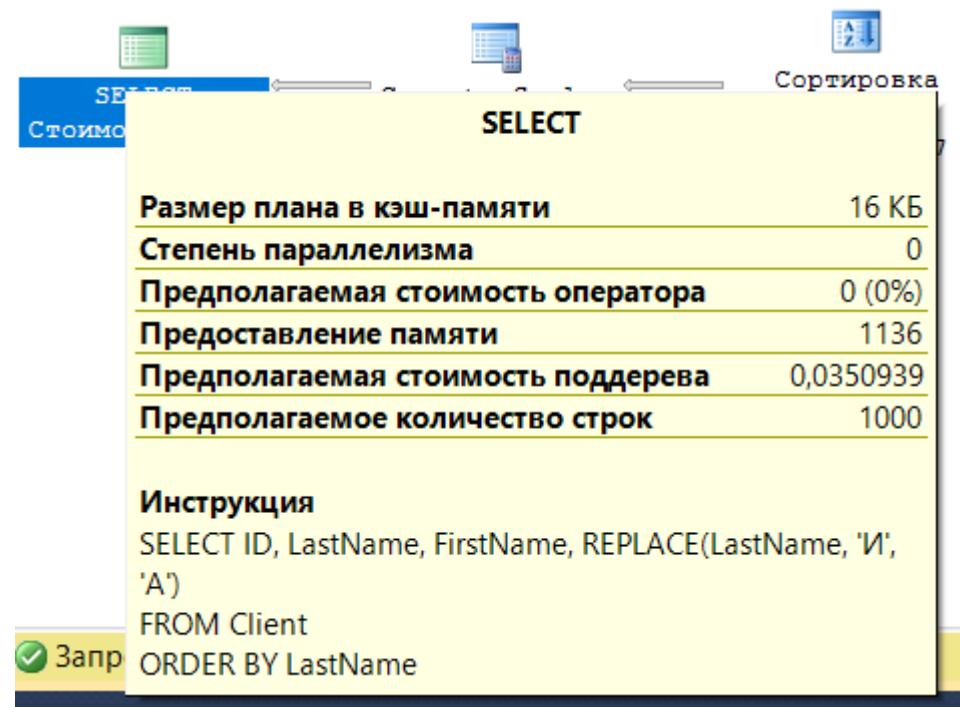


3) Запросы с использованием функций для работы со строками (REPLACE, SUBSTRING, STUFF).

Запрос

```
SELECT ID, LastName, FirstName, REPLACE(LastName, 'И', 'А')
FROM Client
ORDER BY LastName
```

Без индекса



Некластерный:

1) составной

```
CREATE NONCLUSTERED INDEX NonCL_Name --СОСТАВНОЙ
ON Client(LastName, FirstName, ID)
```

Стоймос

Просмотр индекса

SELECT

Размер плана в кэш-памяти 16 КБ
Степень параллелизма 0
Предполагаемая стоимость оператора 0 (0%)
Предполагаемая стоимость поддерева 0,007445
Предполагаемое количество строк 1000

Инструкция

```
SELECT ID, LastName, FirstName, REPLACE(LastName,  
'И', 'А')  
FROM Client  
ORDER BY LastName
```

2) покрывающий

```
CREATE NONCLUSTERED INDEX Cov_Name ON Client( LastName)  
INCLUDE (FirstName, ID)
```

Запрос 1: стоимость запроса (по отношению к п.)

```
SELECT ID, LastName, FirstName, REPLACE(LastName,
```

Стоймос

Просмотр индекса

SELECT

Размер плана в кэш-памяти 16 КБ
Степень параллелизма 0
Предполагаемая стоимость оператора 0 (0%)
Предполагаемая стоимость поддерева 0,007445
Предполагаемое количество строк 1000

Инструкция

```
SELECT ID, LastName, FirstName, REPLACE(LastName,  
'И', 'А')  
FROM Client  
ORDER BY LastName
```

3) уникальный

```
CREATE UNIQUE NONCLUSTERED INDEX Unique_ID --unique  
ON Client(LastName, ID, FirstName)
```

SELECT Compute Scalar Просмотр индекса

Стоимость [U] ость

SELECT

Размер плана в кэш-памяти	16 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,007445
Предполагаемое количество строк	1000

Инструкция

```
SELECT ID, LastName, FirstName, REPLACE(LastName,
    'И', 'А')
FROM Client
ORDER BY LastName
```

Запрос

4) индекс с включенными столбцами

```
CREATE NONCLUSTERED INDEX Include_Name ON Client(LastName) --include
INCLUDE (FirstName, ID)
```

SELECT Compute Scalar Просмотр индекса

Стоимость [In] ость

SELECT

Размер плана в кэш-памяти	16 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,007445
Предполагаемое количество строк	1000

Инструкция

```
SELECT ID, LastName, FirstName, REPLACE(LastName,
    'И', 'А')
FROM Client
ORDER BY LastName
```

Запрос

5) фильтрованный индекс

```
CREATE NONCLUSTERED INDEX Filter_Name ON Client(ID, LastName) --filter
WHERE ID >= 1;
```

The screenshot shows a detailed analysis of a query plan. At the top, there are three icons: a green square for 'SELECT', a blue square for 'Compute Scalar', and a blue square with a white 'A' for 'Сортировка' (Sort). Below these are tabs for 'Стоймост' (Cost) and 'SELECT'. The main pane displays the following information:

Размер плана в кэш-памяти	16 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предоставление памяти	1136
Предполагаемая стоимость поддерева	0,0350939
Предполагаемое количество строк	1000

Инструкция

```
SELECT ID, LastName, FirstName, REPLACE(LastName, 'И', 'А')  
FROM Client  
ORDER BY LastName
```

A green checkmark icon with the text 'Запрос' (Query) is visible at the bottom left.

Кластерный

```
CREATE CLUSTERED INDEX CL_Name  
ON Client(LastName, FirstName, ID)
```

The screenshot shows a detailed analysis of a query plan. At the top, there are three icons: a green square for 'SELECT', a blue square for 'Compute Scalar', and a blue square with a white 'A' for 'Scan'. Below these are tabs for 'Стоймост' (Cost) and 'SELECT'. The main pane displays the following information:

Размер плана в кэш-памяти	16 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0081857
Предполагаемое количество строк	1000

Инструкция

```
SELECT ID, LastName, FirstName, REPLACE(LastName, 'И', 'А')  
FROM Client  
ORDER BY LastName
```

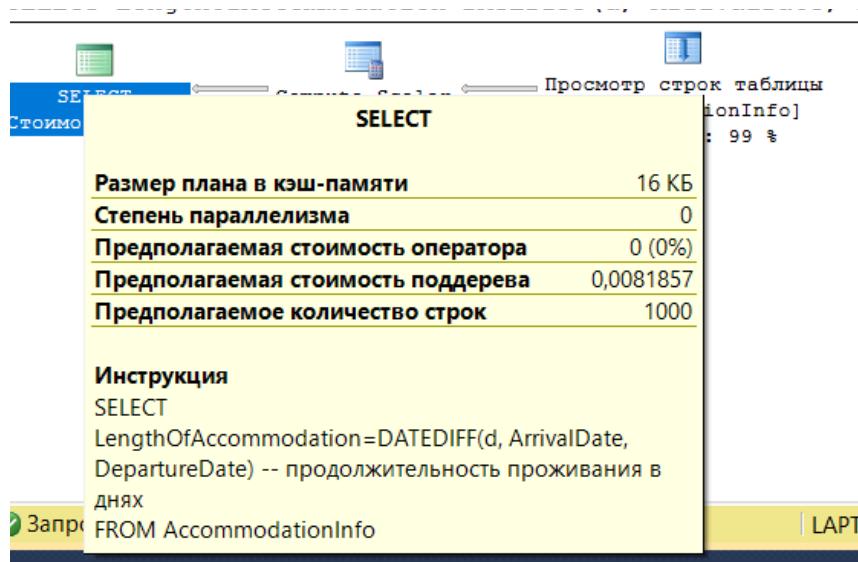
A green checkmark icon with the text 'Запрос успешно выполнен' (Query successfully executed) is visible at the bottom left.

- 4) Запросы с использованием функций даты и времени (DATEPART, DATEADD, DATEDIFF, GETDATE()).

Запрос

```
SELECT
LengthOfAccommodation=DATEDIFF(d, ArrivalDate, DepartureDate) --
продолжительность проживания в днях
FROM AccommodationInfo
```

Без индекса



Некластерный:

1) составной

```
CREATE NONCLUSTERED INDEX NonCL_Acc --СОСТАВНОЙ
ON AccommodationInfo(ID, ArrivalDate, DepartureDate)
```

The screenshot shows the execution plan for a query in SQL Server Management Studio. The plan consists of three main components: a **Compute Scalar** operator, a **Просмотр индекса (NonClustered)** operator, and a **SELECT** operator.

Просмотр индекса (NonClustered) [AccommodationInfo].[NonCL_A...]
Страница: 99 %

SELECT

Размер плана в кэш-памяти	16 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,007445
Предполагаемое количество строк	1000

Инструкция

```
SELECT  
LengthOfAccommodation=DATEDIFF(d, ArrivalDate,  
DepartureDate) -- продолжительность проживания в  
днях  
FROM AccommodationInfo
```

2) покрывающий

```
CREATE NONCLUSTERED INDEX Cov_Acc ON AccommodationInfo(ID)  
INCLUDE (ArrivalDate, DepartureDate)
```

The screenshot shows the execution plan for a query in SQL Server Management Studio. The plan consists of three main components: a **Compute Scalar** operator, a **Просмотр индекса (NonClustered)** operator, and a **SELECT** operator.

Просмотр индекса (NonClustered) [AccommodationInfo].[Cov_Acc]
Страница: 99 %

SELECT

Размер плана в кэш-памяти	16 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,007445
Предполагаемое количество строк	1000

Инструкция

```
SELECT  
LengthOfAccommodation=DATEDIFF(d, ArrivalDate,  
DepartureDate) -- продолжительность проживания в  
днях  
FROM AccommodationInfo
```

3) уникальный

```
CREATE UNIQUE NONCLUSTERED INDEX Unique_AccID  
ON AccommodationInfo(ID, ArrivalDate, DepartureDate)
```

```

SELECT LengthOfAccommodation=DATEDIFF(d, ArrivalDate, DepartureDate)

```

Просмотр индекса (NonClustered) на таблице [AccommodationInfo].[Unique_...]

Стоймость

SELECT

Инструкция

SELECT
LengthOfAccommodation=DATEDIFF(d, ArrivalDate,
DepartureDate) -- продолжительность проживания в
днях

Запрос у | LAPTOP-DJ7

4) индекс с включенными столбцами

```

CREATE NONCLUSTERED INDEX Include_AccID
ON AccommodationInfo(ID)
INCLUDE (ArrivalDate, DepartureDate)

```

```

SELECT LengthOfAccommodation=DATEDIFF(d, ArrivalDate, DepartureDate)

```

Просмотр индекса (NonClustered) на таблице [AccommodationInfo].[Include_AccID]

Стоймость

SELECT

Инструкция

SELECT
LengthOfAccommodation=DATEDIFF(d, ArrivalDate,
DepartureDate) -- продолжительность проживания в
днях

Запро | LAPTOP-DJ7

5) фильтрованный индекс

```

CREATE NONCLUSTERED INDEX Filter_Name ON AccommodationInfo(ID, ArrivalDate,
DepartureDate)
WHERE ArrivalDate > CAST('2021-01-01T00:00:00.000' AS datetime);

```

Страница 1 из 1

Просмотр строк

SELEC	SELECT
Стоимость	
Размер плана в кэш-памяти 16 КБ	
Степень параллелизма 0	
Предполагаемая стоимость оператора 0 (0%)	
Предполагаемая стоимость поддерева 0,0081857	
Предполагаемое количество строк 1000	
Инструкция	
SELECT	
LengthOfAccommodation=DATEDIFF(d, ArrivalDate,	
DepartureDate) -- продолжительность проживания в	
днях	
<input checked="" type="checkbox"/> Запрос у FROM AccommodationInfo	

Кластерный

```
CREATE CLUSTERED INDEX CL_AccID
ON AccommodationInfo(ID, ArrivalDate, DepartureDate)
```

Страница 1 из 1

Просмотр индексов

Страница 1 из 1	Просмотр индексов	Свойства
SELECT		
Размер плана в кэш-памяти 16 КБ		
Степень параллелизма 0		
Предполагаемая стоимость оператора 0 (0%)		
Предполагаемая стоимость поддерева 0,0067042		
Предполагаемое количество строк 1000		
Инструкция		
SELECT		
LengthOfAccommodation=DATEDIFF(d, ArrivalDate,		
DepartureDate) -- продолжительность проживания в		
днях		
<input checked="" type="checkbox"/> Запрос у FROM AccommodationInfo		

5) Запросы с использованием агрегатных функций, группировок (GROUP BY) и фильтрации групп (HAVING).

Запрос

```
SELECT
ClientID, d.FirstName, d.LastName,
COUNT(*) VisitsCount, --кол-во посещений отеля
```

SUM(Cost) SumCost, --сумма, потраченная на проживание, для каждого клиента за всё время

AVG(Cost) AverageCost -- средняя стоимость проживания для каждого клиента
FROM AccommodationInfo

LEFT JOIN Client AS d ON d.ID=ClientID
GROUP BY ClientID, d.FirstName, d.LastName
HAVING ClientID <= 100

Без индекса

The screenshot shows the results of a query execution. On the left, there's a tree view with nodes like 'Результат' (Results), 'Запрос' (Query), and 'Стоимость' (Cost). The main pane displays the query and its execution plan.

SELECT

Размер плана в кэш-памяти	56 KB
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0497603
Предоставление памяти	1632
Предполагаемое количество строк	125,105

Инструкция

```
SELECT
    ClientID, d.FirstName, d.LastName,
    COUNT(*) VisitsCount, --кол-во посещений отеля
    SUM(Cost) SumCost, --сумма, потраченная на
    проживание, для каждого клиента за всё время
    AVG(Cost) AverageCost -- средняя стоимость
    проживания для каждого клиента
FROM AccommodationInfo
LEFT JOIN Client AS d ON d.ID=ClientID
GROUP BY ClientID, d.FirstName, d.LastName
HAVING ClientID <= 100
```

Некластерный:

1) составной

```
CREATE NONCLUSTERED INDEX NonCL_Name
ON Client(ID, LastName, FirstName)
```

```
CREATE NONCLUSTERED INDEX NonCL_ClientAcc
ON AccommodationInfo(ClientID, ID, Cost)
```

Стандартный отчет

Стоимость

Инструкция

```

SELECT
    ClientID, d.FirstName, d.LastName,
    COUNT(*) VisitsCount, --кол-во посещений отеля
    SUM(Cost) SumCost, --сумма, потраченная на проживание, для каждого клиента за всё время
    AVG(Cost) AverageCost -- средняя стоимость проживания для каждого клиента
FROM AccommodationInfo
LEFT JOIN Client AS d ON d.ID=ClientID
GROUP BY ClientID, d.FirstName, d.LastName
HAVING ClientID <= 100
  
```

2) покрывающий

```

CREATE NONCLUSTERED INDEX Cov_Name ON Client(ID)
INCLUDE (FirstName, LastName)
  
```

```

CREATE NONCLUSTERED INDEX Cov_Client ON AccommodationInfo(ClientID)
INCLUDE (ID, Cost)
  
```

Стандартный отчет

Стоимость

Инструкция

```

SELECT
    ClientID, d.FirstName, d.LastName,
    COUNT(*) VisitsCount, --кол-во посещений отеля
    SUM(Cost) SumCost, --сумма, потраченная на проживание, для каждого клиента за всё время
    AVG(Cost) AverageCost -- средняя стоимость проживания для каждого клиента
FROM AccommodationInfo
LEFT JOIN Client AS d ON d.ID=ClientID
GROUP BY ClientID, d.FirstName, d.LastName
HAVING ClientID <= 100
  
```

3) уникальный

```

CREATE UNIQUE NONCLUSTERED INDEX Unique_ID
  
```

```
ON Client(ID, FirstName, LastName)
```

```
CREATE UNIQUE NONCLUSTERED INDEX Unique_AccClientID  
ON AccommodationInfo(ID, ClientID, Cost)
```

The screenshot shows the results of a SQL query execution. The left pane displays the query:`SELECT ClientID, d.FirstName, d.LastName,
COUNT(*) VisitsCount, --кол-во посещений отеля
SUM(Cost) SumCost, --сумма, потраченная на
проживание, для каждого клиента за всё время
AVG(Cost) AverageCost -- средняя стоимость
проживания для каждого клиента
FROM AccommodationInfo
LEFT JOIN Client AS d ON d.ID=ClientID
GROUP BY ClientID, d.FirstName, d.LastName
HAVING ClientID <= 100`

The right pane shows the execution plan details:

Статистика	Значение
Размер плана в кэш-памяти	40 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0421475
Предоставление памяти	1168
Предполагаемое количество строк	125,105

Инструкция:

```
SELECT ClientID, d.FirstName, d.LastName,  
COUNT(*) VisitsCount, --кол-во посещений отеля  
SUM(Cost) SumCost, --сумма, потраченная на  
проживание, для каждого клиента за всё время  
AVG(Cost) AverageCost -- средняя стоимость  
проживания для каждого клиента  
FROM AccommodationInfo  
LEFT JOIN Client AS d ON d.ID=ClientID  
GROUP BY ClientID, d.FirstName, d.LastName  
HAVING ClientID <= 100
```

4) индекс с включенными столбцами

```
CREATE NONCLUSTERED INDEX Include_Name ON Client(ID)  
INCLUDE (LastName, FirstName)
```

```
CREATE NONCLUSTERED INDEX Include_AccClientID  
ON AccommodationInfo(ClientID)  
INCLUDE (Cost)
```

Резу

Запро

SELECT

Стоймос

Размер плана в кэш-памяти	32 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0257605
Предоставление памяти	1024
Предполагаемое количество строк	125,105

Инструкция

```
SELECT
ClientID, d.FirstName, d.LastName,
COUNT(*) VisitsCount, --кол-во посещений отеля
SUM(Cost) SumCost, --сумма, потраченная на
проживание, для каждого клиента за всё время
AVG(Cost) AverageCost -- средняя стоимость
проживания для каждого клиента
FROM AccommodationInfo
LEFT JOIN Client AS d ON d.ID=ClientID
GROUP BY ClientID, d.FirstName, d.LastName
HAVING ClientID <= 100
```

5) фильтрованный индекс

```
CREATE NONCLUSTERED INDEX Filter_Name ON AccommodationInfo(ClientID, ID, Cost)
WHERE ClientID <= 100;
```

100 %

Резу

Запро

SELECT

Стоймос

Размер плана в кэш-памяти	40 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0433105
Предоставление памяти	1024
Предполагаемое количество строк	123,884

Инструкция

```
SELECT
ClientID, d.FirstName, d.LastName,
COUNT(*) VisitsCount, --кол-во посещений отеля
SUM(Cost) SumCost, --сумма, потраченная на
проживание, для каждого клиента за всё время
AVG(Cost) AverageCost -- средняя стоимость
проживания для каждого клиента
FROM AccommodationInfo
LEFT JOIN Client AS d ON d.ID=ClientID
GROUP BY ClientID, d.FirstName, d.LastName
HAVING ClientID <= 100
```

Кластерный

```
CREATE CLUSTERED INDEX CL_Name
ON Client(LastName, FirstName, ID)
```

```
CREATE CLUSTERED INDEX CL_ID  
ON AccommodationInfo(ClientID)
```

The screenshot shows the results of a SELECT query. On the left, there's a sidebar with tabs for 'Результат' (Results), 'Запрос' (Query), and 'Стоимост' (Cost). The 'Стоимост' tab is selected, displaying performance metrics for the query plan:

Размер плана в кэш-памяти	40 KB
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0257607
Предоставление памяти	1024
Предполагаемое количество строк	125,105

Below the metrics, the 'Инструкция' (Instruction) section shows the T-SQL code:

```
SELECT ClientID, d.FirstName, d.LastName,  
COUNT(*) VisitsCount, --кол-во посещений отеля  
SUM(Cost) SumCost, --сумма, потраченная на  
проживание, для каждого клиента за всё время  
AVG(Cost) AverageCost -- средняя стоимость  
проживания для каждого клиента  
FROM AccommodationInfo  
LEFT JOIN Client AS d ON d.ID=ClientID  
GROUP BY ClientID, d.FirstName, d.LastName  
HAVING ClientID <= 100
```

6) Вложенные запросы

Запрос

```
SELECT ID, FirstName, LastName  
FROM Client  
WHERE ID IN  
(SELECT ClientID  
FROM AccommodationInfo  
WHERE Cost = 4000);
```

Без индекса

The screenshot shows the execution plan for a query in SQL Server Management Studio. The plan is labeled 'SELECT' and includes the following statistics:

Размер плана в кэш-памяти	40 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0406998
Предоставление памяти	1136
Предполагаемое количество строк	158,411

Инструкция

```
SELECT ID, FirstName, LastName
FROM Client
WHERE ID IN
    (SELECT ClientID
     FROM AccommodationInfo
     WHERE Cost = 4000)
```

Некластерный:

1) составной

```
CREATE NONCLUSTERED INDEX NonCL_Name
ON Client(ID, LastName, FirstName)
```

```
CREATE NONCLUSTERED INDEX NonCL_ClientAcc
ON AccommodationInfo(Cost, ID, ClientID)
```

The screenshot shows the execution plan for the same query after creating indexes. The plan is labeled 'Составною ссыпкой' (Covering index) and includes the following statistics:

Размер плана в кэш-памяти	24 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0321025
Предоставление памяти	1024
Предполагаемое количество строк	158,411

Инструкция

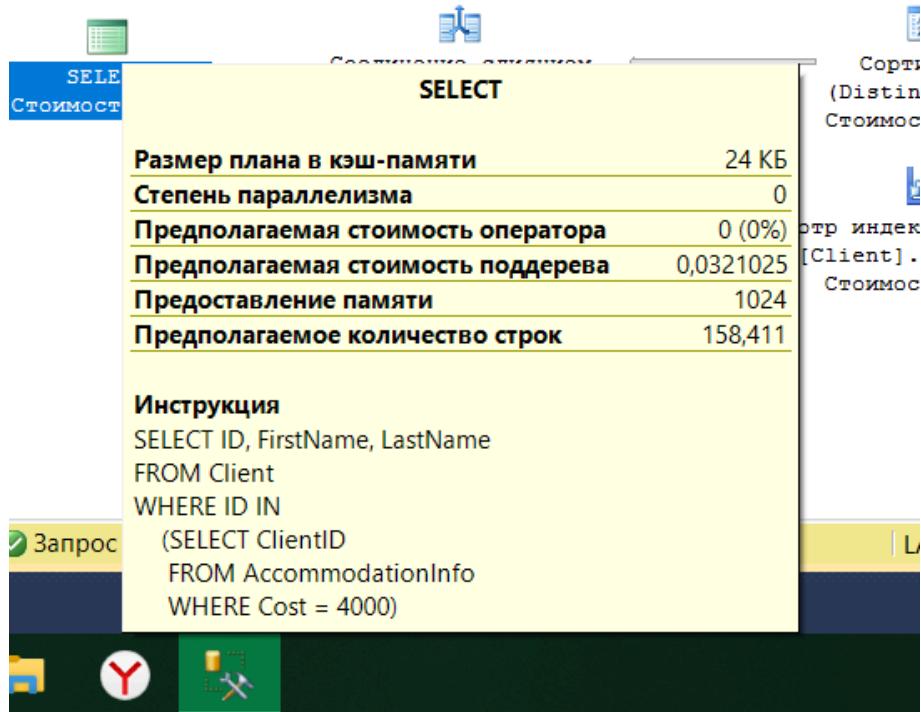
```
SELECT ID, FirstName, LastName
FROM Client
WHERE ID IN
    (SELECT ClientID
     FROM AccommodationInfo
     WHERE Cost = 4000)
```

2) покрывающий

```
CREATE NONCLUSTERED INDEX Cov_Name ON Client(ID)
INCLUDE (FirstName, LastName)
```

```
CREATE NONCLUSTERED INDEX Cov_Client ON AccommodationInfo(Cost)
--ПОКРЫВАЮЩИЙ
INCLUDE (ID, ClientID)
```

```
SELECT ID, FirstName, LastName FROM Client WHERE ID IN
```



3) уникальный

```
CREATE UNIQUE NONCLUSTERED INDEX Unique_ID --unique
ON Client(ID, FirstName, LastName)
```

```
CREATE UNIQUE NONCLUSTERED INDEX Unique_AccClientID --unique
ON AccommodationInfo(Cost, ClientID, ID)
```

The screenshot shows the execution plan for a query. At the top, there are three tabs: 'SELECT' (highlighted in blue), 'Соединение с слиянием' (Join with Merge), and 'Просмотр индекса' (Index View). The 'SELECT' tab displays the following information:

Размер плана в кэш-памяти	24 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0188706
Предполагаемое количество строк	158,411

Инструкция

```
SELECT ID, FirstName, LastName
FROM Client
WHERE ID IN
    (SELECT ClientID
     FROM AccommodationInfo
     WHERE Cost = 4000)
```

4) индекс с включенными столбцами

```
CREATE NONCLUSTERED INDEX Include_Name ON Client(ID) --include
INCLUDE (LastName, FirstName)
```

```
CREATE NONCLUSTERED INDEX Include_AccClientID
ON AccommodationInfo(ClientID)
```

The screenshot shows the execution plan for a query. At the top, there are three tabs: 'SELECT' (highlighted in blue), 'Соединение с слиянием' (Join with Merge), and 'Просмотр индекса' (Index View). The 'SELECT' tab displays the following information:

Размер плана в кэш-памяти	24 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0224924
Предполагаемое количество строк	158,411

Инструкция

```
SELECT ID, FirstName, LastName FROM Client WHERE :
```

```
SELECT ID, FirstName, LastName
FROM Client
WHERE ID IN
    (SELECT ClientID
     FROM AccommodationInfo
     WHERE Cost = 4000)
```

5) фильтрованный индекс

```
CREATE NONCLUSTERED INDEX Filter_ID ON AccommodationInfo(Cost, ClientID, ID)
WHERE Cost = 4000;
```

Запрос 1: стоимость запроса (по отношению к пакету)

```
SELECT ID, FirstName, LastName FROM Client WHERE
```

SELECT	
Размер плана в кэш-памяти	40 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0355964
Предоставление памяти	1136
Предполагаемое количество строк	158,411

Инструкция

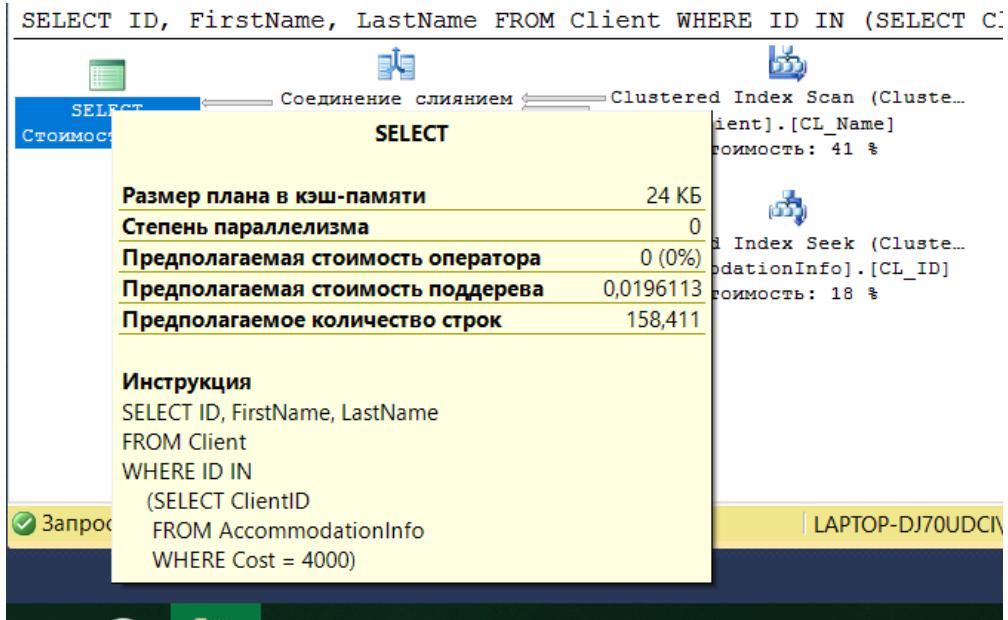
```
SELECT ID, FirstName, LastName
FROM Client
WHERE ID IN
    (SELECT ClientID
     FROM AccommodationInfo
     WHERE Cost = 4000)
```

Запрос

Кластерный

```
CREATE CLUSTERED INDEX CL_Name
ON Client(ID, LastName)
```

```
CREATE CLUSTERED INDEX CL_ID
ON AccommodationInfo(Cost, ClientID, ID)
```

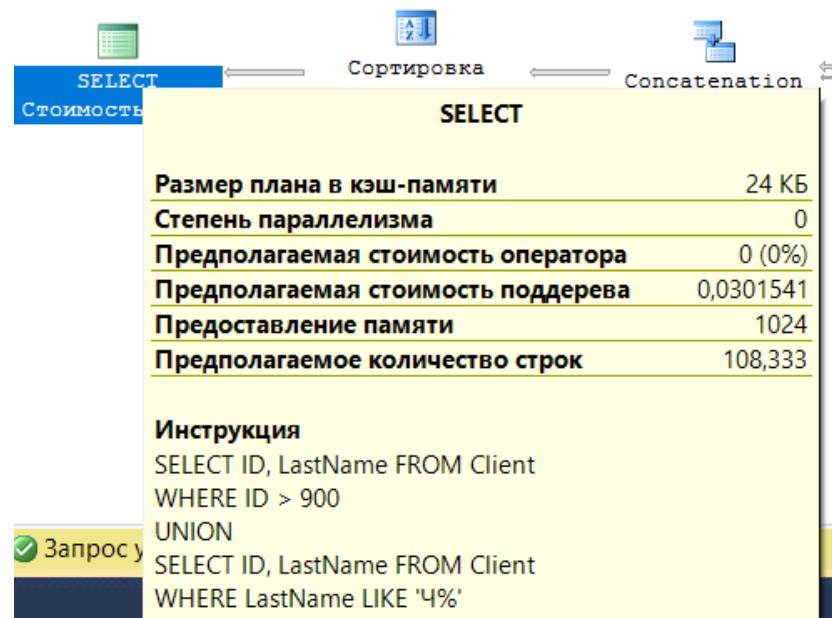


7) Запросы с использованием UNION и INTERSECT

Запрос

```
SELECT ID, LastName FROM Client
WHERE ID > 900
UNION
SELECT ID, LastName FROM Client
WHERE LastName LIKE 'Ч%'
```

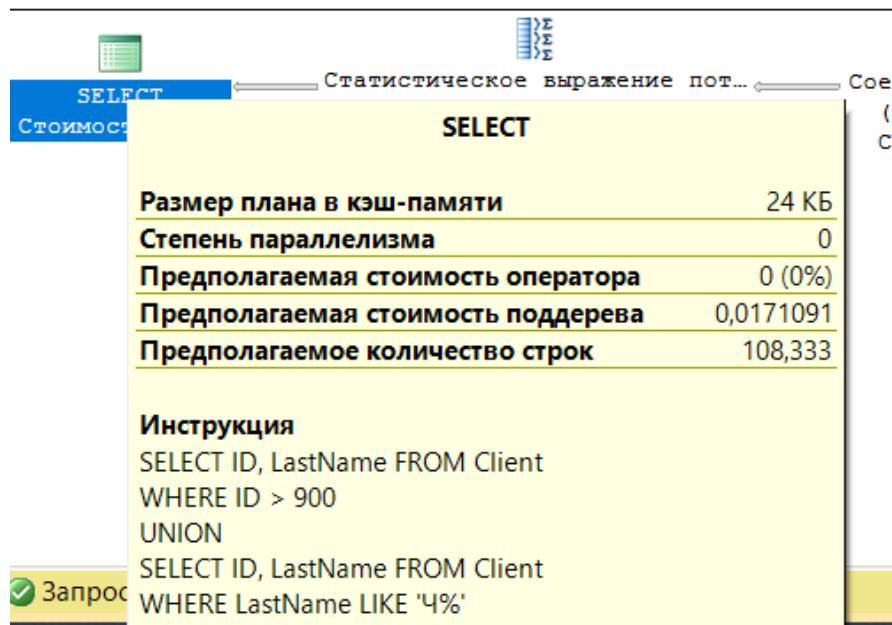
Без индекса



Некластерный:

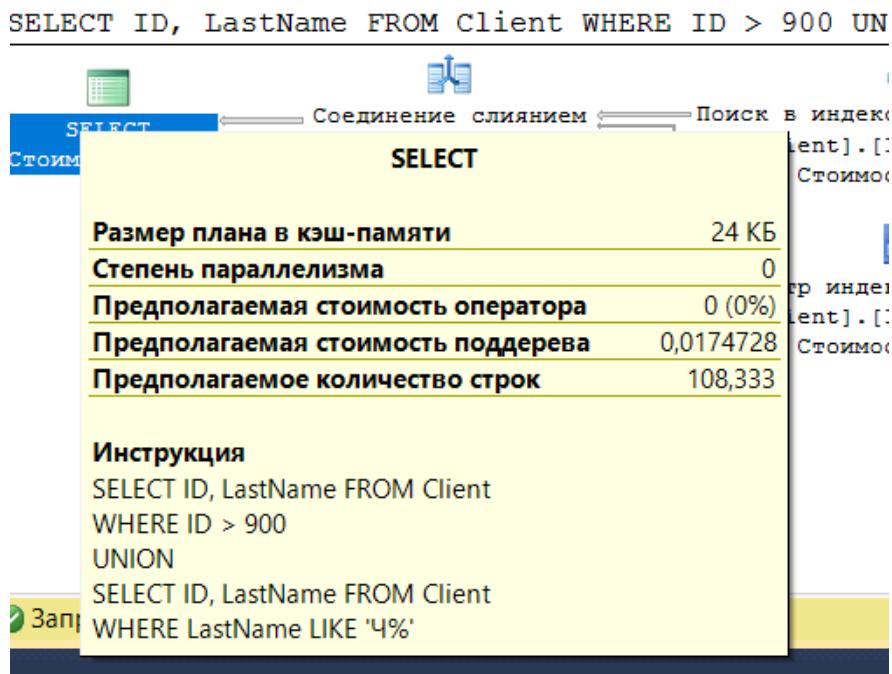
1) составной

```
CREATE NONCLUSTERED INDEX NonCL_Name --СОСТАВНОЙ  
ON Client(LastName, ID, FirstName)
```



2) покрывающий

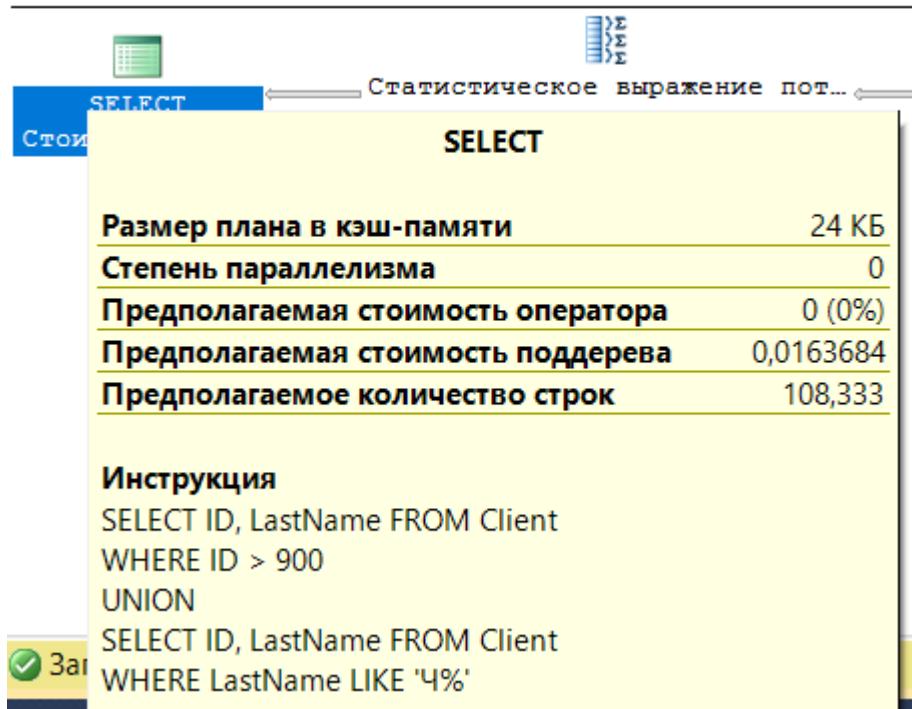
```
CREATE NONCLUSTERED INDEX Include_Name ON Client(ID)  
INCLUDE (LastName, FirstName)
```



3) уникальный

```
CREATE UNIQUE NONCLUSTERED INDEX Unique_ID  
ON Client(LastName, ID)
```

Запрос 1: стоимость запроса (по отношению к
SELECT ID, LastName FROM Client WHERE ID > 900)



4) индекс с включенными столбцами

```
CREATE NONCLUSTERED INDEX Cov_Name ON Client(LastName)  
INCLUDE (ID)
```

SELECT ID, LastName FROM Client WHERE ID > 900 UNION SELECT ID, LastName FROM Client WHERE LastName LIKE 'Ч%'

Стоимость	
Размер плана в кэш-памяти	24 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0230219
Предоставление памяти	1024
Предполагаемое количество строк	108,333

Инструкция

```
SELECT ID, LastName FROM Client
WHERE ID > 900
UNION
SELECT ID, LastName FROM Client
WHERE LastName LIKE 'Ч%'
```

Запрос

5) фильтрованный индекс

```
CREATE NONCLUSTERED INDEX Filter_Name ON Client(ID, LastName) --filter
WHERE ID > 900;
```

```
SELECT ID, LastName FROM Client WHERE ID > 900
```

The screenshot shows the execution plan for the query `SELECT ID, LastName FROM Client WHERE ID > 900`. The plan consists of three main stages: **SELECT**, **Сортировка** (Sort), and **Concatenation**. The **SELECT** stage is highlighted in blue. The **Сортировка** stage is shown with a sorting icon. The **Concatenation** stage is shown with a merge icon.

Стоймость оператора (Operator Cost): 24 КБ

Инструкция (Query):

```
SELECT ID, LastName FROM Client  
WHERE ID > 900  
UNION  
SELECT ID, LastName FROM Client  
WHERE LastName LIKE 'Ч%'
```

Кластерный

```
CREATE CLUSTERED INDEX CL_Name  
ON Client(LastName, ID)
```

The screenshot shows the execution plan for the query `SELECT * FROM Client WHERE ID > 900` after creating a clustered index on `Client(LastName, ID)`. The plan consists of three main stages: **SELECT**, **Сортировка** (Sort), and **Concatenation**. The **SELECT** stage is highlighted in blue. The **Сортировка** stage is shown with a sorting icon. The **Concatenation** stage is shown with a merge icon.

Стоймость оператора (Operator Cost): 24 КБ

Инструкция (Query):

```
SELECT * FROM Client  
WHERE ID > 900  
UNION  
SELECT * FROM Client  
WHERE LastName LIKE 'Ч%'
```

Запрос

```
SELECT ID, LastName FROM Client
```

```
WHERE ID > 300
INTERSECT
SELECT ID, LastName FROM Client
WHERE LastName LIKE 'Ч%'
```

Без индекса

The screenshot shows the execution plan and query details for the provided SQL statement. The execution plan is a yellow box containing performance metrics and the query text. The metrics include:

Параметр	Значение
Размер плана в кэш-памяти	24 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0307286
Предполагаемое количество строк	13,9443

Below the metrics, the query text is displayed:

```
SELECT ID, LastName FROM Client
WHERE ID > 300
INTERSECT
SELECT ID, LastName FROM Client
WHERE LastName LIKE 'Ч%'
```

A green checkmark icon next to the word "Запро" indicates the query was successfully executed.

Некластерный:

1) составной

```
CREATE NONCLUSTERED INDEX NonCL_Name
ON Client(ID, LastName, FirstName)
```

```

SELECT ID, LastName FROM Client WHERE ID > 300 INTERSECT SELECT

```

Стоймость: 32 %

Client].[NonCL_Name]

Стоимость: 32 %

в индексе (NonClustered)

Client].[NonCL_Name]

Стоимость: 32 %

Инструкция

```

SELECT ID, LastName FROM Client
WHERE ID > 300
INTERSECT
SELECT ID, LastName FROM Client
WHERE LastName LIKE 'Ч%'

```

2) покрывающий

CREATE NONCLUSTERED INDEX Cov_Name ON Client(ID) --ПОКРЫВАЮЩИЙ
INCLUDE (FirstName, LastName)

```

SELECT ID, LastName FROM Client WHERE ID > 300 INTERSECT SEL

```

Стоймость: 32 %

индексе (NonClustered)

Client].[Cov_Name]

Стоимость: 32 %

индексе (NonClustered)

Client].[Cov_Name]

Стоимость: 32 %

Инструкция

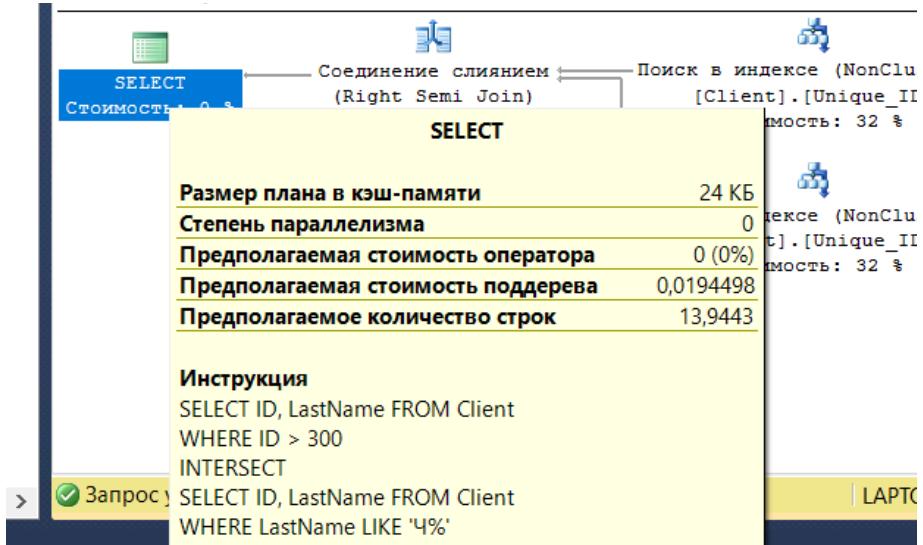
```

SELECT ID, LastName FROM Client
WHERE ID > 300
INTERSECT
SELECT ID, LastName FROM Client
WHERE LastName LIKE 'Ч%'

```

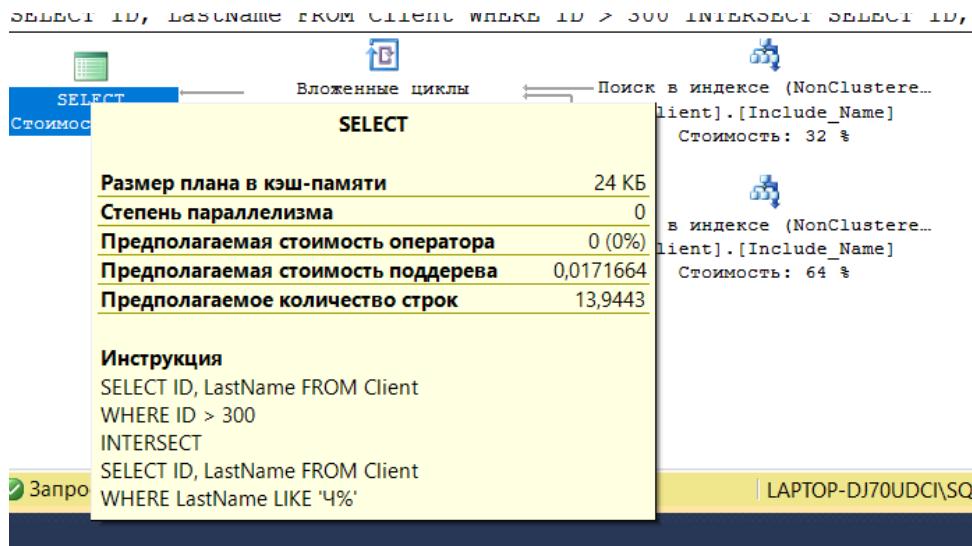
3) уникальный

CREATE UNIQUE NONCLUSTERED INDEX Unique_ID
ON Client(ID, LastName, FirstName)



4) индекс с включенными столбцами

CREATE NONCLUSTERED INDEX Include_Name ON Client(ID)
INCLUDE (LastName)



5) фильтрованный индекс

CREATE NONCLUSTERED INDEX Filter_Name ON Client(ID, LastName)
WHERE ID > 300;

SELECT ID, LastName FROM Client WHERE ID > 300

Стоймост

Вложенные циклы (Высокое значение)

Просмот

SELECT

Размер плана в кэш-памяти	24 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,0149856
Предполагаемое количество строк	13,9443

Инструкция

```
SELECT ID, LastName FROM Client
WHERE ID > 300
INTERSECT
SELECT ID, LastName FROM Client
WHERE LastName LIKE 'Ч%'
```

Запро

Кластерный

CREATE CLUSTERED INDEX CL_Name
ON Client(LastName, ID)

SELECT ID, LastName FROM Client WHERE ID > 300 План запроса

Стоймост

Вложенные циклы

Clustered Index Seek (Cluste...)

[Client].[CL_Name]
Стоимость: 38 %

Clustered Index Seek (Cluste...)

[Client].[CL_Name]
Стоимость: 61 %

SELECT

Размер плана в кэш-памяти	24 КБ
Степень параллелизма	0
Предполагаемая стоимость оператора	0 (0%)
Предполагаемая стоимость поддерева	0,008726
Предполагаемое количество строк	13,9443

Инструкция

```
SELECT ID, LastName FROM Client
WHERE ID > 300
INTERSECT
SELECT ID, LastName FROM Client
WHERE LastName LIKE 'Ч%'
```

Запро

LAPTOP-DJ70UDCI\SC