



كلية الهندسة المعلوماتية جامعة دمشق

DAMASCUS UNIVERSITY | Faculty of Informatics Engineering



الجمهورية العربية السورية  
جامعة دمشق  
وزارة التعليم العالي

## Data Base 2

تقديم الطلاب:

محمد حسين السقعان

بسام محمود الخطيب

رغد وليم ابو فخر

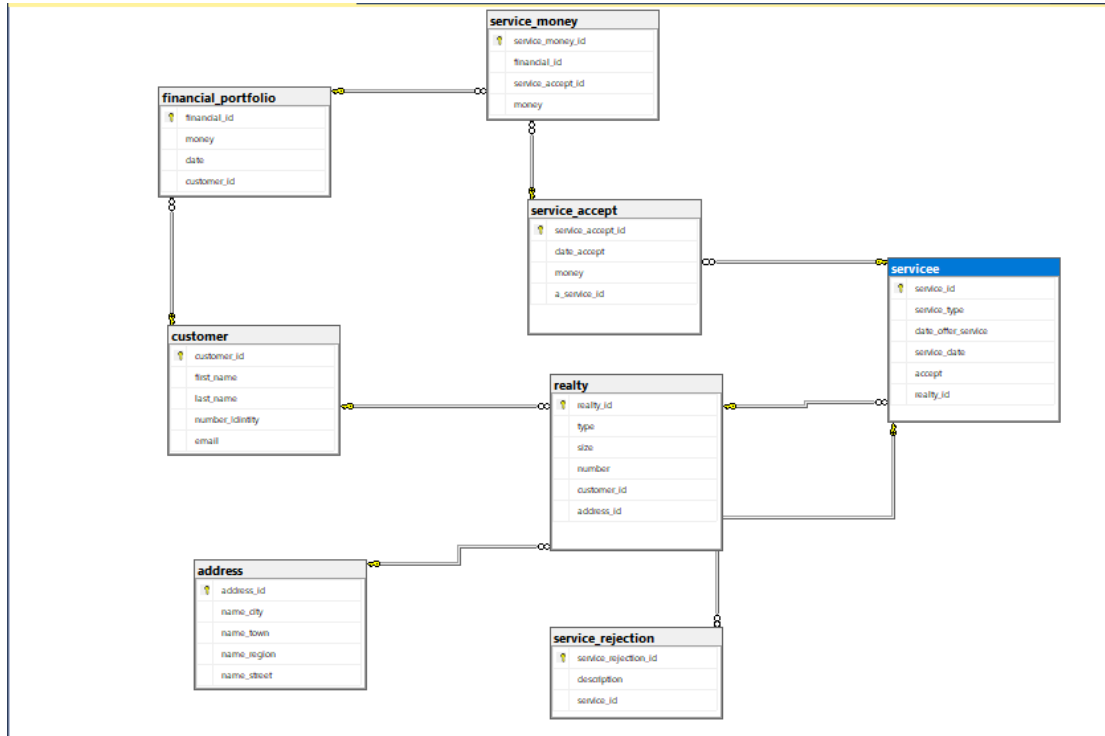
اية أنور ابو فخر

روان عبدالله أبو خزام

اشراف المهندس:

عبد البديع مراد

## Question (1)



لدينا جدول ال servicee وهو جدول الخدمة مرتبط مع جدول ال service\_rejection بعلاقة one to many وأيضا مع جدول ال service\_accept بعلاقة one to many حيث يمثل جدول ال service\_accept طلبات الخدمة المقبولة وال service\_rejection هي طلبات الخدمة غير المقبولة ويرتبط جدول ال servicee أيضا بجدول ال realty وهو جدول العقارات حيث يكون لكل عقار أكثر من خدمة فجدول العقارات مربوط بعلاقة one to many مع جدول ال servicee وأيضا مرتبط مع جدول ال customer حيث يكون جدول ال customer مرتبط بعلاقة one to many بجدول ال realty بالإضافة أنه مرتبط بجدول ال financial\_portfolio وهو جدول المحفظة حيث يكون جدول المحفظة مرتبط بعلاقة one to one بالإضافة انه يرتبط مع جدول ال service\_money حيث يكون هذا الجدول كسر بين جدول المحفظة و جدول الخدمات المقبولة واخيراً أننا نملك جدول ال address وهو جدول العناوين مرتبط بجدول العقار realty حيث يكون جدول العقار مرتبط بجدول ال address بعلاقة one to many وبهذا نكون اكملنا الشرح عن ال erd .

## Question (2)

```
CREATE DATABASE office_for_home_services
ON
PRIMARY
(
    name = office_for_home_services,
    filename = 'C:\Program Files\Microsoft
SQL
Server\MSSQL15.SQLEXPRESS\MSSQL\DATA\
office_for_home_services.mdf',
    size = 10MB,
    MAXSIZE = UNLIMITED,
    FILEGROWTH = 50MB
),
(
    NAME = HW_file1,
    FILENAME = 'C:\Program Files\Microsoft
SQL
Server\MSSQL15.SQLEXPRESS\MSSQL\DATA\
HW_file1.ndf',
    SIZE = 8,
    MAXSIZE = UNLIMITED,
    FILEGROWTH = 10%
),
(
    NAME = HW_file2,
    FILENAME = 'C:\Program Files\Microsoft
SQL
Server\MSSQL15.SQLEXPRESS\MSSQL\DATA\
HW_file2.ndf',
    SIZE = 10MB,
    MAXSIZE = UNLIMITED,
    FILEGROWTH = 10%
),
```

```

FILEGROUP HW_FileGroup
(
    NAME = HW_file3,
    FILENAME = 'C:\Program Files\Microsoft
SQL
Server\MSSQL15.SQLEXPRESS\MSSQL\DATA\HW_f
ile3.ndf',
    SIZE = 20MB,
    MAXSIZE = UNLIMITED,
    FILEGROWTH = 10% ),
(
    NAME = HW_file4,
    FILENAME = 'C:\Program Files\Microsoft
SQL
Server\MSSQL15.SQLEXPRESS\MSSQL\DATA\HW_f
ile4.ndf',
    SIZE = 20MB,
    MAXSIZE = UNLIMITED,
    FILEGROWTH = 10%
)
LOG ON
(
    NAME = HW_Log1,
    FILENAME = 'C:\Program Files\Microsoft
SQL
Server\MSSQL15.SQLEXPRESS\MSSQL\DATA\HW_L
og1.ldf',
    SIZE = 5MB,
    MAXSIZE = UNLIMITED,
    FILEGROWTH = 10%
),

(
    NAME = HW_Log2,
    FILENAME = 'C:\Program Files\Microsoft
SQL
Server\MSSQL15.SQLEXPRESS\MSSQL\DATA\HW_L
og2.ldf',
    SIZE = 5MB,
    MAXSIZE = UNLIMITED,
    FILEGROWTH = 10%
)

```

### Question (3)

----- address table

```
CREATE TABLE address
(
  address_id INT identity(1,1) PRIMARY KEY ,
  name_city varchar(50) not null,
  name_town varchar(50) not null,
  name_region varchar(50) not null,
  name_street varchar(50),
)
ON [HW_FileGroup]
```

----- customer table

```
CREATE TABLE customer
(
  Customer_id INT identity(1,1) PRIMARY KEY ,
  first_name varchar(20) not null,
  last_name varchar(20) not null,
  number_Idintity bigint not null,
)
ON [HW_FileGroup]
```

----- financial\_portfolio table

```
CREATE TABLE financial_portfolio
(
  financial_id INT identity(1,1) PRIMARY KEY ,
  customer_id INT FOREIGN KEY REFERENCES customer(customer_id)
  unique NOT NULL ,
  money int NOT NULL,
  date date NOT NULL,
)
ON [HW_FileGroup]
```

----- realty table

```
CREATE TABLE realty
(
    realty_id INT identity(1,1) PRIMARY KEY ,
    type varchar(50) NOT NULL,
    size int NOT NULL,
    number int NOT NULL,
    address_id INT FOREIGN KEY REFERENCES address(address_id)
    not null,
    customer_id INT FOREIGN KEY REFERENCES customer(customer_id)
    not null,
)
ON [HW_FileGroup]
```

----- service table

```
CREATE TABLE service
(
    service_id INT PRIMARY KEY ,
    service_type varchar(50) NOT NULL,
    date_offer_service date NOT NULL,
    service_date date NOT NULL,
    accept INT NOT NULL,
    name varchar(50) NOT NULL,
    realty_id INT FOREIGN KEY REFERENCES realty(realty_id) NOT
    NULL,
)
ON [HW_FileGroup]
```

----- service\_accept table

```
CREATE TABLE service_accept
(
    service_accept_id INT PRIMARY KEY ,
    date_accept date NOT NULL,
    money bigint NOT NULL,
    service_id INT FOREIGN KEY REFERENCES service(service_id)
    not null,
)
ON [primary]
```

----- service\_money table

```
CREATE TABLE service_money
(
service_money_id INT PRIMARY KEY ,
money bigint NOT NULL,
service_accept_id INT FOREIGN KEY REFERENCES
service_accept(service_accept_id) NOT NULL,
financial_id INT FOREIGN KEY REFERENCES
financial_portfolio(financial_id) NOT NULL,
)
ON [primary]
```

----- service\_rejection table

```
CREATE TABLE service_rejection
(
service_rejection_id INT PRIMARY KEY ,
description text NOT NULL,
service_id INT FOREIGN KEY REFERENCES service(service_id) NOT
NULL,
)
ON [primary]
```



## Question (4)

التعليمة المختارة :

`select service_id , accept from servicee inner join service_accept on  
service_id = a_service_id and accept =1;`

: Non Clustered Index

تم بناء الفهرس باسم ix\_service و ذلك على عمود service\_id في جدول الخدمات (servicee) و ذلك لأننا نقوم بالاستعلام عنه ، الأمر الذي سوف يسرع من عملية الاستعلام بالإضافة إلى تحديد معامل المليء من أجل ترك مساحة 80% فارغة من ال pages .

التعليمة :

`CREATE NONCLUSTERED INDEX ix_service ON servicee(service_id) with  
(FILLFACTOR = 80);`

شرح :

لقد قام بدمج الجدولين من خلال Nested loop ثم استعمل ال Clustered Index للوصول إلى ال service\_accept\_id اما بالنسبة إلى جدول ال service لقد قام ب Nested loop من ثم قام بمسحه عن طريق ال Non Clustered للوصول إلى الأعمدة المطلوبة

SQLQuery3.sql - MO...ology Center (677)\* SQLQuery1.sql - MO...ology Center (56)\* MOHAMDI\SQLEXPRESS - dbo.servicee

select service\_id , accept from servicee inner join service\_accept on service\_id = a\_service\_id and accept =1;

Query 1: Query cost (relative to the batch): 100%  
select service\_id , accept from servicee inner join service\_accept on service\_id = a\_service\_id and accept =1;

Execution plan

100 %

SELECT  
Cost: 0 %

Nested Loops  
(Inner Join)  
Cost: 0 %

Clustered Index Scan (Clustered)  
[service\_accept].[PK\_service\_accept]  
Cost: 44 %

Index Seek (NonClustered)  
[servicee].[ix\_service]  
Cost: 56 %

Properties

Current connection parameters

Aggregate Status

Connection failure:

Elapsed time 00:00:00.162

Finish time 12/2/2023 3:19:17 PM

Name MOHAMDI\SQLEXPRESS

Rows returned 0

Start time 12/2/2023 3:19:17 PM

State Open

Connection

Connection name MOHAMDI\SQLEXPRESS

Connection Details

Connection elapsed 00:00:00.162

Connection encrypt Not encrypted

Connection finish time 12/2/2023 3:19:17 PM

Connection rows 0

Connection start time 12/2/2023 3:19:17 PM

Connection state Open

Display name MOHAMDI\SQLEXPRESS

Login name MOHAMDI\Technology

Server name MOHAMDI\SQLEXPRESS

Server version 15.0.2000

Session Tracing ID

SPID 56

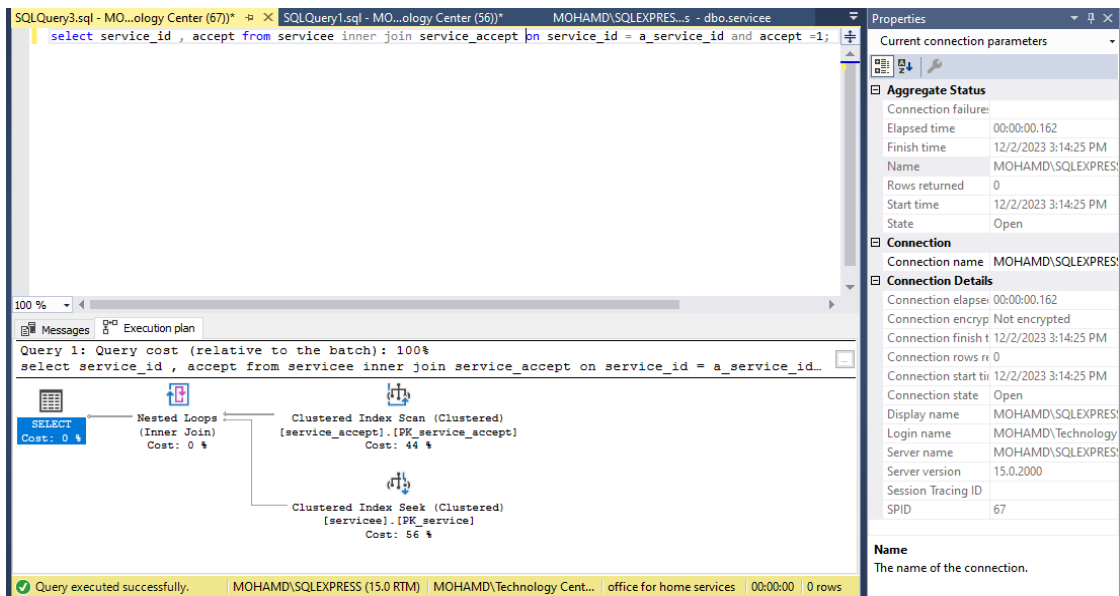
Name

The name of the connection.

Query executed successfully. MOHAMDI\SQLEXPRESS (15.0 RTM) MOHAMDI\Technology Cent... office for home services 00:00:00 0 rows

## : Clustered Index

الفهارس اتم إنشاؤها تلقائياً من قبل ال DBMS عند خلق المفاتيح الأولية في الطلبات السابقة و في استعلامنا قد تم اعتباره الفهرس الأمثل لتنفيذ العملية بالنسبة ل Execution Plan فإنه بالبداية قام بعملية Nested Loop لدمج الجدولين ثم استخدم الفهارس المعنقدة الموجودة في الجدولين للوصول إلى الأعمدة المطلوبة حيث أنه حصل على ال ID الخاص بالخدمة ال service والأعمدة المطلوبة بجدول الخدمة من خلال Index Scan .



## : Non Clustered Index with Include

نلاحظ أننا في الاستعلام الذي تم اعتماده عند استخدام الفهارس غير المعنقدة أننا قد استعرضنا عمود accept لجدول الخدمة أيضا..  
لذلك تم بناء فهرس غير معنقد آخر ix\_include\_service على ذات الجدول و قد احتوت أوراق شجرة هذا الفهرس على عمود accept الأمر الذي يسهم في تسريع عملية الجلب

### : التعليمية :

```
CREATE NONCLUSTERED INDEX ix_include_service ON servicee(accept)
include (service_date);
```

### : شرح :

نلاحظ أن الخطة كانت من خلال استعمال المعنقد في جدول service\_accept والوصول المباشر إلى ID الخاص به Index Seek واستعمال الفهرس ix\_include\_service في جدول الخدمة حيث قام بالوصول مباشرة إلى service\_date و accept ولكنه قام بعملية Index Scan للوصول إلى ID الخاص بال service .

عند إعادة الاستعلام مرة أخرى كانت لدينا النتائج التالية:

The screenshot displays the SQL Server Enterprise Manager interface. The main window shows the execution plan for a query. The query is: `select service_id , accept from servicee inner join service_accept on service_id = a_service_id and accept =1;`. The execution plan shows a Nested Loops (Inner Join) operation with a cost of 0%. The inner join is performed by a Clustered Index Scan (Clustered) on the service\_accept table, with a cost of 44%. The outer join is performed by an Index Seek (NonClustered) on the servicee table, with a cost of 56%. The status bar at the bottom indicates that the query executed successfully.

Properties window details:

- Current connection parameters
- Aggregate Status
  - Connection failure:
  - Elapsed time: 00:00:00.188
  - Finish time: 12/2/2023 3:21:59 PM
  - Name: MOHAMD\SQLEXPRES
  - Rows returned: 0
  - Start time: 12/2/2023 3:21:59 PM
  - State: Open
- Connection
  - Connection name: MOHAMD\SQLEXPRES
- Connection Details
  - Connection elapsed: 00:00:00.188
  - Connection encrypt: Not encrypted
  - Connection finish t: 12/2/2023 3:21:59 PM
  - Connection rows r: 0
  - Connection start t: 12/2/2023 3:21:59 PM
  - Connection state: Open
  - Display name: MOHAMD\SQLEXPRES
  - Login name: MOHAMD\Technology
  - Server name: MOHAMD\SQLEXPRES
  - Server version: 15.0.2000
  - Session Tracing ID
  - SPID: 56
- Name: The name of the connection.

## : Non Clustered Index with filter

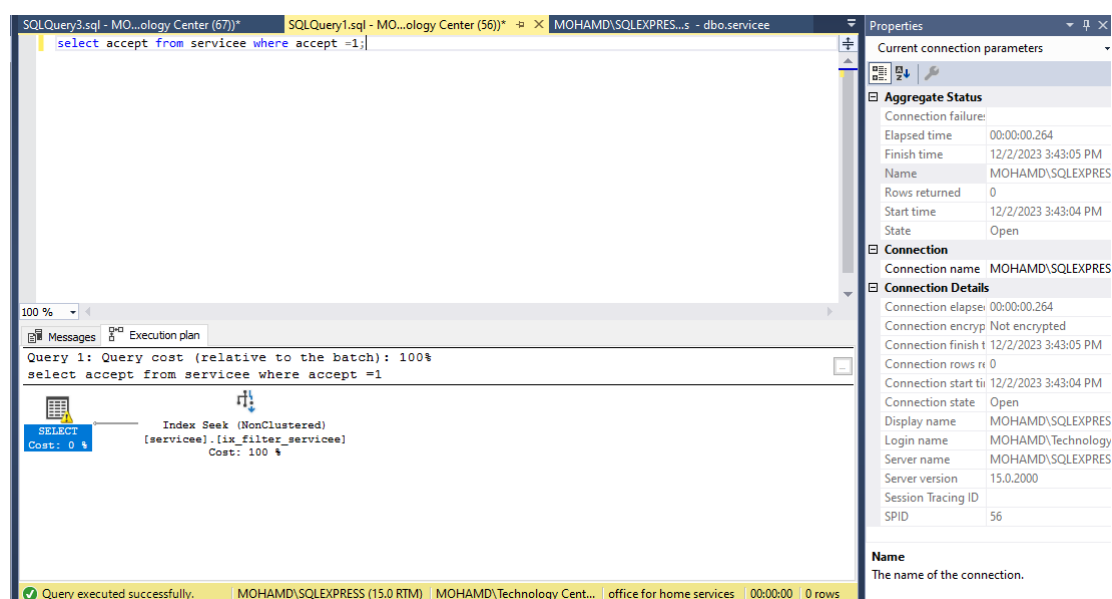
الفهرس الذي تم اعتماده هنا كان بأن نبني فهرس غير معنقد ثالث Filter Index على عمود القبول (accept) في جدول الخدمة (servicee) بحيث لا يتم استخدامه إلا في حال كانت القبول موافق عليها أي قيمته 1 مع تعديل الاستعلام قليلا لإظهار طريقة تطبيق الفهرس فنقوم باستخدام الاستعلام التالي :

```
select accept from servicee where accept =1;
```

التعليمة :

```
CREATE NONCLUSTERED INDEX ix_filter_servicee ON servicee(accept)
where accept = 1;
```

نلاحظ أن تم الوصول إلى النتيجة بشكل مباشر Index Seek :



## Question (5)

```
select service_id , accept
```

```
from servicee
```

```
with(index(ix_include_service))
```

```
inner join service_accept on service_id = a_service_id and accept =1;
```

## Question (6)

set statistics time on

```
select service_id , accept from servicee inner join service_accept on service_id =  
a_service_id and accept =1
```

option(fast 4)

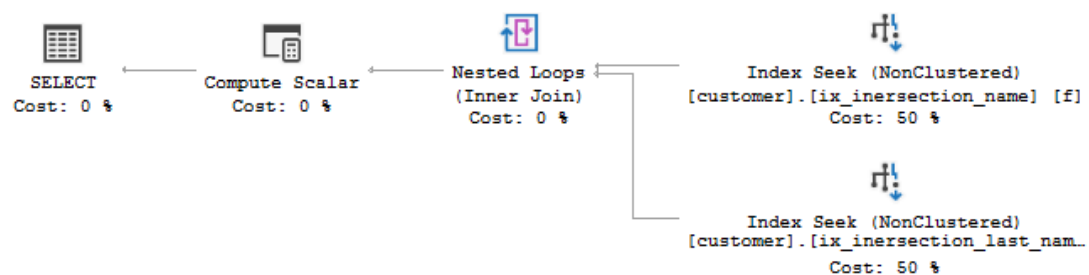
set statistics time off

## Question (7)

```
CREATE NONCLUSTERED INDEX ix_inersection_name  
ON customer (first_name)
```

```
CREATE NONCLUSTERED INDEX ix_inersection_last_name  
ON customer (last_name)
```

```
SELECT f.first_name + l.last_name As 'Full Name' FROM customer l CROSS  
JOIN customer f  
where f.first_name = 'mohamd' and l.last_name ='als';
```



## Question (8)

- Merge Join

تحقيق هذا النوع من الدمج تمت تعبئة جدول العملاء بـ 10000 سطر باستعمال التعليمة التالية :

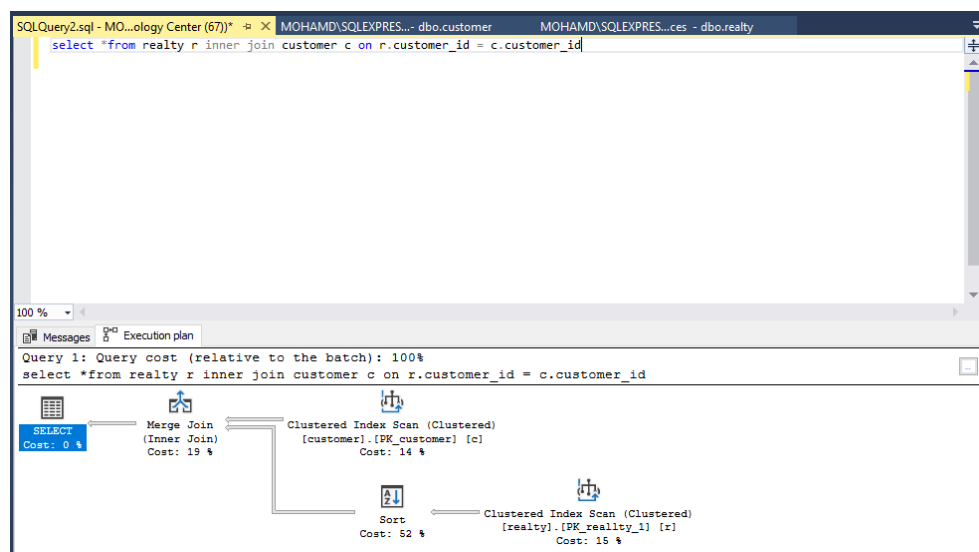
```
DECLARE @i int = 1
WHILE @i < 10000
BEGIN
SET @i=@i+1
INSERT INTO customer VALUES (@i,'mo','al',18932932,'uo@kss')
END
```

وقمنا بتعبئة جدول العقارات بـ 10000 سطر باستعمال التعليمة التالية:

```
DECLARE @i int = 1
WHILE @i < 10000
BEGIN
SET @i=@i+1
INSERT INTO realty VALUES (@i,'clean',102,182,1,3)
END
```

وهنا نجد أننا نحاول دمج بين جدولين يمتلكان عدد كبير من الاسطر ويتوفر شرط مساواة واحدة على الأقل باستخدام الاستعلام التالي :

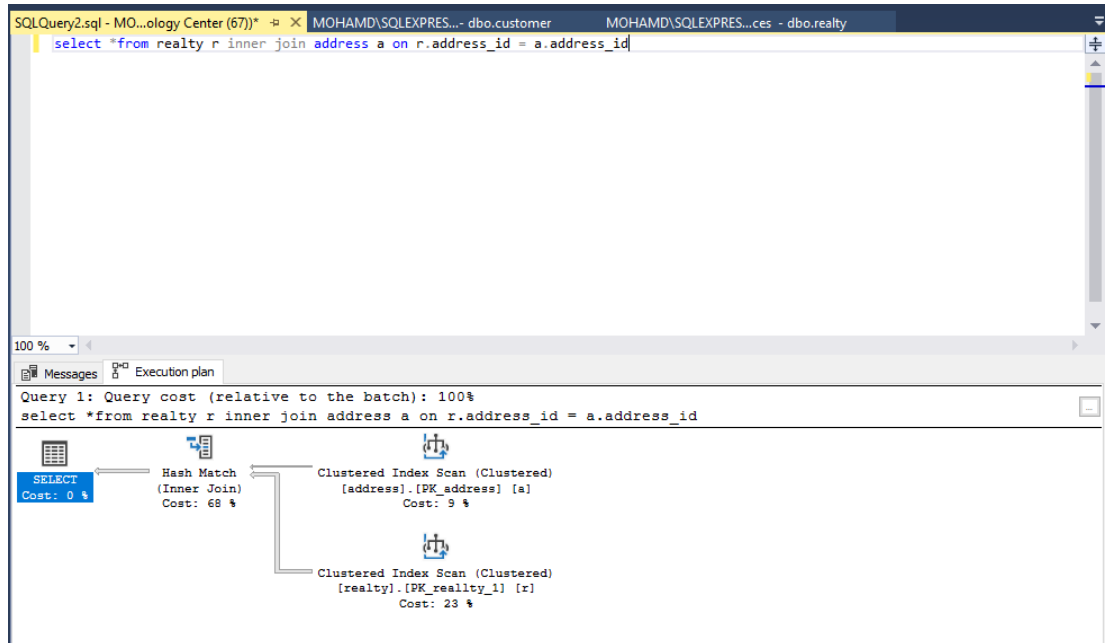
```
select *from realty r inner join customer c on r.customer_id = c.customer_id
```



## • Hash Join

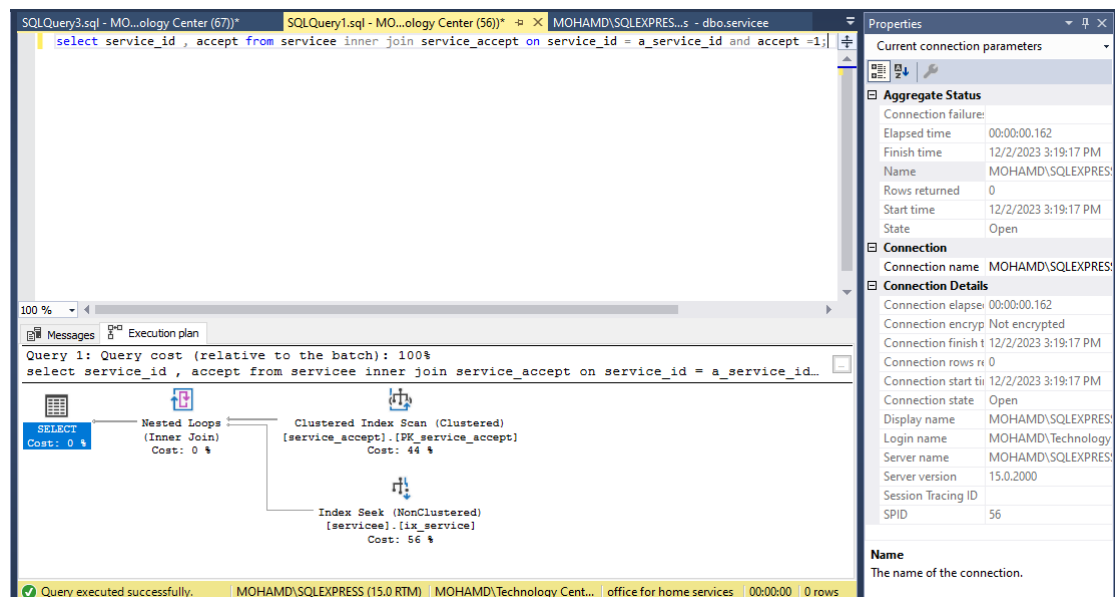
هنا نجد أننا نحاول دمج بين جدولين الأول ضخماً جداً بالنسبة للثاني وهذا ما جعل DBMS يختار عملية Hash Join عند قيامنا باختيار استعمال يدمج الجدولين معاً.. وكان الاستعلام كالتالي:

```
select *from reality r inner join address a on r.address_id = a.address_id
```



## • Nested Loop Join

نلاحظ أن هذا النوع من الدمج يحصل بشكل أمثل عندما يكون لدينا جدولين بعدد قليل من الأعمدة ونلاحظ أيضاً استخدام هذا النوع من الدمج في كل الاستعلامات السابقة .



## Question (9)

تمت تجربة الاستعلام اكثر من 100 مرة وتوضيح استعلامات اختبار الفهارس الجيدة:

```
DECLARE @i int = 1
WHILE @i < 1000
BEGIN
SET @i=@i+1
select * from customer c inner join realty r on c.customer_id = r.customer_id
End
```

```
DECLARE @i int = 1
WHILE @i < 1000
BEGIN
SET @i=@i+1
select * from realty c inner join address r on c.address_id = r.address_id
End
```

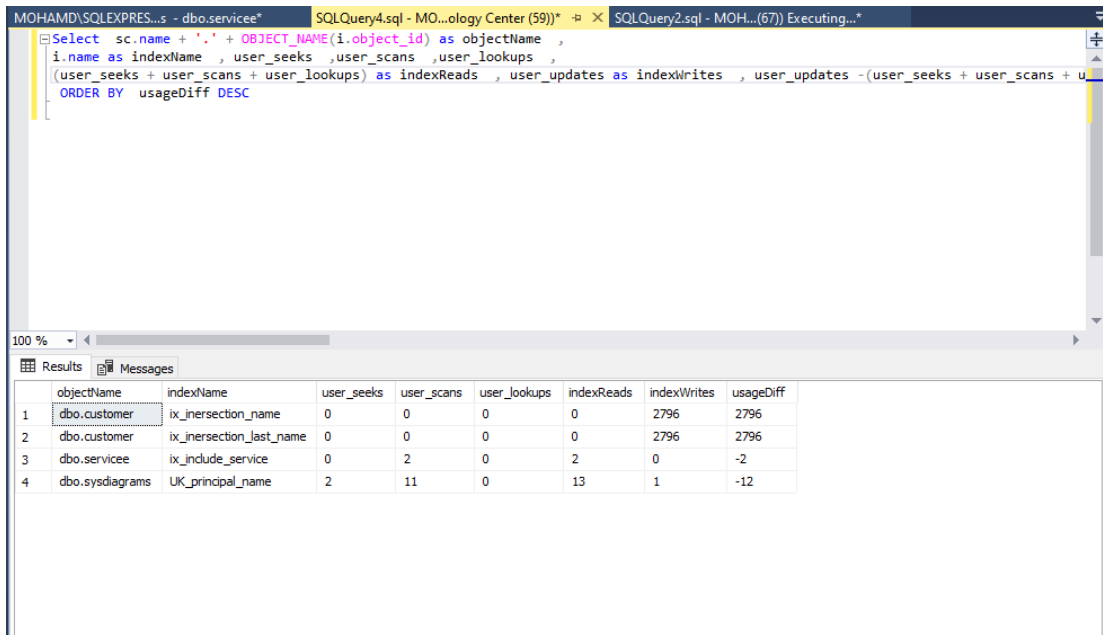
و عند التنفيذ تمت كتابة الاستعلامات التي توضح مدى جودة الفهارس المستخدمة وكانت الاستعلامات التالي

- الاستعلام الأول : يدل على الإنديكس الأكثر أهمية وهو الإنديكس الذي يحوي على أكبر قيمة سالبة وبالتالي هذا الاستعلام يوضح طرح العمليات (Scan , Seek) من عمليات (UpDate) والاستعلام هو التالي:

```
Select
    sc.name + '.' + OBJECT_NAME(i.object_id) as objectName
    , i.name as indexName
    , user_seeks
    , user_scans
    , user_lookups
    , (user_seeks + user_scans + user_lookups) as indexReads
    , user_updates as indexWrites
    , user_updates - (user_seeks + user_scans + user_lookups) as usageDiff
FROM
    sys.dm_db_index_usage_stats s
    INNER JOIN
    sys.indexes i
    on
    i.index_id = s.index_id
    INNER JOIN
    sys.objects o
    ON
    o.object_id = i.object_id
    INNER JOIN
    sys.schemas sc
    ON
    o.schema_id = sc.schema_id
    WHERE
    database_id = db_id()
```



AND  
objectproperty(s.object\_id, 'IsUserTable')=1  
AND  
i.object\_id = s.object\_id  
AND  
i.type\_desc = 'NONCLUSTERED'  
ORDER BY  
usageDiff DESC



The screenshot shows a SQL Server Enterprise Manager window with a query window titled 'SQLQuery4.sql - MOH...ology Center (59))' and another titled 'SQLQuery2.sql - MOH...(67)) Executing...'. The query window contains the following SQL statement:

```

Select sc.name + '.' + OBJECT_NAME(i.object_id) as objectName ,
i.name as indexName , user_seeks , user_scans , user_lookups ,
(user_seeks + user_scans + user_lookups) as indexReads , user_updates as indexWrites , user_updates - (user_seeks + user_scans + u
ORDER BY usageDiff DESC

```

Below the query window, the 'Results' tab is active, displaying a table with 8 columns: objectName, indexName, user\_seeks, user\_scans, user\_lookups, indexReads, indexWrites, and usageDiff. The table contains 4 rows of data:

	objectName	indexName	user_seeks	user_scans	user_lookups	indexReads	indexWrites	usageDiff
1	dbo.customer	ix_intersection_name	0	0	0	0	2796	2796
2	dbo.customer	ix_intersection_last_name	0	0	0	0	2796	2796
3	dbo.servicee	ix_include_service	0	2	0	2	0	-2
4	dbo.sysdiagrams	UK_principal_name	2	11	0	13	1	-12

- الاستعلام الثاني: يدل على الإنديكس الذي لم يتم استخدامه و يتم استخدام هذا الاستعلام بعد مرور فترة زمنية معينة على استخدام الفهارس (ولهذا قد تم تنفيذ الاستعلامات عدد كبير من المرات) و الاستعلام هو التالي :

```

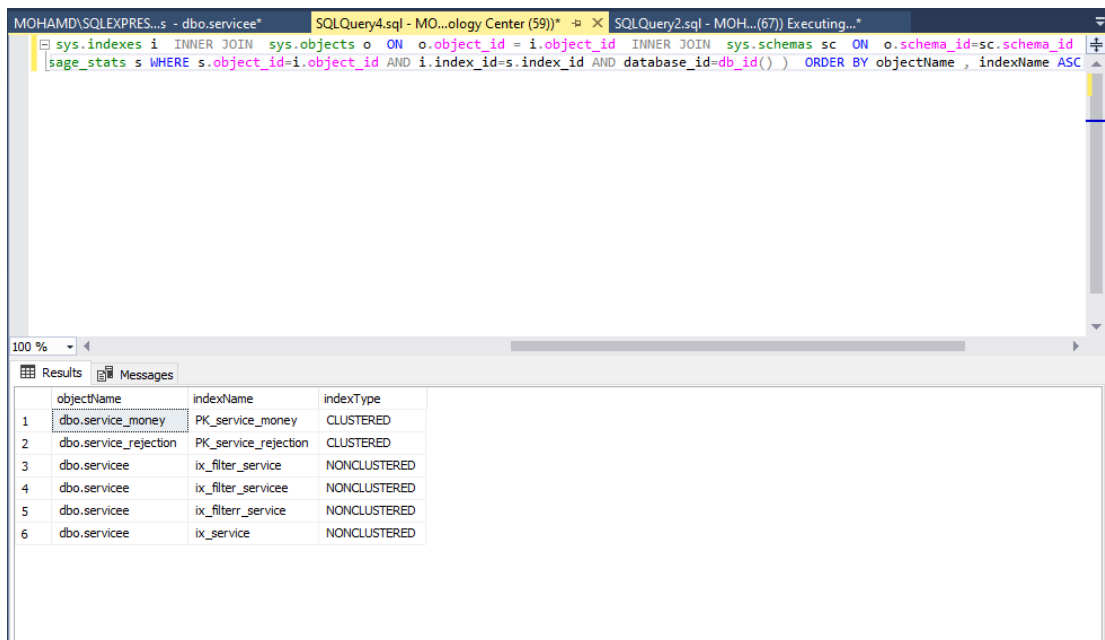
SELECT
sc.name + '.' + object_name(i.object_id) as objectName
,i.name as indexName
,i.type_desc as indexType
FROM
sys.indexes i
INNER JOIN
sys.objects o
ON
o.object_id = i.object_id
INNER JOIN
sys.schemas sc
ON
o.schema_id=sc.schema_id

```

```

WHERE
OBJECTPROPERTY(i.object_id , 'IsUserTable')=1
AND
i.index_id not in
(
SELECT s.index_id
FROM sys.dm_db_index_usage_stats s
WHERE
s.object_id=i.object_id
AND
i.index_id=s.index_id
AND
database_id=db_id()
)
ORDER BY objectName , indexName ASC

```



The screenshot shows a SQL Server Enterprise Manager interface. The top pane displays a query window with the following SQL code:

```

sys.indexes i INNER JOIN sys.objects o ON o.object_id = i.object_id INNER JOIN sys.schemas sc ON o.schema_id=sc.schema_id
sys.dm_db_index_usage_stats s WHERE s.object_id=i.object_id AND i.index_id=s.index_id AND database_id=db_id() ) ORDER BY objectName , indexName ASC

```

The bottom pane shows the results of the query, which is a table with three columns: objectName, indexName, and indexType. The results are as follows:

	objectName	indexName	indexType
1	dbo.service_money	PK_service_money	CLUSTERED
2	dbo.service_rejection	PK_service_rejection	CLUSTERED
3	dbo.serviceee	ix_filter_service	NONCLUSTERED
4	dbo.serviceee	ix_filter_serviceee	NONCLUSTERED
5	dbo.serviceee	ix_filterrr_service	NONCLUSTERED
6	dbo.serviceee	ix_service	NONCLUSTERED

وكما هو متوقع النتيجة كانت هي الفهارس التي لم يتم استخدامها و ذلك لأن الاستعلامات السابقة لم تحتوي على هذه الجداول.

## Question (10)

: Procedure

```
CREATE PROCEDURE [dbo].[Action_on_table_customer]
@first_name varchar(50) = null,
@last_name varchar(50) = null,
@email varchar(50) = null,
@number_Idintity bigint = null

AS BEGIN

DECLARE @counter int
SET @counter = 10002 while (@counter < 10500)
Begin
insert into customer values (@counter,'oi','om',2000021,'re@of')
SET @counter = @counter + 1
END

SET @counter = 1 while (@counter < 100)
Begin
delete from customer where customer_id = @counter
SET @counter = @counter + 1
END

SET @counter = 100 while (@counter < 1000)
Begin
update customer set first_name = @first_name ,
last_name = @last_name ,
email = @email,
number_Idintity = @number_Idintity where customer_id = @counter
SET @counter = @counter + 5
END

END

EXEC Action_on_table_customer
@first_name = 'h',
@last_name = 'r',
@number_Idintity = 32342332,
@email = 'iow@gmail'
GO

select * from customer
```

: insert

```
AS BEGIN
DECLARE @counter int
SET @counter = 10002 while (@counter < 10500)
Begin
insert into customer values (@counter,'oi','om',2000021,'re@of')
SET @counter = @counter + 1
END

SET @counter = 95 while (@counter < 100)
Begin
delete from customer where customer_id = @counter
SET @counter = @counter + 1
END

SET @counter = 100 while (@counter < 1000)
Begin
```

	customer_id	first_name	last_name	number_idinty	email
974	10002	oi	om	2000021	re@of
975	10003	oi	om	2000021	re@of
976	10004	oi	om	2000021	re@of
977	10005	oi	om	2000021	re@of
978	10006	oi	om	2000021	re@of
979	10007	oi	om	2000021	re@of
980	10008	oi	om	2000021	re@of
981	10009	oi	om	2000021	re@of
982	10010	oi	om	2000021	re@of
983	10011	oi	om	2000021	re@of

Query executed successfully. | MOHAMD\SQLXPRESS (15.0 RTM) | MOHAMD\Technology Cent... | office for home services | 00:00:00 | 1,471 rows

: update

```
END

SET @counter = 95 while (@counter < 100)
Begin
delete from customer where customer_id = @counter
SET @counter = @counter + 1
END

SET @counter = 100 while (@counter < 1000)
Begin
update customer set first_name = @first_name ,
last_name = @last_name ,
email = @email,
number_idinty = @number_idinty where customer_id = @counter
SET @counter = @counter + 5
END
END
```

	customer_id	first_name	last_name	number_idinty	email
70	100	h	r	32342332	iow@gmail
71	101	mo	al	18932932	uo@kss
72	102	mo	al	18932932	uo@kss
73	103	mo	al	18932932	uo@kss
74	104	mo	al	18932932	uo@kss
75	105	h	r	32342332	iow@gmail
76	106	mo	al	18932932	uo@kss
77	107	mo	al	18932932	uo@kss
78	108	mo	al	18932932	uo@kss
79	109	mo	al	18932932	uo@kss

Query executed successfully. | MOHAMD\SQLXPRESS (15.0 RTM) | MOHAMD\Technology Cent... | office for home services | 00:00:00 | 1,471 rows

من خلا استخدام الاستعلام التالي أيضاً :

```
EXEC Action_on_table_customer
@first_name = 'h',
@last_name = 'r',
@number_Identity = 32342332,
@email = 'iow@gmail'
GO

select * from customer
```

**: delete**

The screenshot shows a SQL Server Enterprise Manager interface. The top pane displays a query window with the following script:

```
SET @counter = 10002 while (@counter < 10500)
Begin
insert into customer values (@counter, 'oi', 'om', 2000021, 're@of')
SET @counter = @counter + 1
END

SET @counter = 1 while (@counter < 100)
Begin
delete from customer where customer_id = @counter
SET @counter = @counter + 1
END

SET @counter = 100 while (@counter < 1000)
Begin
update customer set first_name = @first_name ,
last_name = @last_name ,
email = @email.
```

The bottom pane shows the 'Results' tab with a table of 11 rows:

customer_id	first_name	last_name	number_Identity	email
101	mo	al	18932932	uo@kss
102	mo	al	18932932	uo@kss
103	mo	al	18932932	uo@kss
104	mo	al	18932932	uo@kss
105	h	r	32342332	iow@gmail
106	mo	al	18932932	uo@kss
107	mo	al	18932932	uo@kss
108	mo	al	18932932	uo@kss
109	mo	al	18932932	uo@kss
110	h	r	32342332	iow@gmail
111	...	...	...	...

The status bar at the bottom indicates: 'Query executed successfully. | MOHAMD\SQLXPRESS (15.0 RTM) | MOHAMD\Technology Cent... | office for home services | 00:00:00 | 1,471 rows'.

```
SELECT S.name as 'Schema',
T.name as 'Table',
I.name as 'Index',
DDIPS.avg_fragmentation_in_percent,
DDIPS.page_count
FROM sys.dm_db_index_physical_stats (DB_ID(), NULL, NULL, NULL, NULL) AS DDIPS
INNER JOIN sys.tables T on T.object_id = DDIPS.object_id
INNER JOIN sys.schemas S on T.schema_id = S.schema_id
INNER JOIN sys.indexes I ON I.object_id = DDIPS.object_id
AND DDIPS.index_id = I.index_id
WHERE DDIPS.database_id = DB_ID()
and I.name is not null
AND DDIPS.avg_fragmentation_in_percent > 0
ORDER BY DDIPS.avg_fragmentation_in_percent desc
```

SQLQuery5.sql - MO...ology Center (67) | SQLQuery new.sql -...nology Center (60)\* | MOHAMD\SQLEXPRES...- dbo.customer

```

select s.name as 'Schema',
t.name as 'Table',
i.name as 'Index',
DDIPS.avg_fragmentation_in_percent,
DDIPS.page_count
from sys.dm_db_index_physical_stats(DB_ID(),NULL,NULL,NULL,NULL) AS DDIPS
INNER JOIN sys.tables t on t.object_id=DDIPS.object_id
INNER JOIN sys.schemas s on t.schema_id=s.schema_id
INNER JOIN sys.indexes i on i.object_id=DDIPS.object_id
AND DDIPS.index_id=i.index_id
WHERE DDIPS.database_id = DB_ID()
and i.name is not null
AND DDIPS.avg_fragmentation_in_percent >0
ORDER BY DDIPS.avg_fragmentation_in_percent desc

```

100 %

	Schema	Table	Index	avg_fragmentation_in_percent	page_count
1	dbo	customer	ix_inserction_last_name	80	5
2	dbo	customer	ix_inserction_name	75	4
3	dbo	realty	PK_realty_1	50	6
4	dbo	customer	PK_customer	16.6666666666667	12

Query executed successfully. | MOHAMD\SQLEXPRESS (15.0 RTM) | MOHAMD\Technology Cent... | office for home services | 00:00:00 | 4 rows

نلاحظ هنا أن الفهارس غير المستخدمة (إلى حد الآن) هي الفهارس المعنقدة و غير المعنقدة الموجودة في الجداول التي لم يتم استعمالها في الاستعلامات المتبعة سابقاً.

و إعادة ترتيب البنية الفيزيائية للفهارس المتضررة يكون من خلال :

**DBCC INDEXDEFRAG('office for home services ', 'service\_rejection')**

و يتم تطبيق ذات التعليمة على كل الجداول غير المستخدمة .

SQLQuery5.sql - MO...ology Center (67) | SQLQuery new.sql -...nology Center (60)\* | MOHAMD\SQLEXPRES...- dbo.customer

```

DBCC INDEXDEFRAG('office for home services ', 'service_rejection')
go

```

100 %

	Index Name	Pages Scanned	Pages Moved	Pages Removed
1	PK_service_rejection	0	0	0

Query executed successfully. | MOHAMD\SQLEXPRESS (15.0 RTM) | MOHAMD\Technology Cent... | office for home services | 00:00:00 | 1 rows

## Question (11)

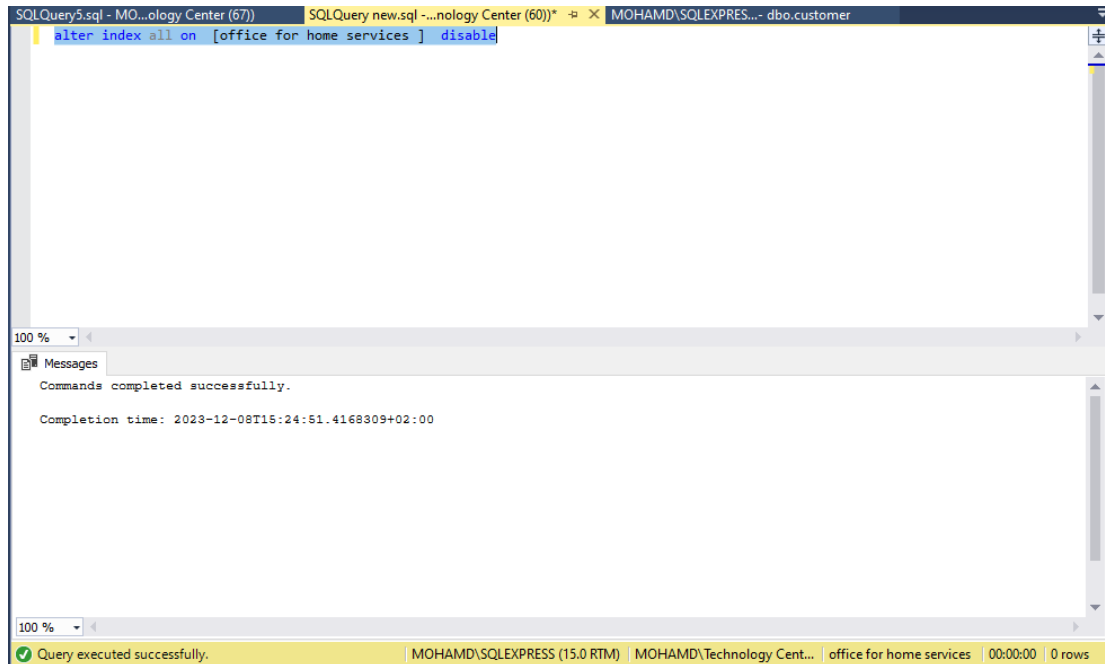
تجميد الفهارس:

من أجل تجميد الفهارس تم استخدام التعليمة التالية:

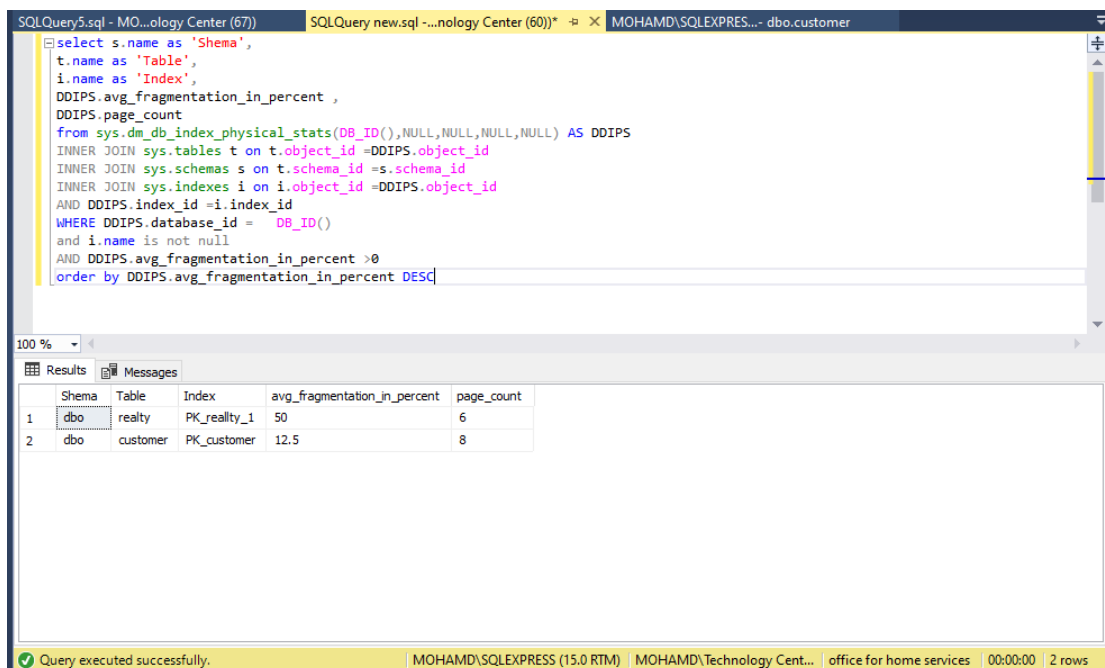
```
ALTER INDEX ix_inersection_last_name ON customer REBUILD;  
ALTER INDEX ix_inersection_name ON customer REBUILD;
```

أو

```
alter index all on [office for home services ] disable
```



وإعادة اختبار التجزئة السابق كالتالي:

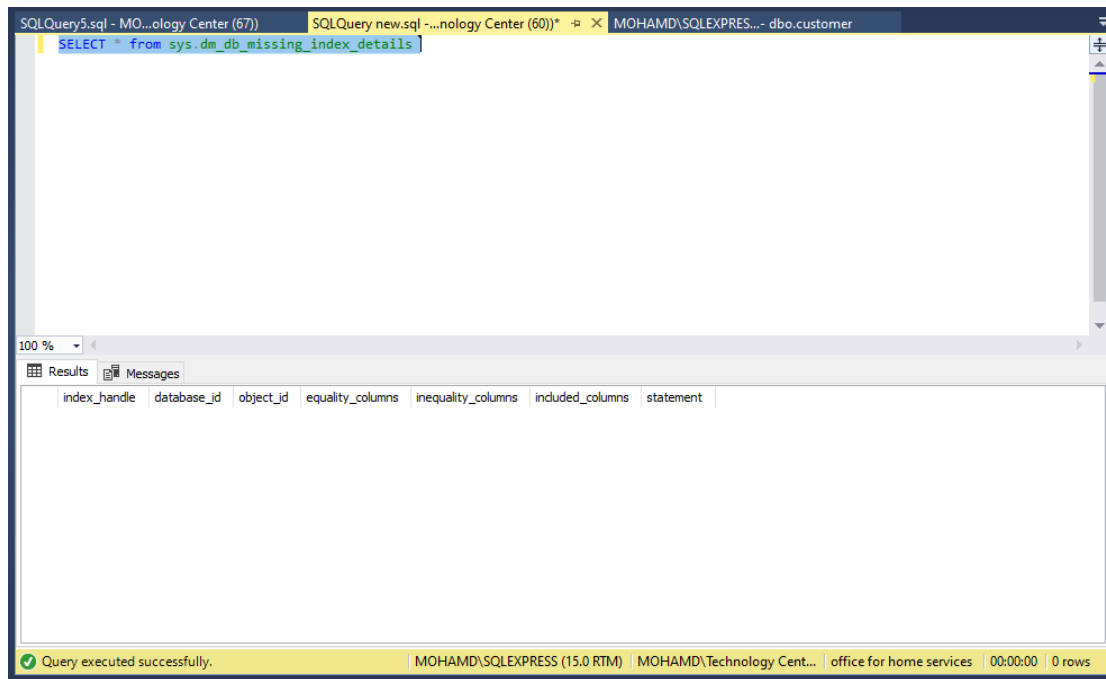


الاستعلام الذي يساعد على تحديد الفهارس المقترحة و تحسين الأداء:

التعليمة المستخدمة هي :

```
SELECT * from sys.dm_db_missing_index_details
```

و بما أن عدد الفهارس المستخدمة على الجداول قليل نسبيا ف بالتالي من الطبيعي أن يظهر لدينا جدول بدون أي نتائج كالتالي:



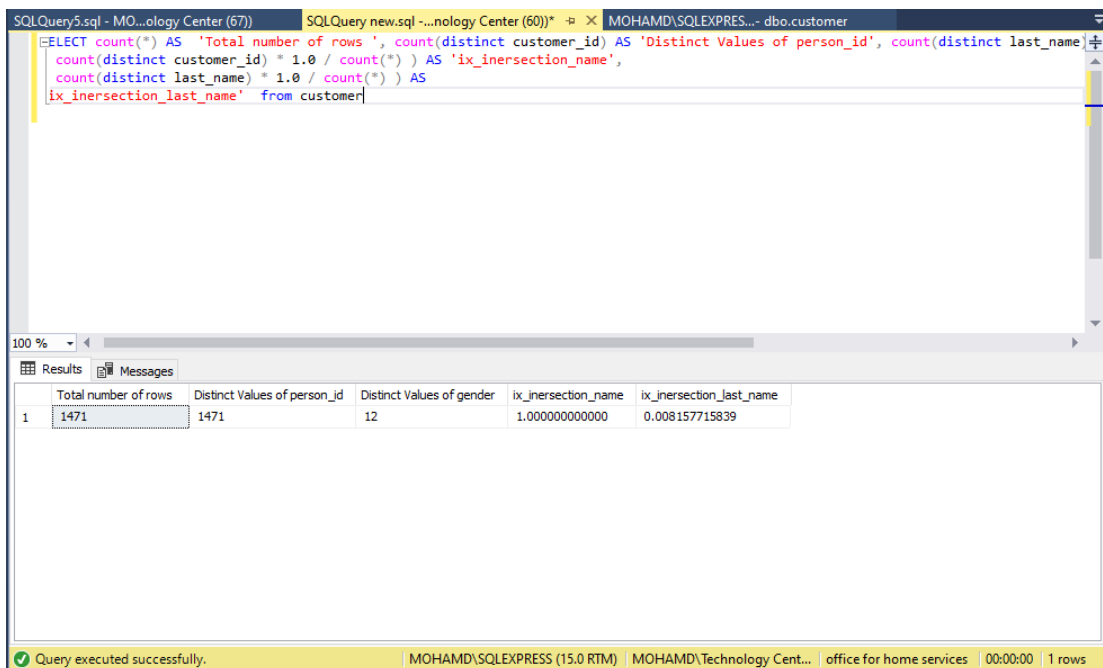


## Question (12)

ان عامل الانتقائية للفهارس على جدول ما يعطي بالعلاقة التالية :

```
SELECT count(*) AS 'Total number of rows ', count(distinct customer_id) AS  
'Distinct Values of person_id', count(distinct last_name) AS 'Distinct Values of  
gender',  
( count(distinct customer_id) * 1.0 / count(*) ) AS 'ix_intersection_name',  
( count(distinct last_name) * 1.0 / count(*) ) AS  
'ix_intersection_last_name' from customer
```

نلاحظ أن انتقائية **ix\_intersection\_last\_name** هي 100 % لأن القسمة هو 1 بينما انتقائية **ix\_intersection\_name** ضئيلة جدا أقل من 0.5%



The screenshot shows a SQL Server Enterprise Manager window with a query executed against the 'customer' table. The query calculates various statistics and intersection values. The results are displayed in a table with 6 columns: 'Total number of rows', 'Distinct Values of person\_id', 'Distinct Values of gender', 'ix\_intersection\_name', and 'ix\_intersection\_last\_name'. The first row shows the results for the query.

	Total number of rows	Distinct Values of person_id	Distinct Values of gender	ix_intersection_name	ix_intersection_last_name
1	1471	1471	12	1.000000000000	0.008157715839

Query executed successfully. MOHAMD\SQLEXPRESS (15.0 RTM) MOHAMD\Technology Cent... office for home services 00:00:00 1 rows

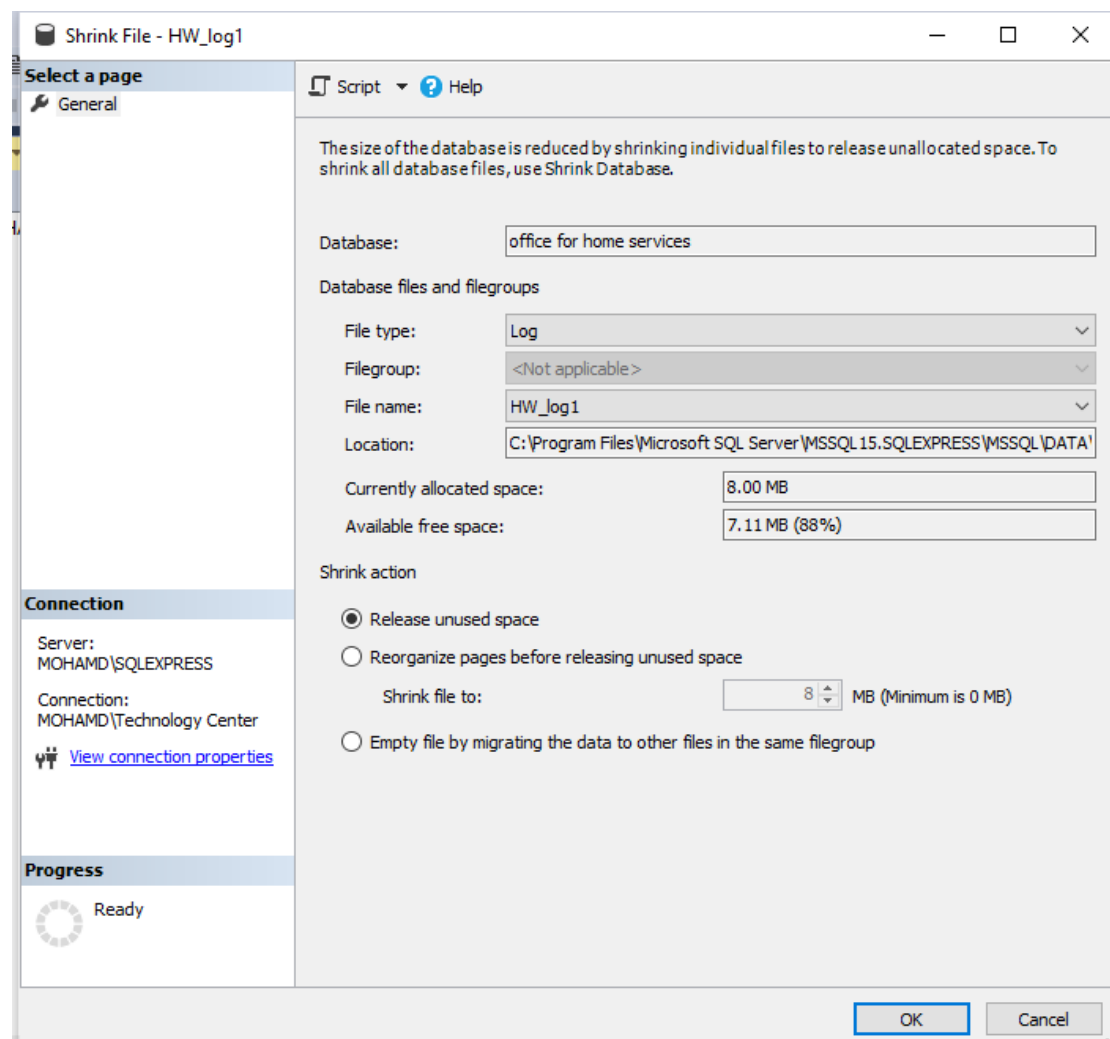
## Question (13-14)

### :Shrink Files

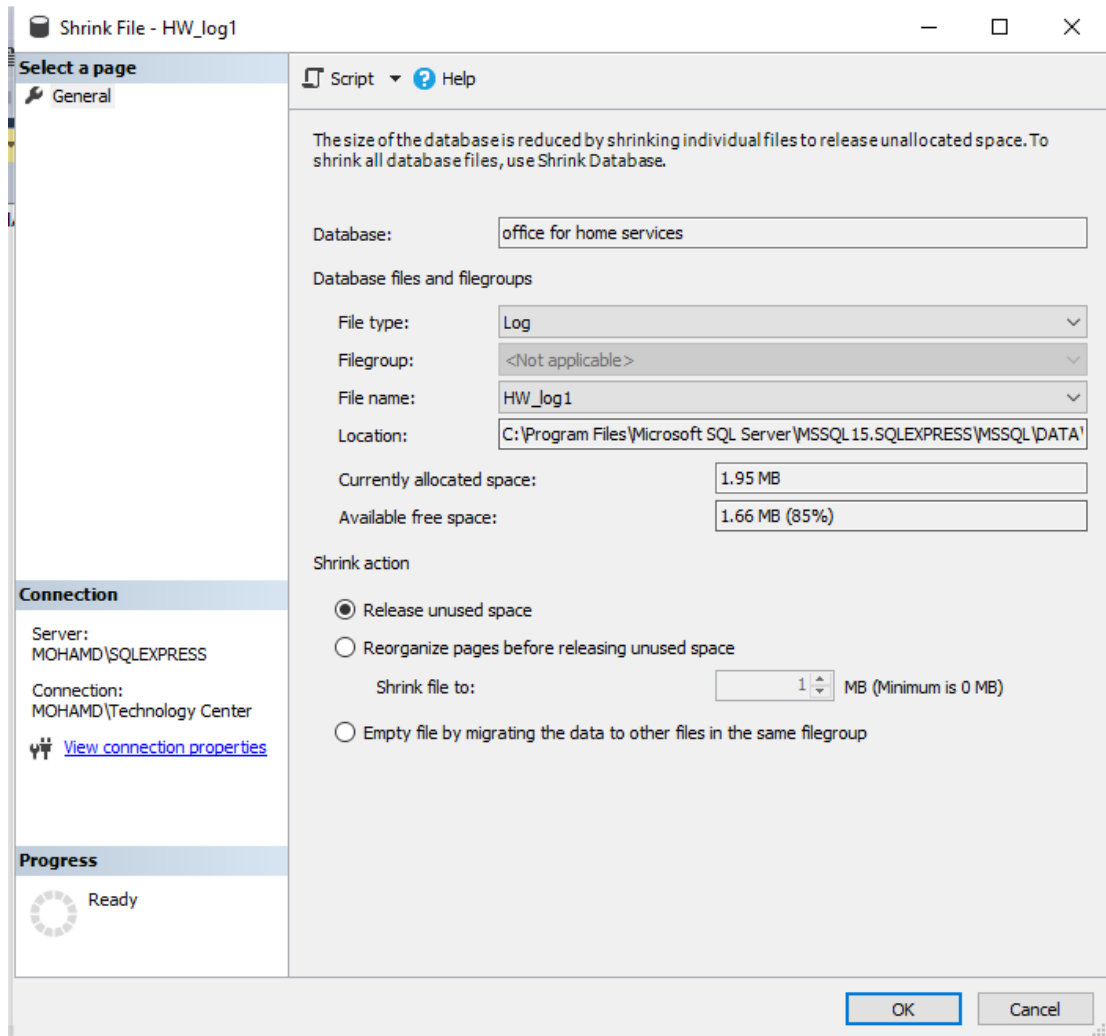
تم تطبيق عملية shrink على ملف log .. ومن المعروف أن ال shrink تساعد على تخفيف حجم الملف عن طريق إزالة الصفحات الفارغة من ملفات قاعدة البيانات و تحرير المساحة الفارغة إما إلى نظام التشغيل أو إلى Sql Server وهي لا تعني ضغط الملفات . و تم استخدام التعليمية التالية :

```
USE [office for home services ]  
GO  
DBCC SHRINKFILE('HW_log1',1)  
GO
```

قبل تنفيذ هذه التعليمية كان لدينا الملف يأخذ مساحة حالية حجمه 8mp كالتالي:



و بعد تطبيق التعليمة تحول الحجم من 8MP إلى 1.95MP كالتالي:



## :Shrink Data Base

الفرق بين القيام بعملية للملفات أو للداتا بيز هو أنه في الداتا بيز يتم تحديد النسبة المئوية المسموحة أن تكون كفاغ و تم استخدام التعليمة التالية:

```
USE [office for home services ]
GO
DBCC SHRINKDATABASE('office for home services ',6)
GO
```

حيث كان حجم الداتا بيز قبل هذه العملية 53.88MP :

Shrink Database - office for home services

Select a page

General

Script Help

The size of a database is reduced by collectively shrinking the database files, releasing unused space. To shrink individual database files, use Shrink Files instead.

Database: office for home services

Database size

Currently allocated space: 53.88 MB

Available free space: 31.06 MB (57%)

Shrink action

☐ Reorganize files before releasing unused space. Selecting this option may affect performance.

Maximum free space in files after shrinking: 0 %

OK Cancel

Connection

Server: MOHAMD\SQLEXPRESS

Connection: MOHAMD\Technology Center

[View connection properties](#)

Progress

Ready

وعند تنفيذ التعليمة تم تخفيض الحجم إلى 53.70MP :

Shrink Database - office for home services

Select a page

General

Script Help

The size of a database is reduced by collectively shrinking the database files, releasing unused space. To shrink individual database files, use Shrink Files instead.

Database: office for home services

Database size

Currently allocated space: 53.70 MB

Available free space: 30.88 MB (57%)

Shrink action

☐ Reorganize files before releasing unused space. Selecting this option may affect performance.

Maximum free space in files after shrinking: 0 %

## Question (15)

```
BEGIN TRY
  BEGIN TRANSACTION
  DECLARE @i INT

  SET @i = 5000
  INSERT INTO customer VALUES(@i,'Osama','Othman',2121111,'sdsa@sa')
  UPDATE financial_portfolio SET customer_id = 7 WHERE customer_id = 10

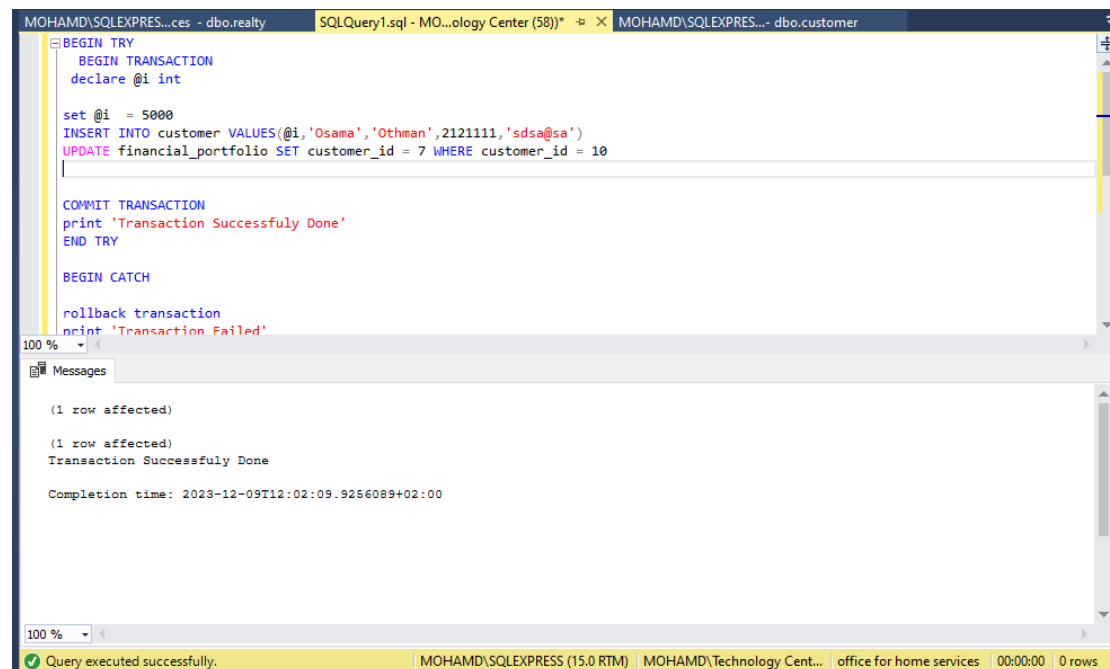
  COMMIT TRANSACTION
  PRINT 'Transaction Successfully Done'
END TRY

BEGIN CATCH

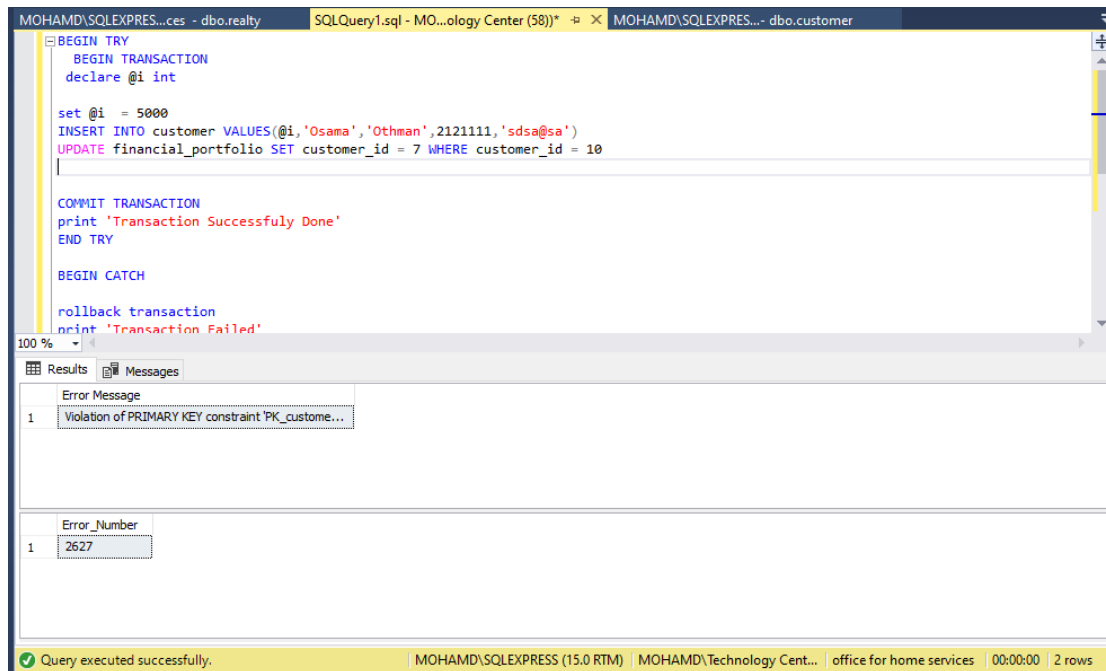
  ROLLBACK TRANSACTION
  PRINT 'Transaction Failed'
  SELECT ERROR_MESSAGE() AS [Error Message]
  SELECT ERROR_NUMBER() AS [Error_Number]

END CATCH
```

والتنفيذ كان كالتالي ففي البداية قام بتنفيذ العملية بنجاح لان المعلومات والداता المدخلة صحيحة:



وعند إعادة تنفيذ هذه التعليمة سوف تظهر لنا رسالة خطأ ال ID وكانت نتيجة الخطأ 27627 مع رسالة  
(Violation of PRIMARY KEY constraint) كالتالي :



## Question (16)

كانت التعليمة كالتالي :

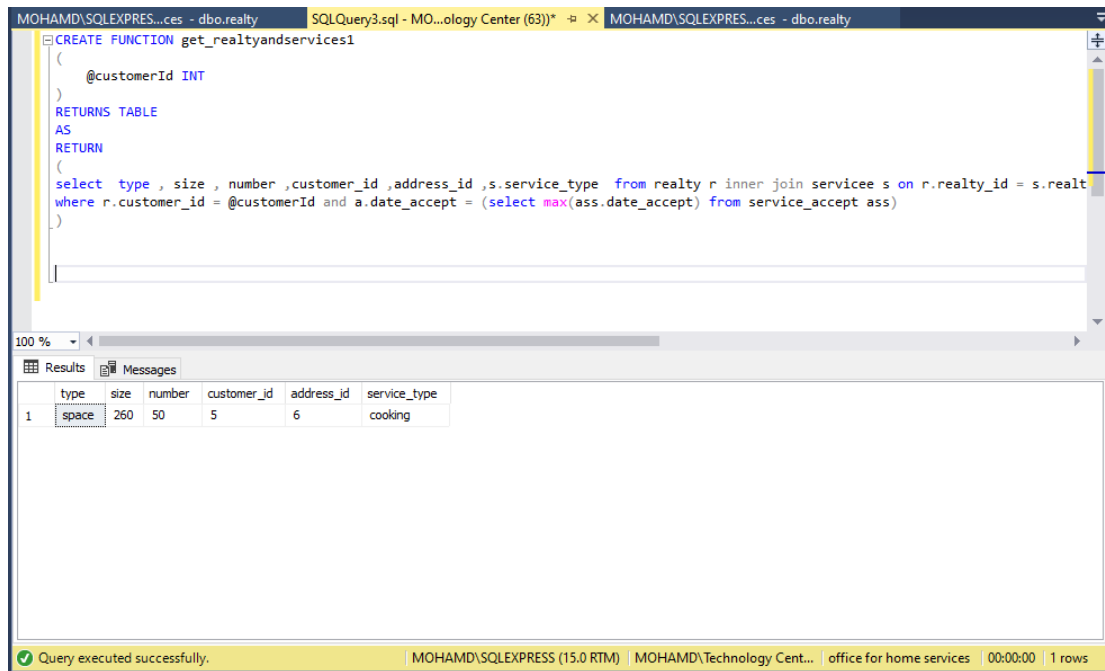
```

CREATE FUNCTION get_realtyandservices1
(
    @customerId INT
)
RETURNS TABLE
AS
RETURN
(
    select type , size , number ,customer_id ,address_id ,s.service_type from realty r
    inner join servicee s on r.realty_id = s.realty_id inner join service_accept a on
    s.service_id = a.a_service_id
    where r.customer_id = @customerId and a.date_accept = (select max (ass.date_accept)
    from service_accept ass)
)

```

والاستدعاء يكون كالتالي:

```
select * from get_realtyandservices1(5)
```



## Question (17)

كانت التعليمة كالتالي :

```

CREATE FUNCTION get_realtyandservicecsss
(
    @customerId INT
)
RETURNS TABLE
AS
RETURN
(
    select type , size , number ,customer_id ,address_id ,s.service_type from realty r
    inner join service s on r.realty_id = s.realty_id inner join service_accept a on
    s.service_id = a.a_service_id
    where r.customer_id = @customerId
)

```

والاستدعاء يكون كالتالي:

```
select * from get_realtyandservicecsss(1);
```

MOHAMD\SQLEXPRES...ces - dbo.realty | SQLQuery3.sql - MO...ology Center (63)) \* | MOHAMD\SQLEXPRES...ces - dbo.realty

```
CREATE FUNCTION get_realtyandserviceess
(
    @customerId INT
)
RETURNS TABLE
AS
RETURN
(
    select type , size , number ,customer_id ,address_id ,s.service_type from realty r inner join servicee s on r.real
    where r.customer_id = @customerId
)

select * from get_realtyandserviceess(1);
```

110 %

Results Messages

	type	size	number	customer_id	address_id	service_type
1	space	190	21	1	3	cooking
2	space	200	20	1	8	cleaning
3	space	200	20	1	8	cleaning
4	space	190	21	1	3	babysitter
5	space	200	20	1	8	cooking

Query executed successfully. | MOHAMD\SQLEXPRESS (15.0 RTM) | MOHAMD\Technology Cent... | office for home services | 00:00:00 | 5 rows



## Question (18)

```
CREATE TRIGGER deduct_service_fee_trigger
on service
for INSERT
as
BEGIN

DECLARE
    @realty_id int;
DECLARE @acceptable_fee int;
-- استخراج معلومات العميل والرسوم المقبولة --
SELECT @realty_id=realty_id ,@acceptable_fee= 5
FROM INSERTED

-- قطع قيمة الخدمة المقبولة من محفظة العميل --
UPDATE customer
SET number_idintity = number_idintity - @acceptable_fee
WHERE customer_id = (
    select realty_id from realty r
    where r.realty_id = (
        select realty_id from service where realty_id =@realty_id
    ) )
end
```