

**Министерство науки и высшего образования Российской Федерации  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«Национальный исследовательский университет ИТМО»  
(Университет ИТМО)**

**Факультет  
Образовательная программа**

**Инфокоммуникационных технологий  
11.03.02 Программирование в  
инфокоммуникационных системах**

**ОТЧЕТ**  
по лабораторной работе 4  
по дисциплине «Разработка баз данных»

**Выполнил: студент группы K33202  
Рогозина Вероника Сергеевна  
Проверил: ст. преподаватель Осетрова И.С.**

Санкт-Петербург  
2024

## 1. Цель работы

Целью данной лабораторной работы является обработка данных.

## 2. Задачи, решаемые при выполнении работы

- 2.1. Добавление данных.
- 2.2. Обновление данных.
- 2.3. Удаление данных.

## 3. Объект исследования

Объектом исследования в данной лабораторной работе являются методы обработки данных (добавление, обновление и удаление) с помощью инструмента Microsoft SQL Server.

## 4. Исходные данные

Инструкция к лабораторной работе, инструмент Microsoft SQL Server Management Studio 2019, база данных, таблицы, индексы, внешние и первичные ключи, созданные в лабораторных работах №1,2,3, файлы, содержащие готовые коды запросов, идущие в приложение к инструкции для лабораторной работы №4.

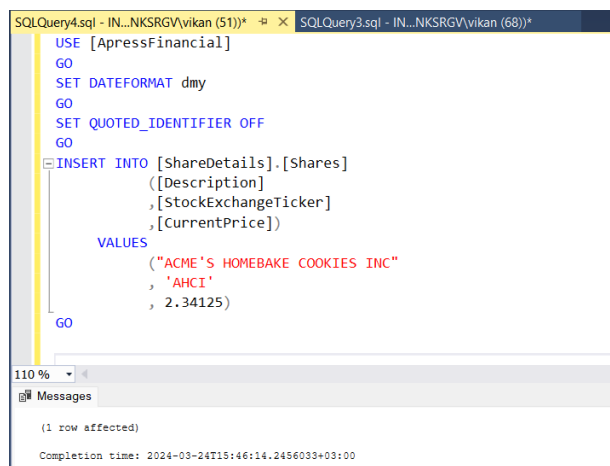
## 5. Выполнение работы

### 5.1. Добавление данных

В ходе выполнения данной задачи необходимо ознакомиться со способами добавления данных, создания ограничений на добавление данных и извлечения данных.

#### 5.1.1. Создание сценария в Query Editor.

Код скрипта, который заполняет одну строку данных представлен на рисунке 1. Для корректной работы сценария была добавлена инструкция SET QUOTED\_IDENTIFIER OFF.




```
USE [ApressFinancial]
GO
SET DATEFORMAT dmy
GO
SET QUOTED_IDENTIFIER OFF
GO
INSERT INTO [ShareDetails].[Shares]
(
    [Description]
    , [StockExchangeTicker]
    , [CurrentPrice]
)
VALUES
(
    "ACME'S HOMEBAKE COOKIES INC"
    , 'AHCI'
    , 2.34125
)
GO
```

110 %  
Messages  
(1 row affected)  
Completion time: 2024-03-24T15:46:14.2456033+03:00

Рисунок 1 – Код скрипта, предназначенного для заполнения данными одной строки

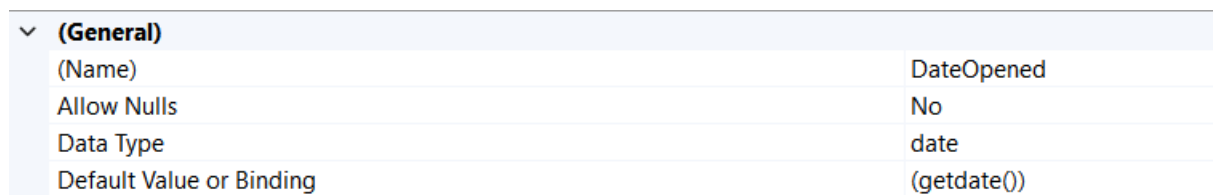
### 5.1.2. Отметки NULL

Для начала необходимо было убедиться, что свойство Is Identity для поля CustomerId установлено в Yes (Рисунок 2), а для поля DateOpened указано значение по умолчанию (getdate()), что показано на рисунке 3.



Identity Specification	Yes
(Is Identity)	Yes
Identity Increment	1
Identity Seed	1

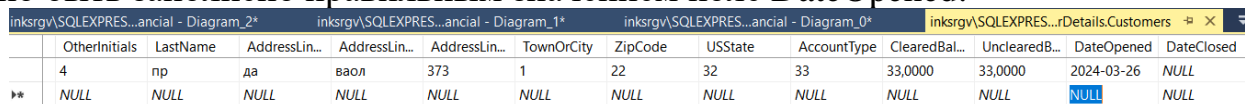
Рисунок 2 – Проверка свойств поля CustomerId



(General)	
(Name)	DateOpened
Allow Nulls	No
Data Type	date
Default Value or Binding	(getdate())

Рисунок 3 - Проверка свойств поля DateOpened

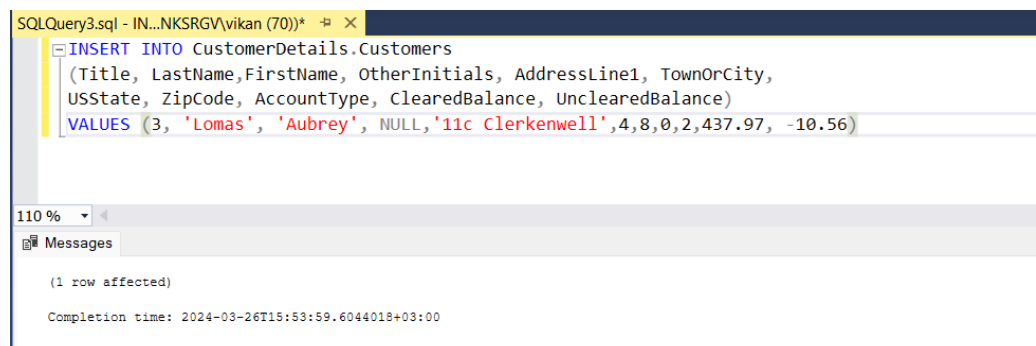
Далее были введены значения для одной строки в таблицу CustomerDetails.Customers, что показано на рисунке 4. Пропуск в заполнении допускают поля OtherInitials, AddressLine2, AddressLine3, ZipCode, AccountType, ClearedBalance, UnclearedBalance, DateClosed. Обязательно должно быть заполнено правильным значением поле DateOpened.



	OtherInitials	LastName	AddressLin...	AddressLin...	AddressLin...	TownOrCity	ZipCode	USState	AccountType	ClearedBal...	UnclearedB...	DateOpened	DateClosed
	4	np	да	ваол	373	1	22	32	33	33,0000	33,0000	2024-03-26	NULL
»»	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Рисунок 4 – Заполнение полей таблицы вручную

Затем был выполнен запрос, который заполнит одну строку таблицы CustomerDetails.Customers. Код данного запроса представлен на рисунке 5. Успешное добавление данных в таблицу с помощью запроса представлено на рисунке 6.



```
SQLQuery3.sql - IN...NKSARGV\vikar (70)) * X
INSERT INTO CustomerDetails.Customers
(Title, LastName, FirstName, OtherInitials, AddressLine1, TownOrCity,
USState, ZipCode, AccountType, ClearedBalance, UnclearedBalance)
VALUES (3, 'Lomas', 'Aubrey', NULL, '11c Clerkenwell', 4, 8, 0, 2, 437.97, -10.56)
```

110 %  
Messages  
(1 row affected)  
Completion time: 2024-03-26T15:53:59.6044018+03:00

Рисунок 5 – Код запроса для заполнения одной строки таблицы

inksrgv\SQLEXPRES...rDetails.Customers													
SQLQuery3.sql - IN...NKSRGV\vikan (70))*													
	CustomerId	Title	FirstName	OtherInitials	LastName	AddressLin...	AddressLin...	AddressLin...	TownOrCity	ZipCode	USState	AccountType	ClearedBal...
▶	1	2	3	4	np	да	ваол	373	1	22	32	33	33,0000
	7	3	Aubrey	NULL	Lomas	11c Clerken...	NULL	NULL	4	0	8	2	437,9700
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Рисунок 6 – Результат заполнения одной строки таблицы с использованием запроса

5.1.3. Инструкция проверки целостности DBCC

Код запроса, который сначала удаляет все записи в таблице, затем значение для столбца IDENTITY снова переустанавливает в 0 и заново добавляет информацию о клиентах, представлен на рисунке 7. Результат запроса представлен на рисунке 8

DELETE FROM CustomerDetails.Customers;

DBCC CHECKIDENT ('CustomerDetails.Customers', RESEED,0);

INSERT INTO CustomerDetails.Customers (Title, LastName,FirstName, OtherInitials, AddressLine1, TownOrCity, USState, ZipCode, AccountType, ClearedBalance, UnclearedBalance) VALUES (1, 'Morgala', 'Nola', NULL, '87 Winnie Way',4,8,112,0,NULL,NULL)

INSERT INTO CustomerDetails.Customers (Title, LastName,FirstName, OtherInitials, AddressLine1, TownOrCity, USState, ZipCode, AccountType, ClearedBalance, UnclearedBalance) VALUES (2, 'Lomas', 'Aubrey', NULL, '11c Clerkenwell',4,8,0,2,437.97, -10.56)

GO

110 %

Messages

(2 rows affected)  
Проверка сведений об идентификаторе: текущее значение идентификатора "2".  
Выполнение DBCC завершено. Если DBCC выдает сообщения об ошибках, обратитесь к системному администратору.  
  
(1 row affected)  
  
(1 row affected)  
  
Completion time: 2024-03-26T16:01:24.8304942+03:00

Рисунок 7 – Код запроса

inksrgv\SQLEXPRES...rDetails.Customers													
SQLQuery4.sql - IN...NKSRGV\vikan (52))*													
SQLQuery3.sql - IN...NKSRGV\vikan (70))*													
	CustomerId	Title	FirstName	OtherInitials	LastName	AddressLin...	AddressLin...	AddressLin...	TownOrCity	ZipCode	USState	AccountType	ClearedBal...
▶	1	1	Nola	NULL	Morgala	87 Winnie ...	NULL	NULL	4	112	8	0	NULL
	2	2	Aubrey	NULL	Lomas	11c Clerken...	NULL	NULL	4	0	8	2	437,9700
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Рисунок 8 - Результат выполнения запроса

#### 5.1.4. Вставка значений в столбцы Identity

Код запроса, который выполняет вставку значений в столбцы Identity представлен на рисунке 9.

```
USE ApressFinancial
GO

CREATE TABLE dbo.T1 (column_1 int IDENTITY, column_2 VARCHAR(30));

INSERT T1 VALUES ('Row #1');
INSERT T1 (column_2) VALUES ('Row #2');

SET IDENTITY_INSERT T1 ON;

INSERT INTO T1 (column_1, column_2)
VALUES (-99, 'Explicit identity value');

SELECT column_1, column_2 FROM T1;

SET IDENTITY_INSERT T1 OFF;
```

Results

	column_1	column_2
1	1	Row #1
2	2	Row #2
3	-99	Explicit identity value

Results

	column_1	column_2
1	1	Row #1
2	2	Row #2
3	-99	Explicit identity value
4	3	What Row?

Results

	column_1	column_2
1	1	Row #1
2	2	Row #2
3	-99	Explicit identity value
4	3	What Row?
5	100	Explicit identity value

Results

	column_1	column_2
1	1	Row #1
2	2	Row #2

Рисунок 9 – Код запроса

#### 5.1.5. Ограничения для столбцов

Код запроса, который добавляет три ограничения к таблице CustomerDetails.CustomerProducts представлен на рисунке 10. Результат создания всех ограничений представлен на рисунке 11.

```
ALTER TABLE CustomerDetails.CustomerProducts
ADD CONSTRAINT PK_CustomerProducts
PRIMARY KEY CLUSTERED (CustomerFinancialProductId)
ON [SECONDARY]
GO

ALTER TABLE CustomerDetails.CustomerProducts
WITH NOCHECK
ADD CONSTRAINT CK_CustProds_AmtCheck
CHECK (AmountToCollect > 0)
GO

ALTER TABLE CustomerDetails.CustomerProducts
WITH NOCHECK
ADD CONSTRAINT DF_CustProds_Renewable
DEFAULT (0) FOR Renewable
GO
```

Messages

Commands completed successfully.

Completion time: 2024-03-26T16:11:15.0439276+03:00

Рисунок 10 – Код запроса

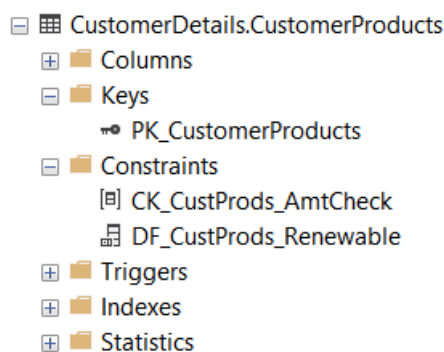


Рисунок 11 – Результат создания 3-х ограничений

Далее с помощью ОД «Проверочные ограничения» было создано еще одно ограничение. Настройки, выполненные для создания нового ограничения представлены на рисунке 12. Успешное создание нового ограничения представлено на рисунке 13.

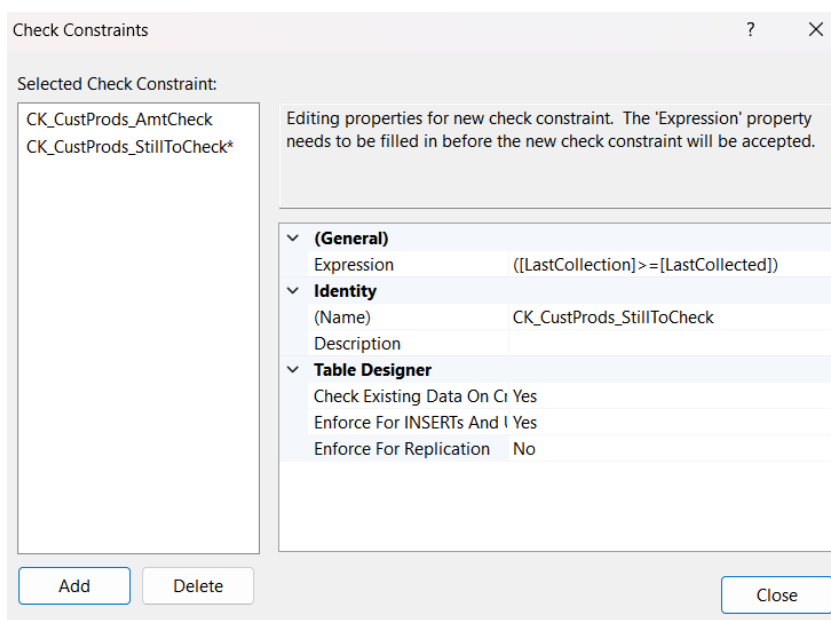


Рисунок 12 – Настройки, необходимые для создания ограничения

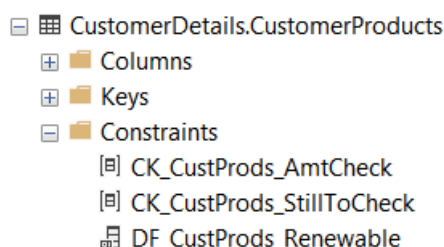


Рисунок 13 – Создание нового ограничения

Далее была выполнена проверка работоспособности ограничений. Код, позволяющий проверить работоспособность ограничений представлен на рисунке 14. По результатам выполнения запросов ограничения работают верно.

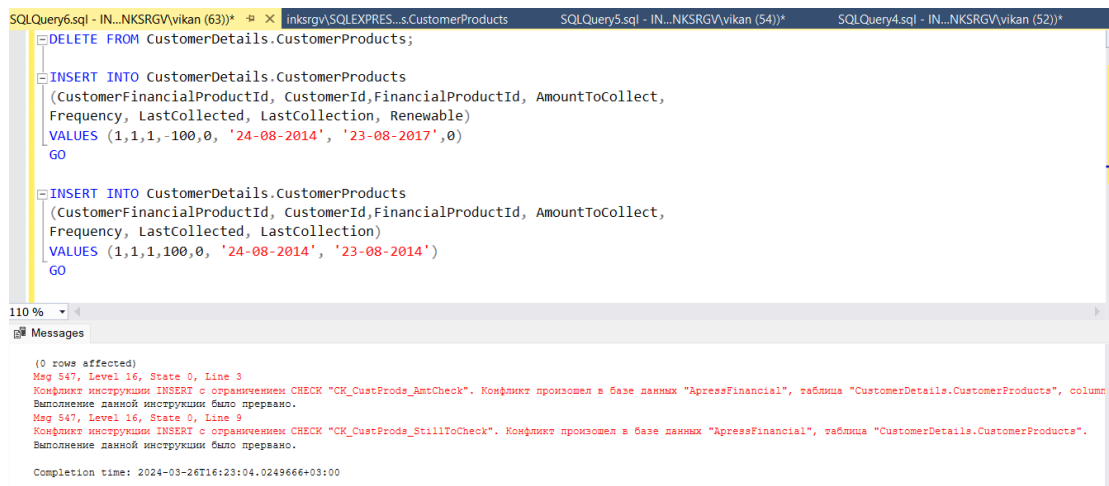


Рисунок 14 – Проверка работоспособности ограничений

### 5.1.6. Одновременная вставка нескольких записей

Код, выполняющий вставку нескольких записей представлен на рисунке 15.

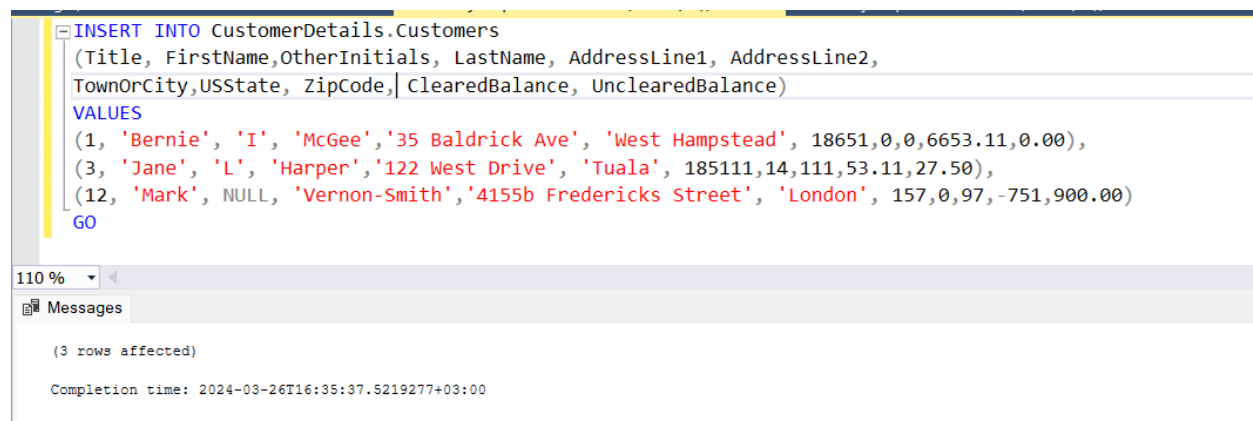


Рисунок 15 – Код запроса для вставки нескольких записей

### 5.1.7. Заполнение таблиц учебной БД тестовыми значениями

В первую очередь был выполнен скрипт, который заполняет таблицу данными. Код скрипта представлен на рисунке 16.

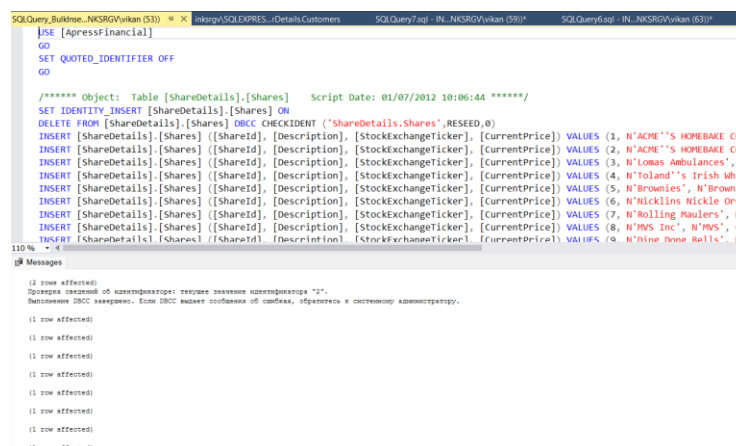


Рисунок 16 – Код скрипта, заполняющего таблицу данными

### 5.1.8. Извлечение данных: отображение результатов

Код запроса, позволяющего извлечь данные из таблицы представлен на рисунке 17.

The screenshot shows a SQL query in the editor and its results in two formats: a grid and a text view.

**Query:**

```
SELECT * FROM CustomerDetails.Customers
GO

SELECT FirstName, LastName, ClearedBalance
FROM CustomerDetails.Customers
GO

SELECT FirstName AS N'Имя',
       LastName AS N'Фамилия',
       ClearedBalance AS N'Баланс'
FROM CustomerDetails.Customers
```

**Grid Results:**

CustomerId	Title	FirstName	OtherInitials	LastName	AddressLine1	AddressLine2	AddressLine3	TownOrCity	ZipCode	USState	AccountType	ClearedBalance	UnclearedBalance
1		Noel	NULL	Morgala	87 Winnie Way	NULL	NULL	4	112	8	0	NULL	NULL
2		Aubrey	NULL	Lomas	11c Clerkenwell	NULL	NULL	2654	0	0	2	437.97	-10.94
3		Bernie	I	McGee	35 Baldrick Ave	West Hampstead	NULL	18651	0	0	NULL	6653.11	0.00
4		Jane	L	Harper	122 West Drive	Tusla	NULL	185111	111	14	NULL	53.11	27.50
5		Mark	NULL	Vernon-Smith	4155b Fredericka Street	London	NULL	157	97	0	NULL	-751.00	900.0
6		Terence	S	Madden	625 Second Blvd	18 North Old Way	NULL	2107	56835	43	1	917.0126	NULL
7		Anne	J	Aguirre	45 South First Road	64 Oak Road	436 Old Freeway	1502	32457	50	19	NULL	NULL
8		Mickey	AB	Ferguson	228 Second Street	NULL	NULL	902	15476	47	21	187.2192	NULL

**Text Results:**

Имя	Фамилия	Баланс
Noel	Morgala	NULL
Aubrey	Lomas	437.97
Bernie	McGee	6653.11
Jane	Harper	53.11
Mark	Vernon-Smith	-751.00
Terence	Madden	917.0126
Anne	Aguirre	NULL
Mickey	Ferguson	187.2192

Рисунок 17 – Код запроса, позволяющего извлечь данные из таблицы

Далее результаты вывода были переведены в текстовый формат. Последний запрос из предыдущего пункта был выполнен еще раз. Результат выполнения запроса представлен на рисунке 18.

The screenshot shows the same SQL query as in Figure 17, but the results are displayed in a text format.

**Query:**

```
SELECT FirstName AS N'Имя',
       LastName AS N'Фамилия',
       ClearedBalance AS N'Баланс'
FROM CustomerDetails.Customers
GO
```

**Text Results:**

Имя	Фамилия	Баланс
Noel	Morgala	NULL
Aubrey	Lomas	437,97
Bernie	McGee	6653,11
Jane	Harper	53,11
Mark	Vernon-Smith	-751,00
Terence	Madden	917,0126
Anne	Aguirre	NULL
Mickey	Ferguson	187,2192
Deanna	Palmer	148,0513
Anne	Mann	299,8416
Carlos	Liu	544,7332
Hector	Ramirez	287,4799
Keri	Steele	831,9561
Arturo	Guerrero	453,9708
Molly	Rivas	153,0041
Kristie	Baker	34,0155
Alfred	Bright	354,3668
Ruth	McKinney	314,6499
Luz	Drake	722,1014
Roderick	Espinoza	458,85
Donnell	Burgess	470,7855
Edwin	Stewart	16,8704
Erich	Kerr	NULL
Irma	Black	954,9562
Sophia	Schroeder	233,8611
Kisha	Strong	863,883

Рисунок 18 – Результат вывода текстовых данных



## 5.2. Обновление данных

В ходе выполнения данной задачи необходимо ознакомиться со способами обновления данных в таблицах.

5.2.1. Код запроса, который обновит данные в строке, которая имеет `CustomerId = 7` представлен на рисунке 19. Результат обновления данных представлен на рисунке 20.

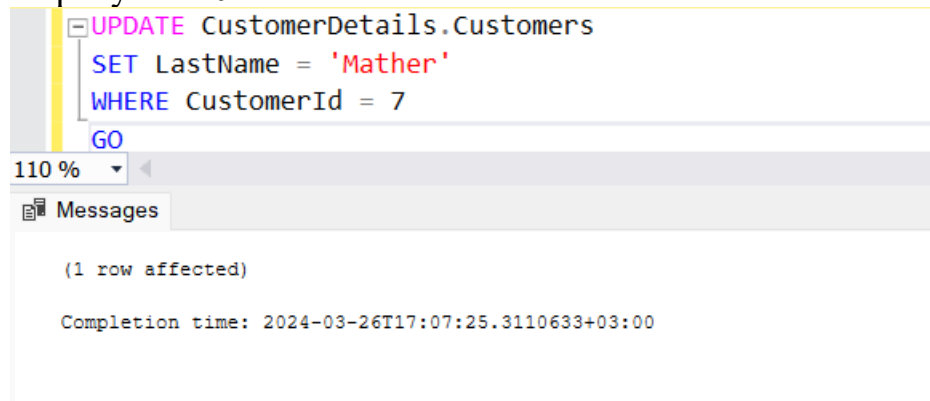


Рисунок 19 – Код запроса, обновляющего данные в таблице

	CustomerId	Title	FirstName	OtherInitials	LastName	AddressLine1	AddressLine2	AddressLine3	TownOrCity
1	1	1	Noel	NULL	Morgala	87 Winnie Way	NULL	NULL	4
2	2	2	Aubrey	NULL	Lomas	11c Clerkenwell	NULL	NULL	2654
3	3	1	Bernie	I	McGee	35 Baldrick Ave	West Hampstead	NULL	18651
4	4	3	Jane	L	Harper	122 West Drive	Tuala	NULL	185111
5	5	12	Mark	NULL	Vernon-Smith	4155b Fredericks Street	London	NULL	157
6	6	1	Terence	S	Madden	625 Second Blvd.	18 North Old Way	NULL	2107
7	7	0	Anne	J	Mather	45 South First Road	64 Oak Road	436 Old Freeway	1502

Рисунок 20 – Результат обновления значений

5.2.2. Код запроса, обновляющего данные с помощью информации из другого столбца таблицы представлен на рисунке 21.

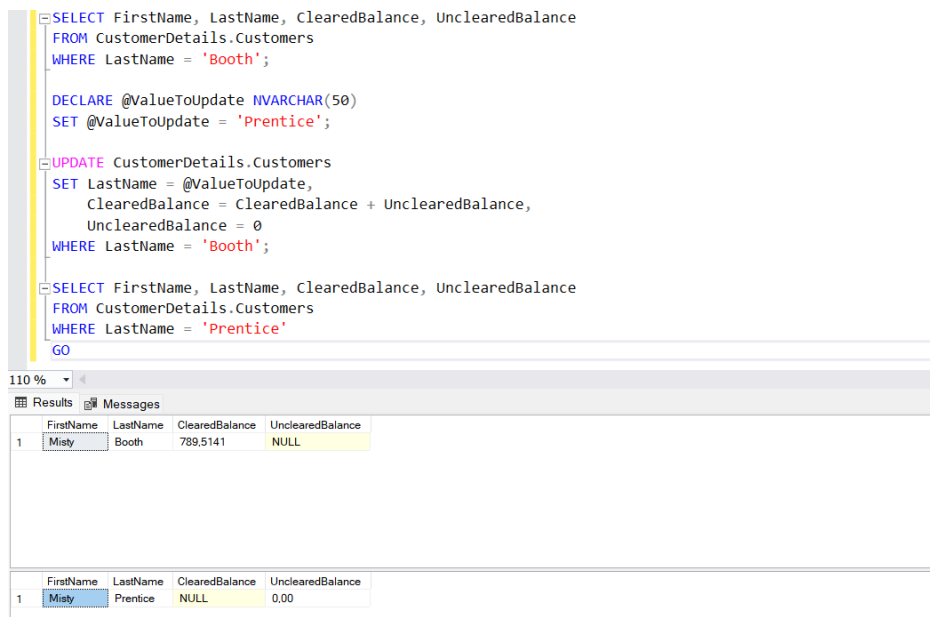


Рисунок 21 – Код запроса, обновляющего данные с помощью информации из другого столбца таблицы

5.2.3. Код запроса, обновляющего столбцы значениями из столбцов, типы данных которых не совпадают, представлен на рисунке 22.

```
SELECT FirstName, LastName, ClearedBalance, UnclearedBalance
FROM CustomerDetails.Customers
WHERE LastName = 'Prentice';

DECLARE @WrongDataType NVARCHAR(50)
SET @WrongDataType = '1422.26';

UPDATE CustomerDetails.Customers
SET ClearedBalance = @WrongDataType
WHERE LastName = 'Prentice';

SELECT FirstName, LastName, ClearedBalance, UnclearedBalance
FROM CustomerDetails.Customers
WHERE LastName = 'Prentice'
GO
```

110 %

Results Messages

	FirstName	LastName	ClearedBalance	UnclearedBalance
1	Misty	Prentice	NULL	0,00

	FirstName	LastName	ClearedBalance	UnclearedBalance
1	Misty	Prentice	1422.26	0.00

Рисунок 22 – Код запроса

5.2.4. Код запроса, который выполняет обновление столбцов значениями из других столбцов, типы данных которых не совпадают, представлен на рисунке 23. Выполнение завершится ошибкой, так как тут SQL Server не может выполнить неявное преобразование данных.

```
SELECT FirstName, LastName, ClearedBalance, UnclearedBalance
FROM CustomerDetails.Customers
WHERE lastname = 'Prentice';

DECLARE @WrongDataType NVARCHAR(50)
SET @WrongDataType = 'Invalid';

UPDATE CustomerDetails.Customers
SET ClearedBalance = @WrongDataType
WHERE LastName = 'Prentice';

SELECT FirstName, LastName, ClearedBalance, UnclearedBalance
FROM CustomerDetails.Customers
WHERE lastname = 'Prentice'
GO
```

110 %

Results Messages

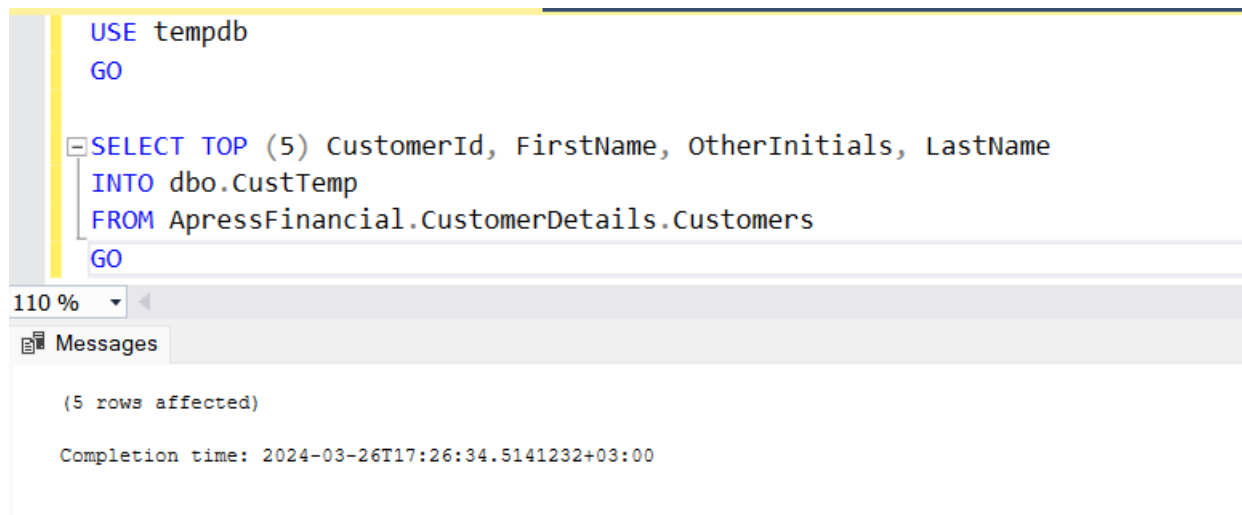
(1 row affected)  
Msg 235, Level 16, State 0, Line 8  
Не удалось преобразовать значение типа char к money. Значение типа char имеет неправильный синтаксис.  
Completion time: 2024-03-26T17:16:43.5196553+03:00

Рисунок 23 – Код запроса

### 5.3. Удаление данных

В ходе выполнения данной задачи необходимо ознакомиться с особенностями удаления значений из таблиц.

5.3.1. Код запроса, который создает новую таблицу в базе данных представлен на рисунке 24.



```
USE tempdb
GO

SELECT TOP (5) CustomerId, FirstName, OtherInitials, LastName
INTO dbo.CustTemp
FROM ApressFinancial.CustomerDetails.Customers
GO
```

110 %

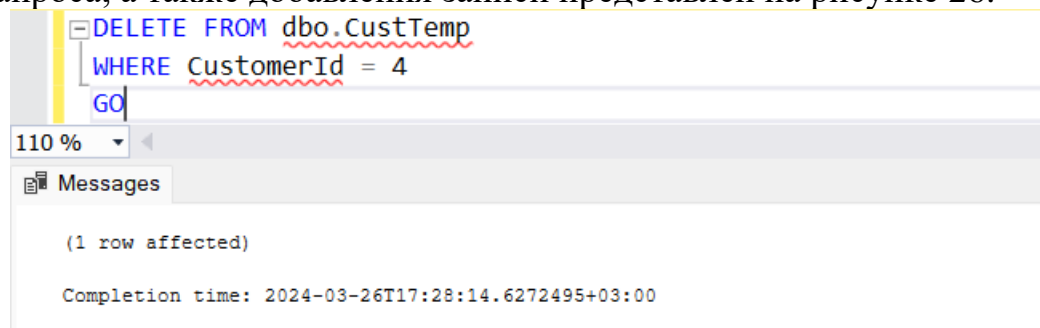
Messages

(5 rows affected)

Completion time: 2024-03-26T17:26:34.5141232+03:00

Рисунок 24 – Код запроса, создающего новую таблицу

5.3.2. Код запроса, который удаляет запись с таблицы по определенному параметру представлен на рисунке 25. Далее было выполнено добавление записи в таблицу. Ей был присвоен CustomerId = 6. Результат работы запроса, а также добавления записи представлен на рисунке 26.



```
DELETE FROM dbo.CustTemp
WHERE CustomerId = 4
GO
```

110 %

Messages

(1 row affected)

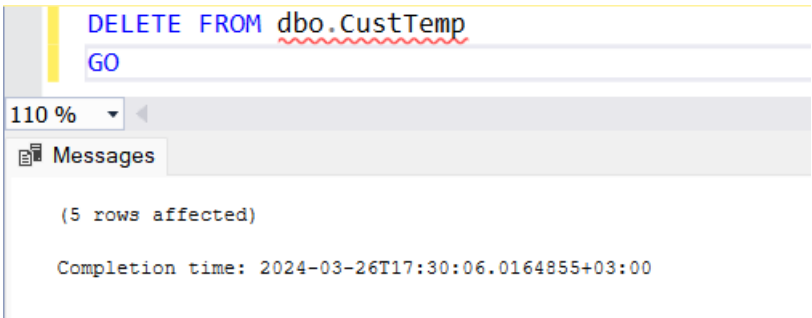
Completion time: 2024-03-26T17:28:14.6272495+03:00

Рисунок 25 – Код запроса, удаляющего запись с таблицы по определённом параметру

	CustomerId	FirstName	OtherInitials	LastName
	1	Noel	NULL	Morgala
	2	Aubrey	NULL	Lomas
	3	Bernie	I	McGee
	5	Mark	NULL	Vernon-Smith
	6	Nika	NULL	Llo

Рисунок 26 – Результат работы запроса и добавления записи

5.3.3. Код запроса, удаляющего все существующие записи из таблицы представлен на рисунке 27. Далее в таблицу было добавлено новое значение, которому был присвоен CustomerId = 7. Результат добавления значения и предшествующего ему удаления всех значений представлено на рисунке 28.



The screenshot shows a SQL query window with the following text:

```
DELETE FROM dbo.CustTemp
GO
```

Below the query, the 'Messages' pane displays:

(5 rows affected)

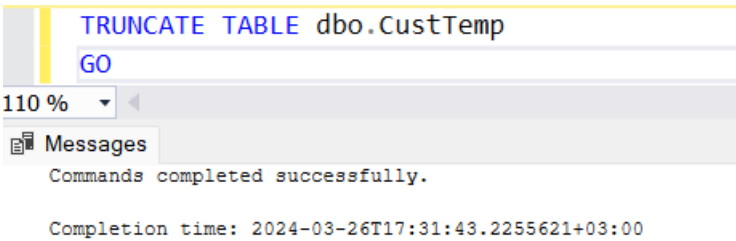
Completion time: 2024-03-26T17:30:06.0164855+03:00

Рисунок 27 – Код запроса

	CustomerId	FirstName	OtherInitials	LastName
	7	Bill	I	Gtyu
▶*	NULL	NULL	NULL	NULL

Рисунок 28 – Результат добавления нового значения и удаления всех предыдущих значений

5.3.4. Код запроса, удаляющего из таблицы все записи и создающего заново начальное значение для столбцов IDENTITY, представлен на рисунке 29. Далее было выполнено добавление нового значения в таблицу. Данной записи был присвоен CustomerId = 1, из-за использования инструкции TRUNCATE TABLE. Добавленная запись представлена на рисунке 30.



The screenshot shows a SQL query window with the following text:

```
TRUNCATE TABLE dbo.CustTemp
GO
```

Below the query, the 'Messages' pane displays:

Commands completed successfully.

Completion time: 2024-03-26T17:31:43.2255621+03:00

Рисунок 29 – Код запроса, удаляющего все значения из таблицы

	CustomerId	FirstName	OtherInitials	LastName
	1	Kkkk	U	Hhhhh
▶*	NULL	NULL	NULL	NULL

Рисунок 30 – Результат добавления записи

## **6. Выводы и анализ результатов работы**

В ходе выполнения данной лабораторной работы были получены практические и теоретические навыки по обработке данных в таблицах. Были изучены три основных операции: добавление, удаление и обновление данных. Также были изучены способы создания ограничений на добавление данных в таблицу.

Все поставленные задачи были выполнены с помощью подробной инструкции, представленной в тексте лабораторной работы, поэтому в процессе выполнения работы никаких трудностей не возникло.