**Лабораторная работа № 2. Проектирование баз данных. Нормализация**

**Вариант 13**

**Цель:** Спроектировать базу данных в соответсвии со своим вариантом

**Задание:** провести нормализацию исходной информации из таблицы, представленной ниже, создав как минимум три таблицы. При необходимости использовать дополнительные поля. Создать базу данных с помощью команд Server Management Studio, определить структуру таблиц, установить связи и заполнить таблицы информацией

**Задание:** провести нормализацию исходной информации из таблицы, представленной ниже, создав как минимум три таблицы. При необходимости использовать дополнительные поля. Создать базу данных с помощью команд **Server Management Studio**, определить структуру таблиц, установить связи и заполнить таблицы информацией.

**Условие: Интернет-магазин.** На сайте компании выставлены на продажу товары. В случае приобретения товаров на сумму свыше некоторого значения клиент получает в дальнейшем скидку на каждую покупку в размере 2%. Информационные поля: Название товара, Количество на складе, Цена, Единица измерения, Фамилия клиента, Имя, Отчество, Адрес, Телефон, E-mail, Признак скидки, Количество заказанного товара, Дата продажи.

**Теория**

**Нормализация данных** - это процесс, в результате выполнения которого таблицы базы данных проверяются на наличие зависимостей между столбцами и, если необходимо, то исходная таблица разделяется на несколько таблиц.

Существует 8 нормальных форм (**1NF – 6NF**), но первые 3 являются самыми необходимыми.

Чтобы таблица соответствовала **1-й нормальной форме** (1NF), необходимо, чтобы все значения ее полей были неделимыми и не вычисляемыми, а все записи – уникальными (не должно быть полностью совпадающих строк).

В исходном варианте будущие поля таблицы не противоречат первой нормальной форме, следовательно, находится в первой нормальной форме.

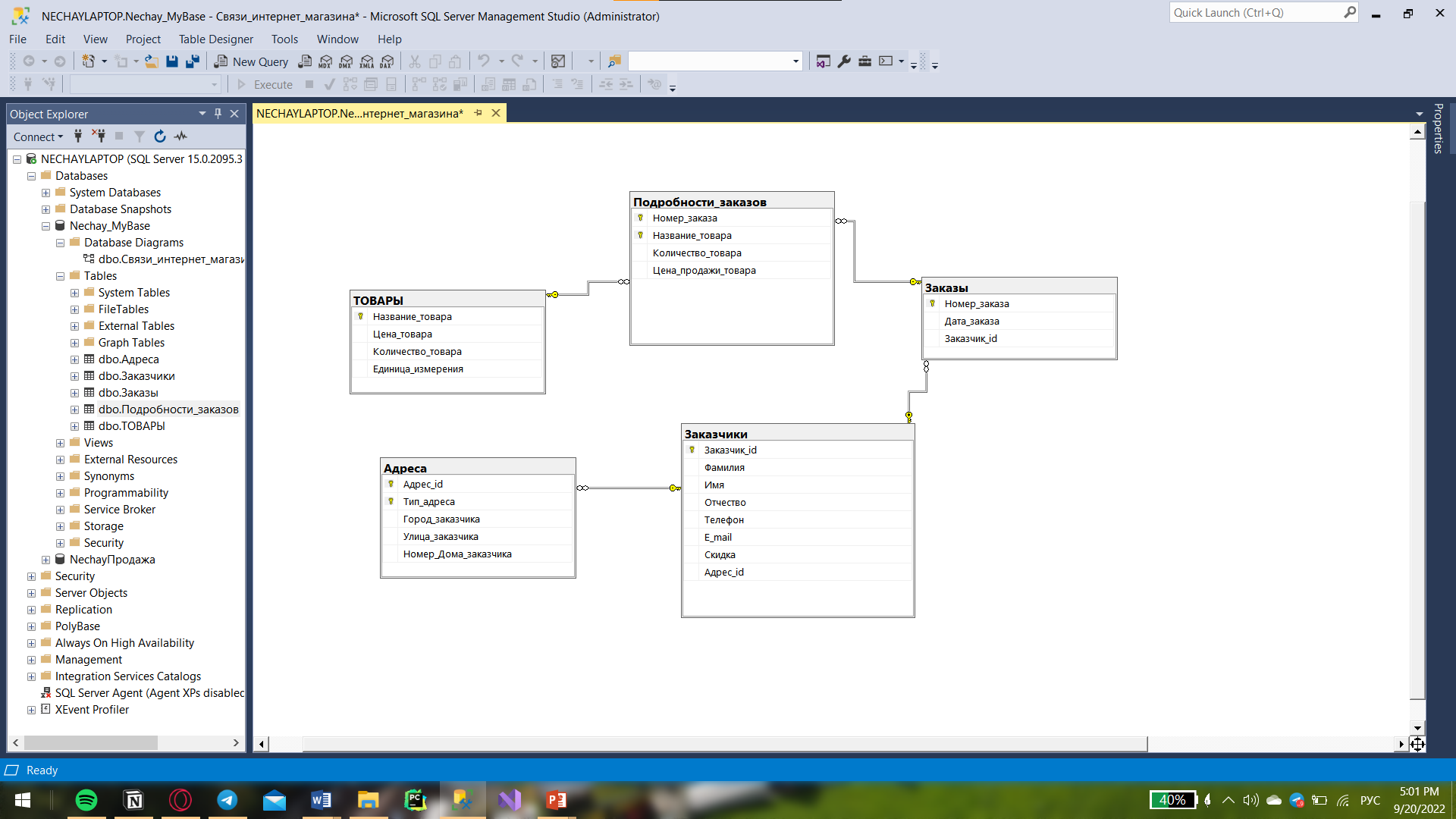
Чтобы таблица соответствовала **2-й нормальной форме** (2NF), необходимо, чтобы она находилась в 1-й нормальной форме и все не ключевые поля полностью зависели от ключевого.

Для таблицы «Заказчики» было создано поле «Заказчик\_id» для установки ему первичного ключа. Для таблицы «ТОВАРЫ» было создано поле «Название\_товара» для установки ему первичного ключа. Для таблицы «Подробности\_заказов» были созданы поля «Название\_товара» и «Номер\_заказа» для установки им составного первичного ключа. Для таблицы «Адреса» были созданы поля «Адрес\_id» и «Тип\_адреса» для установки им составного первичного ключа. Для связи с таблицей «Заказы» в неё было добавлено поле «Номер\_заказа» для связи с таблицами «Подробности\_заказов» и «Заказчики» соответственно.

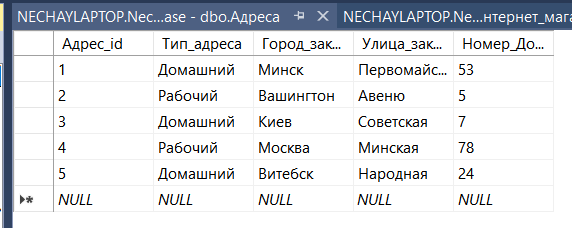
Таблица находится во второй нормальной форме.

Для перехода к **3-й нормальной форме** (3NF), необходимо обеспечить, чтобы все таблицы находились во 2-й нормальной форме и все не ключевые поля в таблицах не зависели взаимно друг от друга.

Таблица находится в третьей нормальной форме.



После этого были заполнены все поля таблиц.

Таблица «Адреса»: 

Таблицы «ТОВАРЫ»:

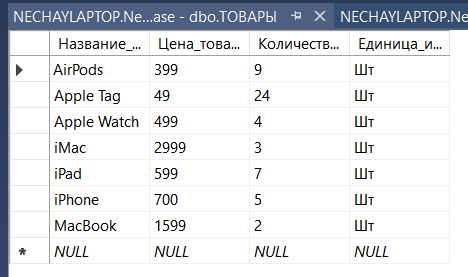


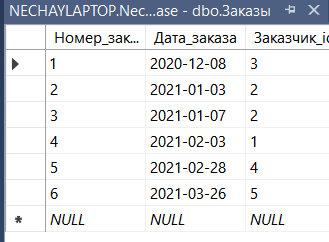
Таблица «Заказы»:  


Таблица «Заказчики»:

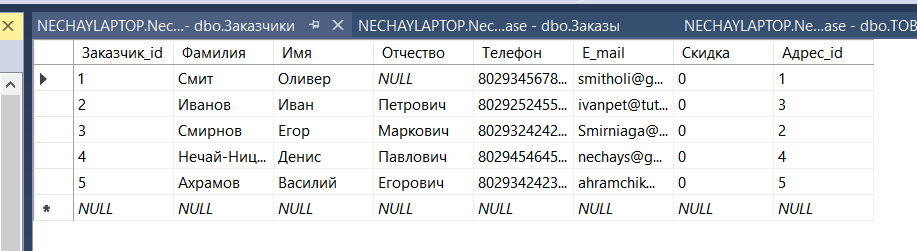
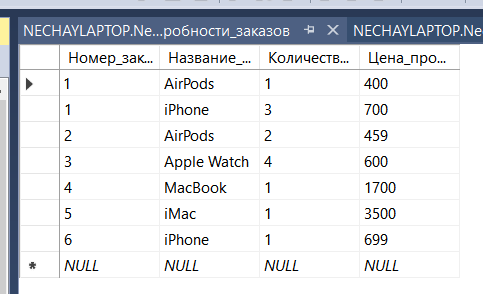


Таблица «Подробности\_заказов»



Типы данных:

- числовые (точные)

Tinyint (0-255 1 бyte)

Bit (0|1 1 byte)

Smalint (-32768 - 32768 2 byte)

Int (-2^31 - 2^31-1 4 byte)

Bigint(-2^63 – 2^63-1 8byte)

Decimal(p,s) p max= 1-38, s max = 0-p (-10^38+1) – (10^38+1) 5-17 byte

(приближенные)

Float (1-53) 4 or 8 byte

Real=float(24) (-3.4\*10^38 - -1.18\*10^-38 4byte )

- символьные

Char(n) 1-8000 n byte

Varchar(n) 1-4000 n+2 byte max=2^31-1

Nchar(n) 1-8000 2n byte

Nvarchar(n) 1-4000 2n+2 byte

Nvarchar(max) 1-(2^30-1) 2n+2 byte

- денежные

Smallmoney (-214 748,3648 – 214748, 3647 4byte)

Money (От -922 337 203 685 477,5808 до 922 337 203 685 477,5807 8 byte)

- Дата и время

Date 01.01.1753-31.12.9999 3 byte

Time(0-7) 00:00:00.000000- 23:59:59.999999 3-5byte

Smalldatetime 01.01.1900 00:00 – 06.06.2079 23:59 4byte

Datetime time+date 8 byte

Datetime2 – 6 - 8 byte

Datetimeofset(p) 8 - 10

- Двоичные

Binary(n) 1-8000 n byte

Varbinary(n) 1-8000 n+2 byte

Varbinary(max) 1-(2^31-1) n+2 byte

**Вывод:** В ходе лабораторной работы мы научились создавать базы данных, приведённые к третьей нормальной форме, используя СУБД MS SQL 2019.

А также узнали, какие и сколько сущевтвует нормальных форм и типов данных.