

Лабораторный практикум

МДК 11.01 Технология разработки
и защиты баз данных

SQL

Куропаткина О.П.

ЛАБОРАТОРНАЯ РАБОТА №4

Основы DDL

Data Definition Language (язык описания данных)

DDL и DML — подмножество языка SQL:

Язык DDL служит для **создания и модификации структуры базы данных, т.е. для создания/изменения/удаления таблиц и связей.**

Язык DML позволяет осуществлять манипуляции с данными таблиц, т.е. с ее строками. Он позволяет делать выборку данных из таблиц, добавлять новые данные в таблицы, а также обновлять и удалять существующие данные.

Продолжаем работу с базой данных CarServis.

Имеющийся функционал базы данных для компании, предоставляющей услуги по ремонту автомобилей необходимо дополнить. **Заказчик вам предоставил следующую информацию:**

Подсистема работы с товарами

Подсистема работы с товарами позволяет администратору следить за актуальным ассортиментом продукции, производить продажи, а также контролировать наличие товаров на складе. Для компании важно, чтобы характеристики товара были точно описаны, а также товары имели качественные фотографии. Качественное описание характеристик позволит делать удобный поиск и фильтрацию товаров.

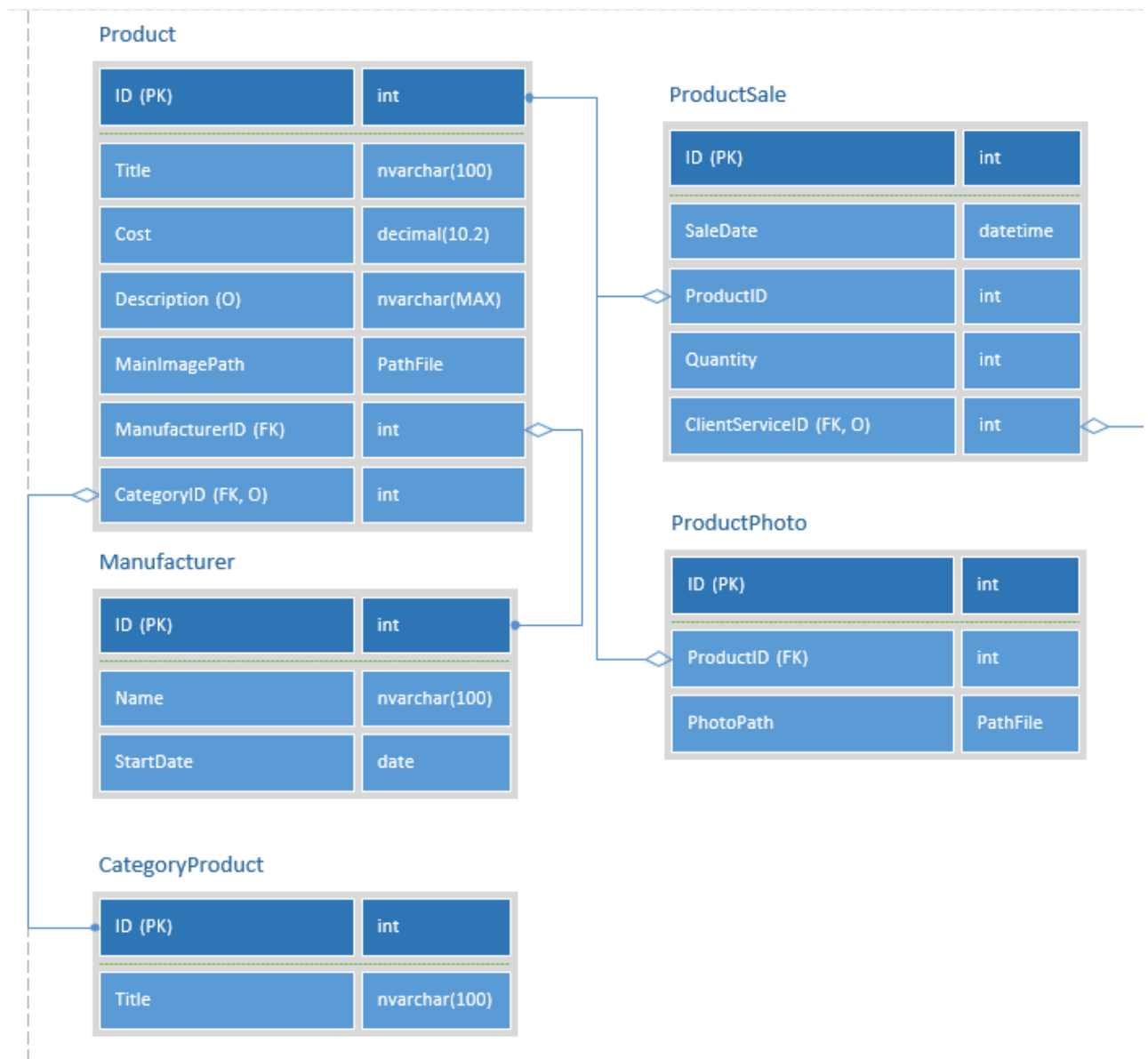
Запись о товаре содержит следующие данные: наименование, категорию, стоимость, описание, вес, размеры трех сторон (ширина, высота, длина), производитель (у каждого из которых может быть указана дата начала работы), главное изображение, а также дополнительные фотографии товара. Некоторые товары могут терять свою актуальность, поэтому их нужно помечать соответствующим флагом.

В рамках системы очень важно понимать какие товары были куплены, когда, в каком количестве, а также кем и в рамках оказания какой услуги. В связи с этим необходимо для работы необходимо хранить историю покупок.

Исходя из представленного описания нам необходимо дополнить функционал базы данных.

Реализовывать будем с применением языка DDL.

Для начала определимся со структурой таблиц.



Откройте окно создания запроса.

Для начала обратимся к своей базе данных (имя базы данных укажите свое).

```
1 USE ForEducationSQL
2 GO
```

Далее необходимо создать таблицу Производителей (Manufacturer).

Синтаксис создания любого объекта базы данных начинается с CREATE.

Пример реализации:

```
/* СОЗДАДИМ ТАБЛИЦУ, СОДЕРЖАЩУЮ СВЕДЕНИЯ О ПРОИЗВОДИТЕЛЯХ ТОВАРОВ */
```

```
CREATE TABLE Manufacturer
(
-- Ключевое слово IDENTITY создает столбец идентификации (начиная с 1, с шагом в 1)
-- Ключевое слово PRIMARY KEY задает первичный ключ
ID int IDENTITY(1,1) PRIMARY KEY NOT NULL,
[Name] nvarchar(100) NOT NULL,
StartDate date NULL
)
GO
```

Самостоятельно создайте таблицу CategoryProduct.

Далее создадим таблицу **Product**.

```
/* ТАБЛИЦА ТОВАРОВ */
```

```
CREATE TABLE Product
(
ID int IDENTITY(1,1) PRIMARY KEY NOT NULL,
Title nvarchar(100) NOT NULL,
Cost decimal(10,2) NOT NULL,
[Description] nvarchar(max) NULL,
MainImagePath PathFile NULL,
-- Ключевое слово DEFAULT присваивает значение по умолчанию
IsActive bit NOT NULL DEFAULT 'TRUE',
ManufacturerID int NULL,
CategoryID int NULL
)
GO
```

Выполните написанный код. Убедитесь, что таблицы созданы.

Они на данный момент имеют первичные ключи, но внешние ключи не указаны. Изменение объектов базы данных осуществляется при помощи инструкции ALTER.

```
-----
--                               Создание внешнего ключа
-----
-- FOREIGN KEY (внешний ключ, ограничение внешнего ключа)
-- Создание ограничения внешнего ключа на существующей таблице

ALTER TABLE Product
ADD CONSTRAINT FK_Product_Manufacturer -- Название ограничения
FOREIGN KEY (ManufacturerID) -- Внешний ключ
REFERENCES Manufacturer(ID); -- Название таблицы первичного ключа (имя поля первичного ключа)
```

Также это можно реализовать на этапе создания таблицы.

```

CREATE TABLE Product
(
    ID int IDENTITY(1,1) PRIMARY KEY NOT NULL,
    Title nvarchar(100) NOT NULL,
    Cost decimal(10,2) NOT NULL,
    [Description] nvarchar(max) NULL,
    MainImagePath PathFile NULL,
    -- Ключевое слово DEFAULT присваивает значение по умолчанию
    IsActive bit NOT NULL DEFAULT 'TRUE',
    ManufacturerID int NULL,
    CategoryID int NULL FOREIGN KEY REFERENCES CategoryProduct(ID)
)
GO

```

Справочно: если вам необходимо удалить какой-либо объект базы данных, например, таблицу, воспользуйтесь инструкцией DROP.

```

DROP TABLE Product

```

Справочно: если вам необходимо создать составной первичный ключ, то реализовать это можно только при помощи конструкции ALTER.

```

ALTER TABLE TagOfClient
ADD -- Имя ограничения не является обязательным.
PRIMARY KEY (TagID, ClientID)

```

Самостоятельно создайте таблицы ProductSale и ProductPhoto с применением DDL. Задайте первичные и внешние ключи. На поле Date таблицы ProductSale задайте значение по умолчанию – текущая дата.

Создадим ограничение на уникальность на поле Title таблицы CategoryProduct.

```

ALTER TABLE CategoryProduct
ADD CONSTRAINT CN_Title UNIQUE (Title)
GO

```

Справочно: если вам необходимо создать ограничение на ввод данных, воспользуйтесь следующей инструкцией.

```
-- Создание пользовательского ограничения на существующей таблице
ALTER TABLE Client
ADD CONSTRAINT CN_ClientPhone
-- Ограничение CHECK
CHECK (Phone LIKE '([0-9][0-9][0-9])[0-9][0-9][0-9][0-9][0-9][0-9]')
GO
```

Далее заполним вновь созданные таблицы тестовыми данными с использованием инструкции INSERT. Для начала заполним вспомогательные таблицы-справочники.

```
INSERT INTO Manufacturer -- Наименования таблицы
([Name], StartDate) -- Наименования атрибутов
VALUES
('Производитель 1', '01.01.2021'),
('Производитель 2', '02.02.2021');
```

Если порядок столбцов в инструкции INSERT совпадает с порядком столбцов в таблице, то их можно не указывать.

```
INSERT INTO Manufacturer -- Наименования таблицы
VALUES
('Производитель 1', '01.01.2021'),
('Производитель 2', '02.02.2021');
```

Самостоятельно заполните таблицы CategoryProduct, Product, ProductSale, ProductPhoto тестовыми данными (не менее 2 строк).

Постройте диаграмму базы данных, убедитесь, что все таблицы связаны, имеют первичные ключи.

Далее воспользуйтесь данными для импорта. Самостоятельно подготовьте данные и импортируйте их в таблицы: Manufacturer, Product и ProductSale.