1. **Module 05. Stored procedures, functions, views, triggers, cursors, constraints**
   1. Написать процедуру, которая возвращает самый крупный заказ для каждого из продавцов за определенный год.

В результатах не может быть несколько заказов одного продавца, должен быть только один и самый крупный.

В результатах запроса должны быть выведены следующие колонки: колонка с именем и фамилией продавца (FirstName и LastName – пример: Nancy Davolio), номер заказа и его стоимость.

В запросе надо учитывать Discount при продаже товаров.

Процедуре передается год, за который надо сделать отчет, и количество возвращаемых записей.

Результаты запроса должны быть упорядочены по убыванию суммы заказа.

Процедура должна быть реализована 2-мя способами с использованием оператора SELECT и с использованием курсора.

Название процедур соответственно GreatestOrders и GreatestOrdersCur.

Необходимо продемонстрировать использование этих процедур.

Также помимо демонстрации вызовов процедур в скрипте Query.sql надо написать отдельный ДОПОЛНИТЕЛЬНЫЙ проверочный запрос для тестирования правильности работы процедуры GreatestOrders.

Проверочный запрос должен выводить в удобном для сравнения с результатами работы процедур виде для определенного продавца для всех его заказов за определенный указанный год в результатах следующие колонки: имя продавца, номер заказа, сумму заказа.

Проверочный запрос не должен повторять запрос, написанный в процедуре, - он должен выполнять только то, что описано в требованиях по нему.  
ВСЕ ЗАПРОСЫ ПО ВЫЗОВУ ПРОЦЕДУР ДОЛЖНЫ БЫТЬ НАПИСАНЫ В ФАЙЛЕ Query.sql

* 1. Написать процедуру, которая возвращает заказы в таблице Orders, согласно указанному сроку доставки в днях (разница между OrderDate и ShippedDate).

В результатах должны быть возвращены заказы, срок которых превышает переданное значение или еще недоставленные заказы.

Значению по умолчанию для передаваемого срока 35 дней.

Название процедуры ShippedOrdersDiff.

Процедура должна высвечивать следующие колонки: OrderID, OrderDate, ShippedDate, ShippedDelay (разность в днях между ShippedDate и OrderDate), SpecifiedDelay (переданное в процедуру значение).

Необходимо продемонстрировать использование этой процедуры.

* 1. Написать функцию, которая определяет, есть ли у продавца подчиненные. Возвращает тип данных BIT. В качестве входного параметра функции используется EmployeeID. Название функции IsBoss.

Продемонстрировать использование функции для всех продавцов из таблицы Employees.

* 1. Написать запрос, который должен вывести следующие поля:
* OrderID (dbo.Orders),
* CustomerID (dbo.Orders),
* LastName + FirstName (dbo.Employees),
* OrderDate (dbo.Orders),
* RequiredDate (dbo.Orders),
* ProductName (dbo.Products),
* цену товара с учетом скидки.

Результат запроса необходимо представить в виде представления.

Отсортировать по цене товара.

* 1. Создать таблицу dbo.OrdersHistory, которая будет хранить историю изменений по таблице dbo.Orders.

Необходимо подумать какие бы поля могла бы содержать данная таблица.

Обосновать свой выбор. Почему именно такой набор полей должен быть в таблице dbo.OrdersHistory?

Затем для таблицы dbo.Orders необходимо создать триггер, который при любом изменении данных в таблице dbo.Orders будет записывать значения в новую таблицу dbo.OrdersHistory.

Написать проверочный запрос, который будет изменять/удалять данные из таблицы dbo.Orders.