**Практическая работа № 10. Разработка функций.**

**Программное обеспечение: Windows 8 и выше, Microsoft Office Word 2016 и выше, Microsoft SQL Server 2017 и выше, Microsoft SQL Server Management Studio 2017.**

**Цель работы:**

**На основании разработанной структуры базы данных, реализовать статистические функции.**

**Время на выполнение практической работы:** 2 часа.

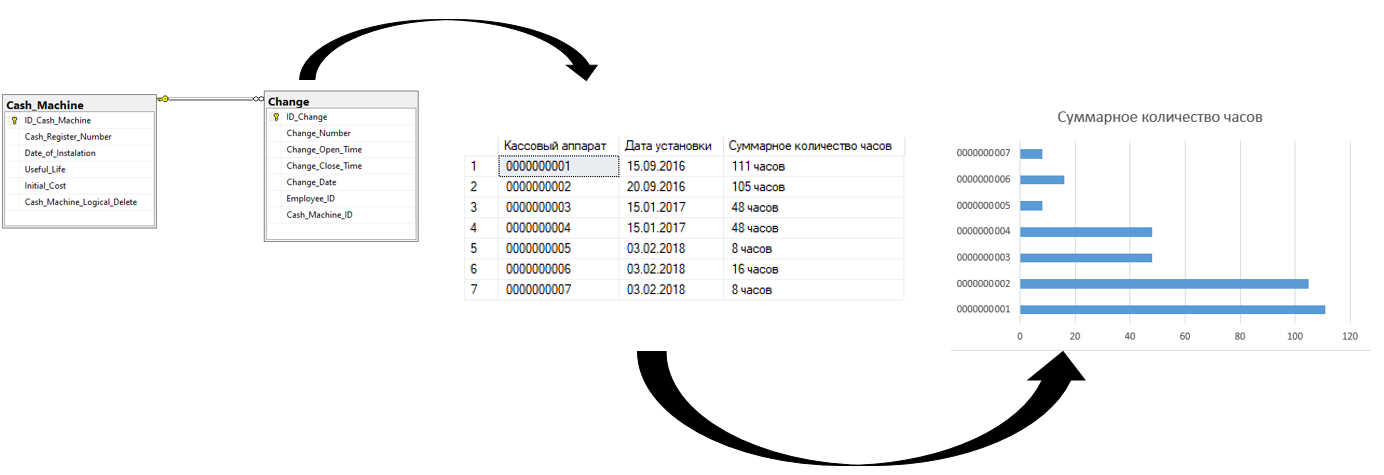
**Получение навыков:**

* **В работе с агрегатными, встроенными, функциями языка SQL;**
* **В работе со встроенными функциями, даты и времени, языка SQL;**
* **В работе со строками и их форматированием, с помощью встроеных функция языка SQL;**
* **В разработке пользовательских агрегатных и таблицных функций.**

**Теоретические сведения к выполнению практической работы:**

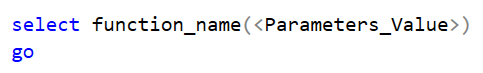
Статистика.

**Статистика** – процесс сбора, анализа и обработки больших объёмов данных, с дальнейшим выводом в отчётный графический вид.



Принципы вызова функций.

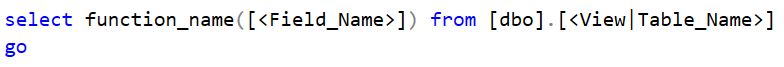
1. Вызов системной и прочей функции без привязке к структуре БД:



1. Вызов табличной функции:



1. Вызов функции с привязкой к структуре БД:



1. Вложенность функций:

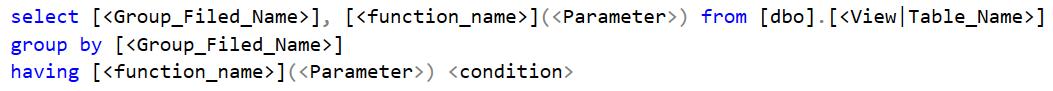


Агрегатные функции.

**Group by** – ключевое слово позволяет, выполнять агрегатную функцию столько раз, сколько указано в группирующем столбце.

**Having** – ключевое слово, задаёт любой функции условия её вывода информации.

Конструкция.



**Count** – функция возвращает количество строк из результирующих таблиц.

Конструкция:



**Sum** – функция возвращает суммарное значение, числовых столбцов.

Конструкция:



**Avg** – функция возвращает среднеарифметическое значение, числовых столбцов.

Конструкция:



**Max, Min** – функции возвращают максимальное (минимальное) значение, столбцов.

Конструкции:



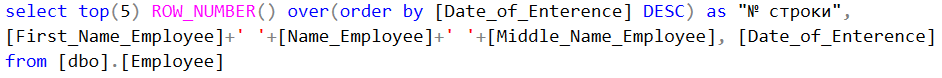
**Top** – функция возвращает определённое количество записей в результате запроса.

**Row\_Number** – оконная функция, позволяет выводить в отдельном столбце нумерацию строк. ***Более подробно о конструкциях оконных функций, можете ознакомиться на официальном сайте Microsoft.***

Конструкции:



Пример:



**Distinct** – возвращает из указанного столбца, только уникальные записи, сводя все повторения к единичному экземпляру.

Конструкция:



**Round** – округляет значение, вещественного поля, с указанным количеством округляемых разрядов после запятой, в качестве второго формального параметра.

Конструкция:



Строковые функции.

**String\_Agg** – функция объединяет в одну ячейку, несколько выбранных строк, где в качестве второго формального параметра выступает разделитель.

Конструкция:



**Len** – функция возвращает, количество символов из указанной строки.

Конструкция:



**Concat** – позволяет конкатенировать строковые значения и поля в один результирующий столбец.

Конструкция:



**Lower, upper** – возвращают текстовое значение в нижнем (верхнем) регистре.

Конструкции:



**Substring** – возвращает строку из подстроки, где второй формальный параметр, начало подстроки, третий формальный параметр длинна подстроки.

Конструкция:



Функции даты и времени.

**Date Part Value** – коллекция значений, для частей даты и времени.

|  |  |
| --- | --- |
| **Значение** | **Логическое значение** |
| YEAR | Год |
| QUARTER | Квартал |
| MONTH | Месяц |
| DAYOFYEAR | День года |
| DAY | День |
| WEEK | Номер недели |
| WEEKDAY | День недели |
| HOUR | Часы |
| MINUTE | Минуты |
| SECOND | Секунды |
| MILLISCOND | Миллисекунды |
| MICROSECOND | Микросекунды |
| NANOSECOND | Наносекунды |

**Format** – возвращает отформатированное поле, по указанному паттерну, в качестве второго формального параметра.

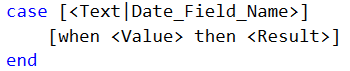
**DateAdd** – возвращает дату, к которой присвоена часть даты в качестве первого формального параметра и значение в качестве второго формального параметра..

Конструкции:

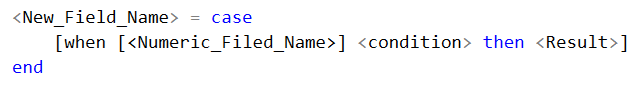


**Case** – условный оператор, позволяет на основании полученного значения поля, получить новые данные в результате запроса.

Конструкция для текстовых, даты и времени значений:



Конструкция для числовых значений:



**DatePart** – возвращает часть даты или времени.

Конструкция:

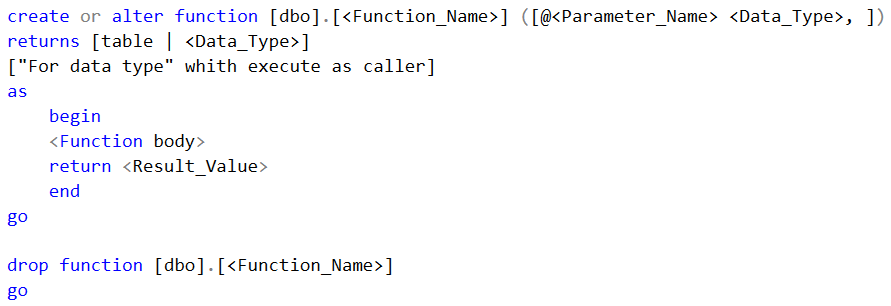


**DateDiff** – возвращает разницу в между датами, где первый формальный параметр – возвращаемая часть даты, второй формальный параметр старшая дата, а третий параметр – младшая дата.

Конструкция:



Создание объекта – функция.



**Этапы выполнения:**

*Для выполнения данной работы, создайте новый файл в Microsoft SQL Server Management Studio, для работы со сценариями функций.*

1. Описание и реализация статистических задач;

|  |  |  |  |
| --- | --- | --- | --- |
| **№ Задачи** | **I** | **Задача:** | Вывести количество поставляемой продукции, каждого производителя. |
| **Скрипт** | | | select [Name\_Manufacturer] as "Производитель", STRING\_AGG([Product\_Name], ', ') as "Cписок постовляемой продукции", count(Manufacturer\_ID) as "Количество поставляемой продукции" from [dbo].[Product]  inner join [dbo].[Manufacturer] on [ID\_Manufacturer] = [Manufacturer\_ID]  group by [Name\_Manufacturer]  go |
| **Результат SSMS** | | |  |
| **График MS Excel** | | |  |
| **№ Задачи** | **II** | **Задача:** | Вывести 3 первые позиции, максимально часто, проданного товара. |
| **Скрипт** | | | select top(3) [Product\_Name] as "Самый продоваемый продукт", sum([Sale\_Check\_View\_Ammount]) as "Количество проданных позиций" from [dbo].[Product]  inner join [dbo].[Sale\_Check\_View] on [Product\_ID] = [ID\_Product]  group by [Product\_Name]  having sum([Sale\_Check\_View\_Ammount]) > 1  go |
| **Результат SSMS** | | |  |
| **График MS Excel** | | |  |
| **№ Задачи** | **III** | **Задача:** | Вывести информацию о держателе карты, и об общей сумме затраченной на покупки. |
| **Скрипт** | | | select Concat('Номер карты: ',[Card\_Number], ', держатель карты: ', [Card\_Holder], 'ФИО Покупателя ', [First\_Name\_Buyer],' ',SUBSTRING([Second\_Name\_Buyer],1,1),'.',SUBSTRING([Middle\_Name\_Buyer],1,1),'.') as "Данные покупателя", Convert([decimal](38,2), Round(avg([Whith\_VAT]),2)) as "Средняя цена чека" from [dbo].[Buyer]  inner join [dbo].[Buyer\_Card] on [ID\_Buyer] = [Buyer\_ID]  inner join [dbo].[Sale\_Check] on [ID\_Buyer\_Card] = [Buyer\_Card\_ID]  group by Concat('Номер карты: ',[Card\_Number], ', держатель карты: ', [Card\_Holder], 'ФИО Покупателя ', [First\_Name\_Buyer],' ',SUBSTRING([Second\_Name\_Buyer],1,1),'.',SUBSTRING([Middle\_Name\_Buyer],1,1),'.')  go |
| **Результат SSMS** | | |  |
| **График MS Excel** | | | Не возможно представит в графическом виде |
| **№ Задачи** | **IV** | **Задача:** | Вывести название продукции, и вместо числового количества, вывести троковое обозначение количества. |
| **Скрипт** | | | select [Product\_Name] as "Название продукта", "Степень дороговизны" = case  when [Price\_Product] > 100 then 'Дешёвый товар'  when [Price\_Product] between 101 and 1000 then 'Доступный товар'  when [Price\_Product] <1001 then 'Дорогой товар'  end from [dbo].[Product]  go |
| **Результат SSMS** | | |  |
| **График MS Excel** | | | Не возможно представит в графическом виде |
| **№ Задачи** | **V** | **Задача:** | Вывести самый дешевый и самый догорой товар. |
| **Скрипт** | | | select Max([Price\_Product]) as "Самый дорогой товар", MIN([Price\_Product]) as "Самый дешёвый товар" from [dbo].[Product]  go |
| **Результат SSMS** | | |  |
| **График MS Excel** | | | Не возможно представит в графическом виде |
| **№ Задачи** | **VI** | **Задача:** | Вывести информацию, о договорах с поставщиками, датой заключения договора, датой окончания договора, сколько ещё будет действовать договор и квартал окончания. |
| **Скрипт** | | | select LOWER([Name\_Supplier]) as "Вид Юр. Лица", UPPER([Full\_Name\_Suppler]) as "Название огранизации", FORMAT([Date\_Create\_Term],'dd.MM.yyyy') as "Дата заключения договора" , FORMAT(DATEADD(YEAR,CONVERT([int], SUBSTRING([Term], 0,3)),[Date\_Create\_Term]),'dd.MM.yyyy') as "Дата окончания договора" , DATEDIFF(YEAR,GETDATE(),DATEADD(YEAR,/,[Date\_Create\_Term])) as "Продолжительность действия договора в годах" , DATEPART(QUARTER,DATEADD(YEAR,CONVERT([int], SUBSTRING([Term], 0,3)),[Date\_Create\_Term])) as "Квартал окончания договора" from [dbo].[Treaty]  inner join [dbo].[Representative] on [ID\_Representative] = [Representative\_ID]  inner join [dbo].[Suppler] on [ID\_Suppler] = [Suppler\_ID]  inner join [dbo].[Type\_Supplier] on [ID\_Type\_Supplier] = [Type\_Suppler\_ID]  go |
| **Результат SSMS** | | |  |
| **График MS Excel** | | | Не возможно представит в графическом виде |
| **№ Задачи** | **VII** | **Задача:** | Вывести уникальные даты заключения договора о поставке. |
| **Скрипт** | | | select distinct(Date\_Create\_Supply) from [dbo].[Supply]  go |
| **Результат SSMS** | | |  |
| **График MS Excel** | | | Не возможно представит в графическом виде |
| **№ Задачи** | **VIII** | **Задача:** | Вывести нуперованный список всех фискальных чеков. |
| **Скрипт** | | | select ROW\_NUMBER() over (order by [Date\_Create\_Check]) as "№ строки", [Sale\_Check\_Number] as "№ чека" from [dbo].[Product]  inner join [dbo].[Sale\_Check\_View] on [Product\_ID] = [ID\_Product]  inner join [dbo].[Sale\_Check] on [ID\_Sale\_Check] = [Sale\_Check\_ID]  go |
| **Результат SSMS** | | |  |
| **График MS Excel** | | |  |
| **№ Задачи** | **IX** | **Задача:** | Разработать функцию, подсчитывающая суммарное значение потраченных денег на закупки. |
| **Скрипт** | | | create or alter function [dbo].[Buyer\_Card\_Money](@Card\_Number [varchar] (19))  returns [decimal] (38,2)  with execute as caller  as  begin  return(select sum([Without\_VAT]) from [dbo].[Sale\_Check]  inner join [dbo].[Buyer\_Card] on [Buyer\_Card\_ID] = [ID\_Buyer\_Card]  where [Card\_Number] = @Card\_Number)  end  go  select [dbo].[Buyer\_Card\_Money]('5542 5674 9974 8974')  go |
| **Результат SSMS** | | |  |
| **График MS Excel** | | | Не возможно представит в графическом виде |
| **№ Задачи** | **X** | **Задача:** | Реазработать функцию, с выводом информаци о номере конкретного чека, составе чека, внесённой суммы, итоговой цены и сдачи за чек. |
| **Скрипт** | | | create or alter function [dbo].[Sale\_Check\_History](@Sale\_Check\_Number [varchar] (13))  returns table  as  return(select [Date\_Create\_Check] as "№ Чека",  STRING\_AGG('Позиция: '+[Product\_Name]+', кол-во: '+CONVERT([varchar](4),[Sale\_Check\_View\_Ammount])+', цена за единицу: '+CONVERT([varchar](41),[Price\_Product])+' руб, общая цена: '+CONVERT([varchar] (max), [Sale\_Check\_View\_Ammount]\*[Price\_Product]),', ') as "Детализация чека",  [Deposited\_Ammount] as "Всего внесено", [Whith\_VAT] as "Итоговая стоимость", [Deposited\_Ammount] - [Whith\_VAT] as "Сдача" from [dbo].[Sale\_Check]  inner join [dbo].[Sale\_Check\_View] on [Sale\_Check\_ID] = [ID\_Sale\_Check]  inner join [dbo].[Product] on [Product\_ID] = [ID\_Product]  where [Sale\_Check\_Number] = @Sale\_Check\_Number  group by [Date\_Create\_Check], [Deposited\_Ammount], [Whith\_VAT])  go  select \* from [dbo].[Sale\_Check\_History]('0000000001-ФК') |
| **Результат SSMS** | | |  |
| **График MS Excel** | | | Не возможно представит в графическом виде |

1. Отчёт об используемых функциях.

| Название функции и ключевых слов | № задач(и) |
| --- | --- |
| Group by | I, II, III, X |
| Having | II |
| Count | I |
| Sum | II, IX |
| Avg | III |
| Max | V |
| Min | V |
| Top | II |
| Row\_Number | VII |
| Disctinct | VII |
| Round | III |
| String\_Agg | I, X |
| Concat | III |
| Lower | VI |
| Upper | VI |
| Substring | III |
| Format | VI |
| DateAdd | VI |
| Case | IV |
| DatePart | VI |
| DateDiff | VI |
| Пользовательская агрегатная функция | IX |
| Пользовательская табличная функция | X |

1. Скриншот обозревателя объектов

