**МИНОБР НАУКИРОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра САПР**

отчет

**по лабораторной работе №2**

**по дисциплине «Базы Данных»**

**Группировка и агрегирование данных**

|  |  |  |
| --- | --- | --- |
| Студенты гр. 7301 |  | Бурков М.П. |
|  |  | Гарцев Е.А. |
|  |  | Литвинов К.Л. |

|  |  |  |
| --- | --- | --- |
| Преподаватель: |  | Каримов А.И. |

Санкт-Петербург

2019

*Цель работы:* Знакомство с опциями GROUP BY и HAVING, а также агрегированием данных.

*Используемая база данных:* AdventureWorks

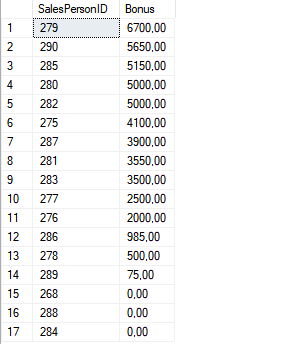
**Упражнение 1 – использование ключевого слова TOP в команде SELECT.**

Запрос 1:

SELECT SalesPersonID, Bonus FROM AdventureWorks.Sales.SalesPerson

ORDER BY Bonus DESC;

Результаты выполнения запроса

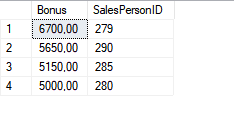


Запрос возвращает записи полей SalesPersonID и Bonus из таблицы SalesPerson, отсортированные по убыванию по полю Bonus.

Запрос 2. Модификация предыдущего запроса:

SELECT TOP(4) Bonus, SalesPersonID FROM AdventureWorks.Sales.SalesPerson ORDER BY Bonus DESC;

Результат выполенения запроса

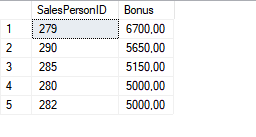


Запрос возвращает только 4 записи о значениях самых больших премий (бонусов).

Запрос 3. Модификация предыдущего запроса:

SELECT TOP(4) WITH TIES SalesPersonID, Bonus FROM AdventureWorks.Sales.SalesPerson ORDER BY Bonus DESC;

Результат выполенения запроса



Запрос возвращает вместе со строками из предыдущего запроса ещё данные по тем продавцам, которые имеют то же значение, что и последнее значение, полученное в предыдущем задании.

**Упражнение 2 – использование агрегатных функция и конструкций GROUP BY и HAVING.**

Запрос 1. Использование агрегатных функций:

SELECT COUNT(\*) FROM AdventureWorks.HumanResources.Employee;

SELECT COUNT(ManagerID) FROM AdventureWorks.HumanResources.Employee;

Результат выполнения запроса



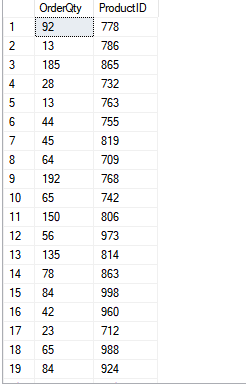


Запрос возвращает общее количество строк в таблице Empoyee схемы HumanResources и общее количество менеджеров (поле ManagerID).

Запрос 2. Использование опции GROUP BY:

SELECT SUM(OrderQty) AS OrderQty,ProductID FROM AdventureWorks.Sales.SalesOrderDetail GROUP BY OrderQty,ProductID;

Результат выполнения запроса



(Обработано строк: 2613)

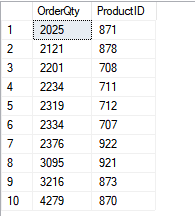
Возвращает суммарное количество заказанного товара (поле OrderQty) для каждого продукта (поле ProductID), отсортированное по полю OrderQty.

Запрос 2.1. Модификация предыдущего запроса.

SELECT SUM(OrderQty) AS OrderQty, ProductID FROM AdventureWorks.Sales.SalesOrderDetail

GROUP BY OrderQty,ProductID HAVING SUM(OrderQty)>=2000 ORDER BY OrderQty;

Результат выполнения запроса



Запрос возвращает только те товары, суммарное значение заказов по которым не менее 2000.

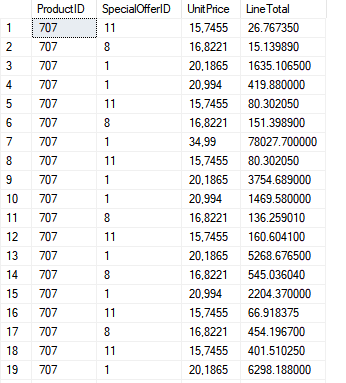
Запрос 3. использование предложения GROUP BY для формирования нескольких групп:

SELECT ProductID, SpecialOfferID, AVG(UnitPrice) AS UnitPrice, SUM(LineTotal) AS LineTotal FROM AdventureWorks.Sales.SalesOrderDetail

GROUP BY LineTotal, ProductID, SpecialOfferID, UnitPrice

ORDER BY ProductID;

Результат выполнения запроса



(Обработано строк: 3770)

Запрос возвращает поля ProductID, SpecialOfferID, среднее значение цены за единицу товара (поле UnitPrice) и суммарное значение по полю LineTotal. Полученный результат отсортирован по полю ProductID по возрастанию.

**Упражнение 3 – использование операторов ROLLUP и CUBE:**

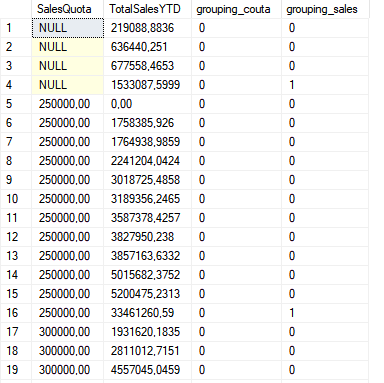
Запрос 1. Использование оператора ROLLUP для создания сводных результатов:

SELECT SalesQuota, SUM(SalesYTD) AS TotalSalesYTD,

GROUPING(SalesQuota) AS grouping\_couta, GROUPING(SalesYTD) AS grouping\_sales FROM AdventureWorks.Sales.SalesPerson

GROUP BY ROLLUP (SalesQuota,SalesYTD);

Результат выполнения запроса



(Обработано строк: 21)

Запрос возвращает поле SalesQuota, суммарное значение по полю SalesYTD, а так же сводный результат по полученной выборке.

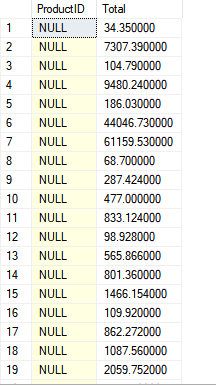
Запрос 2. Использование оператора CUBE для создания сводных результатов:

SELECT ProductID, SUM(LineTotal) AS Total FROM AdventureWorks.Sales.SalesOrderDetail

WHERE UnitPrice < 5.00 GROUP BY CUBE(ProductID, LineTotal, OrderQty)

ORDER BY ProductID;

Результат выполнения запроса



(Обработано строк: 439)

Запрос возвращает поле ProductID и сумму по полю LineTotal, причём выводятся только те значения, для которых UnitPrice < $5.00, а все значения отсортированы по полю ProductID, а вместе с этим выводятся сводные значения по полученной выборке.

**Упражнение 4 – использование предложений COMPUTE и COMPUTE BY в команде SELECT для создания отчетов:**

В MS SQL функция COMPUTE перестала поддерживаться с 2012 года, поэтому выполнить данное упражнение не представляется возможным.

**Вывод:** В ходе лабораторной работы мы ознакомились со способами группировки и агрегирования данных, в частности с операциями GROUP BY, HAVING и другими.