

Министерство образования Республики Беларусь
Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ
Факультет информационных технологий и управления
Кафедра информационных технологий автоматизированных систем

Отчет по лабораторной работе № 6
«Основы MS SQL Server»
дисциплины «Банки и базы данных»

Выполнили:
ст. группы 120602

П.С. Анашкевич
Р.И. Будный
А.А. Мельник

Проверил:
ассистент

А.Ф.Трофимович

Минск 2013

1 Цель работы

Изучение основ MS SQL Server.

2 Задание

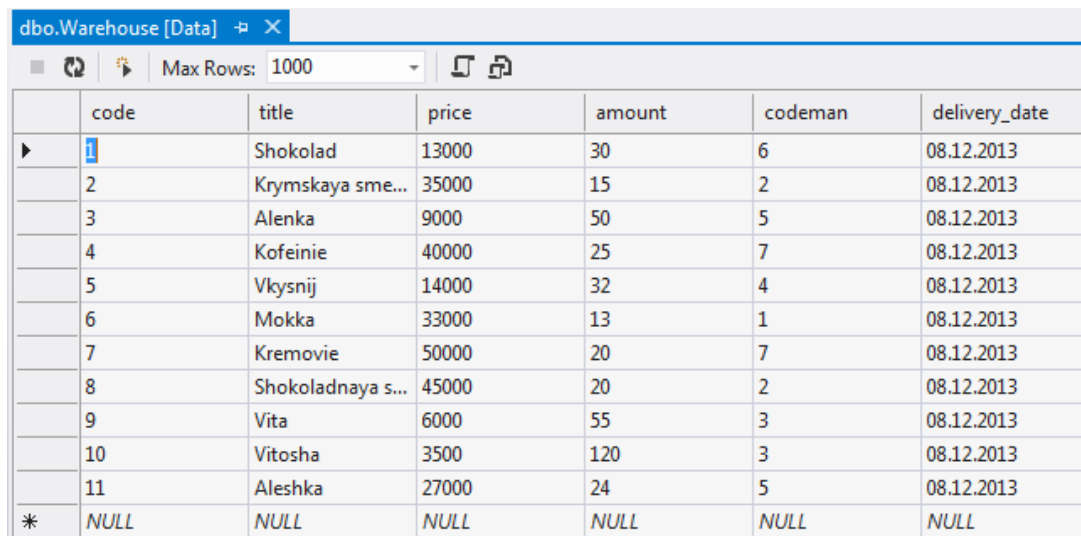
Написать хранимые процедуры для следующих видов запросов для таблицы Warehouse со структурой, определённой в лабораторной работе №1:

- определение цены товара по переданному в процедуру названию товара;
- определение числа товаров, цена которых не превосходит указанную в качестве параметра вызова процедуры;
- вывести название и цену товара, причем название товара упорядочить по алфавиту;
- определить самый дорогой товар;
- вычислить налоги на выручку. Для вычисления налога использовать функцию ПФ (налог считать так: если выручка от продажи больше 10000, то налог равен 20%, иначе — 13% от выручки);
- написать процедуру, возвращающую название всех товаров, выручка от реализации которых превосходит указанное значение передаваемого в процедуру параметра;
- написать процедуру, изменяющую структуру таблицы Warehouse путём добавления нового столбца, соответствующего дате поставки. В качестве даты использовать текущую дату для всех записей;
- написать процедуру, изменяющую цену указанного товара на новую цену, также устанавливаемую в качестве параметра;
- вывести название фирмы, производящей самый дешевый товар;
- продемонстрировать использование курсора для подсчета суммарной выручки от реализации всех товаров.

3 Ход работы

Пусть имеется база данных, содержащая:

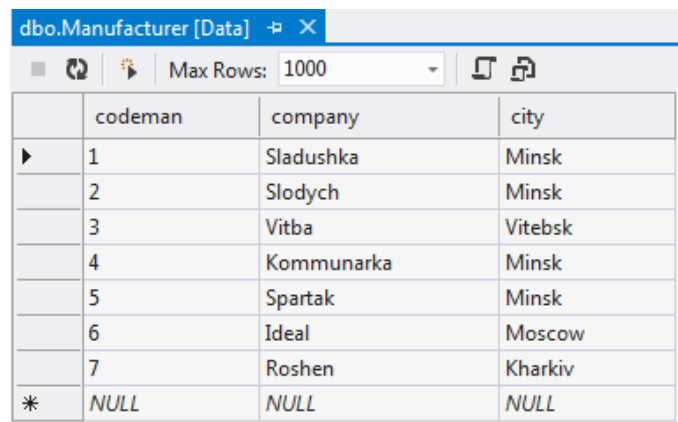
- таблицу Warehouse с полями code (Код товара), codeman (Код производителя), title (Название товара), price (Цена товара), amount (Количество товара).



	code	title	price	amount	codeman	delivery_date
▶	1	Shokolad	13000	30	6	08.12.2013
	2	Krymskaya sme...	35000	15	2	08.12.2013
	3	Alenka	9000	50	5	08.12.2013
	4	Kofeinie	40000	25	7	08.12.2013
	5	Vkysnij	14000	32	4	08.12.2013
	6	Mokka	33000	13	1	08.12.2013
	7	Kremovie	50000	20	7	08.12.2013
	8	Shokoladnaya s...	45000	20	2	08.12.2013
	9	Vita	6000	55	3	08.12.2013
	10	Vitoshka	3500	120	3	08.12.2013
	11	Aleshka	27000	24	5	08.12.2013
*	NULL	NULL	NULL	NULL	NULL	NULL

Рисунок 3.1 — Таблица Warehouse

- таблицу Manufacturer с полями codeman (Код производителя), company (Название компании), city (расположение компании).



	codeman	company	city
▶	1	Sladushka	Minsk
	2	Slodych	Minsk
	3	Vitba	Vitebsk
	4	Kommunarka	Minsk
	5	Spartak	Minsk
	6	Ideal	Moscow
	7	Roshen	Kharkiv
*	NULL	NULL	NULL

Рисунок 3.2 — Таблица Manufacturer

Опишем хранимые процедуры, используемые в лабораторной работе.

```
1 CREATE PROCEDURE [dbo].[1_procedure]
2 @curTitle varchar(20)
3 AS
4 declare price_cursor cursor scroll for
5     SELECT price from Warehouse where title=@curTitle for read only
6 open price_cursor
7 declare @curPrice int, @n int
8 set @n = 0
9 while (@n < @@CURSOR_ROWS)
10 begin
11     set @n = @n + 1
12     fetch next from price_cursor into @curPrice
13     print 'Price of selected title: ' + @curTitle + ' = ' + str(@curPrice)
14 end
15 close price_cursor
16 deallocate price_cursor
17 RETURN 0
```

Рисунок 3.1 — Определение цены товара по переданному в процедуру названию товара.

```
1 CREATE PROCEDURE [dbo].[2_procedure]
2 @price int
3 AS
4 declare @count int
5 select @count = COUNT(title) from Warehouse
6     where price <= @price
7 print 'Number =' + str(@count)
8 RETURN 0
```

Рисунок 3.2 — Определение числа товаров, цена которых не превосходит указанную в качестве параметра вызова процедуры.

```
1 CREATE PROCEDURE [dbo].[3_procedure]
2 AS
3 declare temp_cursor cursor scroll for
4     select title, price from Warehouse
5     order by title asc for read only
6 open temp_cursor
7 declare @n int, @title varchar(20), @price int
8 set @n = 0
9 while (@n < @@CURSOR_ROWS)
10 begin
11     set @n = @n + 1
12     fetch next from temp_cursor into @title, @price
13     print @title + str(@price)
14 end
15 close temp_cursor
16 deallocate temp_cursor
17 RETURN 0
```

Рисунок 3.3 — Вывести название и цену товара, причем название товара упорядочить по алфавиту.

```

1 CREATE PROCEDURE [dbo].[4_procedure]
2 AS
3     declare @price int, @title varchar(20)
4     select @title = title, @price = price from Warehouse
5         where price = (select max(price) from Warehouse)
6     print 'The most expensive item: ' + @title + str(@price)
7 RETURN 0

```

Рисунок 3.4 — Определить самый дорогой товар.

```

1 CREATE PROCEDURE [dbo].[5_procedure]
2 AS
3     declare myCursor cursor
4     scroll for
5     select title, price*amount as cost,
6         tax = IIF(price*amount > 10000, amount*price*0.2, amount*price
7             *0.13)
8     from Warehouse
9     open myCursor
10    declare @n int, @title varchar(20), @cost int, @tax int
11    set @n = 0
12    while (@n < @@CURSOR_ROWS)
13    begin
14        set @n = @n + 1
15        fetch next from myCursor into @title, @cost, @tax
16        print @title + str(@cost) + str(@tax)
17    end
18    close myCursor
19    deallocate myCursor
20 RETURN 0

```

Рисунок 3.5 — Вычислить налоги на выручку. Для вычисления налога использовать функцию ИФ (налог считать так: если выручка от продажи больше 10000, то налог равен 20%, иначе — 13% от выручки).

```

1 CREATE PROCEDURE [dbo].[6_procedure]
2 @etalonCost int
3 AS
4     declare myCursor cursor scroll for
5     select title, price*amount as cost
6     from Warehouse where price*amount > @etalonCost
7     open myCursor
8     declare @n int, @title varchar(20), @cost int
9     set @n = 0
10    while (@n < @@CURSOR_ROWS)
11    begin
12        set @n = @n + 1
13        fetch next from myCursor into @title, @cost
14        print @title + str(@cost)
15    end
16    close myCursor
17    deallocate myCursor
18 RETURN 0

```

Рисунок 3.6 — Написать процедуру, возвращающую название всех товаров, выручка от реализации которых превосходит указанное значение передаваемого в процедуру параметра.

```
1 CREATE PROCEDURE [dbo].[7_1_procedure]
2 AS
3     alter table Warehouse add delivery_date date
4     select * from Warehouse
5 RETURN 0
6
7 CREATE PROCEDURE [dbo].[7_2_procedure]
8 AS
9     update Warehouse set delivery_date = GETDATE()
10    select * from Warehouse
11 RETURN 0
```

Рисунок 3.7 — Написать процедуру, изменяющую структуру таблицы Warehouse путём добавления нового столбца, соответствующего дате поставки. В качестве даты использовать текущую дату для всех записей

```
1 CREATE PROCEDURE [dbo].[8_procedure]
2 @title varchar(20), @newPrice int
3 AS
4     update Warehouse set price = @newPrice
5     where title = @title
6     select * from Warehouse where title = @title
7 RETURN 0
```

Рисунок 3.8 — Написать процедуру, изменяющую цену указанного товара на новую цену, также устанавливаемую в качестве параметра.

```
1 CREATE PROCEDURE [dbo].[9_procedure]
2 AS
3     declare @price int, @codeman int, @company varchar(20)
4     select @codeman = codeman, @price = price from Warehouse
5     where price = (select MIN(price) from Warehouse)
6     select @company = company from Manufacturer where codeman = @codeman
7     print @company + 'produces the cheapest item at the price' + str(@price
8     )
9 RETURN 0
```

Рисунок 3.9 — Вывести название фирмы, производящей самый дешевый товар.

```
1 CREATE PROCEDURE [dbo].[10_procedure]
2 AS
3     declare myCursor cursor scroll for
4     select price, amount from Warehouse for read only
5     open myCursor
6     declare @price int, @n int, @amount int, @sum int
7     set @n = 0
8     set @sum = 0
9     while (@n < @@CURSOR_ROWS)
10    begin
11        set @n = @n + 1
```

```
12      fetch next from myCursor into @price, @amount
13      set @sum = @sum + @price*@amount
14  end
15  close myCursor
16  deallocate myCursor
17  RETURN @sum
```

Рисунок 3.10 — Продемонстрировать использование курсора для подсчета суммарной выручки от реализации всех товаров.

4 Вывод

В ходе лабораторной работы были изучены возможности MS SQL Server по созданию базы данных, использованию хранимых процедур.