Вариант № 3

1. Разработать процедуру, которая по описанию товара ([Товар], [Единица измерения], [Вес ЕдИзм(Кг)], [Группа товаров]) выполняет поиск в таблице «Товары», если такой записи нет, то она добавляется. Процедура возвращает параметр – код найденного или добавленного товара.

use [ИС-20 Апасова]

go

create procedure Task1(@product [nvarchar](50), @EdIzm [nvarchar](10), @Weight [real], @group [nvarchar](30), @id [smallint] output)

as

if exists (select Товар, [Единица измерения], [Вес ЕдИзм(Кг)], [Группа товаров] from Товары

where Товар=@product and [Единица измерения]=@EdIzm and [Вес ЕдИзм(Кг)]=@Weight and [Группа товаров]=@group)

begin

set @id=(select [Код товара] from Товары where Товар=@product and [Единица измерения]=@EdIzm and [Вес ЕдИзм(Кг)]=@Weight and [Группа товаров]=@group)

end

else

begin

set @id=(select top 1 [Код товара] from Товары order by [Код товара] desc)+1

insert into Товары ([Код товара], Товар, [Единица измерения], [Вес ЕдИзм(Кг)], [Группа товаров])

values(@id, @product, @EdIzm, @Weight, @group)

end

вызов функции: declare @newid [smallint]=1

execute Task1 N'Конфеты "Птичье молоко"', N'Коробка', 0.6, N'Кондитерские изделия', @newid output

select @newid

USE [ИС-20 Апасова]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[Task1] Script Date: 16.02.2022 15:01:04 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER procedure [dbo].[Task1](@product nvarchar(50), @EdIzm nvarchar(10), @Weight real, @group nvarchar(30), @id smallint output)

as

if exists (select Товар, [Единица измерения], [Вес ЕдИзм(Кг)], [Группа товаров] from Товары

where Товар=@product and [Единица измерения]=@EdIzm and [Вес ЕдИзм(Кг)]=@Weight and [Группа товаров]=@group)

set @id=(select [Код товара] from Товары where Товар=@product and [Единица измерения]=@EdIzm and [Вес ЕдИзм(Кг)]=@Weight and [Группа товаров]=@group)

else

begin

set @id=(select top 1 [Код товара] from Товары order by [Код товара] desc)+1

insert into Товары ([Код товара], Товар, [Единица измерения], [Вес ЕдИзм(Кг)], [Группа товаров])

values(@id, @product, @EdIzm, @Weight, @group)

end

1. Разработать процедуру с параметрами год и месяц, которая для указанного месяца указанного года строит таблицу со сводными данными: Продавец, Покупатель, Товар, Суммарная стоимость продаж, Суммарный вес продаж, Суммарное количество проданных единиц, Единица измерения товара, Средняя цена, Количество продаж (договоров).

use [ИС-20 Апасова]

go

create procedure Task2(@year int, @month int)

as

select distinct Продавцы.Продавец, Покупатели.Покупатель, Товары.Товар, sum(ТоварыВдоговорах.Количество\*ТоварыВдоговорах.Цена) as СуммарнаяСтоимостьПродаж,

sum(ТоварыВдоговорах.Количество\*Товары.[Вес ЕдИзм(Кг)]) as СуммарныйВесПродаж, sum (ТоварыВдоговорах.Количество) as КоличествоПроданныхЕдиниц,

Товары.[Единица измерения], avg(ТоварыВдоговорах.Цена) as СредняяЦена, count(ТоварыВдоговорах.[Номер договора]) as КоличествоДоговоров

from Продавцы

inner join Договоры on Продавцы.[Код продавца]=Договоры.[Код продавца]

inner join Покупатели on Договоры.[Код покупателя]=Покупатели.[Код покупателя]

inner join ТоварыВдоговорах on Договоры.[Номер договора]=ТоварыВдоговорах.[Номер договора]

inner join Товары on ТоварыВдоговорах.[Код товара]=Товары.[Код товара]

where year(Договоры.Дата)=@year and MONTH(Договоры.Дата)=@month

group by Продавцы.Продавец, Покупатели.Покупатель, Товары.Товар, Товары.[Единица измерения]

1. Разработать функцию, которая для поставщика создает список номеров его договоров через запятую.

Create FUNCTION Task4(@ID smallint)

RETURNS VARCHAR(1000) AS

BEGIN

DECLARE @list varchar(1000)

SELECT @list = COALESCE(@list + ', ', '') +CONVERT(varchar, [Номер договора])

FROM Договоры Д

JOIN Продавцы П ON

Д.[Код продавца] = П.[Код продавца]

Where Д.[Код продавца] = @ID

Return @list

END