

## РАЗРАБОТКА БАЗЫ ДАННЫХ И СОЗДАНИЕ ХРАНИМЫХ ПРОЦЕДУР В СРЕДЕ MSSQLSERVER 2008 R2 ДЛЯ ТОНКОГО КЛИЕНТА.

*Изучаются возможности сервера MSSQLServer2008 R2 для создания базы данных с использованием ERStudio .Созданы хранимые процедуры для реализации тонкого клиента в системе EmbarcaderoERStudioXE2.*

## ИЧКЕ КЛИЕНТ УЧУН MSSQLSERVER 2008 R2 ЧОЙРОСУНДО БЕРИЛИШТЕР БАЗАСЫН ЖАНА САКТАЛУУЧУ ПРОЦЕДУРАЛАРДЫ ТУРГУЗУУ.

*Берилиштер базасын тургузууда ERStudio системасын колдонуп MSSQLServer2008 R2 серверинин мүмкүнчүлүктөрү иштелет. Ичке клиент учун EmbarcaderoERStudioXE2 системасында сакталуучу процедуралар тургузулат.*

## DATABASE DEVELOPMENT AND CREATE A STORED PROCEDURE IN MS SQL SERVER ENVIRONMENT 2008 R2 FOR THIN CLIENTS.

*We study the capabilities of the server MS SQL Server2008 R2 to create a database using the ER Studio. Created stored procedures for the implementation of a thin client in the Embarcadero ER Studio XE2.*

В работы мы, создадим базу данных управления библиотечной системой в среде сервера MSSQLServer 2008 R2. Создадим хранимые процедуры на указанном сервере для работы с редактируемыми запросами в клиентской части сетевого приложения. В работе [1], мы спроектировали с помощью системы EmbarcaderoERStudio работу с базой данных библиотечной системы. Были получены исходные коды работы для данного сервера. После регистрации исходного кода базы данных [1] на сервере MSSQLServer2008 R2 получаем следующую структуру базы данных

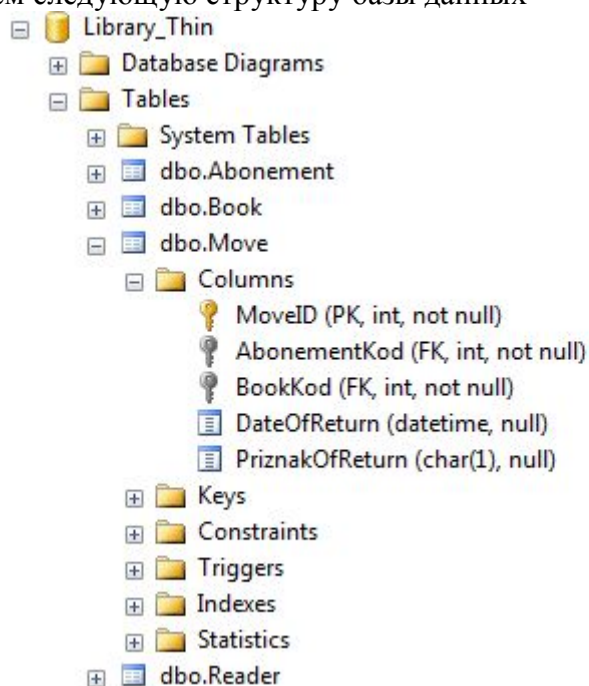


Рис.1. Структура базы данных «Библиотечная система » в среде MSSQLServer 2008 R2.

В структуру базы данных библиотечной системы входят таблицы Abonement (Абонемент), Book (Книги), Move (Движения книг) и Reader (Читатели). В дальнейшем для создания тонкого клиента на сервере необходимо создать бизнес логику управления данными. Для этой цели на сервере необходимо создать хранимые процедуры. Обратимся к схеме данных базы данных. Данная схема отражает ссылочную целостность изучаемой базы данных. Таблицы Reader, Book и Abonement являются элементами справочной системы. Работу с данными таблицами мы подробно остановимся на клиентской части сетевого приложения. Таблица Move(Движения) является основной таблицей для управления данными. Для данной таблицы Move (Движения) мы создадим набор хранимых процедур, в которых будет реализован механизм добавления,изменения и удаления данных, а также некоторые дополнительные действия .

Выпишем теперь несколько процедур работы с таблицами базы данных .Первая процедура –**CloseAbonement**-которая предназначена для закрытия **Абонемента** выпишем следующим образом.

```
USE [Library_Thin]
GO
ALTER PROCEDURE [dbo].[CloseAbonement]
    @AbonementID_Par int,
    @CountOfBook int,
    @Result_Output int Output
AS
BEGIN
    SET NOCOUNT ON;
    SET @Result_Output=1;
    Set      @CountOfBook=(Select      COUNT(*)      From      Move      Where
(AbonementKod=@AbonementID_Par and PriznakOfReturn ='0' ));
    if @CountOfBook>0 begin Set @Result_Output =0; end else
    begin
        Update Abonement Set PriznakOfEnd='1' where AbonementID=@AbonementID_Par
    end
END
```

Цель второй сложной процедуры -**GiveBook**предназначена для выдачи книг читателю.

```
USE [Library_Thin]
GO
ALTER PROCEDURE [dbo].[GiveBook]
    @BookKod_Par int,
    @AbonementKod_Par int,
    @DateOfReturn_par DateTime,
    @PriznakOfReturn_par char(1)
AS
BEGIN
    SET NOCOUNT ON;
    Insert Into Move(BookKod,AbonementKod, DateOfReturn,PriznakOfReturn )values
    (@BookKod_Par,@AbonementKod_Par,@DateOfReturn_Par,'0');
    Update Book Set OnHand='1' where BookID=@BookKod_Par
END
```

В данной процедуре мы добавляем новую запись (InsertInto) в таблицу Move, значения атрибутов берутся из входных параметров.Далее командой Update мы устанавливаем выданной книге атрибут OnHand в единицу.Это нужно для того , чтобы знать , что книга не в библиотеке. Входными являются параметры @BookKod\_Parint, @AbonementKod\_Parint и @DateOfReturn\_parDateTime.

Третья процедура –**ModifyAbonement**-будет предназначена для добавления,изменения и удаления абонементов

```
USE [Library_Thin]
GO
ALTER PROCEDURE [dbo].[ModifyAbonement]
```

```

    @AbonementID_Par int,
    @ReaderKod_par int,
    @DateOfEnd_Par DateTime,
    @PriznakOfEnd_Par char(1),
    @Choose_Par int
AS
BEGIN  if( @Choose_Par=1 ) Begin
INSERT  INTO  Abonement  (ReaderKod,  DateOfEnd,  PriznakOfEnd)  Values
(@ReaderKod_Par,@DateOfEnd_Par,@PriznakOfEnd_Par);
end
if (@Choose_Par=2)
begin
Update Abonement SET
ReaderKod=@ReaderKod_par,
DateOfEnd=@DateOfEnd_Par,
PriznakOfEnd=@PriznakOfEnd_Par
where
AbonementID=@AbonementID_Par;
EndEnd
Четвертая процедура–ModifyBook–будетпредназначена для редактирования таблицы Book
USE [Library_Thin]
GO
ALTER PROCEDURE [dbo].[ModifyBook]
-- Add the parameters for the stored procedure here
@BookID_Par int,
@NameOfBook_par varchar(50),
@Shifr_par int,
@BookAbout_Par varchar(500),
@Onhand_Par char(1),
@Choose_par int
AS
BEGIN
SET NOCOUNT ON;
if @Choose_par=1
begin
Insert Into Book(NameOfBook,Shifr,BookAbout,Onhand) values
(@NameOfBook_par,@Shifr_par,@BookAbout_Par,@Onhand_Par );
end
if @Choose_par=2
Begin Update Book Set
NameOfBook=@NameOfBook_par,
Shifr=@Shifr_par,
BookAbout=@BookAbout_Par,
Onhand =@Onhand_Par
where BookID=@BookID_Par
end
if @Choose_par=3
begin
Delete From Book where BookID=@BookID_Par;
end
END
Пятая процедура–ModifyReader–предназначено для работы с читателями
USE [Library_Thin]
GO
ALTER PROCEDURE [dbo].[ModifyReader]
@FIO_par varchar(50),
@Pasport_Par varchar(25),

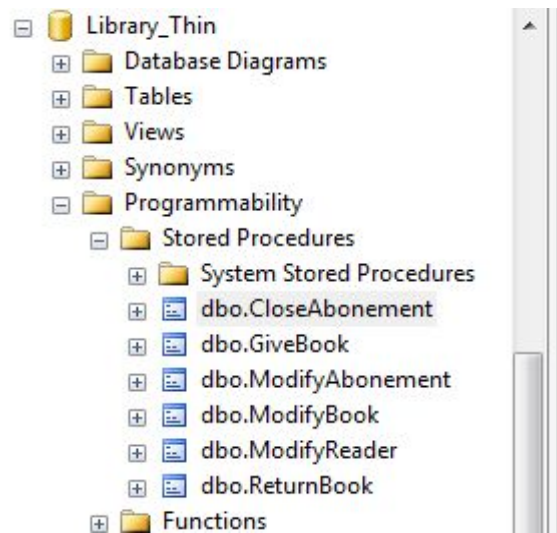
```

```

    @ReaderID_par int,
    @Choose_Par int,
    @Result_Output int output
AS
declare
    @CountOfAbonement int
BEGIN
    SET NOCOUNT ON;
    SET @Result_Output=1;
    if ( @Choose_Par=1 )
Begin
INSERT INTO Reader (FIO,Pasport) Values(@FIO_Par,@Pasport_Par);
end
if (@Choose_Par=2)
begin
Update Reader SET
FIO=@FIO_Par,
Pasport=@Pasport_Par
where
ReaderID=@ReaderID_Par;
end
if (@Choose_Par=3)
begin
SET      @CountOfAbonement=(Select      COUNT(*)      From      Abonement      Where
ReaderKod=@ReaderID_Par);
Select @CountOfAbonement
If ( @CountOfAbonement<0) Delete From Reader Where ReaderID=@ReaderID_Par else SET
@Result_Output=0;
end
END
Шестая процедура–ReturnBook-предназначена для возврата книг
USE [Library_Thin]
GO
ALTER PROCEDURE [dbo].[ReturnBook]
@BookID_Par int,
@AbonementKod_Par int
AS
BEGIN
update Book Set OnHand='0' where BookID=@BookID_Par;
update Move Set PriznakOfReturn='1' where (BookKod=@BookID_Par) and
(AbonementKod=@AbonementKod_Par);
END

```

Вот таким образом создаются хранимые процедуры на сервере MSSQLSERVER 2008 R2, которые существенным образом повышают быстродействие любой БД. К ним мы будем обращаться с клиентской программы. На сервере MSSQLSERVER 2008 R2 созданные процедуры будут, выглядит следующим образом.



### Литература:

1. Сабитов Б.Р, Алмазбекова З. Использование системы EmbarcaderoERStudio для проектирования базы данных сетевых информационных систем. Настоящий сборник.
2. Фаронов В.А. Система программирования Delphi 2007, БХВ, СПб. 2009 г.
3. Фленов М. "Библия Embarcadero RAD Studio XE2". Москва 2011 г.
4. Марко Кэнту, Delphi 7. Для профессионалов. – М.: SYBEX, ПИТЕР 2009. – 1100 с.