Министерство образования и науки Российской Федерации

Севастопольский Государственный Университет

Кафедра ИС

Отчет

по лабораторной работе №1

Изучение архитектуры сервера MS SQL Server.

SQL. Создание таблицы и элементарные выборки

Выполнил:

ст. гр. ИС22-д

Серегин А.В.

Проверил:

Тимофеева Т.А.

Севастополь

2016

1. Цель работы

1.1. Изучить основы организации сервера MS SQL Server; научится устанавливать соединение с сервером. Научиться создать базу данных и производить элементарные действия над ней.

1.2. Изучить формы оператора CREATE TABLE и простейшие формы оператора SELECT.

2. Вариант задания

Вариант 1:



Рисунок 1 – Структура системы, содержащая информацию

о публикациях по ряду выбранных тем.

3. Создание базы данных и таблиц

--Создание базы данных

CREATE DATABASE University ON(

NAME = publication\_dat,

FILENAME = 'D:\DataBase\University\publication.mdf',

SIZE = 10MB,

MAXSIZE = 100MB,

FILEGROWTH = 5MB

) LOG ON (

NAME = publication\_log,

FILENAME = 'D:\DataBase\University\publication.ldf',

SIZE = 10MB,

MAXSIZE = 50MB,

FILEGROWTH = 5MB

)

--Создание таблицы авторов

CREATE TABLE Author

(

AuthorID int IDENTITY(1,1) PRIMARY KEY,

Sername varchar(20) not null,

Name varchar(20) not null,

Patronymic varchar(20) not null,

Adress varchar(100) null

)

--Создание таблицы статей

CREATE TABLE Article

(

ID int IDENTITY(1,1) PRIMARY KEY,

Publishing varchar(30) not null

)

--Создание таблицы журналов

CREATE TABLE Magazine

(

ID int IDENTITY(1,1) PRIMARY KEY,

Title varchar(30) not null,

Number int not null

)

--Создание таблицы тем публикаций

CREATE TABLE Theme

(

ID int IDENTITY(1,1) PRIMARY KEY,

Title varchar(50) not null

)

--Создание таблицы хранящей информацию о публикациях

CREATE TABLE Publication

(

ThemeID int null FOREIGN KEY REFERENCES dbo.Theme([ID]),

AuthorID int null FOREIGN KEY REFERENCES dbo.Author([AuthorID]),

Title varchar(100) not null,

ReleaseDate date not null,

PublicationType varchar(30) not null check (PublicationType in('Magazine','Article')),

)

4. Заполнение таблиц исходными данными

-- Заполнение таблицы Theme

INSERT INTO dbo.Theme VALUES ('Политика')

INSERT INTO dbo.Theme VALUES ('Экономика')

INSERT INTO dbo.Theme VALUES ('Наука')

INSERT INTO dbo.Theme VALUES ('Культура')

-- Заполнение таблицы Author

INSERT INTO dbo.Author(Sername, Name, Patronymic, Adress) VALUES('Иванов','Василий','Борисович','г.Увельский, ул.Авиационная, д.26, кв.40')

INSERT INTO dbo.Author(Sername, Name, Patronymic, Adress) VALUES('Качалов','Тимур','Альбертович',null)

INSERT INTO dbo.Author(Sername, Name, Patronymic, Adress) VALUES('Шалдыбин','Василий','Витальевич','г.Сосновоборск, ул.Глинки, д.81, кв.80')

INSERT INTO dbo.Author(Sername, Name, Patronymic, Adress) VALUES('Сидоров','Матвей','Васильевич','г.Усть-Абакан, ул.Маяковского, д.52, кв.3')

INSERT INTO dbo.Author(Sername, Name, Patronymic, Adress) VALUES('Качалов','Ануфри','Тимурович',null)

INSERT INTO dbo.Author(Sername, Name, Patronymic, Adress) VALUES('Муравьев','Александр','Борисович','г.Урень, ул.Бардина, д.14, кв.176')

INSERT INTO dbo.Author(Sername, Name, Patronymic, Adress) VALUES('Муравьева','Светлана','Витальевна','г.Усть-Большерецк, ул.Весенняя, д.78, кв.163')

INSERT INTO dbo.Author(Sername, Name, Patronymic, Adress) VALUES('Иванова','Александра','Тимуровна','г.Подпорожье, ул.Абрамцевская, д.80, кв.53')

INSERT INTO dbo.Author(Sername, Name, Patronymic, Adress) VALUES('Шалдыбин','Александр','Семенович','г.Саратов, ул.Баррикадная, д.61, кв.170')

INSERT INTO dbo.Author(Sername, Name, Patronymic, Adress) VALUES('Сидоров','Василий','Семенович',null)

-- заполнение таблицы Publication

INSERT INTO dbo.Publication VALUES(1,1,'8testTitle','2013-07-14','Article')

INSERT INTO dbo.Publication VALUES(2,2,'15testTitle','2034-03-17','Magazine')

INSERT INTO dbo.Publication VALUES(3,3,'23testTitle','2022-06-14','Article')

INSERT INTO dbo.Publication VALUES(4,4,'6testTitle','2054-12-17','Magazine')

INSERT INTO dbo.Publication VALUES(null,null,'20testTitle','2011-11-14','Article')

INSERT INTO dbo.Publication VALUES(2,null,'11testTitle','2011-02-17','Magazine')

INSERT INTO dbo.Publication VALUES(3,7,'4testTitle','2056-06-14','Article')

INSERT INTO dbo.Publication VALUES(null,8,'18testTitle','2042-04-17','Magazine')

INSERT INTO dbo.Publication VALUES(1,9,'13testTitle','2013-05-14','Article')

INSERT INTO dbo.Publication VALUES(2,null,'12testTitle','2044-11-17','Magazine')

INSERT INTO dbo.Publication VALUES(null,null,'2testTitle','2017-10-14','Article')

INSERT INTO dbo.Publication VALUES(4,2,'24testTitle','1963-03-17','Magazine')

INSERT INTO dbo.Publication VALUES(1,3,'26testTitle','1922-07-14','Article')

INSERT INTO dbo.Publication VALUES(2,null,'9testTitle','2043-06-17','Magazine')

INSERT INTO dbo.Publication VALUES(3,5,'16testTitle','2005-07-14','Article')

INSERT INTO dbo.Publication VALUES(null,6,'21testTitle','1117-03-17','Magazine')

INSERT INTO dbo.Publication VALUES(1,7,'7testTitle','1992-02-14','Article')

INSERT INTO dbo.Publication VALUES(null,8,'14testTitle','2031-12-17','Magazine')

INSERT INTO dbo.Publication VALUES(3,9,'1testTitle','2002-07-14','Article')

INSERT INTO dbo.Publication VALUES(4,null,'10testTitle','2023-11-17','Magazine')

INSERT INTO dbo.Publication VALUES(1,1,'5testTitle','2022-12-14','Article')

INSERT INTO dbo.Publication VALUES(2,null,'17testTitle','2017-10-17','Magazine')

INSERT INTO dbo.Publication VALUES(3,3,'19testTitle','1987-07-14','Article')

INSERT INTO dbo.Publication VALUES(null,4,'3testTitle','2013-08-14','Article')

INSERT INTO dbo.Publication VALUES(1,5,'22testTitle','1948-07-17','Magazine')

INSERT INTO dbo.Publication VALUES(2,6,'25testTitle','2003-09-17','Magazine')

5. Запросы и результаты

--Вывод списка всех авторов

SELECT \* FROM dbo.Author

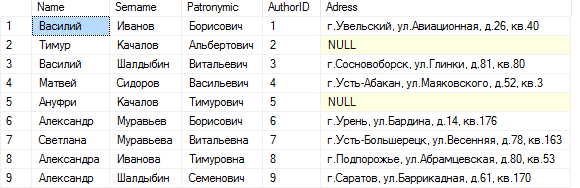
Результат:



--Вывод данных об каждом авторе без ID

SELECT [Name], [Sername], [Patronymic], [AuthorID], [Adress] FROM dbo.Author

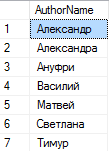
Результат:



--Вывод всех разных имен автора

SELECT DISTINCT [Name] as AuthorName FROM dbo.Author

Результат:



--Вывод всех авторов имеющих запись о месте жительства

SELECT \* FROM dbo.Author

WHERE [Adress] is NULL

Результат:



--Вывод авторов, чей ID находится в промежутке [3,8], имеет инф. о месте жительтсва

--и имя начинающееся на "Алекс"

SELECT \* FROM dbo.Author

WHERE [AuthorID] between 3 and 8 and [Name] like 'Алекс%' and [Adress] is not null

Результат:

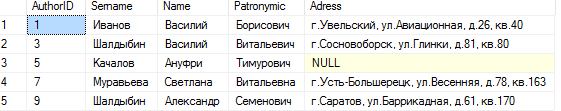


--Вывод всех авторов, чьи ID не находятся в множестве {2,4,6,8,10}

SELECT \* FROM dbo.Author

WHERE [AuthorID] NOT IN(2,4,6,8,10)

Результат:



Выводы

В ходе лабораторной работы мы научились устанавливать связь с сервером с помощью MS SQL Server Management Studio. Научились создавать запросы к серверу для создания баз данных, таблиц и простых манипуляций с ними.