**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное бюджетное   
образовательное учреждение высшего образования   
**«КУЗБАССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ**

**УНИВЕРСИТЕТ ИМЕНИ Т. Ф. ГОРБАЧЕВА»**

филиал КузГТУ в г. Прокопьевске

Кафедра информационных технологий,   
машиностроения и автотранспорта (ИТМА)

**Методические указания  
ДЛЯ выполнениЯ практическОЙ работЫ**   
по дисциплине «Информационные технологии» по теме «Реляционные базы данных» для студентов очной формы обучения направления подготовки 09.03.02 «Информационные системы и технологии», профиль – 01 «Системная интеграция и автоматизация информационных процессов»

Составитель: **А.С.Абрамович**

Рассмотрены и утверждены  
на заседании кафедры  
Протокол № X от X.X.2021 г.

Рекомендованы к печати  
учебно-методической комиссией

Протокол № X от X.X.2021 г.

Электронная копия находится  
в библиотеке филиала КузГТУ  
в г. Прокопьевске

Прокопьевск 2021

**Рецензент:**  
старший преподаватель кафедры информационных технологий, машиностроения и автотранспорта в филиале КузГТУ в г. Прокопьевске

**Иноземцев М.А.**

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 4](#_Toc92756413)

[Загрузка. Установка. Запуск системы управления базами данных Microsoft SQL Server 2012 Express 5](#_Toc92756414)

[Установка Microsoft SQL Server 2012 Express 5](#_Toc92756416)

[Создание базы данных в Microsoft SQL Server 2012 Express 16](#_Toc92756423)

[Задание № 1 20](#_Toc92756430)

[Разработка даталогической модели реляционных баз данных 20](#_Toc92756431)

[Задание № 2 26](#_Toc92756432)

[Создание и заполнение таблиц средствами языка SQL 26](#_Toc92756433)

[Список литературы 32](#_Toc92756434)

ВВЕДЕНИЕ

Данные методические указания разработаны для обучающихся ВО по направлению подготовки 09.03.02 «Информационные системы и технологии», профиль – 01 «Системная интеграция и автоматизация информационных процессов» в рамках изучения темы «Реляционные базы данных» в рамках дисциплины «Информационные технологии».

Загрузка. Установка. Запуск системы управления базами данных Microsoft SQL Server 2012 Express

## Скачайте Microsoft SQL Server 2012 Express, пройдя по ссылке <https://www.microsoft.com/ru-RU/download/details.aspx?id=55994>

## 

Рисунок 1 –Загрузка Microsoft SQL Server 2012 Express.

## Установка Microsoft SQL Server 2012 Express

## Запускаете установщик с расширением .exe.

## 

## 

Рисунок 2 – Начало установки.

## Далее выбираете новую установку изолированного экземпляра.

## 

Рисунок 3 – Установка изолированного экземпляра.

## 

Рисунок 4 – Условия лицензии.

## 

Рисунок 5 – Обновления продукта.

## 

Рисунок 6 – Установка файлов установки.

## 

Рисунок 7 – Установка файлов установки.

## 

Рисунок 8 – Установка файлов установки.

## 

Рисунок 9 – Выбор компонентов.

## 

Рисунок 10 – Настройка экземпляра.

## 

Рисунок 11 – Конфигурация сервера.

## 

Рисунок 12 – Настройка компонента Database Engine.

## 

Рисунок 13 – Настройка служб Reporting Services.

## 

Рисунок 14 – Отчет об ошибках.

## 

Рисунок 15 – Ход выполнения установки.

## 

Рисунок 16 – Завершение установки.

## На этом установка завершена. Далее переходим к запуску сервера. Находите Microsoft SQL Server в меню пуск и открываете приложение «Среда SQL Server Management Studio».

## 

Рисунок 17 – Пуск. Среда SQL Server Management Studio.

## 

Рисунок 18 – Стартовое окно SQL Server Management Studio 2012 Express.

## Вместо Desktop-6FNDHPG вводите имя своего компьютера.

## 

Рисунок 19 – Рабочая среда SQL Server Management Studio 2012 Express.

## Для того чтобы создать базу данных достаточно правой кнопкой мыши нажать на папку Базы данных и создать новую базу.

## 

Рисунок 20 – Создание новой базы данных.

## 

Рисунок 21 – Создание базы данных.

## 

Рисунок 22 –Дерево баз данных в Microsoft SQL Server 2012 Express.

## На этом создание базы данных завершено.

# Создание отношений в базе данных в СУБД Microsoft SQL Server 2012 Express

## Для того чтобы создать таблицы в базе данных можно использовать или конструктор или создавать таблицы программным кодом-запросом –Create table Table1 (Atr1 int Primary key identity(1,1), Atr2 nvarchar(50))- это есть в теории, повторите.

## Создание таблицы с помощью конструктора.

## 

Рисунок 23 – Создание таблицы.

## 

Рисунок 24 – Создание атрибутов таблицы базы данных.

## Написание запросов в Microsoft SQl Server 2012 Express.

## Для того чтобы создать и запустить запрос необходимо создать новый экземпляр запроса – Нажимаете в меню кнопку «Создать запрос».

## 

Рисунок 25 – Создание запроса.

## 

Рисунок 26 – Создание нового запроса.

## После того, как вы написали запрос, необходимо его выполнить, для этого в панели меню нажимаете на кнопку «Выполнить».

## 

Рисунок 27 – Выполнение созданного запроса.

## 

Рисунок 28 – Результат выполнения созданного запроса.

## Для того, чтобы проверить, действительно ли создалась таблица, которую мы создали кодом, откройте в дереве базы данных папку Таблицы и посмотрите, была ли создана таблица. Иногда необходимо обновлять сервер или базу данных. Для этого правой кнопкой мыши нажмите на базу данных и нажмите кнопку «Обновить».

## 

Рисунок 29 – Результат выполнения созданного запроса.

## 

Рисунок 30 – Обновление базы данных.

Задание № 1

Разработка даталогической модели   
реляционных баз данных

**Цель:** получить навыки разработки баз данных в среде MS SQL SERVER Management Studio 2012, изучить архитектуру СУБД Microsoft SQL Server, изучить принципы создания моделей баз данных на основе анализа и выявления объектов предметной области.

**Теория**

Базовым элементом баз данных, построенных на основе реляционной модели является отношение. Отношение реализуется в среде различных СУБД как таблица. **Таким образом, таблица** это объект, предназначенный для хранения информации в реляционной БД. Информация об единичном экземпляре данных представляется как **запись (кортеж)** или строка в таблице. **Поля (атрибуты)** объекта представляются как – столбцы в табличном виде.

Поля в реляционных базах данных характеризуются следующими свойствами:

* имя поля – идентификатор поля, по которому организуется программный доступ к нему;
* тип поля – тип данных, находящихся в этом поле;
* размер поля - величина в байтах, выделяемая для хранения данных в поле. Например: если тип поля СТРОКОВЫЙ, а размер будет равен 10-ти, то это значит, что в ячейку такого поля нельзя будет записать строку более 10 символов. Если задать ЦЕЛЫЙ ЧИСЛОВОЙ тип и установить размер в 4 байта, то числа в ячейке будут принимать значения от 0 до 65535;
* инкриментность (счетчик) – автозаполнение поля в добавленной записи неким значением (как правило, числового целого типа);
* ключ – уникальный идентификатор, характеризующий запись;
* необходимость заполнения – если поле не обязательно для заполнения, то при добавлении записи (в случае отсутствия данных в поле) оно автоматически заполняется значением по умолчанию, если таковое имеется. Если значения по умолчанию нет, записывается псевдопустое значение «NULL», которое определено в системе специальным идентификатором.

**Microsoft SQL Server**

Microsoft SQL Server — это масштабируемая высокопроизводительная система управления реляционными базами данных для платформ на базе Windows Server. Она разработана с учетом требований к современным распределенным клиент-серверным вычислениям и тесно интегрирована с серверными продуктами семейства Microsoft Office.

Включает в себя библиотеки и службы ядра сервера СУБД. Служба Microsoft SQL Server представляет собой движок СУБД, обрабатывающий все запросы, приходящие на сервер.

В стандартный пакет Microsoft SQL Serverвходят несколько приложений, служащих для администрирования и разработки клиент-серверных приложений:

Для разработки таблиц и серверных механизмов используется приложение Microsoft SQL Server Management Studio 2012 и более новые версии.

При запуске приложения открывается окно соединения приложения с сервером. Приложение можно использовать для работы серверами, установленными не зависимо от MICROSOFT SQL SERVER Management Studio.

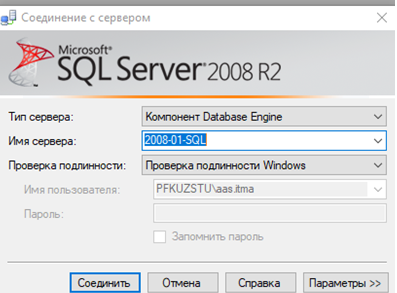


Рисунок 31 - Окно соединения с сервером.

Для соединения с сервером необходимо знать его имя, имя записи, зарегистрированной на сервере и пароль для этой записи. Если используется авторизация на основе учетной записи Windows, данная учетная запись должна быть зарегистрирована на сервере БД. После соединения с сервером открывается окно приложения Microsoft SQL Server Management Studio 2012 Express.

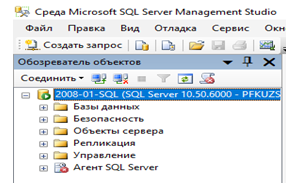


Рисунок 32 - Рабочее окно Microsoft SQL Server Management Studio 2012 Express.

Левую часть окна занимает рабочее окно обозревателя объектов сервера. Объекты сервера представлены в виде древовидной структуры. Корнем дерева является соединение. Management Studio может быть одновременно соединено с несколькими серверами. Работа с любыми объектами сервера может осуществляться через контекстное меню на соответствующем узле дерева.

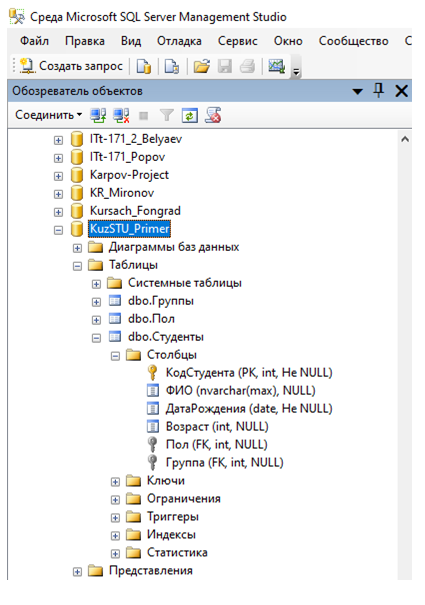


Рисунок 33 - Работа с таблицей в режиме её модификации.

**Выявление отношений в базе данных**

В качестве отношений реляционной базы данных отображаются объекты предметной области, обеспечивающие получение информации, определенной в требованиях к системе.

Для выявления сущностей предметной области необходимо её проанализировать и выявить объекты, обладающие свойствами, на основе которых может быть получена информация, определённая в требованиях для базы данных. Состав объектов должен быть достаточным, но не избыточным. Обычно выделяются объекты оперативные и справочные.

Оперативные объекты содержат некоторую текущю информацию, они часто обновляются. Это могут быть данные о единичной покупке, например:

- таблица «Покупка» (покупатель, товар, количество товара);

Справочные объекты содержат информацию которая может использоваться в качестве значений атрибутов для оперативных объектов. Например данные о товаре:

таблица «Товар»(наименование, цена, производитель);

Атрибуты справочных таблиц т могут определяться значениями, других справочных таблиц, например атрибут производитель в таблице таблица «Товар» может определяться значениями таблицы таблица «Производитель»(Наименование, номер счёта, юр. адрес)

Для выявленных отношений устанавливаются атрибуты и требования к ним.

Для каждого отношения необходимо сформулировать бизнес – правила соответствующей предметной области.

Бизнес – правила характеризуют поведение объекта в предметной области, значение его атрибутов.

Необходимо проанализировать атрибуты, выявленные для отношений, на предмет их атомарности. Не атомарный атрибут подразумевает некоторое множество составных атрибутов, а следовательно его можно представить в виде другого отношения.

**Например:**

Студент – объект, выполняющий обучение на предметах.

**Характеризуется:**

фамилией, именем, отчеством (отдельные атрибуты типа строка);

номером зачетной книжки (атрибут целого типа).

Студент обучается на **учебном курсе** (**учебный курс** – это отдельное отношение, так как может иметь свои характеристики).

Для выявленных объектов и их атрибутов необходимо выявить бизнес правила, определяющие требования целостности сущности, то есть обязательность значения данного атрибута, уникальность значения данного атрибута, его допустимые значения.

**Например:**

Фамилия студента состоит из символов, это обязательный атрибут.

Номер зачетной книжки – число, минимальное значение -10000,

Максимальное 99999.

Для выявленных отношений необходимо определить бизнес-правила их функционирования в предметной области определяющие их с другими отношениями

В бизнес-правилах, характеризующих связи должна быть дана следующая информация:

* содержания связи;
* множественность связи с одной и другой стороны;
* обязательности и дополнительных ограничений, ограничений накладываемых на связь.

**Например**: отношение «Покупка» связано с отношением «Товар» так как покупка должна всегда содержать товар. Данная связь имеет множественность «один к многим», так как одна покупка может содержать много товаров.

Каждое выявленное бизнес-правило реализуются в виде фрагмента диаграммы сущность-связь.

**Порядок выполнения**

1. Выделить отношения согласно заданию, описать отношения и их атрибуты.
2. Описать связи между отношениями с точки зрения их множественности с одной и другой стороны, обязательности, соответствия бизнес правилу предметной области.
3. Построить в среде SQL server Management Studio таблицы в соответствии с заданием.
4. Построить диаграмму отношений в среде SQL server Management Studio.
5. Заполнить отношения тестовыми данными.

**Требование к базе данных:** построенная база данных должна содержать не менее 7 отношений.

**Варианты заданий**

1. Разработка информационной системы управления магазином по продаже компьютерной техники.
2. Разработка информационной системы управления магазином по продаже спортивного инвентаря.

Задание № 2

Создание и заполнение таблиц   
средствами языка SQL

**Цель:** получить практический навык создания объектов БД и заполнение их данными с использованием языка SQL .

**Задачи:**

* изучение основных операторов языка определения и манипулирования данными;
* создание и выполнение запросов на создание, заполнение таблиц данными.

**SQL (**Structured Query Language, структурированный язык запросов) — это язык программирования, предназ­наченный для выборки и обработки информации, содержащейся в реляционной базе данных. SQL является стандартным языком для работы с реляционными базами данных, его основа реляционная алгебра и реляционное исчисление. SQL содержит набор стандартных операторов доступа к данными и манипулирования.

Существуют следующие версии SQL:

* SQL1 (принята в 1986 году, дополнена в 1989 году стандарт ANSI) ,
* SQL2 (SQL-92 принята в 1992 году),
* SQL3 (SQL-99) расширяет SQL2 за счет включения объектно-реляционных инструментов и новых функциональных возможностей.

Существуют версии SQL предлагаемые основными поставщиками СУБД, они, как правило, удовлетворяют требованиям ANSI, реализуют многие возможности SQL2 и имеют некоторые особенности.

SQL — это слабо структурированный язык, особенно по сравнению с такими высокоструктурированными языками, как С, Pascal или Java. В нем нет инструкции IF..THEN для проверки условий, нет инструкции GOTO для организации переходов и нет инструкций DO или FOR для создания циклов.

SQL обеспечивает независимость от конкретных СУБД: реляционную базу данных и программы, которые с ней работают, можно перенести с одной СУБД на другую с минимальными доработками и переподготовкой персонала. Все ведущие поставщики СУБД используют SQL.

Поставщики СУБД предлагают различные диалекты SQL позволяющие создавать самостоятельные программные модули, например, PL/SQL и Transact-SQL. В этих диалектах стандартный SQL дополнен инструкциями IF..THEN, GOTO и др., однако эти диалекты не получили статус стандарта и являются частными разработками отдельных компаний (PL/SQL применяется в СУБД Oracle, a Transact SQL - в СУБД MS SQL Server).

SQL не является отдельным прог­раммным продуктом. SQL — это неотъемлемая часть СУБД ее *Манипуляционная часть*, инструмент, с помощью которого осуществляется связь пользователя с БД.

**Различают несколько групп операторов (подъязыки):**

I. Язык определения данных DDL.

К языку запросов относятся операторы

CREATE TABLE - создания нового отношения;

DROP TABLE - удаление отношения;

ALTER TABLE - изменение структуры таблицы;

CREATE VIEW - создания представления;

DROP VIEW - удаления представления;

CREATE INDEX - удаление индексов.

II. Язык манипулирования данными DM (команды, DELETE, INSERT, UPDATE )

III. Язык запросов DQL (оператор SELECT)

IV. Средства управления транзакциями.

V. Средства администрирования данными.

**Типы данных В SQL**

Используемы в SQL типы данных аналогичны применяемым в других языках программирования:

char(n) – символьный длиной n символов;

varchar(n) – символьный переменной длинны, n – максимально возможная длинна;

bit(n);

varingbit(n);

logical – логический тип, в отличии от логического типа в языках программирования кроме возможных значений *false/true* может иметь значение *unknown* (неопределенно), соответственно изменены таблицы истинности базовых операций (*false*  считается наименьшим значением *true* наибольшим, *unknown* - промежуточное значение *)*;

int;

real(float) – вещественный, возможно использование для задания типа, как имени real, так и float;

datetime;

**Значение NULL и его применение**

Атрибутам отношения или переменным SQL допускает присвоение специального значения NULL.

**Значение NULL имеет следующий смысл:**

* значение не известно, то есть когда создается новый кортеж, а значение некоторого атрибута явно не задается и не заданно по умолчанию, то ему присваивается данное значение;
* значение не может быть заданно, то есть когда значения некоторого атрибута быть не может (например атрибут Супруг, для некоторого кортежа отношения Сотрудники, когда рассматриваемый сотрудник не женат);
* значение умалчивается, то есть, если значение атрибута, выдается по запросу, но данный атрибут запрещен для просмотра для данного источника запроса.

Если атрибут или выражение со значением NULL участвует в арифметической операции, то результат операции будет иметь значение NULL.

При сравнении выражения, имеющего значение NULL с другим выражением с помощью операций сравнения (=, !=, <>, <, >, >=, <=, !> ,!<) результат будет иметь значение *unknown*.

Для проверки выражения на значение NULL операция сравнения не используется. Для Этого необходимо использовать специальный предикат IS NULL (IS NOT NULL), он будет рассмотрен ниже.

**Операторы языка определения данных DDL и модификации отношений**

**Оператор задания схемы** отношения, то есть,  создаёт отношения (таблицы) – и их атрибуты.

 Общий формат оператора

*Create table ИмяОтн(аtr1 ТипАтр [ЗначПоУмолч]* [*огрАтр*]

*[,аtr2 …]*

…..

                 ОгрКортежа), где

* *atr1, atr2* … - идентификаторы (имена) атрибутов отношения;
* *ЗначПоУмолч* – значение, присваиваемое атрибуту по умолчанию;
* *огрАтр* – ограничения на значение атрибута (будут рассмотрены позже ;
* *ОгрКортежа* - ограничения на значение корежа (будут рассмотрены позже ;

**Оператор удаления отношения**

*Drop table ИмяОтн*

**Модификация отношений**

Модификация отношения может быть следующих разновидностей.

              Удаление атрибута    -

*Alter table ИмяОтн drop ИмяАтр1* – из отношения ОТН будет удалён атрибут именем *ИмяАтр1*;

Вставка атрибута -

Alter table *ИмяОтн* add *ИмяАтр1* *типАтр1* *ЗначПоУмолч].. ,*

где   *ИмяАтр1* *типАтр1*  *[ЗначПоУмолч]..*– описание атрибута, аналогичное используемому в операторе *Create table*.

**Операторы SQL манипулирования данными**.

Операторы данной группы позволяют модифицировать существующие кортежи отношений. То есть, вставлять новые кортежи отношения, удалять, изменять значений атрибутов.

**Вставка кортежей** **(INSERT)**

*insert into ИмяОтн(ИмяАтр1, ИмяАтр2, ..) values(знач1, знач2,…)*

В результате выполнения данной команды в отношение с именем *ИмяОтн* будет вставлен кортеж, при этом атрибутам с именами *ИмяАтр1, ИмяАтр2*… будут присвоены значения *знач1, знач2,…* Атрибутам, не перечисленным в списке, будет присвоено значение по умолчанию. Если значения по умолчанию не заданы, то система попытается присвоить им значения NULL.

**Удаление кортежей**

*delete from имяОтн where условие*

При выполнении данной команды из отношения с именем *имяОтн* будут удалены кортежи, значение атрибутов которых будет соответствовать условию.

**Модификация (обновление) кортежей**

*update отн set ИмяАтр1= знач1, ИмяАтр1= знач1 [,…]*

*where условие*

При выполнении данной команды кортежам, отвечающим заданным условиям будет изменено значение заданных атрибутов.

Изменение атрибутов будет отменено, если они противоречат условиям целостности  базы данных, или другим ограничениям.

Для работы с объектами СУБД MS SQL Server Management Studio предоставляет набор команд контекстного меню, в частности для работы с таблицами может использоваться иерархия контекстных меню, показанная на рисунке.

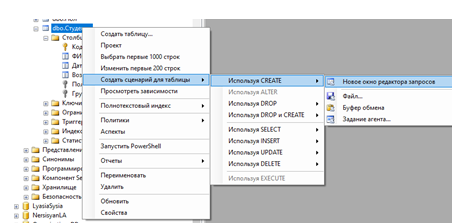


Рисунок 34 - Иерархия команд контекстного меню для работы с таблицами.

Команда меню Script table as позволяет создавать шаблоны команд для создания, удаления, вставки кортежей и для выполнения других действий с таблицами.

Созданные команды можно сохранять во внешних файлах, с расширением SQL

**Задание**

* Создать команды на языке SQL для создания таблиц, аналогичных полученным в результате выполнения практической работы № 3 и заполнение этих таблиц данными. Для создания команд можно воспользоваться шаблонами, получаемыми с помощью контекстного меню. Работу выполняете в Microsoft SQL Server Express 2008, 2012 или более новой версии.

**Порядок выполнения**

1. Создать запросы для создания таблиц с использованием контекстного меню соответствующей таблицы. Скорректировать запросы, изменив имя таблицы (можно добавить в начале или в окончании названия таблицы слово New или Create или \_). Например, таблица Товары, созданная с помощью конструктора и новая таблица ТоварыNew.
2. Создать запросы для вставки в таблицы значений. Использовать контекстное меню для создания запросов.
3. Скорректировать запросы заполнения, так чтобы в таблицы вводились тестовые значения, введенные в существующих таблицах.
4. Сформировать два файла запросов, первый содержит команды по созданию таблиц, второй по заполнению.
5. Создать файл, содержащий команды по удалению созданных таблиц.
6. Выполнить запросы создания и заполнения. Выполнить запросы по удалению.
7. Подготовить отчет о выполненной работе.

# Список литературы

1. Информационные технологии и системы: Учебное пособие [Электронный ресурс]/ Е. Л. Федотова. - М.: ИД ФОРУМ: НИЦ ИНФРА-М, 2014. - 352 с. - Режим доступа: <http://znanium.com/bookread2.php?book=429113>
2. Microsoft SQL Server 2017 [Электронный ресурс].- Режим доступа – Свободный.- URL: <https://www.microsoft.com/ru-RU/download/details.aspx?id=55994> .- Дата обращения: 20.12.2021.

Составитель

**Абрамович Александр Сергеевич**

**Методические указания   
для выполнения практической работы**   
по дисциплине «Информационные технологии» по теме  
«Реляционные базы данных»  
для студентов очной формы обучения направления подготовки 09.03.02 «Информационные системы и технологии», профиль – 01 «Системная интеграция и автоматизация информационных процессов»

Сверстано и отпечатано в филиале КузГТУ в г. Прокопьевске.

653039, г. Прокопьевск, ул. Ноградская, 19а.

Формат 60×84 1/16. Объем X п. л. Заказ X.