

Министерство образования Республики Беларусь

Учреждение образования
«Гомельский государственный технический
университет имени П. О. Сухого»

Кафедра «Информатика»

В. О. Лукьяненко

**МОДЕЛИ ДАННЫХ
И СИСТЕМЫ УПРАВЛЕНИЯ
БАЗАМИ ДАННЫХ**

ПРАКТИКУМ

**по выполнению лабораторных работ
по одноименной дисциплине для студентов
специальности 1-40 04 01 «Информатика
и технологии программирования»
дневной формы обучения**

Электронный аналог печатного издания

Гомель 2019

УДК 004.65(075.8)
ББК 32.973-018.2я73
Л84

*Рекомендовано к изданию научно-методическим советом
факультета автоматизированных и информационных систем
ГГТУ им. П. О. Сухого
(протокол № 5 от 05.03.2018 г.)*

Рецензент: доц. каф. «Информационные технологии» ГГТУ им. П. О. Сухого
канд. техн. наук, доц. *В. В. Комраков*

- Лукьяненко, В. О.**
С12 Модели данных и системы управления базами данных : практикум по выполнению лаборатор. работ по одноим. дисциплине для студентов специальности 1-40 04 01 «Информатика и технологии программирования» днев. формы обучения / В. О. Лукьяненко. – Гомель : ГГТУ им. П. О. Сухого, 2019. – 56 с. – Систем. требования: PC не ниже Intel Celeron 300 МГц ; 32 Mb RAM ; свободное место на HDD 16 Mb ; Windows 98 и выше ; Adobe Acrobat Reader. – Режим доступа: <https://elib.gstu.by>. – Загл. с титул. экрана.

ISBN 978-985-535-400-1.

Цель практикума – помочь освоить основные понятия моделей данных, баз данных и систем управления базами данных. Правильное и последовательное выполнение всех заданий позволит по окончании курса овладеть основными принципами и правилами создания баз данных.

Для студентов специальности 1-40 04 01 «Информатика и технологии программирования» дневной формы обучения.

УДК 004.65(075.8)
ББК 32.973-018.2я73

ISBN 978-985-535-400-1

© Лукьяненко В. О., 2019
© Учреждение образования «Гомельский
государственный технический университет
имени П. О. Сухого», 2019

Содержание

Лабораторная работа № 1. Реляционная алгебра	4
Лабораторная работа № 2. ER-модели.....	17
Лабораторная работа № 3. Создание базы данных	26
Лабораторная работа № 4. Заполнение таблиц базы данных.....	43
Лабораторная работа № 5. Создание запросов.....	46

Лабораторная работа № 1

Реляционная алгебра

Цель работы: изучение основ реляционной алгебры.

Теоретические сведения

Основными понятиями реляционной алгебры являются понятия отношения, кортежа, атрибута, домена.

Не строго данные понятия можно определить следующим образом.

- **Отношение** – это таблица, состоящая из строк и столбцов и имеющая вверху строку, называемую *заголовком отношения*.

- Строки таблицы-отношения называются *кортежами*, а столбцы – *атрибутами*.

- Каждый атрибут в отношении имеет *наименование*, которое указывается в заголовочной части отношения.

- **Домен отношения** – это множество значений, из которого могут браться значения конкретного атрибута. То есть конкретный набор значений атрибута в любой момент времени должен быть подмножеством множества значений домена, на котором определен этот атрибут. Значения атрибута, которые отсутствуют в множестве, задаваемом доменом, являются недопустимыми.

В основе реляционной алгебры лежит идея о том, что поскольку отношение – это множество кортежей, то и средства манипулирования отношениями должны быть такими же, как традиционные теоретико-множественные операции, дополненные специфическими для баз данных операциями.

Основные операторы реляционной алгебры можно разделить на две группы – теоретико-множественные операции и специальные реляционные операции.

Приведем определения теоретико-множественных операций:

- **объединение двух совместных по типу отношений A и B** ($A \cup B$) – это отношение с тем же заголовком, как и в отношениях A и B , и с телом, состоящим из множества всех кортежей t , принадлежащих A и B ;

- **пересечение совместимых по типу отношений A и B** ($A \cap B$) – это отношение с тем же заголовком, как и в отношениях A и B , и с телом, состоящим из множества всех кортежей t , принадлежащих одновременно обоим отношениям A и B ;

– **разность двух совместимых по типу отношений** A и B (A MINUS B) – это отношение с тем же заголовком, как и в отношениях A и B , и с телом, состоящим из множества всех кортежей t , принадлежащих отношению A и не принадлежащих отношению B ;

– **декартово произведение двух отношений** A и B (A TIMES B) – это отношение, заголовком которого является сцепление заголовков отношений A и B , т. е. множество $\{A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_k\}$, при этом требуется, чтобы отношения A и B не имели атрибутов с одинаковыми именами.

Приведем определения специальных реляционных операций:

– **проекция отношения** R по атрибутам X, Y, \dots, Z , обозначается $R[X, Y, \dots, Z]$, где каждый из указанных атрибутов принадлежит отношению R – это отношение с заголовком $\{X, Y, \dots, Z\}$ и телом, содержащим множество всех кортежей $\{ \langle X : x \rangle, \langle Y : y \rangle, \dots, \langle Z : z \rangle \}$ таких, для которых в отношении R значение атрибута X равно x , атрибута Y равно y , ..., атрибута Z равно z ;

– **выборка из отношения** $R[A]$, т. е. RESTRICT $R[A]$ WHERE $F(A)$, – это отношение, заголовок которого совпадает с заголовком отношения R , а тело содержит множество кортежей, являющихся подмножеством множества кортежей отношения R , для которых формула $F(A)$ принимает значение «истина»;

– **естественное соединение отношений** $R1$ и $R2$ ($R1$ JOIN $R2$) по атрибуту Y (возможно составному) – это отношение с заголовком $\{X, Y, Z\}$ и телом, содержащим множество всех кортежей $\{ \langle X : x \rangle, \langle Y : y \rangle, \langle Z : z \rangle \}$ таких, для которых в отношении $R1$ значение атрибута X равно x , а для атрибута Y равно y , в отношении $R2$ значение атрибута Y равно y , а атрибута Z равно z ;

– **операция деления отношения** $R1$ на отношение $R2$, обозначаемая $R1$ DIVIDE BY $R2$, где $R1$ – делимое, а $R2$ – делитель, – это отношение с заголовком $\{X\}$ и телом, содержащим множество всех кортежей

$\{ \langle X : x \rangle \}$ таких, что существует набор кортежей $\{ \langle X : x \rangle, \langle Y : y \rangle \}$, принадлежащий отношению $R1$, для всех кортежей $\{ \langle Y : y \rangle \}$, принадлежащих отношению $R2$.

Поскольку результатом любой реляционной операции является некоторое отношение, можно образовывать реляционные выражения, в которых вместо отношения-операнда некоторой реляционной операции находится вложенное реляционное выражение.

Операции объединения, пересечения взятия разности (вычитания) требуют от отношений-операндов совместимых по типу. Два от-

ношения называются совместимыми по типу, если каждое из них имеет одинаковое множество имен атрибутов и если соответствующие атрибуты определены на одном и том же домене.

Пример. Даны три отношения.

ПОСТАВЩИК

П_НОМ	П_ФАМ	СТАТУС	ГОРОД
П1	Иванов	20	Гомель
П2	Петров	15	Могилев
П3	Сидоров	10	Могилев
П4	Зайцев	30	Гомель
П5	Волков	20	Киев

ДЕТАЛЬ

Д_НОМ	Д_НАЗВ	ЦВЕТ	ВЕС
Д1	Звездочка	Красный	12
Д2	Шкив	Зеленый	17
Д3	Вал	Синий	17
Д4	Вал	Красный	14
Д5	Шестерня	Синий	12
Д6	Рычаг	Красный	19

ПД

П_НОМ	Д_НОМ	ШТ
П1	Д1	300
П1	Д2	200
П1	Д3	400
П1	Д4	200
П1	Д5	100
П1	Д6	100
П2	Д1	300
П2	Д2	400
П3	Д3	200
П4	Д2	200
П4	Д4	300
П4	Д5	400

Необходимо построить выражения реляционной алгебры, которые позволят получить следующую информацию: получить имена поставщиков, которые поставляют деталь Д2; получить имена поставщиков, которые поставляют по крайней мере одну красную деталь; получить имена поставщиков, которые поставляют все детали; получить номера поставщиков, поставляющих по крайней мере все те детали, которые поставяет поставщик П2; получить имена поставщиков, которые не поставляют деталь Д2.

Приведем решение каждой задачи:

1. Получить имена поставщиков, которые поставляют деталь Д2.
T1:= ПД JOIN ПОСТАВЩИК

T1

П_НОМ	Д_НОМ	ШТ	П_ФАМ	СТАТУС	ГОРОД
П1	Д1	300	Иванов	20	Гомель
П1	Д2	200	Иванов	20	Гомель
П1	Д3	400	Иванов	20	Гомель
П1	Д4	200	Иванов	20	Гомель
П1	Д5	100	Иванов	20	Гомель
П1	Д6	100	Иванов	20	Гомель
П2	Д1	300	Петров	15	Могилев
П2	Д2	400	Петров	15	Могилев
П3	Д3	200	Сидоров	10	Могилев
П4	Д2	200	Зайцев	30	Гомель
П4	Д4	300	Зайцев	30	Гомель
П4	Д5	400	Зайцев	30	Гомель

T2:= T1 WHERE Д_НОМ = 'Д2'

T2

П_НОМ	Д_НОМ	ШТ	П_ФАМ	СТАТУС	ГОРОД
П1	Д2	200	Иванов	20	Гомель
П2	Д2	400	Петров	15	Могилев
П4	Д2	200	Зайцев	30	Гомель

T3= T2[П_ФАМ]

T3

П_ФАМ
Иванов
Петров
Зайцев

Искомое выражение выглядит следующим образом:

((ПД JOIN ПОСТАВЩИК) WHERE Д_НОМ = 'Д2') [П_ФАМ]

2. Получить имена поставщиков, которые поставляют по крайней мере одну красную деталь.

**((ДЕТАЛЬ WHERE ЦВЕТ = 'Красный') JOIN ПД) [П_НОМ]
JOIN ПОСТАВЩИК) [П_ФАМ]**

3. Получить имена поставщиков, которые поставляют все детали.

ПД[П_НОМ, Д_НОМ] **DIVIDE BY** ДЕТАЛЬ[Д_НОМ]

4. Получить номера поставщиков, поставляющих по крайней мере все те детали, которые поставяет поставщик П2.

ПД[П_НОМ, Д_НОМ] **DIVIDE BY** (ПД **WHERE** П_НОМ = 'П2')[Д_НОМ]

5. Получить имена поставщиков, которые не поставяют деталь Д2.

((ПОСТАВЩИК[П_НОМ] **MINUS** (ПД **WHERE** Д_НОМ = 'Д2')[П_НОМ]) **JOIN** ПОСТАВЩИК)[П_ФАМ]

Задание

Общее условие для вариантов 1–8

Предметная область – агентство недвижимости. Даны четыре отношения.

ВЛАДЕЛЕЦ

Номер_Влад	Владелец	Улица	Дом	Квартира	Телефон
01	И. Н. Иванов	Комсомольская	165	56	42-26-58
02	П. В. Антонов	Ленина	3а	34	25-78-96
03	ООО «Пилот»	Победы	56	4	73-24-15
04	ЗАО «Метеор»	Горького	22		55-66-59

- Номер_Влад – номер владельца недвижимости.
- Владелец – Ф. И. О. либо наименование владельца.
- Улица, Дом, Квартира – место жительства либо регистрации владельца.
- Телефон – контактные данные владельца.

ОБЪЕКТ

Номер_Об	Тип	Улица	Дом	Квар-тира	Кол_Ком	Стои-мость	Номер_Влад
105	Дом	Железнодорожная	76		3	200	01
107	Дом	Комсомольская	5		3	185	02
108	Дом	Кирпичная	16		2	150	03
15	Коттедж	Комсомольская	34		2	450	04
14	Квартира	Мира	22	17	1	120	01
16	Квартира	Мира	11	122	1	200	04
17	Квартира	Ленина	5	45	1	200	02
18	Квартира	Лермонтова	13	25	3	600	02
19	Квартира	Чехова	23	68	2	400	01
20	Квартира	Горького	35	34	1	250	03
110	Дом	Трудовая	15		2	100	03
111	Дом	Трудовая	26		2	180	01
112	Дом	Лермонтова	67		3	300	02

- Номер_Об – номер объекта недвижимости.
- Улица, Дом, Квартира – место расположения объекта недвижимости.
- Тип – дом, квартира, коттедж или другой вид жилья.
- Кол_Ком – количество комнат.
- Аренда – арендная плата за месяц.
- НомерВлад – номер владельца, которому принадлежит объект.

КЛИЕНТ

Ном_Клиента	ФИО	Улица	Дом	Квар-тира	Телефон	Предп_Тип	Макс_Аренда
13-03	В. И. Петров	Украинская	69	67	73-15-69	Квартира	120
17-02	Н. А. Васильев	Бумажная	24		55-68-96	Квартира	250
01-02	Д. Б. Ветров	Мира	13	19	71-11-87	Дом	150
22-03	Н. Н. Долгова	Мира	58	169	42-63-78	Квартира	100
27-02	Е. И. Перова	Ленина	56	124	73-56-42	Квартира	200
17-03	Н. А. Антонова	Бумажная	25		42-12-45	Квартира	250
18-02	О. Б. Степанов	Мира	95	345	25-32-12	квартира	200

- Ном_Клиента – номер клиента.
- Ф. И. О. – имя клиента.
- Улица, Дом, Квартира – место жительства клиента.
- Телефон – контактные данные клиента.
- Предп_Тип – пожелание клиента относительно количества комнат.
- Макс_Аренда – пожелания клиента по арендной плате за месяц.

ДОГОВОР

Ном_Дог	Дата_нач	Дата_кон	Номер_Об	Ном_Клиента	Комиссия
Д-01-2017-1	17.01.2017	25.04.2017	14	13-03	10
Д-01-2017-2	18.01.2017	18.07.2017	108	17-03	20
Д-01-2017-3	25.01.2017	25.04.2017	107	27-02	15
Д-04-2017-15	26.04.2017	26.07.2017	14	22-03	12
Д-02-2017-4	02.02.2017	02.02.2018	110	18-02	20
Д-02-2017-5	05.02.2017	05.09.2019	111	18-02	22
Д-04-2017-16	25.04.2017	25.10.2018	107	27-02	20
Д-02-2017-6	16.02.2017	16.09.2017	20	17-03	34
Д-02-2017-7	21.02.2017	21.05.2017	16	17-02	12
Д-02-2017-8	29.02.2017	29.08.2018	17	17-03	14
Д-03-2017-9	01.03.2017	01.09.2018	105	27-02	14
Д-03-2017-10	03.03.2017	03.06.2017	112	17-02	12

- Ном_Дог – номер договора.
- Дата_нач – дата подписания договора.
- Дата_кон – срок окончания действия договора.
- Номер_Об – номер объекта недвижимости.
- Номер_Клиента – номер клиента.
- Комиссия – плата агентству за оформление сделки.

Вариант 1

1. Составить список всех домов.
2. Получить список клиентов, арендующих объект с номером 14.
3. Определить тех клиентов, которые арендуют такие же по типу объекты недвижимости, что и клиент Н. Н. Долгова.
4. Определить адреса квартир, при заключении договоров по которым комиссионные составили больше 15 рублей.

5. Получить номера объектов, арендная плата по которым составляет не более 150 рублей или которые расположены на той же улице, где живет клиент Г. Б. Ветров.

Вариант 2

1. Перечислить все 1-комнатные квартиры, арендная плата по которым меньше 200 рублей.

2. Выбрать адреса домов, которые бы устроили клиента Г. Б. Ветрова.

3. Составить список номеров объектов, по которым не было заключено ни одной сделки.

4. Выбрать информацию по договорам, заключенным с клиентом Н. А. Антоновой.

5. Вывести всю информацию по договорам, включая имя клиента, адрес, тип жилья.

Вариант 3

1. Перечислить имена и адреса клиентов.

2. Составить список владельцев, заключивших договора в марте 2017 г.

3. Определить владельцев, которые предлагают для аренды такие же объекты, как и владелец с номером 01.

4. Выбрать объекты, расположенные на той же улице, где живут их владельцы.

5. Вывести список типов жилья с указанием их арендной стоимости.

Вариант 4

1. Перечислить номера объектов, арендованных до 01.01.2018.

2. Составить список владельцев, которые являются собственниками домов.

3. Вывести список владельцев, являющихся собственниками и дома и квартиры.

4. Вывести список клиентов и объектов, находящихся на одной улице.

5. Определить владельца объекта, для которого заключен договор с размером комиссионных 14 рублей.

Вариант 5

1. Составить список адресов объектов недвижимости, находящихся в собственности владельца с номером 03.
2. Выбрать фамилии людей, которым подойдет жилье по адресу: ул. Мира, д. 22, кв. 17.
3. Получить номера объектов недвижимости, размер арендной платы которых равен максимально возможной плате клиента с номером 18-02.
4. Определить имена владельцев, заключивших хотя бы одну сделку.
5. Вывести номера объектов, принадлежащих владельцу И. Н. Иванову, размер платы по которым не меньше 200 рублей.

Вариант 6

1. Составить список всех квартир.
2. Выбрать адреса домов, которые бы устроили клиента О. Б. Степанова.
3. Определить владельцев, которые предлагают для аренды такие же объекты, как и владелец с номером 02.
4. Вывести список клиентов и объектов, находящихся на одной улице.
5. Вывести номера объектов, принадлежащих владельцу И. Н. Иванову, размер платы по которым не больше 150 рублей.

Вариант 7

1. Получить список клиентов, арендующих объект с номером 107.
2. Составить список номеров объектов, по которым была заключена хотя бы одна сделка.
3. Выбрать объекты, расположенные на той же улице, где живут их владельцы.
4. Вывести номера объектов, принадлежащих владельцу И. Н. Иванову, размер платы по которым не больше 160 рублей.
5. Определить имена владельцев, не заключивших ни одну сделку.

Вариант 8

1. Определить тех клиентов, которые арендуют такие же по типу объекты недвижимости, что и клиент Д. Б. Ветров.
2. Выбрать информацию по договорам, заключенным с клиентом Н. А. Васильев.

3. Вывести список типов жилья с указанием их арендной стоимости.
4. Перечислить номера объектов, арендованных до 01.05.2017.
5. Выбрать фамилии людей, которым подойдет жилье по адресу: ул. Трудовая, д. 15.

Общее условие для вариантов 9–15

Предметная область – производственные предприятия. Даны четыре отношения.

ПРЕДПРИЯТИЕ

Номер_предприятия	Название	Рейтинг	Город
180	Электроника	230	Гомель
230	Техномир	300	Москва
150	Бытовая техника	140	Гомель
190	Мастер	300	Минск
270	Строймаркет	240	Москва

- Номер_предприятия – номер предприятия.
- Название – название предприятия.
- Рейтинг – рейтинг предприятия по некоторым показателям.
- Город – город, в котором находится предприятие.

ПРОДУКЦИЯ

Номер_продукции	Наименование	Стоимость	Город_выпуска
10	Микроволновая печь	400	Гомель
20	Посудомойка	1000	Москва
30	Сушилка	650	Гомель
40	Холодильник	800	Могилев
50	Стиральная машина	700	Гомель

- Номер_продукции – номер продукции.
- Наименование – наименование продукции.
- Стоимость – стоимость продукции, выпускаемой в год в данном городе.

- Город_выпуска – город, в котором указанная продукция выпускается.

РАБОТНИК

Номер_работника	Фамилия	Город	Дата_рождения	Номер_предприятия
55	Иванов	Гомель	15.03.98	180
10	Петров	Москва	17.02.95	230
100	Сидоров	Гомель	03.12.93	150
190	Иванов	Минск	18.04.91	190
130	Антонов	Могилев	14.03.94	190

- Номер_работника – номер работника.
- Фамилия – фамилия работника.
- Город – город проживания работника.
- Дата_рождения – дата рождения работника.
- Номер_предприятия – номер предприятия, где числится работник.

ПРЕД_ПРОД

Номер_предприятия	Номер_продукции	Год	Количество
150	30	2016	150
180	10	2016	100
190	50	2015	50
270	50	2016	120
270	20	2016	50
180	20	2017	50
270	40	2017	200
150	10	2016	100
270	30	2017	130
270	10	2017	70

- Номер_предприятия – номер предприятия.
- Номер_продукции – номер продукции.
- Год – год выпуска продукции.
- Количество – количество произведенной продукции.

Вариант 9

1. Получить названия предприятий, производящих продукцию с номером 30.
2. Выбрать информацию обо всех предприятиях, в том числе о работниках и продукции.
3. Выбрать фамилии людей, которые работают на предприятии «Мастер».
4. Определить номера предприятий из Гомеля с рейтингом выше 200.
5. Выбрать названия предприятий, производящих все виды продукции.

Вариант 10

1. Получить имена предприятий, производящих продукцию только одного вида.
2. Выбрать название продукции, стоимость которой находится в диапазоне от 600 до 900 рублей.
3. Выбрать фамилии людей, у которых город проживания совпадает с городом нахождения предприятия.
4. Найти номера работников, работающих на одном предприятии.
5. Определить название предприятий, которые не производят продукцию с номером 50.

Вариант 11

1. Получить номера предприятий, производящих по крайней мере ту продукцию, которую выпускает предприятие с номером 190.
2. Выбрать название предприятий, у которых выработка продукции в 2016 г. составила более 100.
3. Определить фамилии людей, работающих на предприятиях Гомеля.
4. Определить имена предприятий, производящих продукцию с номером 10.
5. Определить номера предприятий, производящих по крайней мере один из видов продукции, производимых предприятием с номером 270.

Вариант 12

1. Выбрать все пары названий городов, для которых предприятие и работники находятся в одном городе.
2. Выбрать название предприятий, которые производят сушилки.
3. Определить название предприятий, производящих продукцию с номером 10 в 2016 г.

4. Определить номера предприятий, имеющих в списке работающих по крайней мере одного Иванова.

5. Получить номера продукции, которая была произведена в год более чем 100 или производится предприятием с номером 190.

Вариант 13

1. Получить названия предприятий, не производящих продукцию с номером 50.

2. Определить номера предприятий, не имеющих в списке работающих человека по фамилии Иванов.

3. Определить фамилии людей, не работающих на предприятиях Гомеля.

4. Найти названия предприятий, производящих по крайней мере одну продукцию с номером 50.

5. Найти названия предприятий, выпускающих продукцию гомельских предприятий.

Вариант 14

1. Определить номера предприятий из Гомеля с рейтингом, не равным 230.

2. Определить название предприятий, которые производят продукцию с номером 30.

3. Выбрать название предприятий, у которых выработка продукции в 2017 г. составила менее 100.

4. Определить фамилии людей, работающих на предприятиях Минска.

5. Выбрать название предприятий, которые не производят холодильники.

Вариант 15

1. Выбрать название продукции, стоимость которой не находится в диапазоне от 700 до 900 рублей.

2. Определить, какие виды продукции производили предприятия «Электроника» и «Техномир».

3. Выбрать название предприятий, которые производят холодильники или стиральные машины.

4. Выбрать название предприятий, которые производили продукцию только в 2016 г.

5. Определить номера предприятий с рейтингом больше 200.

Лабораторная работа № 2

ER-модели

Цель работы: научиться использовать модель «сущность-связь» (ER-модель) для анализа структуры данных предметной области.

Теоретические сведения

Модель «сущность-связь» (*ER-модель*) (англ. entity-relationship model (ERM) или англ. entity-relationship diagram (ERD)) – модель данных, позволяющая описывать концептуальные схемы. Предоставляет собой графическую нотацию, основанную на блоках и соединяющих их линиях, с помощью которых можно описывать объекты и отношения между ними какой-либо другой модели данных. В этом смысле ER-модель является средством описания моделей данных. ER-модель удобна при проектировании информационных систем, баз данных, архитектур компьютерных приложений и других систем. ER-диаграмма содержит информацию о сущностях системы и способах их взаимодействия, включает идентификацию объектов, важных для предметной области (сущностей), свойств этих объектов (атрибутов) и их отношений с другими объектами (связей).

Существует несколько способов визуального отображения (нотаций) ER-модели. Наиболее часто употребляемые из них:

- нотация Питера Чена (первоначальная);
- нотация «воронья лапка» (crow's foot);
- Unified Modeling Language (UML, унифицированный язык моделирования).

Ниже приведены элементы ER-диаграммы в упрощенном варианте нотации «воронья лапка».

Сущность – это класс однотипных объектов, информация о которых должна быть учтена в модели.

Каждая сущность должна иметь наименование, выраженное существительным в единственном числе.

Пример. В качестве сущностей могут выступать такие классы объектов, как «Поставщик», «Сотрудник», «Накладная».

Каждая сущность в модели изображается в виде прямоугольника с наименованием (рис. 2.1).

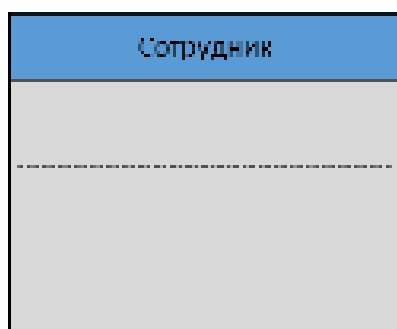


Рис. 2.1

Экземпляр сущности – это конкретный представитель данной сущности.

Экземпляры сущностей должны быть различимы, т. е. сущности должны иметь некоторые свойства, уникальные для каждого экземпляра этой сущности.

Атрибут сущности – это именованная характеристика, являющаяся некоторым свойством сущности.

Наименование атрибута должно быть выражено существительным в единственном числе (возможно, с характеризующими прилагательными).

Примерами атрибутов сущности «Сотрудник» могут быть такие атрибуты, как «Табельный номер», «Фамилия», «Имя», «Отчество», «Должность», «Зарплата» и т. п.

Атрибуты изображаются в пределах прямоугольника, определяющего сущность (рис. 2.2).

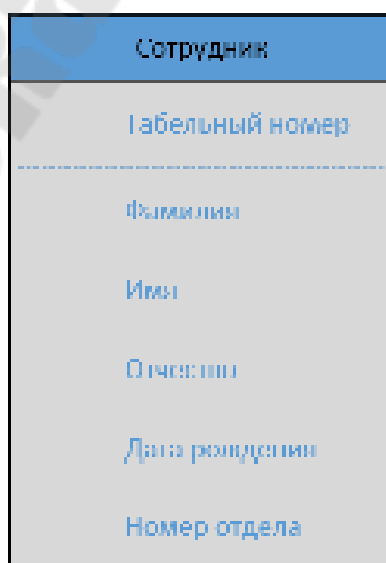


Рис. 2.2

Ключ сущности – это избыточный набор атрибутов, значения которых в совокупности являются уникальными для каждого экземпляра сущности. Избыточность заключается в том, что удаление любого атрибута из ключа нарушается его уникальность. Сущность может иметь несколько различных ключей.

Ключевые атрибуты обозначаются на диаграмме подчеркиванием или символом РК. Экземпляр сущности **Сотрудник** может однозначно идентифицироваться атрибутом «Табельный номер» (рис. 2.3).

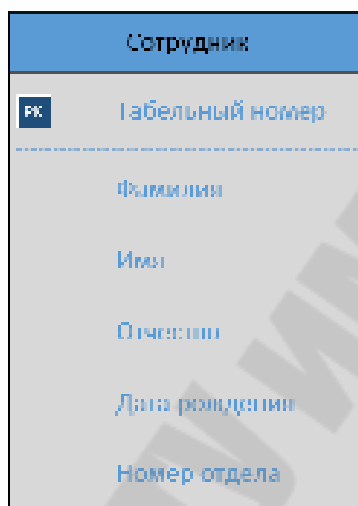


Рис. 2.3

Связь – это некоторая ассоциация между двумя сущностями. Одна сущность может быть связана с другой сущностью или сама с собою.

Связи позволяют по одной сущности находить другие сущности, связанные с ней.

Пример. Связи между сущностями могут выражаться следующими фразами: «Каждый сотрудник должен работать только в одном отделе», «в одной должности может работать несколько сотрудников».

Графически связь изображается линией, соединяющей две сущности (рис. 2.4).

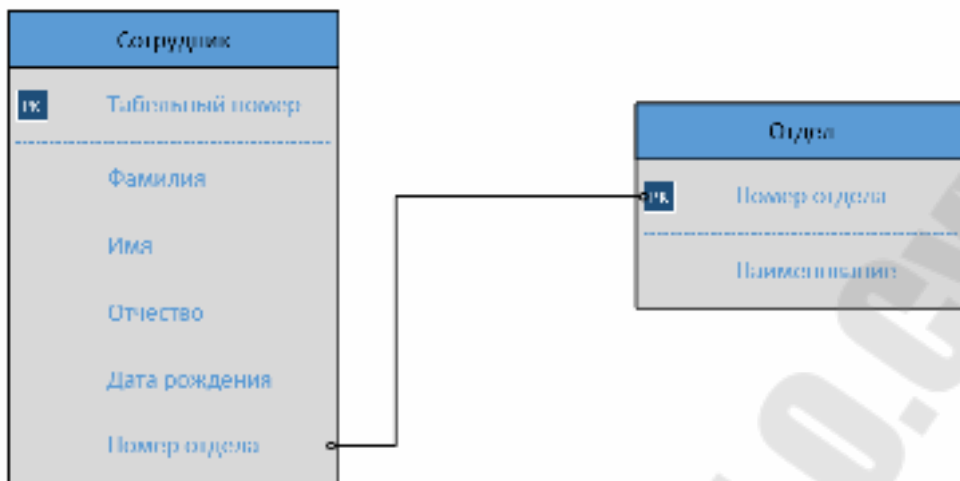


Рис. 2.4

Каждая связь имеет два конца и одно или два наименования. Наименование обычно выражается с помощью глаголов «иметь», «принадлежать» и т. п. Каждое из наименований относится к своему концу связи. Иногда наименования не пишутся ввиду их очевидности.

Каждая связь может иметь один из следующих типов связи (рис. 2.5):

- 1) один-к-одному (1 : 1);
- 2) один-ко-многим (1 : N) или многие-к-одному (N : 1);
- 3) многие-ко-многим (N : M).

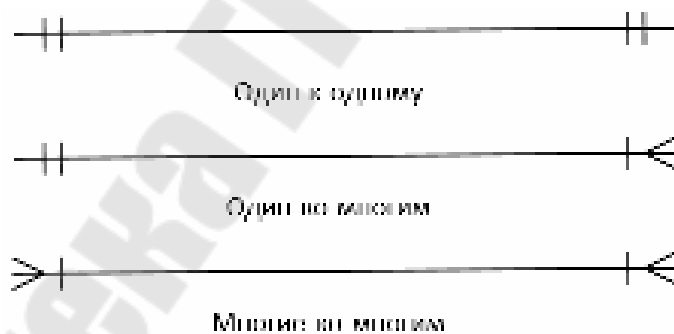


Рис. 2.5

Связь типа «**один-к-одному**» означает, что один экземпляр первой сущности (левой) связан с одним экземпляром второй сущности (правой). Связь «один-к-одному» чаще всего свидетельствует о том, что на самом деле мы имеем всего одну сущность, неправильно разделенную на две.

Связь типа «**один-ко-многим**» означает, что один экземпляр первой сущности (левой) связан с несколькими экземплярами второй сущности (правой). Это наиболее часто используемый тип связи. Ле-

вая сущность (со стороны «один») называется *родительской*, правая (со стороны «много») – *дочерней*.

Связь типа «*многие-ко-многим*» означает, что каждый экземпляр первой сущности может быть связан с несколькими экземплярами второй сущности, и каждый экземпляр второй сущности может быть связан с несколькими экземплярами первой сущности. Тип связи «многие-ко-многим» является временным типом связи, допустимым на ранних этапах разработки модели. В дальнейшем этот тип связи должен быть заменен двумя связями типа «один-ко-многим» путем создания промежуточной сущности.

Каждая связь может быть прочитана как слева направо, так и справа налево.

Связь, изображенная на рис. 2.6, читается следующим образом:

– слева направо: «каждый сотрудник должен работать только в одном отделе»;

– справа налево: «в одном отделе может числиться несколько сотрудников».

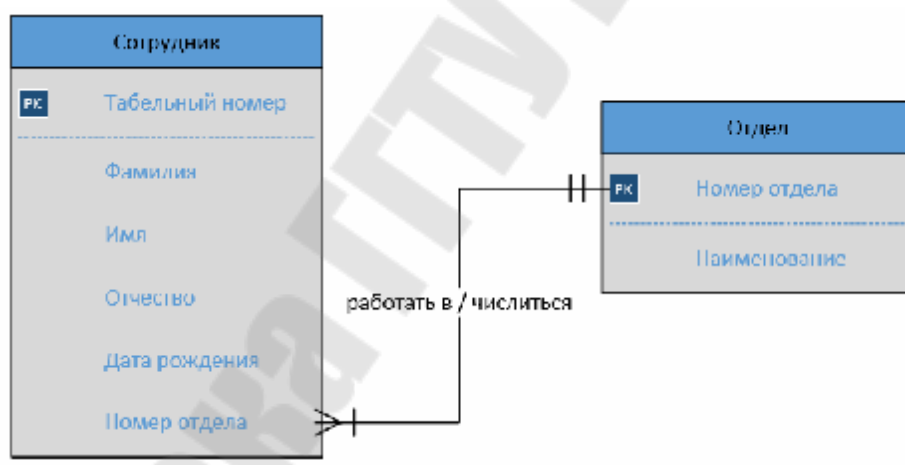


Рис. 2.6

Пример. Разработать ER-модель для предметной области «Сессия».

Проектируемая система должна выполнять следующие функции:

– хранить информацию о студентах, группах, дисциплинах, оценках;

– печатать списки групп, результаты экзаменов.

Объекты **Студент**, **Группа**, **Дисциплины** являются явными сущностями предметной области «Сессия».

Связи между этими сущностями:

– каждый студент должен обучаться в одной группе;

- группа может состоять из нескольких студентов;
- каждая студент обучается по нескольким дисциплинам;
- одна дисциплина может изучаться несколькими студентами.

На первом этапе ER-диаграмма будет иметь вид, представленный на рис 2.7.

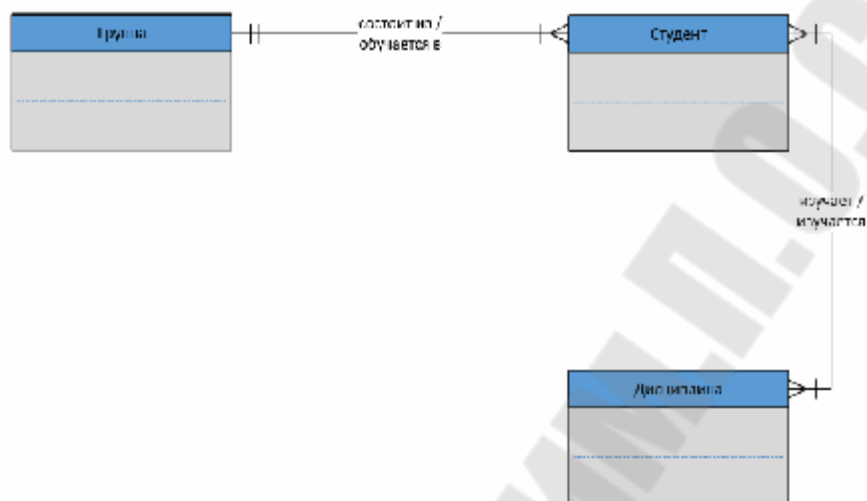


Рис. 2.7

Атрибутами сущности **Студент** могут быть:

- фамилия;
- имя;
- отчество;
- дата рождения;
- номер телефона;
- адрес.

Атрибутами сущности **Группа** могут быть:

- название группы;
- форма обучения (дневная или заочная).

Атрибутом сущности **Дисциплина** может быть:

- название.

Объект **Оценки** не может быть атрибутом никакой из имеющихся сущностей и, скорее всего, представляет собой отдельную сущность. Для этой сущности возникают следующие связи:

- каждый студент может получить несколько оценок;
- каждая оценка должна быть поставлена одному студенту;
- каждый экзамен по дисциплине может оценивать несколько оценок;
- каждая оценка должна быть поставлена за один экзамен.

Атрибутом сущности **Оценка** может быть:
– значение.

На втором этапе ER-диаграмма будет иметь вид, представленный на рис. 2.8.

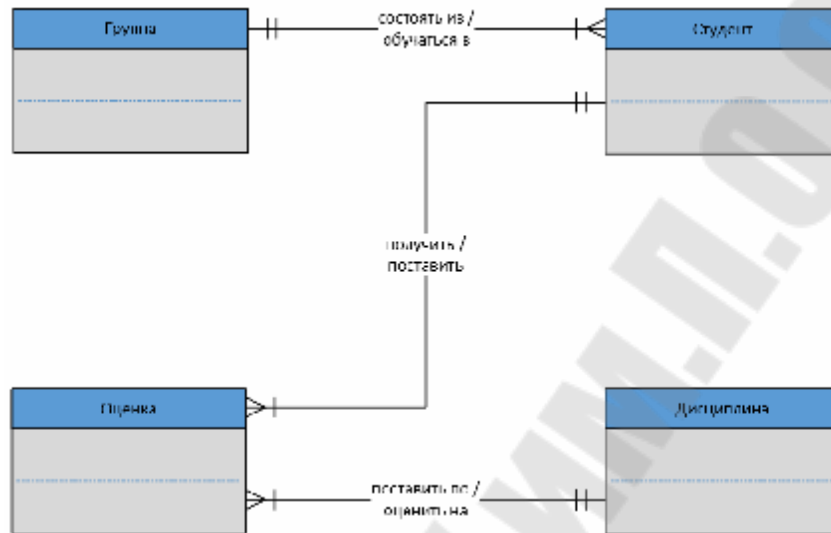


Рис. 2.8

В итоге получаем концептуальную ER-диаграмму (рис. 2.9).

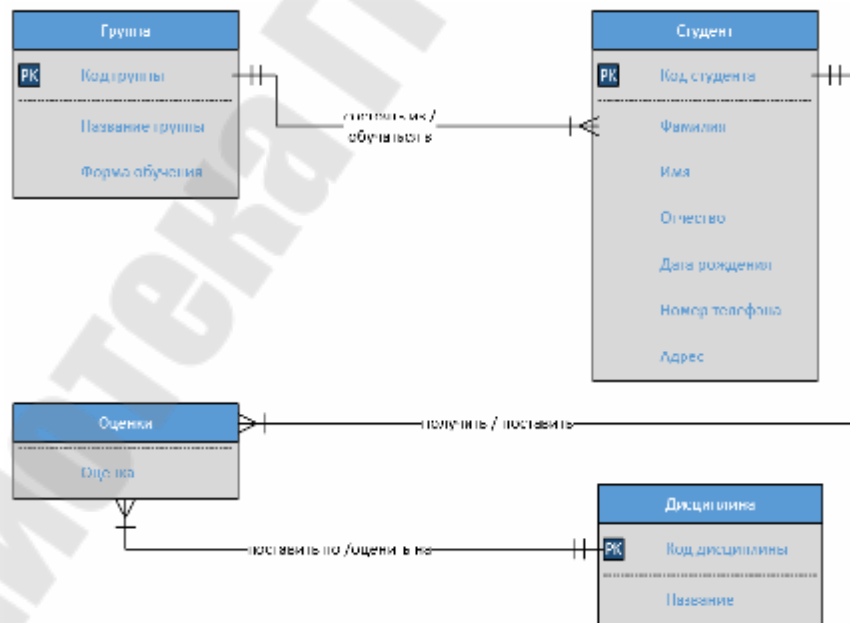


Рис. 2.9

Отметим, что на этапе проектирования в ER-модель могли быть включены такие сущности, как **Факультет, Кафедра, Преподаватель, Специальность** и др.

Контрольные вопросы

1. Опишите область применения ER-моделей.
2. Какие основные элементы используются при построении ER-модели?
3. Дайте определение сущности.
4. Приведите примеры сущностей для разных предметных областей.
5. Дайте определение атрибута. Приведите примеры.
6. Дайте определение связи.
7. Перечислите основные свойства связей.
8. Какие типы связей Вам известны.

Задание

Разработать ER-модель по своему варианту в среде Microsoft Office Visio (вариант выбирается согласно номеру студента по журналу). Спроектированная ER-модель должна состоять не менее чем из пяти сущностей.

Отчет по лабораторной работе должен содержать:

- титульный лист;
- цель работы;
- задание согласно своему варианту;
- словесное описание сущностей и их атрибутов с описанием и указанием ключевых атрибутов;
- перечисление, описание и обоснование связей;
- концептуальную ER-диаграмму;
- выводы.

Вариант	Предметная область
1	Ресторан
2	Банк
3	Больница
4	Гостиница

Окончание табл.

Вариант	Предметная область
5	Аэропорт
6	Видеопрокат
7	Библиотека
8	Радиостанция
9	Таксопарк
10	Туристическое агентство
11	Страховая компания
12	Брачное агентство
13	Сервисный центр
14	Школа
15	Транспортная компания

Лабораторная работа № 3

Создание базы данных

Цель работы: научиться создавать базы данных в среде SQL Server Management Studio.

Теоретические сведения

В СУБД MS SQL Server Management Studio новую базу данных (БД) можно создать, используя стандартные команды языка SQL. Для того чтобы создать новый запрос на панели инструментов, необходимо нажать кнопку **Создать запрос**. Для выполнения команд языка T-SQL на панели инструментов необходимо нажать кнопку **Выполнить** или на вкладке нового запроса набрать команду GO.

В MS SQL Server БД состоит из двух частей:

- Файл данных – файл с расширением mdf, в котором находятся все таблицы и запросы.
- Файл журнала транзакций – файл с расширением ldf, который содержит журнал, где фиксируются все действия с БД. Данный файл предназначен для восстановления БД в случае ее выхода из строя.

Для создания нового файла данных используется SQL-команда CREATE DATABASE, которая имеет следующий синтаксис:

```
CREATE DATABASE <Имя БД>
ON
(Name=<Логическое имя>,
FileName=<Имя файла>
[Size=<Нач. размер>,)
[Maxsize=<Макс. размер>,)
[FileGrowth=<Шаг>])
[LOG ON
(Name=<Логическое имя>,
FileName=<Имя файла>
[Size=<Нач. размер>,)
[Maxsize=<Макс. размер >,)
[FileGrowth=<Шаг>])
GO
```

Здесь:

- **Имя БД** – имя создаваемой БД.
- **Логическое имя** – определяет логическое имя файла данных БД, по которому происходит обращение к файлу данных.
- **Имя файла** – определяет полный путь к файлу данных.
- **Нач. размер** – начальный размер файла данных в МБ.
- **Макс. размер** – максимальный размер файла данных в МБ.
- **Шаг** – шаг увеличения файла данных либо в МБ, либо в процентах.

Параметры в разделе LOG ON аналогичны параметрам в разделе CREATE DATABASE. Однако они определяют параметры журнала транзакций.

Пример. Создать БД «Art», расположенную в файле «D:\Base\Art.mdf» и имеющую начальный размер файла данных – 10 МБ, максимальный размер файла данных – 50 МБ и шаг увеличения файла данных, равный 5 МБ. Файл журнала транзакций данной БД имеет имя «ArtLog» и расположен в файле «D:\Base\Art.ldf». Данный файл имеет начальный размер, равный 5 МБ, максимальный размер, равный 25 МБ, и шаг увеличения, равный 5 МБ.

```
CREATE DATABASE Art
ON
( NAME = Art,
  FILENAME = 'D:\Base\Art.mdf',
  SIZE = 10,
  MAXSIZE = 50,
  FILEGROWTH = 5 )
LOG ON
( NAME = Art_log,
  FILENAME = 'D:\Base\Art.ldf',
  SIZE = 5MB,
  MAXSIZE = 25MB,
  FILEGROWTH = 5MB ) ;
```

Добавление новых файлов, удаление файлов и переименование файлов, входящих в БД, происходит с помощью команды ALTER DATABASE, которая имеет следующий синтаксис:

```
ALTER DATABASE <Имя БД>
ADD FILE (<Параметры>)
```

REMOVE FILE <Логическое имя файла>
MODIFY FILE (<Параметры>)

где раздел **ADD FILE** – добавляет файл, **REMOVE FILE** – удаляет, а раздел **MODIFY FILE** – изменяет параметры файла.

Удаление БД позволяет выполнить команда DROP DATABASE, которая имеет следующий синтаксис:

DROP DATABASE <Имя БД>.

Вышеперечисленные команды используют следующие параметры:

- **Имя БД** – имя БД, с которой производится действие;
- **Параметры** – параметры файла БД, аналогичные параметрам, используемым в команде CREATE DATABASE;
- **Логическое имя файла** – логическое имя файла, входящего в БД.

После запуска среды разработки появится окно подключения к серверу «Соединение с сервером» (рис. 3.1).

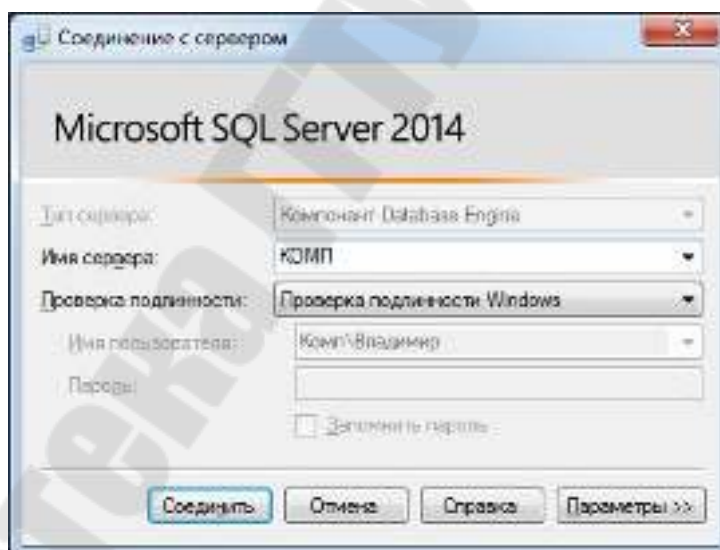


Рис. 3.1

В этом окне необходимо выбрать имя сервера и нажать кнопку «Соединить», после чего появится окно среды разработки «SQL Server Management Studio» (рис. 3.2).

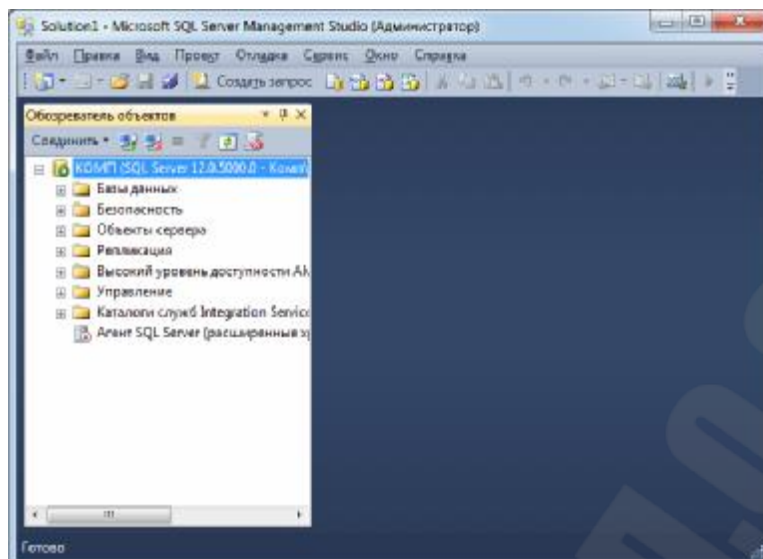


Рис. 3.2

Данное окно имеет следующую структуру:

1. Оконное меню – содержит полный набор команд для управления сервером и выполнения различных операций.

2. Панель инструментов – содержит кнопки для выполнения наиболее часто производимых операций. Внешний вид данной панели зависит от выполняемой операции.

3. Панель «Обозреватель объектов» – это панель с древовидной структурой, отображающая все объекты сервера, а также позволяющая производить различные операции как с самим сервером, так и с БД. Обозреватель объектов является основным инструментом для разработки БД.

4. Рабочая область, в которой производятся все действия с БД, а также отображается ее содержимое.

В обозревателе объектов сами объекты находятся в папках. Чтобы открыть папку необходимо щелкнуть по знаку «+» слева от изображения папки.

Нажатие кнопки «Создать запрос» приводит к открытию окна запросов.

Вся информация в БД хранится в таблицах, которые представляют собой средство хранения данных. Таблицы состоят из строк или записей и столбцов или полей. Каждое поле имеет три характеристики:

1. Имя поля – используется для обращения к полю.
2. Значение поля – определяет информацию, хранимую в поле.
3. Тип данных поля – определяет, какой вид информации можно хранить в поле.

В SQL Server используются следующие основные типы данных:

- **Целочисленные типы данных** – типы данных для хранения целых чисел (в скобках указан диапазон значений типа данных):

- tinyint (0...255),
- smallint (–32768...+32767),
- int (–231...+(231–1)),
- bigint (–263...+(263–1)).

- **Типы данных для хранения чисел с плавающей запятой:**

- real занимает в памяти 4 байта;
- float(*n*), где *n* – это количество битов, используемых для хранения мантиссы числа в формате float при экспоненциальном представлении, определяет точность данных и размер для хранения; значение параметра *n* должно лежать в пределах от 1 до 53; значением по умолчанию для параметра *n* является 53.

- **Типы данных для хранения чисел с фиксированными точностью и масштабом:** decimal(*p*, *s*) и numeric(*p*, *s*), где *p* (точность) – максимальное количество десятичных разрядов числа (как слева, так и справа от десятичной запятой). Точность должна быть значением в диапазоне от 1 до 38. По умолчанию это значение равно (18, *s*) (масштаб) – максимальное количество десятичных разрядов числа справа от десятичной запятой. Масштаб может принимать значение от 0 до *p* и может быть указан только совместно с точностью. По умолчанию масштаб принимает значение 0; поэтому $0 \leq s \leq p$.

- **Символьные типы данных:**

- char(*n*) – строковые данные фиксированной длины не в Юникоде, аргумент *n* определяет длину строки и должен иметь значение от 1 до 8000, размер при хранении составляет *n* байт;

- varchar(*n* | max) – строковые данные переменной длины не в Юникоде, аргумент *n* определяет длину строки и должен иметь значение от 1 до 8000, значение max указывает, что максимальный размер при хранении составляет 231 – 1 байт (2 ГБ);

- text – данные переменной длины не в Юникоде в кодировке сервера и с максимальной длиной строки 231–1.

- **Типы данных даты и времени:**

- date (от 01.01.0001 до 31.12.9999);
- datetime (диапазон даты – от 01.01.1753 до 31.12.1999, диапазон времени – от 00:00:00 до 23:59:59,997);
- smalldatetime (диапазон даты – от 01.01.1900 до 06.06.2079, диапазон времени – от 00:00:00 до 23:59:59);
- time (от 00:00:00.0000000 до 23:59:59.9999999).

• **Денежные типы данных для хранения финансовой информации:** money (8 байт) и smallmoney (4 байта) – типы данных, представляющие денежные (валютные) значения.

Для создания таблиц в SQL Server в первую очередь необходимо сделать активной ту БД, в которой создается таблица. Для этого в новом запросе можно набрать команду USE <Имя БД> либо на панели инструментов необходимо выбрать в выпадающем списке рабочую БД. После выбора БД можно создавать таблицы.

Таблицы создаются командой

```
CREATE TABLE table_name ( { <column_definition> } ) [ ; ]
<column_definition> ::= column_name <data_type> [ NULL | NOT
NULL ] [DEFAULT constant_expression ] | [IDENTITY [ ( seed, incre-
ment ) ] ] [ <column_constraint> [ ...n ] ]
<column_constraint> ::= [ CONSTRAINT constraint_name ]
{ PRIMARY KEY | UNIQUE }
```

Здесь:

- **table_name** – имя таблицы.
- **column_name** – имя столбца в таблице.
- **data_type** – тип данных для столбца.
- **IDENTITY** указывает, что новый столбец является столбцом идентификаторов, при этом seed – значение, используемое для самой первой строки, загружаемой в таблицу, increment – значение приращения, добавляемое к значению идентификатора предыдущей загруженной строки.

- **CONSTRAINT** – необязательное ключевое слово, указывающее на начало определения ограничения, **constraint_name** – имя ограничения.

- **NULL | NOT NULL** определяет, допустимы ли для столбца значения NULL.

- **PRIMARY KEY** – ограничение, которое определяет столбец первичным ключом таблицы.

Если имя поля содержит пробел, то оно заключается в квадратные скобки.

Пример. Создать таблицу «Artworks», содержащую поля:

- 1) код произведения (Artwork_Id);
- 2) название произведения (Title);

- 3) жанр (Genre);
- 4) средства создания (Tools);
- 5) код автора (Author_Id);
- 6) дата создания (Creat_Date);
- 7) цена (Price);
- 8) отдел хранения (Dep_Id).

SQL-запрос для создания этой таблицы имеет следующий вид:

```
USE Art;
CREATE TABLE Artworks
(Artwork_Id BIGINT IDENTITY(1,1) CONSTRAINT
PK_Artworks PRIMARY KEY,
Title VARCHAR(100) NULL,
Genre VARCHAR (50) NULL,
Tools VARCHAR (50) NULL,
Author_Id BIGINT NULL,
Creat_Date DATE NULL,
Price MONEY NULL,
Dep_Id INT NOT NULL);
```

Для каждого атрибута сущности Artworks задается его имя, тип и допустимость значений NULL.

Для столбца **Artwork_Id** определено ограничение в виде первичного ключа (**PK_Artworks**), при этом значения **Artwork_Id** будут начинаться с 1 и увеличиваться при каждом добавлении новых строк в таблицу тоже на 1 (**IDENTITY(1,1)**).

Аналогично создаются и другие таблицы БД Art: **Authors**, **Employees**, **Departments**. SQL-запросы для создания этих таблиц приведены ниже.

```
CREATE TABLE Authors
(Author_Id BIGINT IDENTITY(1,1) CONSTRAINT PK_Authors
PRIMARY KEY,
Last_name VARCHAR(25) NOT NULL,
First_name VARCHAR (25) NOT NULL,
Middle_name VARCHAR (25) NULL,
Date_Of_Birth DATE NULL,
Date_Of_Death DATE NULL,
Country VARCHAR(25) NULL);
```



```
CREATE TABLE Employees
(Emp_Id BIGINT IDENTITY(1,1) CONSTRAINT PK_Employees
PRIMARY KEY,
Last_name VARCHAR(25) NOT NULL,
First_name VARCHAR (25) NOT NULL,
Middle_name VARCHAR (25) NOT NULL,
Position VARCHAR (25) NULL,
Salary MONEY NULL,
Begin_Date DATE NOT NULL,
End_Date DATE NULL,
Dep_Id INT NULL);
```

```
CREATE TABLE Departments
(Dep_Id INT IDENTITY(1,1) CONSTRAINT PK_Departments
PRIMARY KEY,
Name VARCHAR(25) NOT NULL);
```

Для того чтобы обеспечить ссылочную целостность в БД **Art**, нужно добавить в созданные таблицы ограничение по внешним ключам.

Таблицы **Artworks** и **Authors** нужно связать по столбцу **Author_Id**, а таблицы **Artworks** и **Departments** – по столбцу **Dep_Id**. Аналогично должны быть связаны между собой таблицы **Employees** и **Departments**.

```
ALTER TABLE Artworks
ADD CONSTRAINT FK_Artw_Auth
FOREIGN KEY (Author_Id)
REFERENCES Authors (Author_Id);
```

```
ALTER TABLE Artworks
ADD CONSTRAINT FK_Artw_Dep
FOREIGN KEY (Dep_Id)
REFERENCES Departments (Dep_Id);
```

```
ALTER TABLE Employees
ADD CONSTRAINT FK_Emp_Dep
FOREIGN KEY (Dep_Id)
REFERENCES Departments (Dep_Id);
```

Теперь, когда установлены связи между всеми таблицами, можно создать диаграмму, описывающую эти взаимосвязи.

Для создания диаграммы нужно щелкнуть правой кнопкой мыши на элементе БД «Диаграммы баз данных» и в открывшемся контекстном меню выбрать пункт «Создать диаграмму базы данных».

В открывшемся диалоговом окне нужно выбрать таблицы, которые будут добавлены на диаграмму (рис. 3.3).

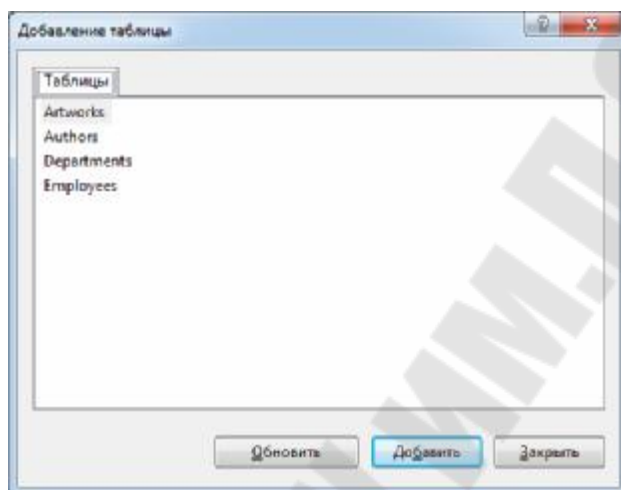


Рис. 3.3

В результате будет получена диаграмма (рис. 3.4).

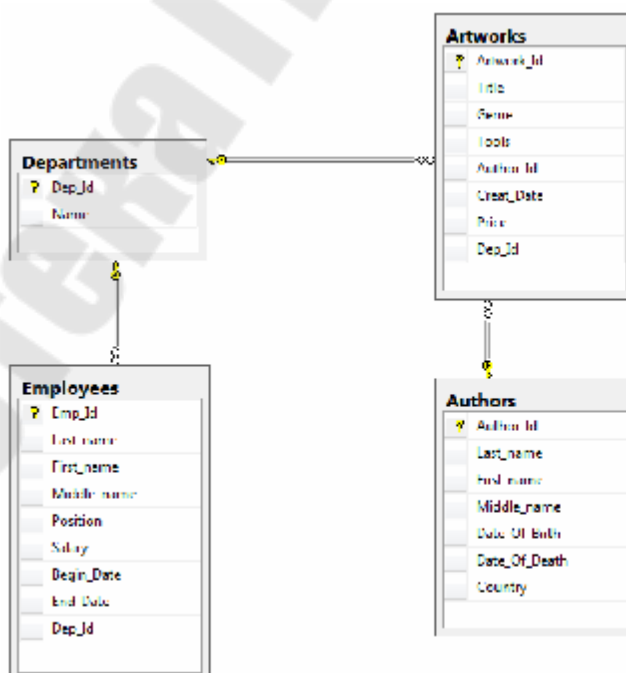


Рис. 3.4

Контрольные вопросы

1. Какая инструкция языка SQL создает БД?
2. Какие параметры задаются при создании БД на языке SQL?
3. Какая инструкция языка SQL создает таблицу?
4. Какие параметры задаются при создании таблиц на языке SQL?
5. Перечислите типы данных, используемые в языке SQL.
6. Как задать ограничение первичного ключа при создании таблицы на языке SQL?
7. Как удалить таблицу?
8. С помощью какой инструкции вносятся изменения в существующие таблицы?
9. Что такое внешний ключ? Для чего он необходим?

Задание

Средствами среды SQL Server Management Studio создать БД и ее таблицы (с обязательным определением первичных и внешних ключей, ссылочной целостности связанных таблиц, с ограничениями на ввод допустимых значений в отдельные поля) в соответствии со своим вариантом. Создать диаграмму БД.

Задание необходимо выполнить с помощью стандартных команд языка SQL.

Отчет по лабораторной работе должен содержать:

- титульный лист;
- цель работы;
- задание согласно своему варианту;
- SQL-запросы для создания БД и ее таблиц;
- итоговую диаграмму БД;
- выводы.

Таблица 3.1

Вариант, предметная область	Примерный перечень таблиц и атрибутов
1. Ресторан	<p>1. Сотрудники (Код сотрудника, Ф. И. О., Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности) [10 записей].</p> <p>2. Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования) [5 записей].</p> <p>3. Склад (Код ингредиента, Наименование ингредиента, Дата выпуска, Объем, Срок годности, Стоимость, Поставщик) [20 записей].</p> <p>4. Меню (Код блюда, Наименование блюда, Стоимость, Время приготовления) [15 записей].</p> <p>5. Заказ (Код заказа, Дата, Время, Ф. И. О. заказчика, Телефон, Стоимость, Отметка о выполнении, Код сотрудника) [15 записей]</p>
2. Банк	<p>1. Сотрудники (Код сотрудника, Ф. И. О., Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности) [10 записей].</p> <p>2. Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования) [5 записей].</p> <p>3. Вклады (Код вклада, Наименование вклада, Минимальный срок вклада, Минимальная сумма вклада, Код валюты, Процентная ставка, Дополнительные условия) [5 записей].</p> <p>4. Валюта (Код валюты, Наименование, Обменный курс) [5 записей].</p> <p>5. Вкладчики (Ф. И. О. вкладчика, Адрес, Телефон, Паспортные данные, Дата вклада, Дата возврата, Код вклада, Сумма вклада, Сумма возврата, Отметка о возврате вклада, Код сотрудника) [15 записей]</p>
3. Больница	<p>1. Сотрудники (Код сотрудника, Ф. И. О., Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности) [10 записей].</p> <p>2. Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования) [5 записей].</p> <p>3. Лекарства (Код лекарства, Наименование, Показания, Противопоказания, Упаковка, Стоимость) [15 записей].</p> <p>4. Болезни (Код болезни, Наименование, Симптомы, Продолжительность, Последствия) [10 записей].</p> <p>5. Пациенты (Ф. И. О. пациента, Возраст, Пол, Адрес, Телефон, Дата обращения, Код болезни, Код сотрудника, Результат лечения) [15 записей]</p>

Вариант, предметная область	Примерный перечень таблиц и атрибутов
4. Гостиница	1. Сотрудники (Код сотрудника, Ф. И. О., Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности) [10 записей]. 2. Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования) [5 записей]. 3. Номера (Код номера, Наименование, Вместимость, Описание, Стоимость) [20 записей]. 4. Услуги (Код услуги, Наименование, Описание, Стоимость) [5 записей]. 5. Клиенты (Код клиента, Ф. И. О., Паспортные данные, Дата заселения, Дата выезда) [10 записей]. 6. Заказ (Код заказа, Код клиента, Код сотрудника, Код номера) [20 записей]
5. Аэропорт	1. Сотрудники (Код сотрудника, Ф. И. О., Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности) [10 записей]. 2. Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования) [5 записей]. 3. Самолеты (Код самолета, Марка, Вместимость, Грузоподъемность, Код типа, Технические характеристики, Дата выпуска, Налетано часов, Дата последнего ремонта, Код сотрудника) [10 записей]. 4. Типы самолетов (Код типа, Наименование, Назначение, Ограничения) [5 записей]. 5. Рейсы (Код рейса, Дата, Время, Пункт вылета, Пункт назначения, Код самолета, Время полета) [10 записей]. 6. Билеты (Код билета, Ф. И. О. пассажира, Паспортные данные, Место, Код рейса, Цена) [10 записей]
6. Видеопрокат	1. Сотрудники (Код сотрудника, Ф. И. О., Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности) [10 записей]. 2. Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования) [5 записей]. 3. Жанры (Код жанра, Наименование жанра, Описание) [5 записей]. 4. Кассеты (Код кассеты, Наименование фильма, Год создания, Производитель, Страна, Режиссер, Код жанра, Цена) [20 записей]. 5. Клиенты (Код клиента, Ф. И. О., Адрес, Телефон, Паспортные данные) [10 записей].

Вариант, предметная область	Примерный перечень таблиц и атрибутов
	6. Заказ (Код заказа, Код клиента, Код сотрудника, Код кассеты, Дата получения, Дата возврата, Отметка об оплате, Отметка о возврате) [15 записей]
7. Библиотека	<p>1. Сотрудники (Код сотрудника, Ф. И. О., Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности) [10 записей].</p> <p>2. Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования) [5 записей].</p> <p>3. Издательства (Код издательства, Наименование, Город, Адрес) [5 записей].</p> <p>4. Жанры (Код жанра, Наименование, Описание) [5 записей].</p> <p>5. Книги (Код книги, Наименование, Код издательства, Год издания, Код жанра) [15 записей].</p> <p>6. Авторы (Код автора, Ф. И. О.) [5 записей].</p> <p>7. Читатели (Код читателя, Ф. И. О., Дата рождения, Пол, Адрес, Телефон, Паспортные данные) [15 записей].</p> <p>8. Выданные книги (Код книги, Код читателя, Дата выдачи, Дата возврата, Отметка о возврате, Код сотрудника) [15 записей]</p>
8. Радиостанция	<p>1. Сотрудники (Код сотрудника, Ф. И. О., Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности) [10 записей].</p> <p>2. Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования) [5 записей].</p> <p>3. Исполнители (Код исполнителя, Наименование, Описание) [10 записей].</p> <p>4. Жанры (Код жанра, Наименование, Описание) [5 записей].</p> <p>5. Записи (Код записи, Наименование, Код исполнителя, Альбом, Год, Код жанра, Дата записи, Длительность) [15 записей].</p> <p>6. График работы (Дата, Код сотрудника, Время начала, Время окончания) [10 записей]</p>
9. Таксопарк	<p>1. Сотрудники (Код сотрудника, Ф. И. О., Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности) [10 записей].</p> <p>2. Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования) [5 записей].</p>

Вариант, предметная область	Примерный перечень таблиц и атрибутов
	<p>3. Марки (Код марки, Наименование, Технические характеристики, Стоимость, Специфика) [5 записей].</p> <p>4. Тарифы (Код тарифа, Наименование, Описание, Стоимость) [5 записей].</p> <p>5. Дополнительные услуги (Код услуги, Наименование, Описание услуги, Стоимость) [5 записей].</p> <p>6. Автомобили (Код автомобиля, Код марки, Регистрационный номер, Номер кузова, Номер двигателя, Год выпуска, Пробег, Код сотрудника-шофера, Дата последнего ТО) [10 записей].</p> <p>7. Вызовы (Дата, Время, Телефон, Пункт отправления, Пункт назначения, Код тарифа, Код услуги, Код автомобиля, Код сотрудника-оператора) [15 записей]</p>
10. Туристическое агентство	<p>1. Сотрудники (Код сотрудника, Ф. И. О., Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности) [10 записей].</p> <p>2. Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования) [5 записей].</p> <p>3. Виды отдыха (Код вида, Наименование, Описание, Ограничения) [5 записей].</p> <p>4. Отели (Код отеля, Наименование, Страна, Город, Адрес, Телефон, Количество звезд) [10 записей].</p> <p>5. Дополнительные услуги (Код услуги, Наименование, Описание, Цена) [5 записей].</p> <p>6. Клиенты (Код клиента, Ф. И. О., Дата рождения, Пол, Адрес, Телефон, Паспортные данные) [15 записей].</p> <p>7. Путевки (Дата начала, Дата окончания, Продолжительность, Код клиента, Код сотрудника, Отметка о бронировании, Стоимость, Отметка об оплате) [15 записей]</p>
11. Страховая компания	<p>1. Сотрудники (Код сотрудника, Ф. И. О., Возраст, Пол, Адрес, Телефон, Паспортные данные, (Код должности) [10 записей].</p> <p>2. Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования) [5 записей].</p> <p>3. Риски (Код риска, Наименование, Описание) [5 записей].</p> <p>4. Виды страхований (Код вида страхования, Наименование, Описание, Условия) [5 записей].</p>

Вариант, предметная область	Примерный перечень таблиц и атрибутов
	<p>5. Клиенты (Код клиента, Ф. И. О., Дата рождения, Пол, Адрес, Телефон, Паспортные данные) [10 записей].</p> <p>6. Страховка (Код страховки, Дата начала, Дата окончания, Стоимость, Сумма выплаты, Код вида страхования, Отметка о выплате, Отметка об окончании, Код клиента, Код сотрудника) [15 записей].</p>
12. Брачное агентство	<p>1. Сотрудники (Код сотрудника, Ф. И. О., Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности) [10 записей].</p> <p>2. Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования) [5 записей].</p> <p>3. Знаки зодиака (Код знака, Наименование, Описание) [12 записей].</p> <p>4. Отношения (Код отношения, Наименование, Описание) [5 записей].</p> <p>5. Национальности (Код национальности, Наименование, Замечания) [5 записей].</p> <p>6. Дополнительные услуги (Код услуги, Наименование, Описание, Цена) [5 записей].</p> <p>7. Клиенты (Код клиента, Ф. И. О., Пол, Дата рождения, Возраст, Рост, Вес, Количество детей, Семейное положение, Вредные привычки, Хобби, Описание, Код знака, Код отношения, Код национальности, Адрес, Телефон, Паспортные данные) [10 записей].</p> <p>8. Услуги (Код клиента, Дата, Стоимость, Код сотрудника) [15 записей]</p>
13. Сервисный центр	<p>1. Сотрудники (Код сотрудника, Ф. И. О., Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности) [10 записей].</p> <p>2. Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования) [5 записей].</p> <p>3. Запчасти (Код запчасти, Наименование, Цена) [20 записей].</p> <p>4. Ремонтируемые модели (Код модели, Наименование, Производитель, Особенности) [10 записей].</p> <p>5. Виды неисправностей (Код вида, Код модели, Описание, Методы ремонта, Стоимость) [5 записей].</p>

Вариант, предметная область	Примерный перечень таблиц и атрибутов
	<p>6. Обслуживаемые магазины (Код магазина, Наименование, Адрес, Телефон) [5 записей].</p> <p>7. Заказы (Код заказа, Дата заказа, Дата возврата, Ф. И. О. заказчика, Серийный номер, Код вида неисправности, Код магазина, Отметка о гарантии, Срок гарантии ремонта, Цена, Код сотрудника) [15 записей]</p>
14. Школа	<p>1. Сотрудники (Код сотрудника, Ф. И. О., Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности) [10 записей].</p> <p>2. Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования) [5 записей].</p> <p>3. Ученики (Ф. И. О., Дата рождения, Пол, Адрес, Ф. И. О. отца, Ф. И. О. матери, Код класса, Дополнительная информация) [15 записей].</p> <p>4. Классы (Код класса, Код сотрудника – классного руководителя, Код вида, Количество учеников, Буква, Год обучения, Год создания) [15 записей].</p> <p>5. Предметы (Код предмета, Наименование, Описание) [15 записей].</p> <p>6. Расписание (Дата, День недели, Код класса, Код предмета, Время начала, Время окончания) [15 записей].</p> <p>7. Журнал (Код ученика, Код дисциплины, Оценка, Количество пропусков) [20 записей]</p>
15. Транспортная компания	<p>1. Сотрудники (Код сотрудника, Ф. И. О., Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности) [10 записей].</p> <p>2. Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования) [5 записей].</p> <p>3. Виды автомобилей (Код вида автомобиля, Наименование, Описание) [5 записей].</p> <p>4. Марки автомобилей (Код марки, Наименование, Технические характеристики, Описание) [5 записей].</p> <p>5. Виды грузов (Код вида груза, Наименование, Код вида автомобиля для транспортировки, Описание) [5 записей].</p> <p>6. Грузы (Код груза, Наименование, Код вида груза, Срок годности, Особенности) [5 записей]</p>

Вариант, предметная область	Примерный перечень таблиц и атрибутов
	<p>7. Автомобили (Код автомобиля, Код марки, Код вида автомобиля, Регистрационный номер, Номер кузова, Номер двигателя, Год выпуска, Код сотрудника-водителя, Дата последнего технического осмотра) [10 записей].</p> <p>8. Рейсы (Код автомобиля, Заказчик, Пункт отправления, Пункт назначения, Дата отправления, Дата прибытия, Код груза, Цена, Отметка об оплате, Отметка о возвращении, Код сотрудника) [10 записей]</p>

Лабораторная работа № 4

Заполнение таблиц базы данных

Цель работы: научиться заполнять таблицы БД в среде SQL Server Management Studio.

Теоретические сведения

Заполнение таблиц в SQL осуществляется с помощью команды INSERT, которая имеет следующий синтаксис:

```
INSERT INTO имя_таблицы(список_столбцов)
VALUES (список_значений);
```

Здесь:

- **Имя_таблицы** – таблица, куда вставляются данные.
- **Список_столбцов** – список полей, в которые вставляются данные, если он не указывается, то подразумевается заполнение всех полей, в списке полей поля указываются через запятую.
- **Список_значений** – значение полей для вставки через запятую.

Из таблицы можно удалить все столбцы, либо отдельные записи. Это осуществляется командой DELETE.

```
DELETE FROM <имя таблицы>
[WHERE <условие>];
```

где <условие> – условие, которым удовлетворяют удаляемые записи, если условие не указано, то удаляются все столбцы таблицы.

Значение полей таблицы можно обновить (изменить), используя следующую команду:

```
UPDATE <имя таблицы>
SET <имя поля1> = <выражение1>,
...
[WHERE <условие>]
```

Здесь <имя поля1>, <имя поля2> – имена изменяемых полей; <выражение1>, <выражение2> – значения, которые должны принять

поля; <условие> – условие, которому должны соответствовать записи, поля которых изменяем.

В качестве выражения можно использовать математические формулы.

Если необходимо из таблицы удалить все записи, но сохранить ее структуру, нужно воспользоваться командой TRUNCATE TABLE <имя таблицы>, при этом все данные будут удалены, но сама таблица останется.

Пример. Заполнить таблицу «Departments», в которой должна храниться информация с именами отделов музея. Идентификатор каждого отдела будет заполняться автоматически, потому что при создании таблицы было использовано свойство IDENTITY.

```
INSERT INTO Departments (Name)
VALUES ('Отдел искусства фотографии'),
('Отдел гравюры и рисунка'),
('Отдел искусства Античного мира'),
('Отдел искусства Древнего Востока'),
('Отдел искусства стран Европы XIX–XX вв.'),
('Отдел реставрации'),
('Отдел русского искусства XIX–XX вв.);
```

Если поле является внешним ключом, то заполнять его следует значениями того поля, которое является первичным ключом для него, как и было сделано в этом случае для поля «Dep_Id» (номер отдела из таблицы «Departments»).

Контрольные вопросы

1. Какая инструкция языка SQL добавляет строки в таблицу?
2. Опишите синтаксис инструкции INSERT.
3. Как добавить несколько строк в таблицу одним вызовом инструкции INSERT?
4. Каково назначение свойства IDENTITY?
5. Какая инструкция языка SQL удаляет строки из таблицы?
6. С помощью какой инструкции языка SQL можно обновить содержимое строк таблицы?

Задание

Заполнить таблицы БД достоверной информацией в соответствии со своим вариантом (см. лабораторную работу № 3).

Отчет по лабораторной работе должен содержать:

- титульный лист;
- цель работы;
- задание согласно своему варианту;
- SQL-запросы для заполнения таблиц;
- результаты заполнения таблиц;
- выводы.

Лабораторная работа № 5

Создание запросов

Цель работы: научиться создавать запросы в среде SQL Server Management Studio.

Теоретические сведения

Запросы могут осуществлять отбор отдельных полей из таблицы и производить фильтрацию данных согласно условию, наложенному на одно или несколько полей. Такие запросы создаются с помощью инструкции SELECT, которая имеет следующий синтаксис:

```
SELECT [ ALL | DISTINCT ]
[ TOP (expression) [ PERCENT ] ]
{
*
    | { table_name }. *
    | {
        [ { table_name }. ]
          { column_name }
        | expression
        [ [ AS ] column_alias ]
    }
| column_alias=expression
}
[ INTO new_table ]
[ FROM { table_name } ]
[ WHERE search_condition ]
[ GROUP BY column_expression ]
[ HAVING search_condition ]
[ ORDER BY { order_by_expression [ ASC | DESC ] } ]
```

Здесь параметры *ALL* | *DISTINCT* показывают, какие записи окажутся в результирующей таблице: *ALL* – дублирующиеся записи могут оказаться в результате, *DISTINCT* – только уникальные записи окажутся в результате; *TOP expression PERCENT* определяет, сколько записей попадет в результат запроса (в количестве записей или в процентах); *table_name* – имя таблицы; *column_name* – имя столбца;

INTO new_table – результат запроса может быть помещен в таблицу с именем *new_table*; *search_condition* определяет условие для осуществления поиска данных в таблице (после *WHERE*) или для группировки данных в результате запроса (после *HAVING*); *ORDER BY* позволяет сгруппировать данные, полученные в результате запроса, по столбцам (*order_by_expression*) в порядке возрастания или убывания (*ASC | DESC*).

Замечания:

1. Если выбираются поля из разных таблиц с одинаковыми именами, нужно указывать и имя таблицы *table_name.column_name*.

2. Полям можно присваивать псевдонимы следующим образом: *column_name AS column_alias*

3. Если необходимо выбрать все поля из таблицы, то их можно заменить значком «*».

4. Раздел *INTO*. Если присутствует этот раздел, то на основе результатов запроса создается новая таблица. Параметр *INTO* – это имя новой таблицы.

5. Раздел *FROM*. Здесь указываются таблицы и запросы, через запятую, которые участвуют в новом запросе. В разделе *FROM* также можно задавать сложные связи: связь поля одной таблицы с несколькими полями другой таблицы.

6. Раздел *WHERE*. Данный раздел используют для создания простых запросов. В этом случае в качестве условия указываются связываемые поля либо этот раздел используют для создания фильтров, и здесь указываются условия отбора. В условиях отбора мы можем использовать стандартные логические операторы *NOT*, *OR*, *AND*.

7. Раздел *GROUP BY* определяет поле для группировки записей в запросе.

8. Раздел *ORDER BY* определяет поле для сортировки записей в запросе. Если указан параметр *ASC*, то будет производиться сортировка по возрастанию, если *DESC* – по убыванию. По умолчанию используется сортировка по возрастанию.

Кроме связывания таблиц и отбора данных, оператор *SELECT* может использоваться для вычислений. В этом случае он имеет синтаксис:

SELECT expression

где *expression* – какое-то математическое выражение или функция.

В SQL Server существуют следующие встроенные агрегатные функции:

AVG	Возвращает среднее значение
COUNT	Возвращает количество значений (результат имеет тип int)
MAX	Возвращает максимальное значение
MIN	Возвращает минимальное значение
SUM	Возвращает сумму всех значений
STDEV	Возвращает среднеквадратичное отклонение всех значений
VAR	Возвращает дисперсию всех значений

Примеры использования агрегатных функций:

```
SELECT AVG(возраст) FROM Студент
```

– выводит средний возраст студента из таблицы «Студенты»;

```
SELECT COUNT(ФИО) FROM Студенты
```

– выводит количество различных Ф. И. О. из таблицы «Студенты».

Пример. Рассмотрим пример создания запросов для БД Art. В лабораторной работе № 3 была приведена диаграмма этой БД.

Создадим запрос «Отдел кадров» (связывает таблицы «**Employees**» и «**Departments**» по полю «**Dep_Id**»), а также на его основе запрос для отображения сотрудников отдельных отделов.

Следующий программный код осуществляет внутреннее соединение таблиц «**Employees**» и «**Departments**» по внешнему ключу «**Dep_Id**»:

```
SELECT E.Last_name, E.First_name, E.Middle_name, E.Position,  
D.Name, E.Begin_Date  
FROM Employees AS E  
JOIN Departments AS D ON E.Dep_Id = D.Dep_Id;
```

Результатом выполнения этого запроса будет таблица, содержащая все строки из таблицы «**Employees**», в которых значение атрибута «**Dep_Id**» равно значению этого же атрибута из таблицы «**Departments**».

Для создания запроса, выводящего данные сотрудников отдельных отделов, необходимо модифицировать предыдущий запрос следующим образом:


```
SELECT E.Last_name, E.First_name, E.Middle_name, E.Position,  
D.Name, E.Begin_Date  
FROM Employees AS E  
JOIN Departments AS D ON E.Dep_Id = D.Dep_Id AND D.Name =  
'Отдел искусства фотографии';
```

Поле «**Name**» таблицы «**Departments**» содержит названия отделов. Сравнивая его с конкретным именем отдела, можно организовать вывод информации о сотрудниках этого отдела.

Контрольные вопросы

1. Для чего предназначен запрос?
2. Запишите синтаксис инструкции SELECT.
3. Для чего предназначен элемент GROUP BY?
4. Для чего предназначен элемент ORDER BY?
5. Что означает символ «*» в списке элемента SELECT?
6. Какие параметры инструкции SELECT являются обязательными?
7. Что такое предикат?
8. Какие предикаты могут использоваться в инструкции SELECT?
9. Как выполнить соединение таблиц в языке SQL?
10. Как осуществляются перекрестное, внутреннее и внешнее соединения?
11. Чем отличаются левое, правое и полное внешние соединения?

Задание

Создать запросы в соответствии с заданием для своего варианта.

Отчет по лабораторной работе должен содержать:

- титульный лист;
- цель работы;
- задание согласно своему варианту;
- созданные SQL-запросы;
- результаты выполнения запросов;
- выводы.

Таблица 4.2

Вариант, предметная область	Запросы
1. Ресторан	<ol style="list-style-type: none"> 1. Отдел кадров – данные о сотрудниках и в каких должностях они работают. 2. Кулинарная книга – информация о каждом блюде с указанием всех ингредиентов. 3. Выполненные заказы на сумму больше 100, но меньше 200 рублей. 4. Параметрический запрос для отображения информации об ингредиентах, входящих в состав определенного блюда. 5. Параметрический запрос для отображения информации об ингредиентах, поставляемых определенными поставщиками. 6. Перекрестный запрос с информацией о стоимости выполненных заказов по каждому сотруднику. 7. Вычислить стоимость всех заказов, выполненных в текущем году. 8. Запрос на добавление нового блюда. 9. Запрос на обновление данных о блюде. 10. Запрос на удаление данных о блюде
2. Банк	<ol style="list-style-type: none"> 1. Отдел кадров – данные о сотрудниках и в каких должностях они работают. 2. Вклады – информация о вкладчиках и сделанных ими вкладах. 3. Список вкладчиков, у которых заканчивается срок вклада в марте текущего года. 4. Параметрический запрос для отображения информации о сотрудниках, работающих на определенной должности. 5. Параметрический запрос для отображения информации о вкладах по заданной валюте. 6. Перекрестный запрос с информацией о сумме вкладов для всех видов валют по каждому вкладчику. 7. Вычислить сумму всех вкладов, сделанных в прошлом году. 8. Запрос на добавление нового вклада. 9. Запрос на обновление данных о вкладе. 10. Запрос на удаление данных о вкладе
3. Больница	<ol style="list-style-type: none"> 1. Отдел кадров – данные о сотрудниках и в каких должностях они работают. 2. Болезни – информация о всех болезнях и лекарствах, применяемых для их лечения.

Вариант, предметная область	Запросы
	3. Список пациентов, которые обращались в больницу не менее двух раз. 4. Параметрический запрос для отображения информации о сотрудниках, работающих на определенной должности. 5. Параметрический запрос для отображения информации о болезнях, которые можно лечить заданным лекарством. 6. Перекрестный запрос с информацией о количестве посещений врачей за каждую дату. 7. Вычислить стоимость лекарств для лечения каждой болезни. 8. Запрос на добавление нового пациента. 9. Запрос на обновление данных о пациенте. 10. Запрос на удаление данных о пациенте
4. Гостиница	1. Отдел кадров – данные о сотрудниках и в каких должностях они работают. 2. Номера – информация о номерах и клиентах, которые в них проживали. 3. Список клиентов, которые хотя бы раз воспользовались дополнительной услугой. 4. Параметрический запрос для отображения информации о сотрудниках, работающих на определенной должности. 5. Параметрический запрос для отображения информации о клиентах, проживавших в определенном номере гостиницы. 6. Перекрестный запрос с информацией о количестве проживавших по каждому номеру на каждую дату. 7. Вычислить стоимость всех дополнительных услуг, сделанных каждым клиентом. 8. Запрос на добавление нового клиента. 9. Запрос на обновление данных о клиента. 10. Запрос на удаление данных о клиенте
5. Аэропорт	1. Отдел кадров – данные о сотрудниках и в каких должностях они работают. 2. Самолеты – информация о самолетах и пилотах, которые на них летали. 3. Список рейсов, которые выполняются только внутри страны. 4. Параметрический запрос для отображения информации о сотрудниках, работающих на определенной должности. 5. Параметрический запрос для отображения информации о самолетах заданного типа.

Вариант, предметная область	Запросы
	<p>6. Вычислить количество сделанных рейсов каждым пилотом.</p> <p>7. Перекрестный запрос с информацией об общей стоимости проданных билетов для различных рейсов на каждую дату.</p> <p>8. Запрос на добавление нового рейса.</p> <p>9. Запрос на обновление данных о рейсе.</p> <p>10. Запрос на удаление данных о рейсе</p>
6. Видеопрокат	<p>1. Отдел кадров – данные о сотрудниках и в каких должностях они работают.</p> <p>2. Кассеты – информация о кассетах и клиентах, которые их брали в прокат.</p> <p>3. Список клиентов, которые брали кассеты в прокат не менее двух раз.</p> <p>4. Параметрический запрос для отображения информации о сотрудниках, работающих на определенной должности.</p> <p>5. Параметрический запрос для отображения информации о кассетах по заданному жанру.</p> <p>6. Перекрестный запрос с информацией о прибыли по каждому фильму для различных дат.</p> <p>7. Вычислить количество фильмов, произведенных в заданной стране.</p> <p>8. Запрос на добавление нового клиента.</p> <p>9. Запрос на обновление данных о клиенте.</p> <p>10. Запрос на удаление данных о клиенте</p>
7. Библиотека	<p>1. Отдел кадров – данные о сотрудниках и в каких должностях они работают.</p> <p>2. Каталог – полная информация обо всех имеющихся в библиотеке книгах.</p> <p>3. Список клиентов, которые брали за все время только одну книгу.</p> <p>4. Параметрический запрос для отображения информации о сотрудниках, работающих на определенной должности.</p> <p>5. Параметрический запрос для отображения информации о книгах по определенному году издания.</p> <p>6. Перекрестный запрос с информацией о количестве книг на руках по каждому читателю.</p> <p>7. Посчитать количество всех должников на текущую дату.</p> <p>8. Запрос на добавление новой книги.</p> <p>9. Запрос на обновление данных о книге.</p> <p>10. Запрос на удаление данных о книге</p>

Вариант, предметная область	Запросы
8. Радиостанция	<ol style="list-style-type: none"> 1. Отдел кадров – данные о сотрудниках и в каких должностях они работают. 2. Музыкальный архив – полная информация об имеющихся записях. 3. Исполнители, работающие не более чем в двух жанрах. 4. Параметрический запрос для отображения информации о сотрудниках, работающих на определенной должности. 5. Параметрический запрос для отображения информации о записях по заданным исполнителям. 6. Перекрестный запрос с информацией о сетке вещания по отдельным датам. 7. Вычислить количество записей для каждого исполнителя. 8. Запрос на добавление нового исполнителя. 9. Запрос на обновление данных об исполнителе. 10. Запрос на удаление данных об исполнителе
9. Таксопарк	<ol style="list-style-type: none"> 1. Отдел кадров – данные о сотрудниках и в каких должностях они работают. 2. Автопарк – полная информация об автомобилях и их водителях. 3. Список вызовов водителей, фамилия которых начинается на букву «М». 4. Параметрический запрос для отображения информации о сотрудниках, работающих на определенной должности. 5. Параметрический запрос для отображения информации об автомобилях определенного года выпуска. 6. Перекрестный запрос для отображения количества вызовов по отдельным датам для каждого сотрудника. 7. Вычислить среднюю стоимость вызовов для каждого клиента. 8. Запрос на добавление нового автомобиля. 9. Запрос на обновление данных об автомобиле. 10. Запрос на удаление данных об автомобиле
10. Туристическое агентство	<ol style="list-style-type: none"> 1. Отдел кадров – данные о сотрудниках и в каких должностях они работают. 2. Список путевок – полная информация о путевке и оказываемых услугах. 3. Информация о клиентах, купивших путевку не менее двух раз. 4. Параметрический запрос для отображения информации о сотрудниках, работающих на определенной должности. 5. Параметрический запрос для отображения путевок в отдельные страны.

Вариант, предметная область	Запросы
	<p>6. Перекрестный запрос с информацией о стоимости путевок по каждому клиенту для разных отелей.</p> <p>7. Вычисление общей стоимости путевок по каждому отелю.</p> <p>8. Запрос на добавление нового отеля.</p> <p>9. Запрос на обновление данных об отеле.</p> <p>10. Запрос на удаление данных об отеле</p>
11. Страховая компания	<p>1. Отдел кадров – данные о сотрудниках и в каких должностях они работают.</p> <p>2. Риски страховок – информация обо всех видах страховок и рисках страхования.</p> <p>3. Список клиентов, застраховавших что-либо на сумму менее 100 рублей.</p> <p>4. Параметрический запрос для отображения информации о сотрудниках, работающих на определенной должности.</p> <p>5. Параметрический запрос для отображения информации о клиентах по заданному виду страхования.</p> <p>6. Перекрестный запрос с информацией о выплаченных суммах для разных видов страхования по каждому клиенту.</p> <p>7. Вычисление стоимости произведенных выплат по каждому виду страхования.</p> <p>8. Запрос на добавление нового вида страхования.</p> <p>9. Запрос на обновление данных о виде страхования.</p> <p>10. Запрос на удаление данных о виде страхования</p>
12. Брачное агентство	<p>1. Отдел кадров – данные о сотрудниках и в каких должностях они работают.</p> <p>2. Список клиентов – полная информация о всех клиентах.</p> <p>3. Список услуг, стоимость которых составляет менее 30 рублей или более 100 рублей.</p> <p>4. Параметрический запрос для отображения информации о сотрудниках, работающих на определенной должности.</p> <p>5. Параметрический запрос для отображения информации о клиентах, родившихся под определенным знаком зодиака.</p> <p>6. Перекрестный запрос с информацией об оплате услуг по каждому клиенту для всех знаков зодиака.</p> <p>7. Вычисление стоимости оказанных услуг по каждому сотруднику.</p> <p>8. Запрос на добавление новой услуги.</p> <p>9. Запрос на обновление данных об услуге.</p> <p>10. Запрос на удаление данных об услуге</p>

Вариант, предметная область	Запросы
13. Сервисный центр	<ol style="list-style-type: none"> 1. Отдел кадров – данные о сотрудниках и в каких должностях они работают. 2. Список неисправностей – информация об отремонтированных моделях. 3. Список заказов, сделанных во втором квартале прошлого года. 4. Параметрический запрос для отображения информации о сотрудниках, работающих на определенной должности. 5. Параметрический запрос для отображения видов неисправностей отдельных моделей автомобилей. 6. Перекрестный запрос с информацией о количестве заказов для каждого сотрудника по всем месяцам прошлого года. 7. Вычислить стоимость всех заказов, выполненных в текущем году определенным сотрудником. 8. Запрос на добавление новой запчасти. 9. Запрос на обновление данных о запчасти. 10. Запрос на удаление данных о запчасти
14. Школа	<ol style="list-style-type: none"> 1. Отдел кадров – данные о сотрудниках и в каких должностях они работают. 2. Список учеников – информация об учениках и классах, в которых они учатся. 3. Расписание занятий для преподавателей, в фамилии которых не более шести букв. 4. Параметрический запрос для отображения информации о сотрудниках, работающих на определенной должности. 5. Параметрический запрос для отображения расписания для определенного класса по заданной дате. 6. Перекрестный запрос с информацией о количестве занятий для всех преподавателей по каждому классу. 7. Вычисление среднего балла всех учеников, которым исполнилось 15 лет. 8. Запрос на добавление нового ученика. 9. Запрос на обновление данных об ученике. 10. Запрос на удаление данных об ученике
15. Транспортная компания	<ol style="list-style-type: none"> 1. Отдел кадров – данные о сотрудниках и в каких должностях они работают. 2. Перевозимые грузы – полная информация о всех перевозимых грузах. 3. Список заказов, перевозимых на прошлой неделе.

Вариант, предметная область	Запросы
	<p>4. Параметрический запрос для отображения информации о сотрудниках, работающих на определенной должности.</p> <p>5. Параметрический запрос для отображения информации о заданном виде груза.</p> <p>6. Перекрестный запрос для отображения стоимости заказов по перевозке отдельных грузов для каждого водителя.</p> <p>7. Вычислить количество автомобилей, прошедших технический осмотр за определенный промежуток времени.</p> <p>8. Запрос на добавление нового автомобиля.</p> <p>9. Запрос на обновление данных об автомобиле.</p> <p>10. Запрос на удаление данных об автомобиле</p>

Учебное электронное издание комбинированного распространения

Учебное издание

Лукьяненко Владимир Олегович

**НАЛАДКА И ДИАГНОСТИКА
АВТОМАТИЗИРОВАННОГО
ЭЛЕКТРОПРИВОДА**

**Практикум
по выполнению лабораторных работ
по одноименной дисциплине
для студентов специальности 1-53 01 05
«Автоматизированные электроприводы»
дневной формы обучения**

Электронный аналог печатного издания

Редактор
Компьютерная верстка

*Н. Г. Мансурова
И. П. Минина*

Подписано в печать 25.02.19.

Формат 60x84/16. Бумага офсетная. Гарнитура «Таймс».

Ризография. Усл. печ. л. 3,49. Уч.-изд. л. 3,45.

Изд. № 20.

<http://www.gstu.by>

Издатель и полиграфическое исполнение
Гомельский государственный
технический университет имени П. О. Сухого.
Свидетельство о гос. регистрации в качестве издателя
печатных изданий за № 1/273 от 04.04.2014 г.
пр. Октября, 48, 246746, г. Гомель