# Лабораторная работа №1. Операторы SDL и DML

## Краткие теоретические данные.

**1.1 Операторы языка SDL**

Команды языка определения схемы данных (Schema Definition Language– SDL) представляют собой инструкции SQL, которые позволяют создавать имодифицировать элементы структуры базы данных. Например, используяSDL, можно создавать, удалять таблицы и изменять их структуру, создаватьи удалять индексы.

**1.1.1 Создание таблицы**. Оператор создания таблицы имеет следующий вид:

CREATE TABLE <tbl\_name> (CREADE\_DEFINITION)

**<tbl\_name>**– имя создаваемой таблицы

**CREADE\_DEFINITION**включает в себя:

1) перечень столбцов, их типов, и ограничений для значений ячейки в столбце:

col\_name type [NOT NULL | NULL] [DEFAULT default\_value] [IDENTITY]

- для текстовых переменных NULL устанавливается автоматически

*- DEFAULT* – определяет значение по умолчанию

*- [*IDENTITY*]* – автоматически увеличивающееся значение

2) набор столбцов участвующих в первичном ключе

[CONSTRAINT <pk\_name>] PRIMARY KEY (index\_col\_name,...)

***CONSTRAINT <pk\_name***>- создает имя для первичного ключа

3) набор столбцов участвующих во внешнем ключе, а также таблица с которой осуществляется связь

[CONSTRAINT <fk\_name>] FOREIGN KEY index\_name (index\_col\_name,...)[reference \_definition]

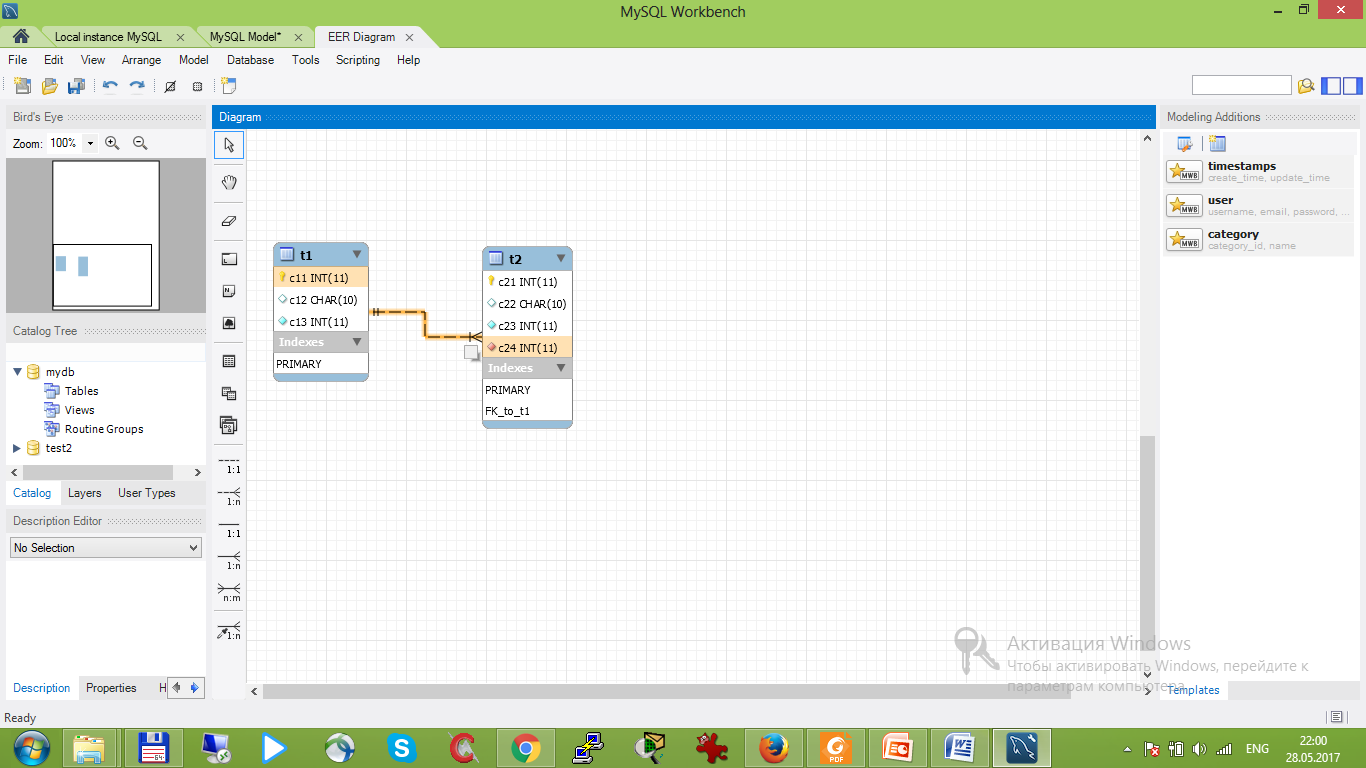
***CONSTRAINT <fk\_name****>-* создает имя для внешнего ключа

Обязательными операндами оператора являются **имя** создаваемой  
таблицы и имя **хотя бы одного столбца** (поля) с указанием типа данных,  
хранимых в этом столбце.

Пример 1.1

|  |
| --- |
| Create table t1  (c11 int not null IDENTITY,  c12 char(10) default 'no\_text',  c13 int not null,  CONSTRAINT PK\_t1 Primary key (c11));  CREATE TABLE t2  (c21 int not null IDENTITY,  c22 char(10) default 'no\_text',  c23 int not null,  c24 int not null,  CONSTRAINT PK\_t2 Primary key (c21),  CONSTRAINT FK\_to\_t1 foreign key (c24) references t1(c11) ); |

Результат 1.1



**1.1.2Изменение таблицы.**Для изменения **структуры таблицы** (добавление, удаление полей,изменения типов полей) используется оператор ALTERTABLE. в зависимости от необходимого изменения, оператор ALERT может иметь один из следующих видов:

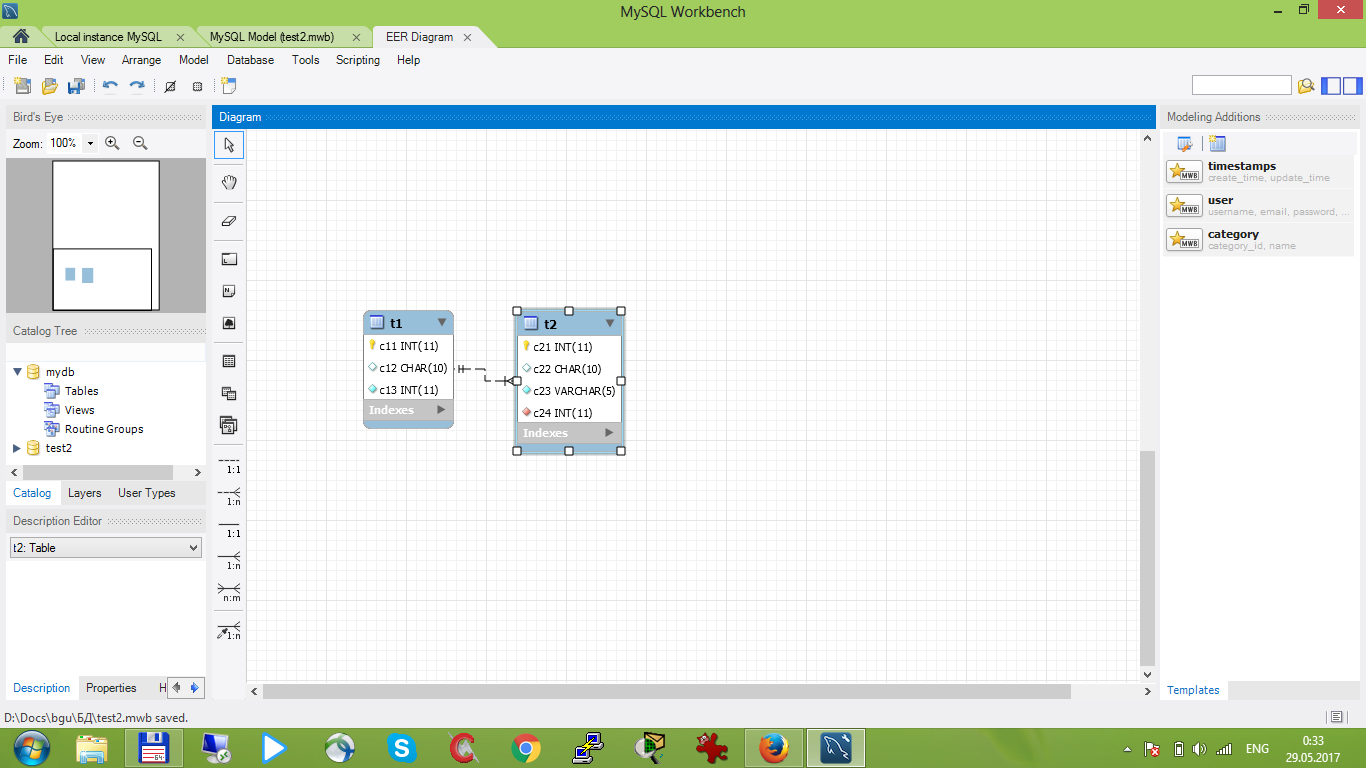
1) Изменениетипастолбца:

ALTER TABLE <tab\_name>ALTER COLUMN<col\_name>< type><NOT NULL>

Пример 1.2

|  |
| --- |
| ALTER TABLE t2 ALTER COLUMNc23 varchar(5) NOT NULL; |

результат 1.2



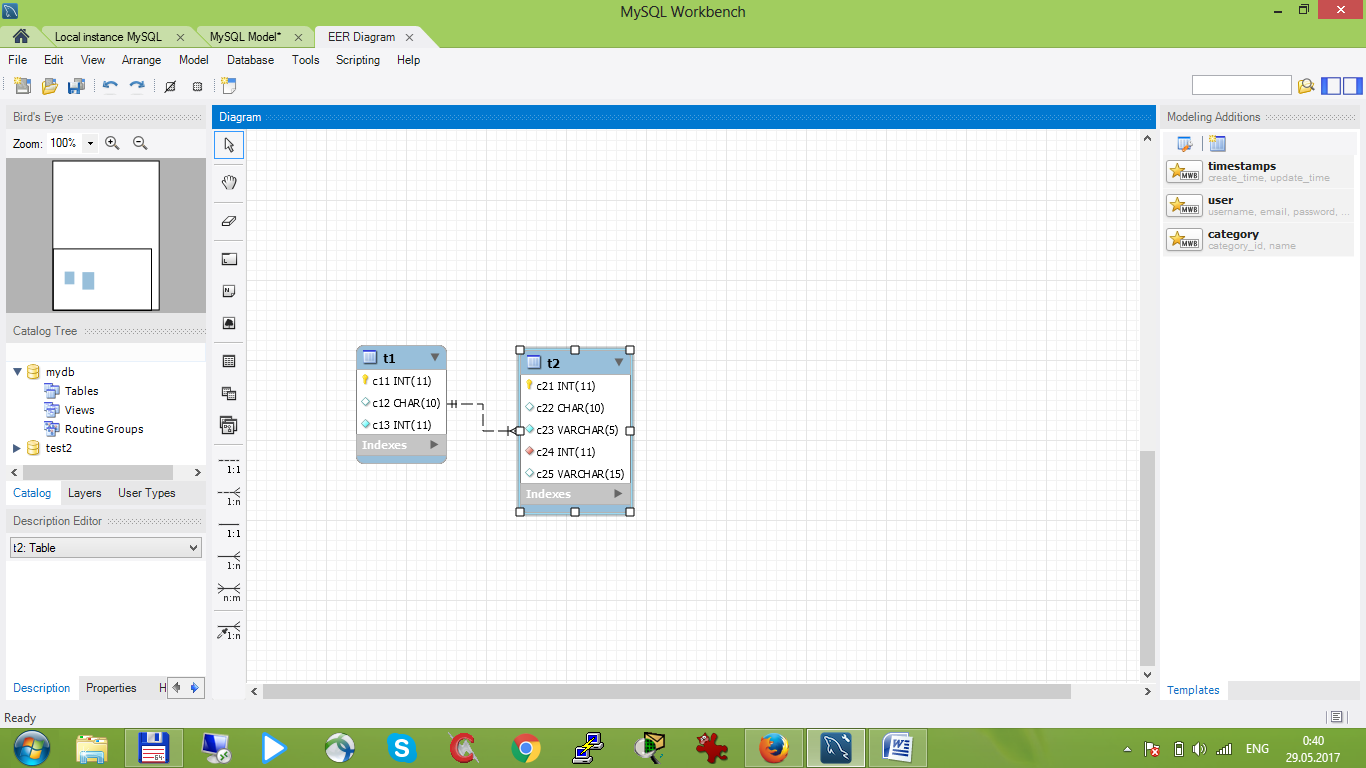
2)Добавление новогостолбца:

ALTER TABLE <table\_name> ADD <col\_name>< type> [<NOT NULL> DEFAULT <val>];

Пример 1.3

|  |
| --- |
| ALTER TABLE t2 ADD c25 varchar(15) default 'no\_text'; |

результат 1.3



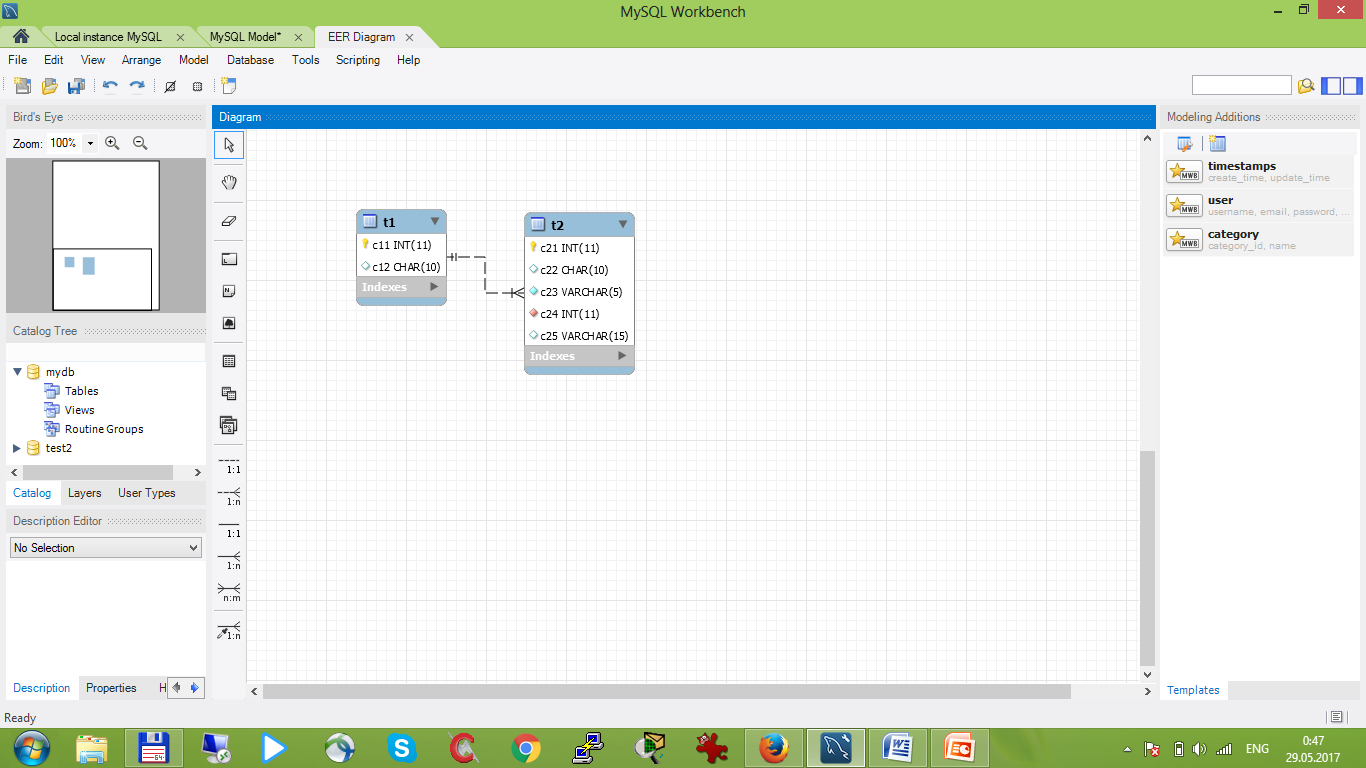
3)Удалениестолбца

ALTER TABLE <tab\_name> DROP COLUMN<col\_name>

Пример 1.4

|  |
| --- |
| ALTER TABLE t1 DROP c13 varchar(15) default 'no\_text'; |

результат 1.4



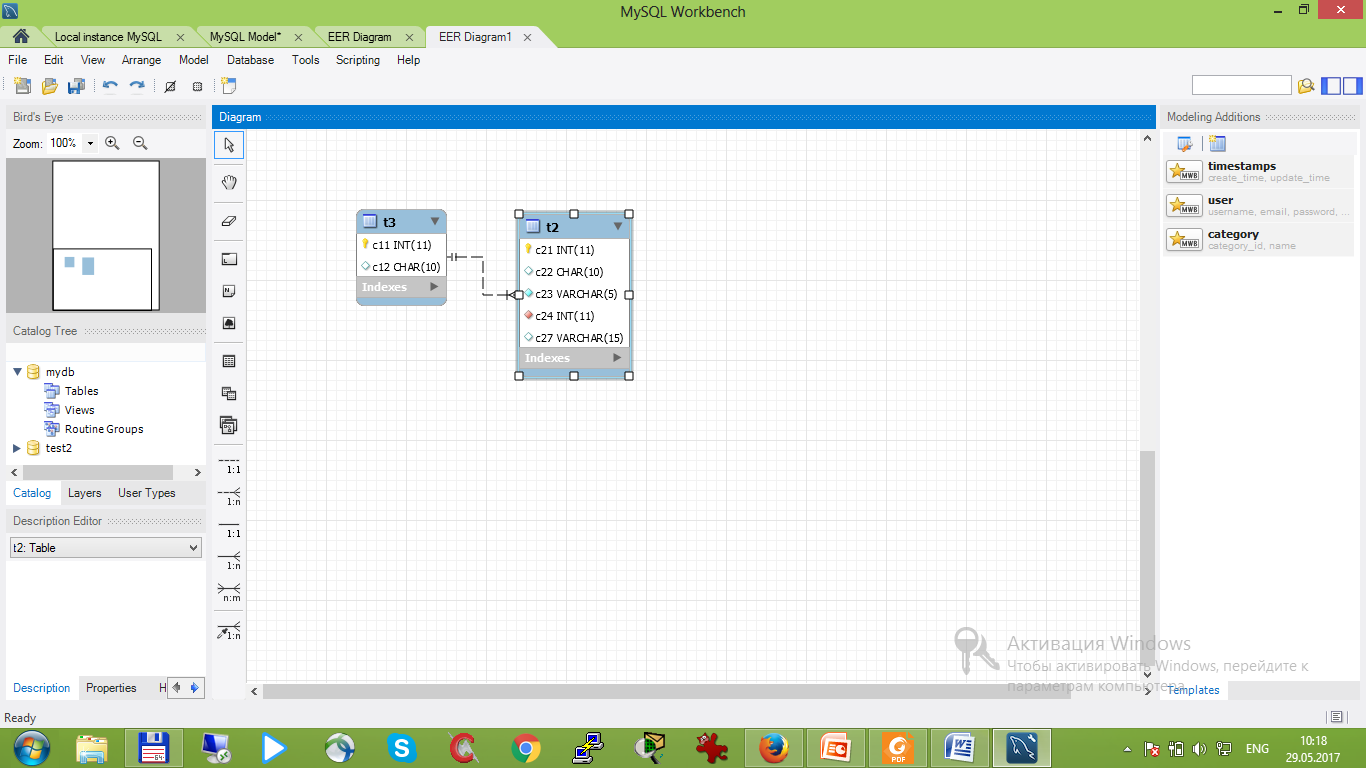
5)Переименование таблицы в MSSQL server возможно через выполнение (exec) хранимой процедуры sp\_rename

exec sp\_rename<tab\_name\_old>,<tab\_name\_new>

Пример 1.6

|  |
| --- |
| exec sp\_rename t1, t3 |

результат 1.6



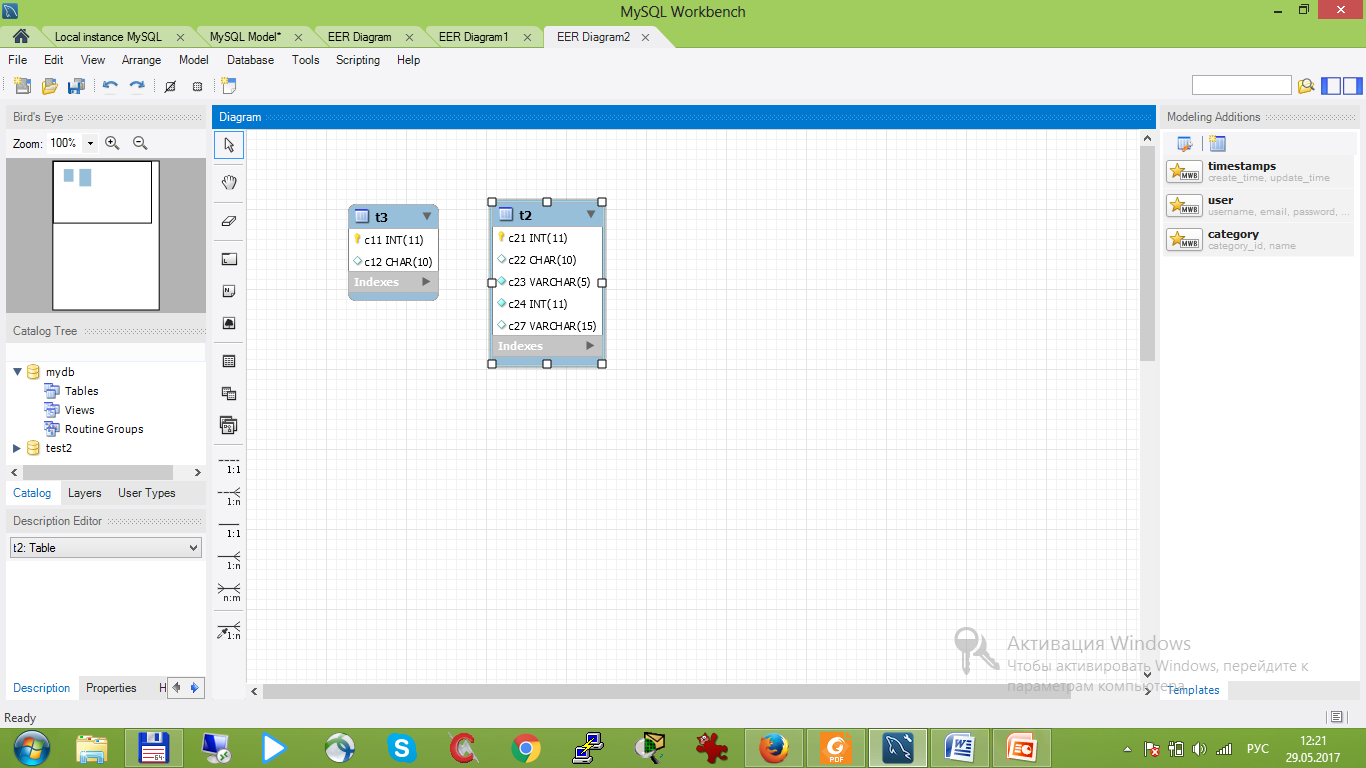
6) Удаление внешнего ключа

ALTER TABLE <table\_name> drop <key\_name>

Пример 1.7

|  |
| --- |
| ALTER TABLE t2 drop FK\_to\_t1; |

результат 1.7



7) назначение нового внешнего ключа:

ALTER TABLE <tab\_name1> add [CONSTRAINT <fk\_name>] foreign key (<col\_name>) references <tab\_name2>(<col\_name3>)

Пример 1.8

|  |
| --- |
| ALTER TABLE t3 add CONSTRAINT FK\_to\_t1 foreign key (c24) references t1(c11); |

8) Назначение нового первичного ключа (невMySQL)

ALTERTABLE <tab\_name1> add [CONSTRAINT<key\_name>]primary key(col\_name1, col\_nam2)

9) Удаление первичного ключа (невMySQL)

ALTER TABLE <tab\_name1>drop<key\_name>

**1.1.3 Удаление таблицы**

DROP TABLE<table\_name>

**1.2 Операторы языка DML**

ЯзыкDML (Data Manipulation Language) является средством управления данными в внутри объектов схемы (БД). К запросам языка манипуляции относятся запросы на добавление, удаление и модификацию кортежей.

**1.2.1 Добавление кортежа** производится командой:

*INSERT INTO имя\_таблицы [(<список столбцов>)] VALUES (<список*  
*значений>)*

Пример 1.8

|  |
| --- |
| INSERT INTO t3 (c12) values ('Петров'), ('Иванов'), ('Сидоров'); |

Результат 1.8:

|  |  |
| --- | --- |
| C11 | C12 |
| 1 | Петров |
| 2 | Иванов |
| 3 | Сидоров |
|  |  |

**1.2.2** Операция **обновления данных** UPDATE требуется при изменении данных, которые надо отразить в базе данных.  
Запрос на обновление может изменить одну или несколько записей.

**-- обновление всех записей в столбце**

*UPDATE tab SET <col\_name>=<val>*

Пример 1.9

|  |
| --- |
| UPDATE t3 SET c12=null |

Результат 1.9:

|  |  |
| --- | --- |
| C11 | C12 |
| 1 | Null |
| 2 | Null |
| 3 | Null |
|  |  |

**-- обновление одной или группы записей (по условию)**

*UPDATE tab SET <col\_name>=<val> WHERE <col\_name1>=<val>*

Пример 1.9

|  |
| --- |
| UPDATE t3 SET c12=null c12='Иванов' where c11=2; |

Результат 1.10:

|  |  |
| --- | --- |
| C11 | C12 |
| 1 | Null |
| 2 | Иванов |
| 3 | Null |

**1.2.3** Оператор **удаления данных** DELETE позволяет удалить одну или  
несколько строк из таблицы в соответствии с условиями, которые задаются  
для удаляемых строк. Синтаксис оператора DELETE следующий:

*DELETE FROM <имя\_таблицы> [WHERE <условия\_отбора>]*

Если условия отбора не задаются, то из таблицы удаляются все строки.

**1.3 Основные типы данных**

Строка переменной длины: varchar(N) и nvarchar(N), где N - максимальная длинна строки

Строка постоянной длины: char(N) и nchar(N), где N - длинна строки

Целое число: int

Число с плавающей точкой: float

Дата и время: datetime(ввод в формате 'ггггммдд чч:мм:сс'')

Дата: date (ввод в формате 'ггггммдд')

Время: time(ввод в формате 'чч:мм:сс')

## Задание к лабораторной работе №1

1) Создайте следующие таблицы

1. таблица STUDENTS содержащая поля

- идентификатор студента

- имя

-фамилию

-отчество

- дату рождения

-идентификатор группы

поле идентификатора студента должно быть первичным ключом в таблице

1. Таблица GROUPS содержащая поля

- идентификатор группы

- наименование

- курс

поле идентификатора группы должно быть первичным ключом в таблице

1. Таблица PLAN содержащая поля

- идентификатор группы

- идентификатор предмета

Все поля образуют первичный ключ

1. Таблица SUBJECTS содержащая поля

- идентификатор предмета

- наименование предмета

- количество часов по предмету

поле идентификатора предмета должно быть первичным ключом в таблице

2. Измените таблицы, создав связи между ними

- STUDENTS.*идентификатор группы* -- GROUPS.*идентификатор группы*

- PLAN.*идентификатор предмета* --SUBJECTS.*идентификатор предмета*

- PLAN.*идентификатор группы* --GROUPS.*идентификатор группы*

3. Заполнить таблицы согласно данным Таблицы 1.1 и Таблицы 1.2

Таблица 1.1

|  |  |  |  |
| --- | --- | --- | --- |
| **ФИО** | **дата рождения** | **Номер группы** | **Курс** |
| Федоренко П.Р. | 25.12.1997 | ПО135 | 1 |
| Зингел О. | 25.12.1985 | ПО135 | 1 |
| Савицкаян Н. | 22.09.1987 | ПО235 | 2 |
| Ковальчук М.Е. | 17.06.1992 | ПО235 | 2 |
| КовригоТ.Р. | 13.05.1992 | ПО335 | 3 |
| Шарапо Н. | 14.08.1992 | ПО335 | 3 |

Таблица 1.2

|  |  |  |
| --- | --- | --- |
| **Предмет** | **Курс** | **Количество часов** |
| Физика | 1 | 200 |
| Математика | 1 | 120 |
| Основы алгоритмизации | 2 | 70 |
| Проектирование БД | 2 | 130 |
| Средства визуального программирования | 3 | 90 |
| Объектно-ориентированное программирование | 3 | 70 |

4) В таблице STUDENTS внесите изменения таким образом, чтобы все студенты, учащиеся в группе ПО135, числились в группе 134.

5) Из таблицы групп удалите запись о группе по 135

6) в таблице SUBJECTS увеличьте количество часов на 30 для предметов

Средства визуального программирования, Объектно-ориентированное программирование

7) в таблицу SUBJECTS добавьте столбец, в который будут внесены данные для формы контроля по предметам (по всем предметам сдаются экзамены, кроме "Основы алгоритмизации" -зачет). Обновите таблицу SUBJECTS с учетом форм контроля по предметам.

8) Из таблицы STUDENTS удалите столбцы с данными отчества.